Mining building information modeling (BIM) event logs for improved project management

Pan, Yue

2021

Pan, Y. (2021). Mining building information modeling (BIM) event logs for improved project management. Doctoral thesis, Nanyang Technological University, Singapore.
<https://hdl.handle.net/10356/152484>

<https://hdl.handle.net/10356/152484>

<https://doi.org/10.32657/10356/152484>

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License (CC BY-NC 4.0).

Downloaded on 26 Apr 2023 07:25:47 SGT

**NANYANG
TECHNOLOGICAL
UNIVERSITY**

SINGAPORE

**MINING BUILDING INFORMATION MODELING
(BIM) EVENT LOGS FOR IMPROVED
PROJECT MANAGEMENT**

**PAN YUE
SCHOOL OF CIVIL AND ENVIRONMENTAL ENGINEERING**

2021

**MINING BUILDING INFORMATION MODELING
(BIM) EVENT LOGS FOR IMPROVED
PROJECT MANAGEMENT**

PAN YUE

School of Civil and Environmental Engineering

A thesis submitted to the Nanyang Technological University
in partial fulfilment of the requirement for the
degree of Doctor of Philosophy

Statement of Originality

I hereby certify that the work embodied in this thesis is the result of original research, is free of plagiarised materials, and has not been submitted for a higher degree to any other University or Institution.

.....March 1, 2021
Date

Yue Pan
Pan Yue

Supervisor Declaration Statement

I have reviewed the content and presentation style of this thesis and declare it is free of plagiarism and of sufficient grammatical clarity to be examined. To the best of my knowledge, the research and writing are those of the candidate except as acknowledged in the Author Attribution Statement. I confirm that the investigations were conducted in accord with the ethics policies and integrity standards of Nanyang Technological University, Singapore and that the research data are presented honestly and without prejudice.

..... March 1, 2021.....
Date

limao Zhang

.....
Zhang Limao

Authorship Attribution Statement

This thesis contains material from 7 papers published in the following peer-reviewed journals, 1 paper accepted at conferences, and 1 paper under review in which I am listed as the first author.

Chapters 2 and 7 are published as **Pan, Y.** and Zhang, L. (2021). "Roles of artificial intelligence in construction engineering and management: A critical review and future trends." Automation in Construction 122: 103517. DOI: <https://doi.org/10.1016/j.autcon.2020.103517>.

The contributions of the co-authors are as follows:

- I was the lead investigator for literature review, formal analysis, and paper writing.
- Prof. Zhang Limao provided the conceptualization and the initial project direction and edited the manuscript drafts.

A part of Chapter 3 is published as **Pan, Y.** and Zhang, L. (2020). "BIM log mining: Learning and predicting design commands." Automation in Construction 112: 103107. DOI: <https://doi.org/10.1016/j.autcon.2020.103107>.

A part of Chapter 4 is published as **Pan, Y.** and Zhang, L. (2020). Sequential Design Command Prediction Using BIM Event Logs. Construction Research Congress 2020: Computer Applications, American Society of Civil Engineers Reston, VA. DOI: <https://doi.org/10.1061/9780784482865.033>.

The contributions of the co-authors are as follows:

- I was the lead investigator for writing, methodology, visualization, investigation, and formal analysis.
- Prof. Zhang Limao provided the conceptualization and the initial project direction and edited the manuscript drafts.

A part of Chapter 4 is published as **Pan, Y.** and Zhang, L. (2020). "BIM log mining: Exploring design productivity characteristics." Automation in Construction 109: 102997. DOI: <https://doi.org/10.1016/j.autcon.2019.102997>.

A part of Chapter 4 is published as **Pan, Y.**, Zhang, L., Li, Z. (2020). "Mining event logs for knowledge discovery based on adaptive efficient fuzzy kohonen clustering network." Knowledge-Based Systems: 106482. DOI: <https://doi.org/10.1016/j.knosys.2020.106482>.

The contributions of the co-authors are as follows:

- I was the lead investigator for writing, methodology, visualization, investigation, and formal analysis.
- Prof. Zhang Limao provided the conceptualization and the initial project direction and edited the manuscript drafts.
- Prof. Li Zhiwu was responsible for reviewing and editing.

A part of Chapter 5 is published as **Pan, Y.**, Zhang, L. and Skibniewski, M. J. (2020). "Clustering of designers based on building information modeling event logs." Computer-Aided Civil and Infrastructure Engineering 35(7): 701-718. DOI: <https://doi.org/10.1111/mice.12551>. A part of Chapter 5 is adopted from a manuscript which is currently under the 1st review as **Pan, Y.** and Zhang, L. "Data-Driven Modeling and Analyzing Dynamic Social Networks for Collaborative Pattern Discovery." Automation in Construction.

The contributions of the co-authors are as follows:

- I was the lead investigator for writing, methodology, visualization, investigation, and formal analysis.
- Prof. Zhang Limao provided the conceptualization and the initial project direction and edited the manuscript drafts.
- Prof. Mirosław J Skibniewski was responsible for reviewing and editing.

A part of Chapter 6 is published as **Pan, Y.** and Zhang, L. (2021). "A BIM-data mining integrated digital twin framework for advanced project management,"

Automation in Construction 124: 103564. DOI:
<https://doi.org/10.1016/j.autcon.2021.103564>. A part of Chapter 6 is published
as **Pan, Y.** and L. Zhang (2021). "Automated process discovery from event logs
in BIM construction projects." Automation in construction 127: 103713.

The contributions of the co-authors are as follows:

- I was the lead investigator for writing, methodology, visualization, investigation, and formal analysis.
- Prof. Zhang Limao provided the conceptualization and the initial project direction and edited the manuscript drafts.

..... March 1, 2021.
Date Pan Yue

ACKNOWLEDGMENTS

First and most importantly, my deep gratitude goes to my supervisor Prof. Zhang Limao for his sincere guidance and kind support throughout the research. Without his advice and encouragement, I cannot move the work forward. His positive attitude towards work encourages me to engage in research with passion. Moreover, his personal generosity makes my time at NTU enjoyable.

I am thankful to Professor Baabak Ashuri and Professor Chuck Eastman at Georgia Institute of Technology, who provide the rich source of data and guide the research. Also, I am grateful to my thesis advisory committee members, Prof Adrian Law and Asst Prof Yi Yaolin, and the precious member Asst Prof Okan Duru, for their constructive advice to my thesis.

I would like to express my greatest regards to my parents. They always give me love and encouragement in whatever I pursue. Their great support makes me have the confidence to keep going and chase my dream of pursuing the Ph.D. degree.

Finally, I am grateful to friends and groupmates to color my life at NTU. It is lucky for me to have them, whose care and help contribute to making my life easier and more pleasant.

TABLE OF CONTENTS

Statement of Originality	I
Supervisor Declaration Statement	II
Authorship Attribution Statement	III
ACKNOWLEDGMENTS	VI
TABLE OF CONTENTS	VII
SUMMARY	XII
LIST OF PUBLICATIONS	XIV
LIST OF TABLES	XVII
LIST OF FIGURES	XIX
LIST OF ABBREVIATIONS	XXIII
CHAPTER 1. INTRODUCTION.....	1
1.1 Research background	1
1.2 Research motivation	5
1.2.1 Challenges and opportunity in BIM data analysis	5
1.2.2 Potentials in BIM event log mining	6
1.3 Research goal and objectives	9
1.4 Thesis outline	11
CHAPTER 2. LITERATURE REVIEW.....	11
2.1 Introduction	11
2.2 BIM adoption in construction project management.....	11
2.3 BIM event log mining	16
2.3.1 Research status	16
2.3.2 Research gap	18
2.4 Studies related to research objectives.....	20
2.4.1 Human behavior prediction.....	20
2.4.2 Work performance assessment.....	22
2.4.3 Social network analysis	24

2.4.4 Process mining	26
2.4.5 Digital twin.....	29
2.5 Chapter Summary.....	33
CHAPTER 3. LEARNING AND PREDICTING DESIGN COMMANDS BY DEEP LEARNING METHODS	35
3.1 Introduction	35
3.2 Methodology	37
3.2.1 Data acquisition and preprocessing.....	37
3.2.2 Data mining.....	39
3.2.2.1 RNN	40
3.2.2.2 LSTM NN	41
3.2.3 Performance evaluation.....	44
3.3 Case study based on RNN.....	45
3.3.1 Data extraction from logs.....	45
3.3.2 RNN model development.....	46
3.3.3 Result analysis.....	48
3.4 Case study based on LSTM NN.....	50
3.4.1 Data preparation	50
3.4.2 Command classification	52
3.4.3 LSTM NN model development.....	56
3.4.4 Result analysis.....	59
3.4.5 Discussions.....	65
3.5 Chapter Summary.....	68
CHAPTER 4. EXPLORING CHARACTERISTICS OF DESIGN PERFORMANCE BY CLUSTERING METHODS	72
4.1 Introduction	72
4.2 Methodology	73
4.2.1 BIM log preprocessing.....	74
4.2.2 Fuzzy Kohonen clustering.....	75
4.2.2.1 Preliminary.....	75
4.2.2.2 EFKCN algorithm	76
4.2.2.3 Proposed AEFKCN algorithm	78

4.2.3 Clustering performance analysis	80
4.2.3.1 Common clustering validity indexes.....	80
4.2.3.2 A new cluster validity index	82
4.3 Case study based on EFKCN	86
4.3.1 Feature extraction	86
4.3.2 Individual-level clustering.....	88
4.3.2.1 Dataset partitioning	88
4.3.2.2 Clustering results analysis.....	92
4.3.3 Team-level clustering.....	98
4.4 Case study based on AEFKCN	101
4.4.1 Experiment setup.....	101
4.4.2 Comparison of results from different clustering algorithms	102
4.4.3 Knowledge discovery from AEFKCN-based log mining	106
4.4.4 Experiments in additional datasets	111
4.5 Chapter Summary	113

CHAPTER 5. DISCOVERING COLLABORATIVE PATTERNS BY SOCIAL NETWORK ANALYSIS 117

5.1 Introduction	117
5.2 Methodology	119
5.2.1 Network development	120
5.2.2 Proposed algorithm for node clustering	120
5.2.2.1 Preliminary.....	120
5.2.2.2 node2vec-GMM algorithm	122
5.2.3 Network analysis	125
5.2.3.1 Common metrics for node importance measurement	125
5.2.3.2 A new defined metric for node importance measurement	127
5.2.3.3 CatBoost regression algorithm for node importance prediction	129
5.2.3.4 Link prediction.....	131
5.3 Case study for community detection.....	132
5.3.1 Construction of social network	132
5.3.2 Implementation of <i>node2vec-GMM</i>	134
5.3.3 Analysis of detected communities.....	139

5.3.4 Validation of <i>node2vec-GMM</i>	143
5.4 Case study for dynamic network analysis	146
5.4.1 Discovery of dynamic social networks	146
5.4.2 Exploration of collaborative patterns	148
5.4.3 Measurement of designers' influence	151
5.4.4 Discussion of structural and behavioral effects on designers' influence	156
5.5 Chapter Summary.....	161
CHAPTER 6. SIMULATING AND INVESTIGATING CONSTRUCTION	
ACTIVITIES BY PROCESS MINING	165
6.1 Introduction	165
6.2 Methodology	168
6.2.1 Current perspective: Process discovery and diagnosis.....	169
6.2.1.1 Algorithms of process discovery.....	169
6.2.1.2 Representations of process models	172
6.2.1.3 Validation of discovered process models.....	173
6.2.1.4 Analysis of discovered process models	175
6.2.2 Future perspective: Process prediction and analysis	176
6.2.2.1 Time series prediction	176
6.2.2.2 Model selection and evaluation.....	178
6.2.3 Digital twin architecture.....	179
6.3 Case study on automated process discovery and analysis.....	182
6.3.1 Data preparation and description.....	182
6.3.2 Process discovery	188
6.3.3 Conformance checking.....	189
6.3.4 Frequency and bottleneck analysis.....	191
6.3.5 Social network analysis	194
6.4 Case study on digital twin implementation	198
6.4.1 Data description.....	198
6.4.2 Modeling of construction process	200
6.4.3 Diagnosis of construction process	203
6.4.4 Prediction of construction process	207
6.4.5 Discussion	212

6.5 Chapter Summary.....	214
CHAPTER 7. CONCLUSIONS AND FUTURE WORKS	218
7.1 Conclusions	218
7.1.1 Key methods.....	219
7.1.2 Key contributions	220
7.2 Future works.....	224
7.3 Future research trends	231
REFERENCE.....	238

SUMMARY

Currently, Building Information Modeling (BIM) serves as a project management tool to inform data-driven decisions in modeling, construction, operation, and maintenance. As BIM is progressively adopted in civil engineering, one kind of important BIM data called event log will be accumulated continuously to bring about some features of “big data”. To be more specific, BIM event logs keep detailed records of timestamp, activity, actor, and others in chronological order to track the evolution of the construction project. Noticeably, a lot of knowledge is hidden behind such an ever-growing data source, which deserves deep exploration. However, it is still a comparatively new development in BIM event log mining due to the difficulty in handling the disordered and non-intuitive log data in unstructured text content. Therefore, the motivation of this thesis is to employ artificial intelligence (AI)-related techniques in massive log data to better comprehend the construction project and shed light on data-driven decision-making. The contributions of this thesis lie in two major aspects. From the technical perspective, it provides an opportunity to fill a gap of data science talent in the Architecture, Engineering, Construction, and Operation (AECO) industry. From the application perspective, it is a significant step beyond existing performance assessment methods heavily relying on subjective judgment, enabling improvements in both the building design and construction process.

In general, the proposed BIM event log framework contains three major steps: (1) Data preparation from massive event logs; (2) AI implementation for log data mining; (3) Knowledge discovery as a smart decision tool. The key findings are summarized as follows: (1) The deep learning-based approach can learn designers’ behavior to make a sequential prediction about the next possible design command class towards automation of the modeling process, and thus following the suggested command classes can potentially accelerate the design and prevent some unwanted mistakes. (2) The clustering-based approach can automatically generate several patterns on behalf of a person’s design behavior characteristics and distinguish design efficiency into the high, medium, and low

level for design performance evaluation, and thus these extracted clusters provide concrete evidence for managers to strategically schedule work. (3) The social network-based approach can graphically understand the collaborative design from discovering potential communities of designers, identifying a designer's role, predicting work transmission and collaboration evolution, which hold the promise of promoting design collaboration through better leadership and work arrangement. (4) The process mining-based approach can simulate and analyze activities of modeling a building with inherent conflicts and uncertainty, which is useful in making process improvement through detecting potential deviations, inefficiencies, and collaboration patterns. Moreover, a digital twin integrating BIM, Internet of Things (IoT), data mining, and process mining is developed for process simulation, bottleneck diagnosis, and performance prediction, which is proven useful in facilitating the better understanding and optimization of physical construction operations. In brief, the proposed BIM event log mining presents a unique opportunity to convert data into meaningful information to provide a variety of value-added services, which is bound to create long-lasting positive impacts on driving construction project management to go through constant innovations towards digitalization and intelligence.

LIST OF PUBLICATIONS

Publication related to this thesis:

- [1] **Pan, Y.** and L. Zhang (2021). "Automated process discovery from event logs in BIM construction projects." Automation in construction 127: 103713.
- [2] **Pan, Y.** and Zhang, L. (2021). "A BIM-data mining integrated digital twin framework for advanced project management," Automation in Construction 124: 103564.
- [3] **Pan, Y.** and Zhang, L. (2021). "Roles of artificial intelligence in construction engineering and management: A critical review and future trends." Automation in Construction 122: 103517.
- [4] **Pan, Y.**, Zhang, L., Li, Z. (2020). "Mining event logs for knowledge discovery based on adaptive efficient fuzzy kohonen clustering network." Knowledge-Based Systems: 106482.
- [5] **Pan, Y.**, Zhang, L. and Skibniewski, M. J. (2020). "Clustering of designers based on building information modeling event logs." Computer-Aided Civil and Infrastructure Engineering 35(7): 701-718.
- [6] **Pan, Y.** and Zhang, L. (2020). "BIM log mining: Learning and predicting design commands." Automation in Construction 112: 103107.
- [7] **Pan, Y.** and Zhang, L. (2020). "BIM log mining: Exploring design productivity characteristics." Automation in Construction 109: 102997.
- [8] **Pan, Y.** and Zhang, L. (2020). Sequential Design Command Prediction Using BIM Event Logs. Construction Research Congress 2020: Computer Applications, American Society of Civil Engineers Reston, VA.

[9] **Pan, Y.** and Zhang, L. "Data-Driven Modeling and Analyzing Dynamic Social Networks for Collaborative Pattern Discovery." Automation in Construction. (Under 1st review)

Other publications:

[1] **Pan, Y.**, Zhang, L., Koh, J. and Deng, Y. (2021). "An adaptive decision making method with copula Bayesian network for location selection." Information Sciences 544: 56-77.

[2] Zhang, G., **Pan, Y.**, and Zhang, L. (2021). "Semi-supervised learning with GAN for automatic defect detection from images." Automation in construction 128: 103764.

[3] **Pan, Y.**, Zhang, G. and Zhang, L. (2020). "A spatial-channel hierarchical deep learning network for pixel-level automated crack detection." Automation in Construction 119: 103357.

[4] **Pan, Y.** and Zhang, L. (2020). "Data-driven estimation of building energy consumption with multi-source heterogeneous data." Applied Energy 268: 114965.

[5] **Pan, Y.**, Zhang, L., Wu, X. and Skibniewski, M. J. (2020). "Multi-classifier information fusion in risk analysis." Information Fusion 60: 121-136.

[6] Zhang, G., **Pan, Y.**, Zhang, L. and Tiong, R. L. K. (2020). "Cross-scale generative adversarial network for crowd density estimation from images." Engineering Applications of Artificial Intelligence 94: 103777.

[7] **Pan, Y.**, Zhang, L., Wu, X., Zhang, K. and Skibniewski, M. J. (2019). "Structural health monitoring and assessment using wavelet packet energy spectrum." Safety Science 120: 652-665.

[8] **Pan, Y.**, Ou, S., Zhang, L., Zhang, W., Wu, X. and Li, H. (2019). "Modeling risks in dependent systems: A Copula-Bayesian approach." Reliability Engineering and System Safety 188: 416-431.

- [9] **Pan, Y.**, Zhang, L., Li, Z. and Ding, L. (2019). "Improved fuzzy Bayesian network-based risk analysis with interval-valued fuzzy sets and DS evidence theory." IEEE Transactions on Fuzzy Systems 28(9): 2063-2077.
- [10] **Pan, Y.**, Zhang, L., Wu, X., Qin, W. and Skibniewski, M. J. (2019). "Modeling face reliability in tunneling: A copula approach." Computers and Geotechnics 109: 272-286.

LIST OF TABLES

Table 3.1. Examples of SQL query in data cleaning.....	39
Table 3.2. Data labeling and examples.....	46
Table 3.3. Prediction results of five continuous command classes.	48
Table 3.4. Comparison of the original dataset and cleaned dataset.....	52
Table 3.5. List of 14 command classes and related Top 5 commands.	54
Table 3.6. Precision, recall, and F1 score for each class.	64
Table 3.7. Comparison of predicted accuracy and training time by different methods.....	68
Table 4.1. Column name and relevant content in the parsed CSV file.....	75
Table 4.2. Detail of dataset for Design #1 targeted in the individual-level clustering.	88
Table 4.3. Detail of dataset for the design team targeted in the team-level clustering.....	88
Table 4.4. Results of regression analysis in cluster 1–3.....	97
Table 4.5. Clustering results and characteristics for datasets of Designer #1–#4.	97
Table 4.6. Clustering results and characteristics for the team-level dataset.....	101
Table 4.7. Description of dataset for Designer #2 (720 data points).....	102
Table 4.8. Parameters setting in five methods.....	102
Table 4.9. Computational cost of five methods.....	106
Table 4.10. Clustering evaluation from new index.	106
Table 4.11. Cluster properties of dataset for Design #2.	110
Table 4.12. Results of the Mann-Whitney U Test.....	111
Table 4.13. Clustering results in three datasets from UCI repository.	112
Table 4.14. Clustering results of three new datasets.	113
Table 5.1. Characteristics of the BIM-based design collaboration.....	134
Table 5.2. Probability assignment for each designer in community #1– #3.....	138
Table 5.3. Top five critical designers in cluster 1-3 by different web-page ranking.....	143
Table 5.4. Comparison of clustering performance from different node clustering methods.....	146
Table 5.5. Characteristics of two collaboration patterns (i.e., large and small groups).	150
Table 5.6. The top-5 most critical designers ranked by the impact score and three centrality metrics in per month.	155
Table 5.7. Comparison of prediction performance from different machine learning algorithms.	159
Table 6.1. Six attributes in the BIM as-planned event logs.....	186
Table 6.2. Evaluation of the discovered process model based on the inductive miner.	191
Table 6.3. Evaluation of the discovered process model based on the fuzzy miner.	194
Table 6.4. Characteristics of the three social networks based on different metrics.....	196
Table 6.5. Cluster detection in the discovered social network based on modularity.....	197
Table 6.6. Example of continuous records from construction event logs in the CSV format. ..	200
Table 6.7. Evaluation of the discovered process model.	203
Table 6.8. Summary of time series data.	209

Table 6.9. Goodness of fit for six candidate ARIMAX models.....	210
Table 6.10. Coefficient estimation of ARIMAX (2, 1, 2) model.	211
Table 6.11. Evaluation of predictions from different time series algorithms.....	214

LIST OF FIGURES

Figure 1.1. Structure of the thesis.	12
Figure 2.1. Examples of data items in BIM design event logs (Yarmohammadi, Pourabolghasem et al. 2017).	17
Figure 2.2. Architecture structure of (a) RNN; (b) LSTM NN.	22
Figure 2.3. Procedure of worker performance evaluation based on mobile sensing data.	24
Figure 2.4. Description of BIM-based collaborative design by a social network.	26
Figure 2.5. Typical tasks in process mining.	29
Figure 2.6. Architecture of digital twin.	33
Figure 3.1. Workflow of the proposed command prediction method. (Note: DL is the abbreviations of deep learning).	37
Figure 3.2. Example of the parsed CSV file.	39
Figure 3.3. General process of RNN.	41
Figure 3.4. Memory block in LSTM NN.	43
Figure 3.5. Pie chart of command number in each class. (The number outside the brackets is the command frequency and the number inside the brackets is the command percentage.)	46
Figure 3.6. Learning curve of: (a) Loss; (b) Accuracy.	48
Figure 3.7. Confusion matrix of prediction results in the testing set.	49
Figure 3.8. ROC and AUC of command class: (a) 1; (b) 2; (c) 3; (d) 4; (e) 5; (f) 6.	50
Figure 3.9. Design command execution frequency in each project.	52
Figure 3.10. Percentage of command number in 14 command classes and three journal events.	56
Figure 3.11. Accuracy curves at training and test sets: (a) training set at different learning rates; (b) test set at different learning rates; (c) training set with different numbers of memory cells; (d) test set with different numbers of memory cells.	58
Figure 3.12. Loss and accuracy curves at training and test sets: (a) Loss curve of training and test set; (b) Accuracy curve of training and test set.	59
Figure 3.13. Histogram of test accuracy.	63
Figure 3.14. Probabilistic results to predict the actual command class 12 in (a); Probability distribution of the actual command class 12 to be predicted as command class (b) 1; (c) 2; (d) 3; (e) 4; (f) 5; (g) 6; (h) 7; (i) 8; (j) 9; (k) 10; (l) 11; (m) 12; (n) 13; (o) 14.	64
Figure 3.15. Example of a command sequence with 11 commands.	65
Figure 3.16. Accuracy at different timesteps based on (a) training set; (b) test set.	67
Figure 3.17. Accuracy about ten experiments after 100 epochs based on (a) training set; (b) test set.	68
Figure 4.1. Flowchart of the proposed clustering method.	74
Figure 4.2. Examples of three continuous records in BIM design log files.	74
Figure 4.3. Clustering results in 3D space.	90
Figure 4.4. Pair plots of four features in the dataset about Designer #1.	91
Figure 4.5. Boxplots of feature x_3 and x_4	92
Figure 4.6. An example of KDE for feature x_3 and x_4	92

Figure 4.7. Violin plots of feature x_1	96
Figure 4.8. Variation with time about (a) Number of commands (x_3); (b) Length of activation time (x_4).	96
Figure 4.9. Regression analysis about x_4 and x_3 in: (a) Cluster 1; (b) Cluster 2; (c) Cluster 3. ..	96
Figure 4.10. Membership value for data in: (a) Cluster 1; (b) Cluster 2; (c) Cluster 3.....	100
Figure 4.11. Boxplots and data scatter of feature: (a) Number of sessions (x_5); (b) Number of activation days (x_6); (c) Number of commands (x_7).	101
Figure 4.12. Visualization of clustering results by (1) KCN; (2) FCM; (3) FKCN; (4) EFKCN; (5) AEFKCN.....	105
Figure 4.13. Comparison of clustering results in the pair of (1) KCN-AEFKCN; (2) FCM-AEFKCN; (3) FKCN-AEFKCN; (4) EFKCN-AEFKCN.....	105
Figure 4.14. Evaluation of clustering results by three CVIs: (1) SI; (2) CHI; (3) DBI.....	105
Figure 4.15. CVI for each cluster number: (a) CE; (b) XB; (c) CHI; (d) DBI.....	109
Figure 4.16. Data distribution of clustering results from AEFKCN.	109
Figure 4.17. Membership value in three clusters: (1) Cluster 1; (2) Cluster 2; (3) Cluster 3. ..	110
Figure 4.18. Boxplots and scatters in cluster 1-3 for feature: (a) Number of executed commands x_3 ; (b) Activation time x_4	110
Figure 5.1. Framework of the network-enabled BIM design event log mining.....	119
Figure 5.2. Example of a simple collaborative network.	120
Figure 5.3. Example of six continuous records in BIM design logs.	133
Figure 5.4. Framework of the network-enabled BIM design event log mining.....	134
Figure 5.5. Node features from (a) Adjacency matrix visualized by a heatmap; (b) node2vec algorithm visualized by t-SNE.....	137
Figure 5.6. AIC and BIC for each cluster number.	137
Figure 5.7. Results of community detection visualized in (a) Gaussian distribution; (b) BIM-based design collaboration network.	138
Figure 5.8. Comparison of clusters measured by (a) Degree centrality; (b) Closeness centrality; (c) Betweenness centrality; (d) Eigenvector centrality.	141
Figure 5.9. Comparison of clusters ranked by (a) PageRank; (b) Authority; (c) Hub.	142
Figure 5.10. Sankey diagram about the design task flows among clusters.	142
Figure 5.11. Top 12 most possible links based on the value of Adamic/Adar index for (a) Designer #31 in cluster #1; (b) Designer #9 in cluster #2; (c) Designer #18 in cluster #3. (The number in brackets are the cluster label.)	142
Figure 5.12. Top 12 most possible links based on the value of SimRank for (a) Designer #31 in cluster #1; (b) Designer #9 in cluster #2; (c) Designer #18 in cluster #3. (The number in brackets are the cluster label.).....	143
Figure 5.13. Visualization of designer clustering results in 2D by: (a) MF-GMM; (b) DeepWalk-GMM; (c) LINE (2nd)-GMM; (d) Node2vec-GMM; (e) MF-Kmeans; (f) DeepWalk-Kmeans; (g) LINE (2nd)-Kmeans; (h) Node2vec-Kmeans.	146
Figure 5.14. Structure of the monthly-based collaborative networks for design work.	148
Figure 5.15. Network structural characteristics: (a) Relationship in network density, modularity, and average shortest path length; (b) Mean value of three centrality metrics and the 95% confidence interval.....	151

Figure 5.16. Results of the impact score and their validity: (a) Designers' impact score in two collaborative groups; (b) The Kendall'tau correlation coefficient between the impact score and three benchmark metrics; (c) Similarities for top-5, 10, and 15 designers between the impact score and three benchmark metrics. (Note: DC, CC, and BC are the abbreviations of the degree centrality, closeness centrality, and betweenness centrality, respectively. IS represents the impact score.).....	154
Figure 5.17. Variation in the role importance of designers based on the impact score for networks in: (a) the large collaborative group; (b) the small collaborative group.	155
Figure 5.18. Relationship between the impact score and features of network structures (degree) and designers' behaviors (number of days, tasks, and commands). (Note: The "pearsonr" is the Pearson correlation coefficient and the "p" is the P-value.)	160
Figure 5.19. Relationship between the centrality metrics and behavioral features.	160
Figure 5.20. Overall performance of the CatBoost model: (a) Predictive results and ground truth of designers' influence; (b) Scatter plots of the standardized residual of the predictions; (c) Distribution of the standardized residual with a kernel density estimate.....	161
Figure 6.1. Process mining-based framework for BIM event log mining.....	169
Figure 6.2. Examples of: (a) Petri nets; (b) BPMN; and (c) Process tree (AND means parallel composition, XOR means exclusive choice, and SEQ means sequential composition).	173
Figure 6.3. Architecture of the proposed digital twin for a BIM-enabled construction project.	182
Figure 6.4. Bubble chart about the relationship in frequency, duration, and task types of cases.	187
Figure 6.5. Dotted chart about cases, events, and the corresponding timestamp in a participant-specific process model.	187
Figure 6.6. Representation of the process model by: (a) Petri net; (b) Process tree.	189
Figure 6.7. Process model from the inductive miner.	191
Figure 6.8. Mode concepts of the discovered process model from the inductive miner: (a) edge and activity; (b) concurrency activities; (c) model move deviation; and (d) log move deviation.	191
Figure 6.9. Process model from the fuzzy miner focusing on: (a) Absolute frequency; (b) Mean duration.	193
Figure 6.10. Three different social networks based on metrics: (a) Handover of Work; (b) Subcontracting; and (c) Working Together. (Note: Number in brackets are the node degree.)	196
Figure 6.11. Importance of participants measured by the PageRank and HITS.	197
Figure 6.12. Comparison of collaboration metrics in three networks.	198
Figure 6.13. 4D snapshots for the virtual model at the end of (a) Feb; (b) May; (c) Aug; and (d) Dec. (Note: Point clouds are also provided in (d).)	202
Figure 6.14. Task-centered process model represented by (a) BPMN; and (b) Petri nets.	203
Figure 6.15. Worker-centered process model represented by (a) BPMN; and (b) Petri nets....	203
Figure 6.16. Fuzzy process model about May for bottleneck detection: (a) Task-centered model; and (b) Worker-centered model.	206
Figure 6.17. 4D model visualization of the certain bottleneck in task "External facade work".	206
Figure 6.18. Plots and the augmented Dickey-Fuller test for: (a) Original time series data; and (b) Stationary data after the first-order difference.	209
Figure 6.19. ACF and PACF plots for stationary data after the first-order difference.....	210

Figure 6.20. Plots of the forecast line and corresponding true value in: (a) Whole dataset; and (b) Test set.....	211
Figure 6.21. Residual errors in: (a) Whole dataset; (b) Training set; and (c) Test set.	212
Figure 6.22. (a) and (b) Variation of task number and worker month by month; and (c) Relationship between the number of tasks and workers.	214
Figure 6.23. Comparisons of predictions from different time series algorithms visualized in: (a) Whole dataset; and (b) Test set.	214
Figure 7.1. Summary of adopted methods	220

LIST OF ABBREVIATIONS

Abbreviations

2D
3D
AI
AIC
ACF
AECO
AEFKCN
AMI
ARI
ARIMAX
AR/VR
AUC
BIC
BIM
BPMN
CAD
CatBoost
CDE
CHI
CI
CNN
CSV
CVI
DBI
DM
EFKCN
EM
FCM
FKCN
FPR
GBDT
GMM
HITS
IFC
IQR
IoT
KCN
KDD
KDE

Full terms

Two-Dimensional
Three-Dimensional
Artificial Intelligence
Akaike Information Criteria
Autocorrelation Function
Architecture, Engineering, Construction, and Operation
Adaptive EFCKN-Based Algorithm
Adjusted Mutual Information
Adjusted Rand Index
Multivariate Autoregressive Integrated Moving Average
Augmented/Virtual Reality
Area under the ROC Curve
Bayesian Information Criteria
Building Information Modeling
Business Process Modeling Notation
Computer-Aided Design
Categorical Boosting
Common Data Environment
Calinski-Harabasz Index
Confident Interval
Convolutional Neural Networks
Comma Separated Values
Cluster Validity Indices
Davies-Bouldin Index
Data Mining
Efficient Fuzzy Kohonen Clustering Network
Expectation-Maximum
Fuzzy C-means
Fuzzy Kohonen Clustering Network
False Positive Rate
Gradient Boosting Decision Tree
Gaussian Mixture Model
Hypertext Induced Topic Search
Industry Foundation Classes
Interquartile Range
Internet of Things
Kohonen Clustering Network
Knowledge Discovery in Databases
Kernel Density Estimation

KNN	K-nearest Neighbors
LiDAR	Light Detection and Ranging
LLE	Locally Linear Embedding
LSTM NN	Long Short-Term Memory Neural Network
MEP	Mechanical, Electrical and Plumbing
MF	Matrix Factorization
MAE	Mean Absolute Error
MSE	Mean Square Error
nD	Multi-dimensional
NLP	Natural Language Processing
O&M	Operation and Maintenance
PACF	Partial Autocorrelation
PC	Principle Component
PCA	Principal Component Analysis
PDF	Probability Density Function
PI	Predictive Interval
RF	Random Forest
RFID	Radio-Frequency Identification
RMSE	Root Mean Square Error
RNN	Recurrent Neural Network
ROC	Receiver Operating Characteristic
SARIMA	Seasonal ARIMA
SARIMAX	Seasonal ARIMAX
SGD	Stochastic Gradient Descent
SI	Silhouette Index
SNA	Social Network Analysis
SQL	Structured Query Language
SVM	Support Vector Machine
SVR	Support Vector Regression
TPR	True Positive Rate
t -SNE	t -distributed Stochastic Neighbor Embedding
UAV	Unmanned Aerial Vehicles

CHAPTER 1. INTRODUCTION

1.1 Research background

Rather than a simple virtual model or software, Buildings Information Modeling (BIM) can be typically defined as a shared digital representation of a built asset to facilitate design, construction, and operation processes to form a reliable basis for decisions according to the British Standard ISO 19650:2019 (ISO 2019). Different researchers have their own conception of BIM. For example, Ding et al. (2014) treated BIM as a process of creating, utilizing, and managing digital representations with semantically rich information in a common data environment (CDE). Belsky et al. (2016) presented that BIM is emerging to accelerate informatization and revolution in Architecture, Engineering, Construction, and Operation (AECO) industry based upon information integration and interoperability. As for me, BIM serves as a rich database for capturing and managing contextual information throughout the whole life cycle of a construction project, including the phase of design, construction, operation, and maintenance (O&M).

As reviewed, BIM is profoundly innovating the construction field worldwide. From an investigation by the McGraw-Hill Construction Company, the industry adoption of BIM has surged from 28% in 2007 to 71% in 2012, and contractors (74%), architects (70%), and engineers (67%) are the top three players reaching highest engagement level in BIM-based projects (Construction 2012). By 2016, BIM has gradually extended to all over the world with a relatively high utilization ratio of 77%-85% (Ghaffarianhoseini, Tookey et al. 2017). Currently, BIM continuous to gain global prominence, and BIM awareness has become universal. Until 2019, the USA and UK have performed as two leading countries in BIM technology, where BIM is forced to use. (Hamma-adama and Kouider 2019). To be more specific, the USA is not only the biggest producer and consumer of BIM products and solutions, but also the hub of technology development nowadays (Zhou, Yang et al. 2019). In the UK, the awareness of BIM utilization has

reached over 90% in 2013, and BIM level 2 has even become mandatory for public sector works (Travaglini, Radujković et al. 2014). Although China is a relatively late starter in BIM, the government has formulated a series of relevant policies and standards to actively promote BIM since 2011, and the recent large-scale projects, including Shanghai tower and Shanghai Disneyland, are the representative cases of the successful BIM application (Liu, Wang et al. 2017). Under the fast development and application of BIM, the annual number of relevant research papers has exhibited an upward tendency. According to a literature review by Yin et al. (2019), the curve for BIM publications increased rapidly year by year since 2005 and two bursts of publications appeared in 2014 and 2017. Another BIM-related review by Mannino et al. (2021) uncovered that there was an increasing interest in the integration of BIM with the emerging technique IoT in the recent two years (2019 and 2020). In this regard, BIM adoption as a hot topic is attracting ever-growing attention from academia to improve AECO practice, which is believed to be the promising future direction for sustainable and smart project management.

Since BIM has shown its potential benefits in information visualization, integration, interoperability, and sharing, the usefulness of successful BIM implementation has been highlighted from the data layer (Li, Wu et al. 2017). To be more specific, BIM incorporating various aspects, disciplines, and systems of a facility within a model is more than a digital representation, which actually serves as a project management tool and process to enhance the automatic information management and knowledge exchange across the project lifecycle (Zhao 2017, Antwi-Afari, Li et al. 2018). Therefore, BIM paves a new way for project participants in different roles like designers, engineers, managers, and others to more accurately and efficiently collaborate for time and cost saving, error and rework reduction, and others. Serving as a shared knowledge center on open standards for interoperability, BIM has been proved to bring great performance improvement in intelligent project management from mixed perspectives (Wu 2013). The value of BIM deployment has been highlighted in transforming the design and construction process, which is particularly beneficial for designers and project managers as presented below.

(1) Designers: One of the most popular uses of BIM is to help designers create semantically enriched and digital multi-dimensional (nD) models with parametric objects by object-oriented modeling software (e.g. *Autodesk Revit*, *Sketchup*) (Volk, Stengel et al. 2014). At the moment, the BIM-based design is gradually replacing the traditional paper-based two-dimensional (2D) Computer-Aided Design (CAD) tools, enabling designers to quickly rectify the model and gain easy access of model information (Merschbrock 2012, Ding, Zhou et al. 2014). Since BIM has been proved to potentially improve the design work in terms of reducing design errors cost and time along with facilitating communication in designers and managers, more and more designers conduct the BIM-based design in recent five years all over the world (Love, Edwards et al. 2011, Petrova, Pauwels et al. 2019). According to a survey in 2012 (Shaikh, Raju et al. 2016), 84% of respondents believe that BIM is useful in visualization. Moreover, nearly half of architects in the United States have applied BIM in more than 60% of projects (Azhar 2011); 55.88% of design tasks in the UK often adopt BIM tools (Eadie, Browne et al. 2013); 74% of designers in South Korea have modeling experience in BIM (Son, Lee et al. 2015).

(2) Project managers: It should be noted that BIM is far more than three-dimensional (3D) parametric models to deliver value from the design-related work. As a digital project management tool, BIM can generate, maintain, and share abundant flows of information to provide a wealth of data sources for project analysis. The time (4D) and cost (5D) dimension of BIM can also be incorporated to offer efficiency and quality insights within the construction project (Bradley, Li et al. 2016). As a result, BIM can assist project managers to plan and simulate the construction progress logistics in a data-driven manner, aiming to smooth the complicated executing process with improved visualization, cooperation, scheduling, productivity, and safety control (Matthews, Love et al. 2015). By 2014, around 60% of project managers in the world have operated BIM implementation at a medium or high level for delivering successful projects in great efficiency, high quality, and cost effectiveness (Construction 2014). Under the full exploration of rich data accumulated in BIM, project managers can therefore form useful guidance to promote collaboration and communication, reduce construction errors, conflicts, reworks, cost, and project duration (Chen and Luo 2014). That is to say, project managers are put in the

position of project leaders in the BIM-based project, who focus on the progress in the job site and check it against the plans to constantly optimize the project delivery. By 2014, around 60% of project managers in the world have operated BIM implementation at a medium or high level, who contribute to the project success.

Moreover, with the growth of BIM applications in the data layer, it is worth noting that massive data is continually accumulated into large sizes. Notably, one of the important BIM data sources is the event logs, which automatically capture a variety of data related to the entire model evolution in chronological order, including timestamps, system environment, modeling operations, designer-software interaction, and others. In other words, BIM event logs are semantically rich data to be gathered passively without human intervention, which presents valuable opportunities in discovering a wealth of hidden knowledge in complex engineering projects. This is similar to a topic called web log mining, which investigates web logs in depth by the means of various data mining (DM) techniques to retrieve navigational patterns and predict users' preferences under steps of data preprocessing, pattern discovery, and result analysis (Srivastava, Cooley et al. 2000). In the same way, the BIM event log is made up of process-specific sequences related to the modeling activities, including cases, persons, time stamps, and others, which is the value-added data to track the executed procedure that has occurred in the entire project session. Proper DM approaches can also be implemented in the huge amount of BIM event logs, which hold the promise to objectively monitor modeling procedures, uncover valuable patterns, and make intelligent predictions for informing strategic decisions in a complicated construction project. However, because of the difficulty in processing the ever-increasing and text-format event logs, there are still few works in BIM event log mining. In other words, BIM event log mining has not reached its full potential yet in latent knowledge discovery for improvement of the design process and construction workflow in a data-driven manner. To further narrow the gap in BIM event logs and data science, I intend to leverage various data mining approaches to investigate the ever-increasing availability of BIM event logs for different purposes in this thesis. It is expected that efforts in BIM event log mining contribute to boosting the high degree of automation and digitalization in construction.

1.2 Research motivation

1.2.1 Challenges and opportunity in BIM data analysis

As the application of BIM grows in the data layer, such as information integration and interoperability in AECO industries, an increasing volume of disordered and non-intuitive data is accumulated automatically and increases exponentially in the BIM platform, bringing about some features of “big data”. For instance, the BIM design data of an airport terminal with 548,300 m² can reach 50 GB (Lin, Hu et al. 2016). The huge accumulation of BIM data will impose heavy burdens on data manipulation. Also, a lot of uncertainty, subjectivity, and ambiguity are inherent in data related to the project execution, which will negatively confuse the data analysis and even return unconvincing results. That is to say, it is not a straightforward task in exploring BIM data due to data overload and diversity. The main challenges come from two aspects (Peng, Lin et al. 2017). For one thing, inexperienced users are likely to feel overwhelmed in handling the massive and complex data records, who will have difficulties in capturing useful information and features. For another, inaccurate data and poor data management will adversely influence the data quality, which will possibly generate unreliable knowledge discovery and decision making. Thus, there is still a huge gap between BIM data and data science talent.

Since it is a comparatively new development in exploring BIM from the data layer, it remains a matter of concern to make the utmost of the massive BIM data. To seek a latent solution, proper artificial intelligence (AI) focusing on data mining (DM), such as statistical model, machine learning, deep learning, process mining, and others, can be carried out, which is also known as Knowledge Discovery in Data (KDD). More specifically, DM is responsible to automatically learn characteristics and patterns from the increasing BIM data to achieve automatic clustering and predicting. As a result, these DM-based solutions can deeply explore the large volumes of raw BIM data to capture meaningful patterns and trends, which can eventually return useful decision-oriented information to instruct the ongoing projects. It is believed that a variety of DM methods can potentially become the next digital frontier to drive the high level of automation and intelligence in construction project management. Currently, some researchers have concentrated on improving the construction, operation, and maintenance phase using DM

methods. For instance, Hu and AbouRizk (2014) explored the historical BIM data by a linear regression model to estimate the man-hour requirements and make cost-effectiveness plans for steel fabrication projects. Peng et al. (2017) developed a novel BIM-based data mining approach under clustering, outlier detection, and pattern mining, in order to enhance resource usage and maintenance efficiency. Kang and Choi (2018) proposed a BIM-based data mining method with data integration and function extension to support building energy management. These existing studies mainly focus on the phase of building operation and maintenance (O&M), which perform data analysis to operate and maintain a constructed facility to meet the anticipated functions over its lifecycle. That is to say, efficient information utilization can add additional value to BIM applications, which have shown benefits in cost reduction, energy optimization, and risk control. However, there are still very few DM-related studies concerning the design and construction processes, where a great deal of uncertainty, subjectivity, and innovations are involved in. Therefore, I intend to apply different DM methods to discover some implicit information and valuable knowledge regarding project evolution particularly embedded in the design and construction stage. It is expected to open a new way to understand the project evolution and evaluate participants' performance, which can potentially optimize the project execution progress in a data-driven manner.

1.2.2 Potentials in BIM event log mining

Great attention should be paid to the BIM data layer. During the project execution, BIM can passively and continually gather mass data concerning all aspects of the BIM-based project, including graphical models, resources, costs, safety issues, time, and others, which paves the way to overcome the limitation of human interference (Boje, Guerriero et al. 2020). As a standard and digital description of the building asset industry, the typical BIM data called the Industry Foundation Classes (IFC) serves for archiving and exchanging project information, which has been supported by a lot of BIM software packages and promotes interoperability among them (Barda, Riesel et al. 2020). Notably, IFC developed by buildingSMART (previously known as, the International Alliance of Interoperability, IAI) is an open and neutral data schema to save digital building

descriptions, mainly serving as a global standard for BIM data exchange (Chen, Papandreou et al. 2017).

So far, more and more BIM-enabled projects have been performed on the level of the IFC schema (Liebich 2010). One of the most commonly used schemas is IFC4, which extends supports for buildings, building services, and structural domains by logically codifying multi-information, including entities, attributes, relationships, abstract concepts, processes, and people (Liebich 2013). However, a problem remains that most of computer algorithms have difficulties in directly handling and understanding IFC data. It becomes a critical task to transform the available IFC data into a proper data structure that could be easily explored to bring additional value in a certain business/engineering context. Notably, IFC4 provides an opportunity for users to directly query the IFC model for information extraction with no need of comprehension in the complex IFC specification (Zhang and Issa 2013). Some studies have developed various algorithms for the convenience of retrieving important information from IFC and reducing information redundancy (Sun, Liu et al. 2015). After data retrieval, extracted IFC entities and other meaningful information can be regularly organized in a Comma Separated Values (CSV) file. The new CSV file is made up of several attributes, including cases, activities, persons, and time, aiming to capture flows of activities in chronological order. Since this prepared CSV contains a set of cases and each case comprises a sequence of events/activities along with the timestamp, it can be reasonably regarded as event logs according to the definition of log data given in (Van der Aalst 2016). Noticeably, this kind of event log is a supplementary BIM data file to offer rich process-specific records. To fully explore these BIM event logs using a variety of DM algorithms has great potential to solve some challenges attributed to the large BIM-model file issues. Nevertheless, relevant studies on BIM event log mining are still rare, which deserve more attention to output actionable insights into the BIM-enabled project.

During the modeling process, Journal file in *Autodesk Revit* is the name of the event log. They are initially utilized by BIM software engineers to diagnose errors and fix bugs of the software. Later on, they are regarded as a rich source of process-specific data to be updated constantly, which document the full range of executed activities and track model

evolution without human intervene, such as the conceptual design, operation steps, and knowledge exchange among various participants. During the construction stage, BIM event logs can be represented by IFC entities, such as *IfcProcess*, *IfcControl*, *IfcActor*, and others. Then, important information associated with cases and events can be retrieved from IFC files and saved in Comma Separated Values (CSV) files. The output of the ideal data structure in CSV is also known as the event log, which is made up of sequential cases and events with typical attributes, like timestamp, activity, actor, and others (Van der Aalst 2016). It should be noted that a lot of valuable knowledge regarding project evolution will be embedded in event logs. A special focus can be on various DM techniques to exploit the growing availability of BIM event logs in a meaningful way, aiming to reveal valuable insights into the real executed processes towards better management.

Notably, these BIM event logs are very similar to web server logs, an automatic recording of activities a user performs in sequence (Shi and Yang 2013, Zhang, Wen et al. 2018). In pursuit of web intelligence, web mining based on logs (Yao, Raghavan et al. 2008, Yu, Huang et al. 2008, Slanzi, Balazs et al. 2017, Slanzi, Pizarro et al. 2017), including web usage mining, web content mining, and web structure mining, has been developed maturely in extracting valuable information from the web. For example, web usage log mining has demonstrated promise in discovering hidden knowledge about user's navigation behavior, which can be utilized for developing recommendation systems and web content personalization to satisfy users' preferences and achieve users' better surfing experience (Géry and Haddad 2003, Guerbas, Addam et al. 2013, Lopes and Roy 2015). Due to the superior performance in web log mining, there are reasons to believe that BIM event logs, a high-fidelity operable dataset with similar characteristics as web logs, are worthy of deep exploration. Likewise, proper AI-related approaches regarding DM can also be implemented in the huge amount of BIM event logs gathered passively, which are effective in objectively monitoring modeling procedures, uncovering valuable features of participants' behavior, and even realizing evidence-based decision making in complicated tasks. In the end, the likelihood of project success can be possibly raised in a data-driven way.

1.3 Research goal and objectives

BIM event logs, which automatically keep detailed records on the project execution process, are the basis for data acquisition and data mining. The overall goal of this thesis is to propose novel frameworks of BIM event log mining for different purposes and verify them in real-world datasets provided by an international construction firm for improved project management. The practical value of this thesis is to evaluate, control, and optimize the complex project evolution under a high degree of automation and intelligence, which can narrow the gap between BIM data and the data science talent. As a solution, various AI techniques, including statistical models, machine learning, deep learning, and process mining, are carried out in the log about an ongoing year-long construction project to realize data mining, and thus useful knowledge can be discovered from different perspectives. Eventually, extensive analytical results provide an insight into BIM event logs to fully understand and assess both the project execution and participants' performance, which can drive the phase of design and construction to be more efficient and reliable. To accomplish the research goal, four research objectives are put forward as follows.

- The first objective of the research is to develop a deep learning-based framework to learn sequential data extracted from BIM design event logs and predict the next possible design command class intelligently towards automation of the design process. It can be realized by two deep learning models, namely the Recurrent Neural Network (RNN) and the Long Short-Term Memory Neural Network (LSTM NN). As a result, the intelligent design command predictions will provide designers with reliable suggestions about the next possible command based on probabilities, which are prone to reduce the likelihood of possible wrong commands and enhance operational efficiency.
- The second objective of the research is to develop a clustering-based method to explore design behavior patterns and evaluate design productivity from both the individual and team level. Since design behavior is non-deterministic and subjective, a novel clustering algorithm named efficient fuzzy Kohonen clustering network

(EFKCN) is utilized to produce informative clusters containing different characteristics. Moreover, for yielding the more satisfactory clustering quality and efficiency, a hybrid clustering algorithm named adaptive efficient fuzzy Kohonen clustering network (AEFKCN) is proposed with a modified learning rate to accelerate the convergence. A new clustering validity index (CVI) only relying on boundary points is designed to reduce computational complexity. Based on the in-depth analysis in clusters, the performance of designers and teams can be evaluated without unnecessary individual bias, which supports project managers to rationalize work allocation and smooth the design process.

- The third objective of the research is to develop a network-enabled event log mining approach for modeling and understanding the BIM-based collaborative design work. A novel algorithm termed *node2vec-GMM* combining a graph embedding algorithm named *node2vec* and a clustering method named Gaussian mixture model (GMM) is proposed to study the network structure and cluster designers into several potential communities. The partitioned communities can be analyzed in terms of node important measurement and link prediction. Besides, the collaborative design can be mapped into dynamic social networks with the notion of time, in order to capture the variation of collaboration patterns during the design process. An emerging machine learning algorithm named Categorical boosting (CatBoost) can be built to predict designers' influence intelligently under the consideration of both network structure and human behavior. Therefore, managers can refer to results from social network analysis (SNA) to monitor the whole course of the BIM-based design and formulate more optimized work plans to increase collaboration opportunities.
- The fourth objective of the research is to implement techniques of process mining to simulate and analyze the end-to-end activities of modeling a building embedded in the BIM event log. To begin with, there is a need to retrieve meaningful information from logs by the inductive mining and fuzzy mining algorithms, which are used to automatically build process models as a succinct description of the complex construction process. Then, the discovered process model is analyzed deeply under the joint use of conformance checking, frequency and bottleneck analysis, and social

network analysis, in order to provide evidence in process improvement through identifying deviations, inefficiencies, and collaboration features. Furthermore, to make full use of event log data, a closed-loop digital twin framework can be created under the integration of BIM, the Internet of Things (IoT), and process mining techniques. Based on fuzzy mining algorithm and multivariate autoregressive integrated moving average (ARIMAX) model, the virtual part of the digital twin can foresee possible bottlenecks in the current process and predict the variation trend of construction progress in the next phase. In the end, data-driven decision making can be achieved to strategically smooth and accelerate the construction process along with increasing collaboration opportunities, which can expectedly reduce the risk of project failure ahead of time.

1.4 Thesis outline

As shown in Figure 1.1, this thesis is organized into seven chapters. To be more specific, Chapter 1 briefly introduces the research background of ever-increasing BIM applications and presents the research motivation behind BIM event log mining for improved project management. Also, the research goal and objectives are clarified, which can be considered to maximize the strength of huge BIM event logs by proper AI techniques. Chapter 2 offers a broad review of the existing researches related to the topic in this thesis from three aspects. Firstly, it summarizes a wide range of BIM adoption in different phases of construction project management for different purposes. Secondly, previous works in BIM event log mining are presented. Thirdly, relevant AI techniques and their applications are reviewed, which will be employed in this research to achieve the research objectives. Chapters 3, 4, 5, and 6 individually realize the four research objectives listed in Section 1.3. The proposed novel approaches are tested in real cases to verify their practicability in optimizing the design and construction process. The structure of these four chapters contains the introduction, methodology, case study, and conclusion. Chapter 7 summarizes the thesis and highlights its contributions from both theoretical and practical perspectives. The limitations are also discussed, which can be addressed in future

works. Also, key directions of future researches are identified to further narrow the gap between AI and CEM for the more advanced project management.

Figure 1.1. Structure of the thesis.

CHAPTER 2. LITERATURE REVIEW

2.1 Introduction

The construction engineering and management inside the scope of the AECO industry is fraught with its own problems and complications, which covers a set of construction-related activities and processes along with human factors and interactions (Jin, Zou et al. 2019). Since construction activities contribute a lot to our society economically, it makes the most sense to take proper construction management for improving the project performance. If the project productivity is enhanced by as much as 50% to 60% or higher, it is estimated to bring an additional \$1.6 trillion into the industry's value each year and further boost the global GDP. It is worth noting that the use of AI is the backbone to launch real digital strategies in project management, which fundamentally changes the way a construction project performs.

This chapter starts with reviewing the board applications of BIM in three main stages of construction project management, which can manifest the necessity of AI implementation in BIM to accelerate the digital transformation in the field of civil engineering. It is followed by a review of previous literature on BIM event log mining to reveal their limitations. Lastly, relevant studies about topics on human behavior prediction, work performance assessment, social network analysis, process mining, and digital twin are reviewed to guide the four identified research objectives.

2.2 BIM adoption in construction project management

It should be noted that BIM with technological, agential, and managerial components can be defined as an integrative technology with parametric intelligence to digitalize the building representation process, which has currently played the leading role in revolutionizing the construction industry (Oraee, Hosseini et al. 2017). As a trend, BIM is going far more than the 3D modeling, which can provide a pool of information to

support project management and exert substantial impacts on aspects of economic, social, and environment across its full lifecycle. According to a survey (Eadie, Browne et al. 2013), BIM brings project delivering benefits in the phase of planning and design, construction, and O&M accounting for approximately 55%, 35%, and 10% of BIM adoption. It is clear that BIM is more pervasively applied in design and construction, since the great advantages of BIM are to provide data-rich 3D visualizations and consolidate information for fast information retrieval. Meanwhile, BIM facilitates closer stakeholder collaboration in these two stages to enhance the performance of the project organization (Arayici, Coates et al. 2011). A brief introduction of BIM applications in the three major phases is presented below.

(1) Planning and design: Before the start of physical construction, it is of necessity to create detailed plans for the project development concerning resources, schedule, budget, dependencies, and others. BIM can be introduced as a design tool to more efficiently formulate well-prepared plans and design schemas fitted to the desired client demand, time scale, and workflow, which is expected to reduce errors, cost, duration, and irrational processes in the practical project. For example, BIM relying on commercial software (i.e., Revit, Synchro, etc.) plays a vital role in transforming the simple drawings to be digital models under the functionalities of visualization, navigation, and parametric modeling (Gu and London 2010). It helps to visualize the schematic design in the detailed 3D model/animation with semantic information, which can eventually offer a comprehensive overview of the project for easier understanding and modification. There have been a few attempts to leverage BIM to automate the design and drafting process at different levels of detail (LoD) (Liu, Singh et al. 2018). Since LoD typically refers to the complexity of a 3D model representation, the growth of LoD from LoD 100 to LoD 500 means that more building information in terms of orientation, location, shape, size, quantity, and some nongraphic information will be enriched in BIM (Ramaji and Memari 2016). An issue in the basic 3D modeling is that it is a little far away from the actual project due to the lack of accurate project plans and estimates. Many efforts have been made to turn the concept of 3D BIM into 4D/5D BIM by incorporating the additional dimension of schedule and cost, enabling the better-planned and more cost-effective

construction (Chen and Tang 2019). Another focus should be on the BIM-based collaborative design for improved project delivery and efficiency, which facilitates the co-design practice through exchanging design information in the standard data format among a group of participants. To this end, Oh et al. (Oh, Lee et al. 2015) developed an integrated design system composed of the BIM module, BIM checker, and BIM Server, which could provide support for collaborative design to significantly improve design quality and productivity.

(2) Construction: This is a phase of executing physical construction. BIM builds a solid link between the design and construction, and thus the plan made at the previous phase is expected to pay off. It is worth noting that BIM creates a collaborative working environment for supporting complicated interactions among participants in various disciplines, such as designers, civil engineers, general contractors, project managers, and others. Based on the effective information dissemination and sharing, BIM spans multi-organizational boundaries in project networks, which helps to inform inter-dependent discipline decisions for reducing unnecessary reworks, conflicts, and errors on site (Liu, Van Nederveen et al. 2017). At present, the BIM-based approaches are experiencing fast growth in the site safety management to proactively address the potential issues and prevent casualties, which have proved to overcome limitations of the manual safety checking, such as inaccurate, discontinuity, inefficiency, and labor intensive. For example, Park et al. (Park and Kim 2013) developed a novel safety management and visualization system under the combination of BIM, location tracking, augmented reality (AR), and game technologies, in order to improve the identification of field safety risks and enhance the real-time communication in managers and workers. Zhang et al. (Zhang, Sulankivi et al. 2015) proposed an automated rule-checking framework based on BIM especially for detecting and visualizing potential fall-related hazards dynamically using the construction schedule, which helped to plan corrective actions for fall prevention ahead of time. Alizadehsalehi et al. (Alizadehsalehi, Yitmen et al. 2018) combined the 4D BIM-based model with on-site data collected from unmanned aerial vehicles (UAVs), and then quantitative analysis was performed in this integrated BIM/UAV model to recognize hazards and produce suitable strategies for safety enhancement. Another thing to notice is

that the 4D BIM simulations of construction schedules and activities are applicable to well handle construction logistics. The adoption of BIM supports the better understanding of logistics information, detection of conflicts, supervision of construction progresses and supply chains, and coordination of different activities, which improves the site safety to run the construction smoothly (Whitlock, Abanda et al. 2018, Bortolini, Formoso et al. 2019).

(3) Operation and Maintenance (O&M): When construction is completed, the project will enter a new phase called O&M to operate and maintain a constructed facility to not only meet the anticipated functions over its lifecycle but also ensure the safety and comfort of users. It is known that O&M takes the most of the time within the lifecycle, leading to a large amount of cost accounting for around 60% of the total project budget (Zhang and Ashuri 2018), but BIM applications for effectively operating and maintaining facilities are still insufficient. To support the relatively new usage of BIM in decision making for facility managers, some studies have integrated the standardized information inheriting from the design and construction phase along with additional information pertaining to the O&M phase into the as-built model (Hu, Tian et al. 2018). For example, Marzouk and Abdelaty (Marzouk and Abdelaty 2014) integrated data collected by wireless sensor networks into the BIM platform, and thus the designed BIM-based system was able to visualize and monitor the thermal comfort at different spaces within the subway for operation enhancement. Kang and Hong (Kang and Hong 2015) proposed an efficient architecture for information extraction, transforming, and loading, whose usefulness had been verified in facility management use cases to automatically integrate from BIM, geographic information system (GIS), and the facility itself for further analysis. Yin et al. (Yin, Liu et al. 2020) developed a generic BIM-based framework encompassing the BIM model, relational database, and monitoring system, and thus data from these three components could be exchanged easily through API to assist with the sustainable O&M of utility tunnels. In short, BIM implementation also provides the opportunities to visualize various aspects of the facility and comprehensively analyze data about the facility's performance, and thus a wide range of O&M activities, like maintenance and repair, emergency management, energy management, and others, can potentially embrace

the benefits of BIM (Gao and Pishdad-Bozorgi 2019). As a result, the day-to-day services can be controlled in an efficient, economical, and reliable manner. Time-based preventive maintenance detects the potential risks and adjusts the ongoing operation prior to unexpected events. Corrective maintenance implemented after the occurrence of problems strives to repair the problematic parts and get them back on the normal status as quickly as possible.

To further facilitate the information digitalization in intelligent construction project management, BIM can be reasonably considered as a digital backbone to work with AI. For BIM, it drives the construction industry into a data-intensive field. It provides a platform for not only collecting large volumes of data about all aspects of the project, but also sharing, exchanging, and analyzing data in real-time to achieve in-time communication and collaboration among various participants. For the AI techniques, they automate and accelerate the process of learning, reasoning, and perceiving the rapid growth of heterogeneous data from BIM through training suitable models to automate and improve the construction process. In the immediate future, the integration of BIM and AI can move the paper-based work towards online management, which assists the traditional construction industry to catch up with the fast pace of automation and digitalization. As expected, it can deliver the most efficient and effective information to keep continuous updating of the ongoing project. The solutions for construction projects are different from one another. Based on the in-depth analysis in a range of ways (i.e., simulation, prediction, and optimization), strategic decisions that are suitable for a certain project will be informed without human intervention under complicated and uncertain environments, which is expected to generate immediate reactions to streamline the complicated workflow, shorten operation time, cut costs, reduce risk, optimize staff arrangement, and others. Meanwhile, this kind of tactical decision making can possibly be adapted to the changeable conditions to optimize the project operation continuously for delivering smarter construction management throughout the full project lifecycle. Hence, it can be reasonably considered that the practical value of the hybrid framework based on BIM and AI lies in addressing challenges arising from characteristics of construction project

management, including uniqueness, labor-intensive, dynamics, complexity, and uncertainty. This topic of BIM and AI integration deserves more attention.

2.3 BIM event log mining

2.3.1 Research status

At present, the revolutionary technology BIM is increasingly applied in both the design and construction phases for project management. To be specific, BIM can passively and continually gather mass data concerning all aspects of a construction project, including graphical models, resources, costs, safety issues, time, and others, which paves the way to overcome the limitations of human interference (Boje, Guerriero et al. 2020). It should be noted that an important BIM data type is the event log in the plain text format (Pan, Zhang et al. 2020). Commonly, event logs contain a set of cases and each case comprises a sequence of events/activities along with the timestamp (Rojas, Munoz-Gama et al. 2016). Thus, the BIM event logs can be defined as a rich source of process-specific information to capture flows of activities in chronological order, which contain several attributes, including cases, activities, persons, and time (Yarmohammadi, Pourabolghasem et al. 2017). Take the BIM design event log as an example. A detailed collection of modeling activities, designer-software interaction, and system information is saved into the growing volumes of design event logs, which can provide affluent evidence for BIM-based design analysis. Figure 2.1 provides an example of data items in BIM design event logs, which are stored in the Program Files directory under the *Autodesk Revit* product folder named journal files (Revit 2017). The selected words in blue color are important information that needs to be extracted and saved in the CSV files.

Remarkably, there is hidden knowledge about productivity, bottlenecks, process deviations, social networks of actors behind such large amounts of event log data. It means that the full potential of BIM event log can be harnessed from the data layer. Some researchers have paid attention to mining design-related event logs towards better management of the design phase. These previous works mainly rely on the techniques of Knowledge Discovery in Databases (KDD) and basic pattern recognition to understand

the complex model development process. For instance, Mirakhorli et al. (2015) explored big data to summarize a large set of architectural design concepts, including design patterns, design tactics, architecture styles, etc. Two studies from Yarmohammadi et al. (2017) and Zhang et al. (2017) adopted pattern retrieval algorithms (i.e., Generalized Suffix Trees, PATRICIA) to simply extract the most frequent patterns of design sequential commands, and thus the performance of different designers could be measured and evaluated by comparing the time they took to conduct the same 3D modeling patterns. Zhang and Ashuri (2018) built a social network based on huge design logs to describe the collaboration among designers and then analyzed the network structure by some fundamental metrics, in order to better understand the level of collaboration, the characteristics of information exchange and sharing, and the relationship in sociological network structures and modeling performance. Petrova et al. (2019) conceptually presented a basic framework of a data-driven sustainable design system relying on operational building data and BIM data repositories, allowing for knowledge discovery in a semantic integration layer. All the promising analysis and results from these existing studies mentioned above show that the exploration of BIM event log in a data-driven and systematic manner offers unprecedented opportunities to understand the BIM-enabled projects and inform suitable decisions toward a more efficient and sustainable modeling process.

Command Name	Example
User Name	Jrn.Directive "Username" _ , "Modeler1"
Date and Time	'E 29-Mar-2014 10:26:05.808; 0:< Jrn.Command "Internal" , "Show or hide recent files , ID_STARTUP_PAGE"
Project Name	Jrn.Data "File Name" _ , "IDOK", "P:\Atlanta\gatech\building.rvt"
General Command Description	Jrn.Command "AccelKey" , "Cut the selection and put it on the Clipboard" ' 0:< <<Begin build CT>>
Specific Command Description	:13.619; 0:< y" , "Cut the selection and put it on the Clipboard , ID_EDIT_CUT"
View	ective "WindowSize" _ , "[2013_03_21_NGHA_TAIF_blochaa.rvt]", "Floor Plan: LEVEL 01 PLANNING"
Software Version	Jrn.Directive "Version" _ , "2014.000", "2.122"

Figure 2.1. Examples of data items in BIM design event logs (Yarmohammadi, Pourabolghasem et al. 2017).

2.3.2 Research gap

Although these existing researches offer unique insights into the model evolution process, four obvious limitations remain to be addressed: (1) These studies directly extract frequently-used command patterns for specific modeling tasks and measure designers' performance by the basic statistical methods, which lack the learning ability and cannot independently adjust to new data. (2) No novel machine learning-based algorithm has been developed to be more flexible and suitable for mining BIM event logs in large volumes and great complexity. (3) It is evident that only event logs associated with the design phase have been taken into account, but the investigation of construction log data is still in the initial stage. Nonetheless, the current penetration of BIM has been expanded to large-size construction projects. Since more than 60% of BIM users from Germany rate very high value of BIM in improving the planning and tracking of schedule, labor, cost, and materials on the construction field (Analytics 2014), it also worth facilitating more intelligent use of such event logs heavily accumulated in the construction phase. (4) Since BIM has a natural interface for IoT implementation, a new way to make the utmost of BIM event logs is to merge them with IoT and various data mining techniques. To be specific, BIM acts as a high-fidelity data repository and IoT provides time-series data about the actual operations, which can provide a significant opportunity for establishing the digital twin. The topics of BIM-IoT integration and digital twins are relatively new in the construction industry, which have not reached their full potential yet.

The primary factor contributing to the difficulty in exploring BIM logs is the nature of ever-increasing and text-format event logs generated in the process of BIM-based project management under characteristics of uniqueness, labor-intensive, dynamics, complexity, and uncertainty. That is to say, BIM will collect growing amounts of disordered, non-intuitive, and heterogeneous log data from different stakeholders and domains, which will impose heavy burdens on data manipulation. What's more, a lot of uncertainty, subjectivity, and ambiguity will be inherent in data related to the design phase, which will negatively confuse the data analysis and even return unconvincing results. It has been found these massive log data with high-dimensionality and incompleteness information challenges the traditional statistical theory significantly in terms of

meaningful feature selections and computational cost (Fan and Li 2006). Therefore, it is necessary to narrow the gap of data science in exploring BIM logs for reliable knowledge discovery and tactical decision making.

The past decades have witnessed the growing interest of AI techniques to bring about unprecedented changes in several data-intensive domains, such as biology, mechanical engineering, transportation, and others, which can present valuable opportunities for producing strategic solutions and decisions (Qiu, Wu et al. 2016). Various AI techniques have been developed to make machines mimic human cognitive processes in terms of learning, reasoning, and self-correcting. For example, machine learning is a great step of AI to teach machines how to discover patterns hidden in large data and realize data-driven predictions on future tasks. As machine learning evolves, deep learning as a new trend has been developed at a higher level. A young discipline named process mining specializes in handling event logs with the aim of monitoring, diagnosing, analyzing, and improving the actual process. There are reasonable prospects that these attractive AI methods can also be utilized to explore the rapid growth of BIM event logs, aiming to easily transform massive BIM data into useful knowledge towards a high degree of automation and intelligence. However, research in this focus is still rare. Although a considerable amount of BIM event log data increases unprecedently in the construction project, the adoption of AI techniques still lags behind the process in other industries. I intend to perform AI-based BIM event log mining to make more objective predictions and evaluations for processes at both the design and construction stages. Therefore, project managers no longer heavily depend on their subjectivity, knowledge, and experience to evaluate participants' performance and adjust the work plan. Ultimately, the gap of data science talent in the AECO industry is supposed to be filled, which drives the traditional construction industry to catch up with the fast pace of automation and digitalization

2.4 Studies related to research objectives

2.4.1 Human behavior prediction

The first research objective is to predict an individual's design command, which belongs to the topic of human behavior prediction. It is known that human behavior is more highly predictable than expected when sufficient observed data is available (Alahi, Ramanathan et al. 2017). Besides, goal-oriented behaviors can be directed based on the prediction results, which will possibly avoid unnecessary human errors and even contribute a lot to better decision making in complex conditions. However, human behavior prediction is not actually a straightforward task, since dynamical changes constantly occur to adapt to diverse situations (Subrahmanian and Kumar 2017). To resolve this kind of issue, the increasingly popular machine learning techniques are becoming a powerful tool to track, learn and predict offline and online human behaviors, which hold the promise to better understand human behavior and make more accurate predictions at a faster speed than human judgment (Kanter and Veeramachaneni 2015). That is because algorithms can learn the most relevant features and discover the causality behind the behavior data automatically with no need of human interference, which can minimize the negative effects of individual bias in data analysis.

Deep learning models, a promising area of machine learning research, have been successfully applied in human behavior prediction for different purposes, such as to explain the social networks (Phan, Dou et al. 2017), to analyze handwriting (Champa and AnandaKumar 2010), to develop smart home services (Choi, Kim et al. 2013), and others. In particular, the Recurrent Neural Network (RNN) (Jordan 1986, Elman 1990), a variant of the feed-forward neural network in Figure 2.2 (a), is developed to intelligently predict sequential data. To be specific, RNN has the memory in the hidden layer to remember the output, which will act as a new input to enter the RNN at the next step. One of the most typical applications of RNN is the natural language processing (NLP) tasks, aiming to predict the next possible word by learning the sequence of input words (Evermann, Rehse et al. 2017). Also, it has expanded to various sequence learning problems. For instance, Choi et al. (2016) developed an RNN-based Doctor AI to predict diagnosis and medication categories by learning the longitudinal time-stamped data in the electronic health record.

Fan et al. (2017) proposed a spatial-temporal prediction framework based on deep RNN to forecast air pollution. Zhang et al. (2014) employed RNN to model the dependency on the user's sequential behaviors and make sequential click prediction for sponsored search.

Another deep learning model named the Long Short-Term Memory Neural Network (LSTM NN) can also well capture the temporal-spatial evolution of events and cope with high-dimensional and non-linear problems (Zhao, Chen et al. 2017). As shown in Figure 2.2 (b), LSTM NN is a variation of RNN to regard the hidden layer as a memory unit, which is superior to RNN in mitigating gradient vanishing and exploding issues in the condition with long-time tags (Ma, Tao et al. 2015). Due to the unique structure of LSTM NN to encode information from multiple frames and generate a sequential action (Liu, Shao et al. 2019), it has been successfully applied in various domains, including computer vision, robot control, speech recognition, transportation, and others. For example, Inoue et al. (2019) proposed a novel robot path planning method for executing autonomous moving robots by rapidly exploring the random tree and LSTM NN. Ma et al. (2015) captured nonlinear traffic dynamics by LSTM NN in an effective manner and achieved great performance in both accuracy and stability. Alahi et al. (2017) encoded complex interactions that one might not be aware of in the LSTM NN model, in order to forecast human trajectories in crowded environments with high accuracy. Makarencov et al. (2019) adopted a bidirectional LSTM tagger for proper word choice in lexical substitution and grammatical error correction to support scientific writing tasks. Lipton et al. (2015) built an LSTM NN model in clinical medical data to solve the multi-label classification problem for early diagnosis diagnoses. Analogously, since designers can display some regularities in the modeling process, the application of RNN and LSTM NN can be extended to the BIM-based design, aiming to learn from design sequences and classify the next possible design commands. Proper guidance for modeling can be therefore offered based on the command prediction with the expectation of raising design quality and efficiency.

Figure 2.2. Architecture structure of (a) RNN; (b) LSTM NN.

2.4.2 Work performance assessment

Since human behavior at work is a combination of personal habits and capabilities, cognitive status, and activities to achieve goals (Sansone, Morf et al. 2003), it is not an easy task to evaluate an individual's performance reasonably. So far, the most common approach for work performance evaluation still relies on the subjective judgment through various kinds of peer assessment and self-assessment. That is to say, people will jump to conclusions after reviewing the archival records, self-reports, rating scales, and work results (Campbell, McHenry et al. 1990). Obviously, this kind of performance evaluation has two considerable drawbacks. For one thing, the assessment process is manual, burdensome, and time-consuming, which is prone to generate subjective and unreliable results with individual bias (Mirjafari, Masaba et al. 2019). For another, the traditional method cannot track the changes of human behavior in real-time and adjust the evaluation results accordingly (Swain, Saha et al. 2019). It means that the traditional assessment is inflexible in adapting the complex and varying situations.

To measure the workers' activities more convincingly, collected data in large volumes can be analyzed deeply to perform the objective assessment, which can be termed as a topic of data mining. The mobile sensing data is taken as an example. It is details of workers' physiological, behavioral, and mobility information continually recorded in the mobile with no human intervention, which is explored to track and model human behavior (Saeb, Zhang et al. 2015, Harari, Wang et al. 2017). That is to say, there is hidden information about the individuals' behavior in the source of sensing data, which has demonstrated the potential in learning a person's work performance. For instance, Matic et al. (2014) extracted features from the mobile sensing data to classify the formal and informal social interactions at the on-going work with around 80% accuracy, which can improve the communications between workers. Wang et al. (2018) reported a mobile data sensing approach to capture and assess the within-person behavior variability patterns, which could be then adopted to offer a great prediction of personality traits. Swain et al. (2019) explored the huge mobile sensing data collected 108 days to explain the worker's performance and understand his organizational personas in the daily activities by classical clustering methods, such as *k*-means and hierarchical methods. Mirjafari et al. (2019) proved that mining the mobile sensing data by the *k*-means clustering provided new insights into patterns to differentiate workers with high and low productivity, which could offer regular feedback and guidance in the workplace.

Remarkably, the BIM event logs are similar to the mobile sensing data, which document the full range of cases and activities passively. During the project progress, participants will definitely display different work habits. Inspired by the developed mobile sensing data analysis as shown in Figure 2.3, it is also meaningful to conduct proper data mining techniques to better understand the unique behavior and productivity of participants. As reviewed, clustering methods have played an important role in grouping similar characteristics of workers or their behavior derived from the mobile data together and generating worker profiles. Similarly, I intend to recognize different design behavioral patterns by learning features from temporal design logs under proper clustering approaches. When the working habits of a particular designer based on a series of features,

like operation time and command information are captured, the results provide references for managers to make rational work arrangements to accelerate the modeling process.

Figure 2.3. Procedure of worker performance evaluation based on mobile sensing data.

2.4.3 Social network analysis

In general, a number of participants sharing common interests and goals will be jointly involved in large-scale projects, which have the nature of high complexity and uncertainty in project size, technology, and personal capability (Šmite, Moe et al. 2017). To better visualize and understand the complicated cooperation relation, a social network can be built to graphically model the interaction structures and characteristics, where vertices standing for people are connected by directed links to clearly represent their relationships. Subsequently, social network analysis (SNA) can be conducted within the established network to study the complex system by examining social roles, information spreading, and behavior interactions in the collaborative team. As a qualitative and quantitative analytical tool, SNA can evaluate participants' performance in a more objective and reliable manner to replace the commonly-used subjective methods (i.e., self-evaluation, peer rating), which could be troublesome and prejudiced.

Due to the strong capability of knowledge discovery in complex networks, the topic of SNA is popular in a wide range of domains since the late 1970s, such as recommendation systems (Palau, Montaner et al. 2004, Sun, Han et al. 2015), bioscience (Sharan, Ulitsky et al. 2007, Kovács, Luck et al. 2019), sociology (Fu, Chai et al. 2012,

Dhand, White et al. 2018), business (Bonchi, Castillo et al. 2011, Neumeyer and Santos 2018), sports (Fransen, Van Puyenbroeck et al. 2015, Wäsche, Dickson et al. 2017), electronics (Basole, Bellamy et al. 2016, Wang, Sun et al. 2019) and others. SNA can offer a more objective and reliable assessment of work performance to take the place of the commonly-used subjective methods (i.e., self-evaluation, peer rating), which could be troublesome and prejudiced. It should be noted that the latest application of SNA is to provide scientific evidence to guide governments and organizations in fighting the global pandemic. For example, SNA can be performed to examine Twitter data related to COVID-19, which helps to capture the emotional changes of citizens (Hung, Lauren et al. 2020) and comprehend characteristics of public key players in offering relevant information (Yum 2020). Also, SNA can intuitively visualize the contact people and transmission of COVID-19 across a country or the world, which paves a simple yet powerful way to evaluate the pandemic risk and formulate appropriate strategies of social distancing/isolation (Block, Hoffman et al. 2020, So, Tiwari et al. 2020). Due to the great practical value of SNA, it is expected to extend its application to the civil engineering field.

Since the great benefit of BIM is towards collaborative project delivery, the BIM-based project can be understood as the comprehensive results of modeling operations, communications, information sharing, and decision making within a group of participants working on a common goal. Through exploring the interdependencies of actors in different roles by SNA, firms can potentially develop better relationship cultivation tactics for more competitive and rationalized construction project management (Lin 2014, Cao, Li et al. 2018). Therefore, it is reasonable to connect the SNA and its dynamic level with the BIM-based collaboration for construction project enhancement. Figure 2.4 gives an example of a social network in describing a BIM-based collaborative design process and revealing hidden insights into both technical and social aspects. Some efforts have also been made on such a topic. For instance, a green building design project, which emphasized the roles of designers on the green feature choosing, established social networks to discover communication patterns among designers for optimizing the design process (El-Diraby, Krijnen et al. 2017). Inter-organization communications in a Greek

construction project were described and examined from a network perspective with the goal of enhancing the team's cohesion (Badi and Diamantidou 2017). What's more, design quality is possible to be improved when SNA detects errors and patterns of error diffusion through tracking and analyzing the structure of communication (Al Hattab and Hamzeh 2015). In other words, SNA is beneficial in monitoring and assessing the BIM-based design objectively, which encourages evidence-based decision making for the pursuit of high-efficient and high-quality design procedures. Meanwhile, SNA opens a new way to process data from design activities filling with subjectivity and uncertainty, which is relatively unexplored and hard to analyze (Bilal, Oyedele et al. 2016). However, the value of SNA in BIM event logs has not been emphasized. It is, therefore, a novel and significant idea to maximize the potential of SNA in mining BIM event logs for complex project management. By building networks and examining the social environment during the project execution, results from SNA based upon BIM event logs can provide strong evidence to objectively formulate proper collaborative strategies, which is expected to facilitate task delivery, knowledge sharing, information interoperability, and technical cooperation.

Figure 2.4. Description of BIM-based collaborative design by a social network.

2.4.4 Process mining

Process mining is relatively a young research discipline belonging to a sub-area of AI techniques. Since process mining is devoted to exploring event logs, it can be regarded

as a connection between event logs and the operational process. It can be seen in Figure 2.5, process mining is a mixture of data mining and process analysis to take control of event log data, which can output a meaningful picture of the entire process for further analysis. That is to say, process mining can well handle the overwhelming event logs to maximizes the potential value of available data from two aspects, namely process discovery and process analytics. For one thing, the true process with a high degree of complexity can be abstracted and visualized in a more comprehensive model by proper algorithms (La Rosa, Wohed et al. 2011). Based on the established process model, it is straightforward to observe process steps that are influential, repeated, overcomplicated, and fallible from the graph directly. For another, a wide range of analytical methods can be implemented on the refined process model to detect possible and capture characteristics of the organization in the process. The revealed insights are especially beneficial in understanding the core process and detecting performance issues (i.e., deviations and bottlenecks), which can present evident-based recommendations in strengthening operations, enhancing efficiency, and resolving the process bottlenecks to reduce the risk of failures beforehand (Rebuge and Ferreira 2012). Consequently, process mining assists managers to quickly point at the key parts of the process and inform data-driven decisions for strengthening operations and accelerating the process.

Some software products for process mining are available to efficiently convert event logs into process-related views and deliver insightful analytics, such as the ProM framework, Disco (Fluxicon), Celonis, ARIS Process Mining, Myivenio, and others. The first task of the software is to create a visual map to clearly describe the step-by-step process, which is followed by more advanced analysis in the model to realize functions of diagnosis, checking, exploration, prediction, recommendation, and others. With the help of software, process mining is not merely a theoretical subject (dos Santos Garcia, Meinheim et al. 2019). It has been put into industrial practice, such as the business (Jans, Van Der Werf et al. 2011, Li, Cao et al. 2013, Dymora, Koryl et al. 2019), healthcare (Rojas, Munoz-Gama et al. 2016, Pika, Wynn et al. 2019), education (Premchaiswadi and Porouhan 2015, Bogarín, Cerezo et al. 2018), and information and communication technology (Gupta, Sureka et al. 2014, Valle, Santos et al. 2017), and others, allowing for

uncovering unwanted behavior, shortening the waiting and service time, and promoting collaboration. According to a recent survey, the top benefits of process mining techniques are associated with objectivity, accuracy, speed, and transparency (Ailenei, Rozinat et al. 2011). It is worth noting that the starting point of process mining is the event log, a special data type containing process-specific information, including cases, activities, persons, and time, to capture flows of activities in the chronological order. Since the growing use of BIM applications can also generate great volumes of computer-generated event logs, it is reasonable to expand process mining to CEM for knowledge discovery and decision making.

Some existing researches have carried out process mining in BIM-enabled projects to effectively examine workflow and collaboration. For instance, Chua and Hossain (Chua and Hossain 2011) simulated the design process to inspect the influence of early information on the redesign and total design duration, but it ignored the inherent role of individual and team behavior in information sharing. Al Hattab and Hamzeh (Al Hattab and Hamzeh 2018) established the agent-based modeling to dynamically integrate design information with social networks and improve design workflow for higher quality and efficiency, which mainly focused on characteristics of persons' behavior and interaction rather than the task itself. Kouhestani and Nik-Bakht (Kouhestani and Nik-Bakht 2020) built process models from both the actor and phase views about the design-authoring phase and made comprehensive analysis for process and collaboration, which ultimately guided BIM managers to monitor, control, and re-engineering the design work. It is well known that BIM can come into play in the phase more than design. However, the scope of all these previous studies is limited to the design process, which means that the construction still remains unexplored. Besides, there is no analysis to associate participants' roles from social networks with their relevant bottleneck from the process. Therefore, more efforts for process mining considering various aspects need to be made by deeply investigating construction-related event logs, which assist in realizing cost-effective troubleshooting to prevent undesirable conflicts, delays, poor collaboration in the complex workflow. By offering a comprehensive view of the complicated process

along with the end-to-end performance analysis, process mining is changing the current way of construction management.

Figure 2.5. Typical tasks in process mining.

2.4.5 Digital twin

The term “digital twin” initially proposed in 2003 is not a new concept, but it gains increasing popularity in the current industrial revolution 4.0 (digitalization). More specifically, the re-emergence of interest in digital twins is largely inspired by the study from the National Aeronautics and Space Administration (NASA) to continuously simulate, forecast, and evaluate the spacecraft state, aiming to mitigate the degradation and failure in the vehicle (Glaessgen and Stargel 2012). Afterward, digital twins have been increasingly recognized by more and more researchers, and Gartner research firm in 2018 even predicted the idea as one of the top ten most promising technology trends over the next ten years (Tao and Zhang 2017). In my opinion, the digital twin can be simply described by Figure 2.6 under the integration of physical products, virtual products, and relevant connection data, which typically refers to a mirror and digital depiction of the actual production process. That is to say, the digital twin can be understood as a cyber-physical system with the help of IoT devices and various AI methods, where a digital replica of a physical counterpart that is enriched with large volumes of data can dynamically imitate, model, and analyze real-world behavior for multiple purposes of simulating, diagnosing, predicting, and optimizing.

To date, digital twins play a crucial role in pursuing the deep cyber-physical integration of intelligent manufacturing towards a greater level of flexibility, adaptability,

and predictability in production management. The digital twin system has been widely applied in product design and production, which can assist in understanding customer demands quickly, identifying or even predicting weaknesses in models early, controlling production processes to respond to the changing environment timely, and making valuable suggestions to optimize plant operation and maintenance before failure occurrence (Schleich, Anwer et al. 2017, Vachálek, Bartalský et al. 2017, Min, Lu et al. 2019, Tao, Sui et al. 2019). Moreover, some leading companies, such as General Electric (GE), Siemens, British Petroleum (BP), and Airbus, have implemented digital twins in the practical production and relevant patents for production technical innovation (Yang, Li et al. 2018). Due to the success of digital twin in manufacturing, some efforts have been devoted to building the cyber-physical model for supporting digital development in the construction industry. It has been proved that a system architecture of digital twin potentially has a wide application prospect in representing, predicting, and managing the current and future conditions of the infrastructure itself, built environment, or city assets. For instance, Yuan et al. (Yuan, Anumba et al. 2016) monitored the temporary structure by the bi-directional coordination between physical and virtual systems, where the virtual components were built by the real-time data from sensors in the physical part to make early warning and immediate instruction for structural failure prevention. Srewil and Scherer (Srewil and Scherer 2013) utilized data from Radio-frequency identification (RFID) to map the actual process into the virtual model, which could provide a comprehensive solution for real-time construction process monitoring. Linares et al. (Linares, Anumba et al. 2019) adopted the advanced equipment of an Augmented/Virtual Reality (AR/VR) coupled with sensors to capture images or videos on the physical site, which was helpful in safety monitoring, risk warning, and remote instruction. Lu et al. (Lu, Parlikad et al. 2020) designed a digital twin at both the building and city levels following data integration, synchronization, and analysis, in order to realize anomaly detection, ambient environment monitoring, maintenance optimization and prioritization, and energy planning. To sum up, the superiority of digital twin lies in its value-added services in automatic data collection, conceptual development, dynamic analysis, problem diagnosis and optimization for smart design, operation, control, and maintenance. In other

words, real-time data derived from the physical products are the basis to align the real world into the virtual parts. Through automatically detecting issues and evaluating performance ahead of time, optimized solutions can be formulated in a data-driven manner and put into operation in time to bring benefits of improved reliability and efficiency. Thus, there are reasons to believe that the concept of digital twins will become increasingly important in the rise and progression of the construction industry revolution.

From these above-mentioned pieces of literature, it can be found that the effectiveness of the virtual part largely depends on the great volumes of collected data and the corresponding data analysis. Commonly, IoT supports more efficient data acquisition to collect time-series data about the actual and continuous operations, and then this information can be shared across the internet enabling real-time data analysis (Tang, Shelden et al. 2019). The 3D point clouds from the IoT device is used as an example. For monitoring the complex construction process in real-time, unmanned aerial vehicles (UAV) can fly over the construction site to take point clouds continually for capturing the actual (as-built) environment. In other words, as-built data about time, space, progress, and others are available in point clouds. Since BIM has evolved into an open platform for information sharing and management, it is able to synchronize with multiple data sources from IoT. That is to say, the integration of BIM and IoT can store and update a variety of information, including object properties, site and facility conditions, physical measurements, time series data about the progress, and others, which offers rich data sources for DM-supported knowledge learning and decision making. Hence, it can be considered to establish a well-defined framework of a digital twin based upon BIM, IoT, and DM, which can be presented as a “physical-data-virtual” paradigm for higher interoperability, automation, and intelligence in delivering smarter construction services (Boje, Guerriero et al. 2020). In existing research, the developed digital twins mainly provide a crucial and analytical edge to BIM-IoT integration. For instance, Lu and Brilakis (Lu and Brilakis 2019) automated the geometric modeling in the digital twin part for existing reinforced concrete bridges from 3D cloud points, which could reach a relatively high spatial accuracy. Stojanovic et al. (Stojanovic, Trapp et al. 2018) reconstructed and visualized the captured state of the built environment using the basic data from 3D point

clouds and related IFC, which could be helpful in enhancing collaboration, decision making, and forecasting among facility management stakeholders. Shim et al. (Shim, Dang et al. 2019) adopted the 3D scanning technology to duplicate an existing bridge structure as the object-based digital twin model, from which data about damage and repair history could be analyzed to orient long-term strategies for bridge assessment and maintenance. However, they mostly emphasize on the 3D geometry and model evaluation in digital twins, while less attention has been paid to knowledge discovery from the DM layer.

It should be noted that BIM-IoT integration can provide a constantly updated and rich data influx about both the functional and performance features of a facility (Ding, Zhou et al. 2014). In particular, BIM is known as an information system demonstrating the powerful ability to efficiently synchronize and store mass data that continuously collected from IoT devices. However, it is important to note that BIM itself lacks data manipulation capabilities to evaluate and predict the real-time status of assets, processes, systems, or even services, which is unable to provide smart services, like automated monitoring, real-time safety detection, accurate prediction, adapted optimization, and others. This is the biggest difference from the digital twin. In this regard, BIM can only be regarded as a start point of digital twins. An open question is that how to integrate BIM-IoT with advanced data analysis methods for creating a closed-loop paradigm as a complete set of digital twins, aiming to continuously update and learn data in an intelligent and efficient manner for real-time decision making. To address issues in information integration and data analysis, Cheng et al. (Cheng, Chen et al. 2020) connected various kinds of information from the as-built BIM models and IoT sensor networks, which were used to train machine learning algorithms (SVM and ANN) to make predictive maintenance planning for building facilities. Ma et al. (Ma, Ren et al. 2020) adopted BIM and GIS in an integrated manner to provide related geometric, attributive, and spatial data, and then Reliability Centered Maintenance (RCM) algorithms were performed on these prepared data for decision-making on equipment maintenance of business parks. In other words, DM techniques can offer a wealth of digital insights into the collected data for making more informed and proactive decisions in condition assessment, prediction, and

improvement, which no longer rely on the subjective judgment with bias and uncertainty. Since a digital twin under BIM-IoT will contain a lot of data with hidden knowledge, appropriate DM methods need to be performed to realize the full value of data for two major purposes. For one thing, DM can promote the bidirectional interaction in the physical and cyber space. For another, DM helps to continuously guide and adjust the construction process towards the project goals using actual data rather than observation or intuition. Despite the importance of DM approaches, the integration of BIM, IoT, and DM for digital twin is still at infancy. For this concern, we intend to develop a data-driven framework of a digital twin, which can be strategically leveraged and integrated with the BIM, IoT, and DM to yield significant value in intelligently improving construction efficiency, collaboration, and reliability.

Figure 2.6. Architecture of digital twin.

2.5 Chapter Summary

This chapter presents an overview of the previous studies on BIM-based construction project management, BIM event log mining, and relevant studies about the proposed research objectives. It has been found that BIM is gaining more and more attention for speeding up the pace of digitalization and revolution in the construction industry. BIM can be interpreted as a digital representation of the physical and functional characteristics of infrastructures and a novel process of creating and managing information during the lifespan of the construction project, which can bring a mass of accumulated BIM data with some apparent features of “big data”. In particular, BIM event log data is an important BIM data type to capture the entire project evolution chronologically with a lot of hidden

knowledge. However, there exists a clear gap between BIM event log data and data science for adding value in data-driven decision making. Since the BIM event log is similar to the web log that has been widely used in web usage mining, it is reasonable to implement proper AI methods to make the utmost of such rich data. As the literature review, various AI techniques have been successfully equipped machines with human-like intelligent behavior and reasoning for different purposes, such as human behavior prediction, work performance assessment, social network analysis, process mining, and digital twin implementation, which can therefore be deployed to handle the ever-increasing and text-format BIM event logs. The purpose of this research is to link AI to the large amount of BIM event log data, which is expected to provide innovative solutions for delivering better design and construction processes.

CHAPTER 3. LEARNING AND PREDICTING DESIGN COMMANDS BY DEEP LEARNING METHODS

3.1 Introduction

This chapter addresses the Research Objective 1 of this thesis. The specific objective is to develop a deep learning-enabled framework to learn a series of designers' subjective commands recorded in BIM event logs and make accurate predictions on the possible design command in the next step. Its ultimate goal is to achieve a reliable data-driven design process, which has the potential to improve modeling efficiency and quality. In this regard, there are three main steps in the proposed approach, including data preparation, deep learning-based model establishment, and classification evaluation. To be more specific, various design commands are categorized into several classes according to their effects and given numerical labels as the preparation of the multi-class classification problem, and thus computers can understand this information directly. Due to the powerful ability to model temporal dependencies, deep learning algorithms, including RNN and LSTM, are then employed to capture the temporal dynamics in the design process. They can learn sequential data with varying lengths from logs to intelligently generate design commands with probability. Finally, the predicted command class verified by the evaluation metrics is expected to serve as the operation reference to guide the modeling process in a data-driven manner under the assumption that the correct class tends to appear owning the top three highest probabilities, enabling an easier and more efficient modeling process. In other words, the proposed deep learning-based framework is helpful in improving the modeling process in both efficiency and quality, which is possible to realize the personalized command recommendations for designers to speed up modeling and avoid unnecessary operation mistakes.

The research questions of this chapter can be summarized as: (1) How to clean the extracted data from BIM design event logs and label data properly to make it more

interpretable, which can prepare high-quality inputs for the deep learning model; (2) How to train the RNN or LSTM NN with optimal parameters for learning the preprocessed data from BIM event logs in a multi-class classification task, which is intended to intelligently predict the potential types of design command by giving exact probability for each command class; and (3) How to explore the influence of network parameters on the predictive accuracy and demonstrate the superiority of the developed deep learning model over some other popular machine learning algorithms in learning and predicting designers' behaviors. In consequence, the design command can be predicted at the category level continually through three steps of data acquisition and preprocessing, data mining, and performance evaluation. By providing the three most possible incoming command classes, designers no longer spend too much time in thinking about the next possible command class. They can easily search for the proper design command in a certain class. It is also worth noting that the deep learning model can capture a designer's modeling preference to realize personalized command prediction. That is to say, the proposed approach in this chapter makes full use of the time-stamped model evolution information embedded in the huge BIM event logs, contributing to the automation, intelligence, and reliability of design processes.

The remaining of this chapter is structured as follows: Section 3.2 introduces the overall framework of the developed RNN/LSTM NN-based intelligent command prediction approach along with detailed steps and methods. Section 3.3 performs RNN in a simple case study with totally 57,915 command records associated with the "Create" function. Acting as a multi-classification task, hundreds of design commands are categorized into six classes and labeled by numbers 1-6 and the RNN with 1 hidden layer and 64 hidden neurons will be trained. Section 3.4 utilizes a more complete dataset of BIM design event log in 4GB and performs a more complex neural network termed LSTM NN for a real case study. After data retrieval from logs, totally 352,056 lines of design commands over 289 projects are remained, which are then categorized into 14 classes for LSTM NN training and testing. Section 3.5 summarizes the conclusions of this chapter.

3.2 Methodology

The motivation of this chapter is to develop a deep learning-based prediction model to explore the sequential design commands based on BIM event logs. Figure 3.1 illustrates the conceptual workflow for the proposed method, which is composed of three main steps: data acquisition and preprocessing, data mining, and performance evaluation. As a whole, the design command prediction mechanism is performed by learning design behaviors from BIM event logs, which provides designers with modeling instruction to facilitate a smooth, high-efficiency, and intelligent design progress

Figure 3.1. Workflow of the proposed command prediction method. (Note: DL is the abbreviations of deep learning)

3.2.1 Data acquisition and preprocessing

As the rich sources for data acquisition, the design logs contain a massive influx of data about multiple designing projects and designers, which are created automatically during the course of building design by *Autodesk Revit* software. The design log data is stored in the Journals under the Program Files directory in the *Revit* Product version folder (Revit 2017). Each *Revit* journal file records a block of operating information associated with design activities, like the user, project, time, command, file path, and others. Since a group of designers works over several projects in the design firm, considerably vast

amounts of *Revit* journal files will be generated to keep detailed records about modeling events, serving as a sufficient premise for further data analysis.

A particular concern is that the original design log data saved in *Revit* journal files are in the text format, which will pose challenges in data mining. In order to make the original data understood by computers easily, the required information is pulled out automatically by a journal file parser and then saved in a CSV file (Revit 2011). Figure 3.2 takes a very small part of the CSV file parsed from journal files as an example, where six continuous commands are displayed and the user name is represented by a common name “Tom” for confidentiality consideration. In reality, the BIM design event log we explore was from 2,647 projects and created by 97 modelers, implying that the resultant RNN or LSTM NN model would be susceptible to the size of the data file. However, the parsed information stored in a CSV file is unable to be directly used for data analysis, since susceptible analysis results will be inevitably produced by the poor data quality arising from the missing, meaningless, irrelevant, and incorrect value in the CSV file. To address the concern, a kind of standard query language named Structured Query Language (SQL) is applied, which is designed to query and extract data. In particular, SQL is helpful to access and manipulate large databases at high speed and efficiency, which is especially effective in identifying and removing noisy data. Table 3.1 lists three examples of SQL queries. For example, Query 1 is executed to remove all rows with the value of “|” in “Internal” column, which is errors with no meaning. Query 2 aims to delete rows with a null value in “Command” column. Query 3 removes all commands executed less than 100 times. It would seem that data cleaning enables us to boost the conciseness and reliability of data, allowing for more accurate and dependable predictions and decision makings. Advantages of the user-friendly SQL lie in fast query processing, no coding requirement, portability, and well-defined standards. Besides, Natural Language Processing (NLP) could be another option for the crucial step of data preprocessing. I will consider NLP techniques to transform text into a more digestible form in the future study.

	User	Session	Start Time	Duration	Project
1	Tom	22	2013-03-05 16:27:20.867	6.786	CoreShell_Tom.rvt
2	Tom	22	2013-03-05 16:27:27.653	7.234	CoreShell_Tom.rvt
3	Tom	22	2013-03-05 16:27:34.887	12.786	CoreShell_Tom.rvt
4	Tom	22	2013-03-05 16:27:47.673	0.007	CoreShell_Tom.rvt
5	Tom	22	2013-03-05 16:27:47.680	3.293	CoreShell_Tom.rvt
6	Tom	22	2013-03-05 16:27:50.973	13.394	CoreShell_Tom.rvt

	View	Event	Command
1	Overall Level 02 Floor Plan North	Other	Command “KeyboardShortcut” Activate this viewpoint
2	Overall Level 02 Floor Plan North	Create	A straight detail line or a detail arc
3	Overall Level 02 Floor Plan North	Create	An arc tangent to existing entity end
4	Overall Level 02 Floor Plan North	Create	An arc by specifying center and end points
5	Overall Level 02 Floor Plan North	Delete	Lines: Detail lines
6	Overall Level 02 Floor Plan North	Other	Command “AccelKey” Save the active project

	File Path
1	\\perkinswill.net\Projects\Atlanta\800654.000_UNT_Student_Union\DESIGN\BIM\REVIT\CoreShell.rvt
2	\\perkinswill.net\Projects\Atlanta\800654.000_UNT_Student_Union\DESIGN\BIM\REVIT\CoreShell.rvt
3	\\perkinswill.net\Projects\Atlanta\800654.000_UNT_Student_Union\DESIGN\BIM\REVIT\CoreShell.rvt
4	\\perkinswill.net\Projects\Atlanta\800654.000_UNT_Student_Union\DESIGN\BIM\REVIT\CoreShell.rvt
5	\\perkinswill.net\Projects\Atlanta\800654.000_UNT_Student_Union\DESIGN\BIM\REVIT\CoreShell.rvt
6	\\perkinswill.net\Projects\Atlanta\800654.000_UNT_Student_Union\DESIGN\BIM\REVIT\CoreShell.rvt

Figure 3.2. Example of the parsed CSV file.

Table 3.1. Examples of SQL query in data cleaning.

No	Query 1	Query 2	Query 3
SQL	DELETE *	DELETE *	DELETE *
Query	FROM Sheet1	FROM Sheet1	FROM Sheet1
Sentence	WHERE Command =' '	WHERE Command = NULL	WHERE Command in (SELECT Command FROM Sheet1 GROUP BY Command HAVING COUNT (num) <100 ORDER BY COUNT (num) DESC)

3.2.2 Data mining

The goal of data mining is to track and predict design commands in sequence at the category level during the design process by exploring the cleaned data obtained from data preprocessing. Owning a more robust performance of classification and stronger memory ability, RNN and its variation LSTM NN are regarded as the basic algorithm to tackle the command sequential problems in this research, which are introduced below.

3.2.2.1 RNN

The RNN is a kind of neural network with a memory-state added in the hidden layer, which has the outstanding capability in handling sequential data. That is to say, the hidden layer with the activation function has internal memory to capture the dynamic sequential state, which allows for sending back the previous hidden state into the RNN model as a part of new inputs at the current state. The basic process of RNN is shown in Figure 3.3 with an input sequence $x = (x_1, x_2, \dots, x_t)$, hidden states of the recurrent layer $h = (h_1, h_2, \dots, h_t)$, and an output sequence $y = (y_1, y_2, \dots, y_t)$. To be more specific, x_t, h_t , and y_t denote the input, the hidden state, and the output at the time step t , respectively. The key feature of RNN lies in its hidden units, which typically obtain feedback from the previous state at time step $t-1$ to affect the current state at t (Graves, Mohamed et al. 2013). It is clear that there are cycles in the hidden layer with activation functions as the memory of the network, and thus the current h_t will become h_{t-1} at the next time step. When an input sequence x is given, h_t expressed in Eq. (3.1) can remember all previous information at time step $t-1$, and the output at time step t can be calculated under Eq. (3.2) (Du, Wang et al. 2015). By remembering important inputs, RNN has a better understanding of sequential data to make more precise predictions for the next possible event.

$$h_t = f_1(h_{t-1}, x_t; b_h) = f_1(W_{hh}h_{t-1} + W_{xh}x_t + b_h) \quad (3.1)$$

$$y_t = f_2(h_t, b_y) = f_2(W_{hy}h_t + b_y) \quad (3.2)$$

where W_{hh} , W_{xh} , and W_{hy} are the input-hidden, hidden-hidden, and hidden-output weight matrix, b_h and b_y stand for the hidden bias vector in the hidden and output layer, respectively, f_1 and f_2 are the activation function in the hidden layer and the output layer, respectively.

In fact, RNN has two drawbacks that should not be neglected. Firstly, RNN only turns out to be effective in short-term dependencies. In other words, the dependency on time in RNN from Eq. (3.1) demonstrates that the prediction h_t at time step t largely relies on the previous information h_{t-1} at time step $t-1$, which can only remember things for a small duration of time. Moreover, the issue of vanishing gradient (Hochreiter 1998) will appear in the backpropagation algorithm, in which weights will be proportionally changed

with the errors (also called gradients of loss). With the gradient becoming smaller, it will slow down or even stop the training process, causing difficulties in training a model well.

Figure 3.3. General process of RNN.

3.2.2.2 LSTM NN

To resolve the problem of RNN, Hochreiter and Schmidhuber (Hochreiter and Schmidhuber 1997) firstly proposed LSTM NN for addressing long-term dependencies by creating memory blocks and gate units as the improvement of the classical RNN. To be specific, LSTM NN can be also computed by Eq. (3.2) as RNN, but takes the place of hidden units in RNN by a more complex structure called a memory block as shown in Figure 3.4, where the information flow is controlled by three gates, namely input gate, forget gate, and output gate. In terms of the block control mechanism, it is effective to memorize long-term information and handle the gradient vanishing problem caused by a long sequence (Wei, Wang et al. 2017). As for the three gates with different sets of weight filter, they constitute the hidden layer of LSTM called memory block, aiming to control information through the block by selectively remembering or forgetting it. More precisely, multiplicative gate units in a memory cell will learn to open and close correctly in reaction to a constant error named Constant Error Carousel (CEC), in order to keep error unchanged for solving the vanishing error problem (Cortez, Carrera et al. 2018). Detailed introductions about information processing in three gates are given as follows.

a. Forget gate

The forget gate layer is responsible for removing irrelevant memory selectively from the cell state. Eq. (3.3) measures how much information will be dropped in the forget gate based on a standard sigmoid function $\sigma(x) = (1 + e^{-x})^{-1}$, which squishes value in the range of $[0,1]$. When Eq. (3.3) returns the value of 1, information from the previous hidden state and current input will be completely reserved. In contrast, 0 from Eq. (3.3) means that the information will be thoroughly forgotten.

$$f_t = \sigma(W_{fh}h_{t-1} + W_{fx}x_t + b_f) \quad (3.3)$$

where, h_{t-1} stands for the output of the previous memory block, x_t represents the current input vector, b_f is the bias vector, and W_{fh} and W_{fx} are the weight matrices from the forgot gate to the hidden layer and the input layer, respectively.

b. Input gate

There are two major parts in the input gate to add new information for memory updating. Firstly, information from the previous hidden state and current input will be fed into a standard sigmoid function σ in Eq. (3.4). A value closer to 1 indicates the higher importance of the information. Secondly, a $\tanh$ activation function scaling value to the range $[-1,1]$ is utilized to generate new memory $\tilde{c}_t$ as illustrated in Eq. (3.5). The new cell state c_t in the current memory block at the top of Figure 3.4 can be updated by Eq. (3.6), where the term $f_t \times c_{t-1}$ represents the information to forget and the term $i_t \times \tilde{c}_t$ controls the important information to be updated.

$$i_t = \sigma(W_{ih}h_{t-1} + W_{ix}x_t + b_i) \quad (3.4)$$

$$\tilde{c}_t = \tanh(W_{ch}h_{t-1} + W_{cx}x_t + b_c) \quad (3.5)$$

$$c_t = f_t \times c_{t-1} + i_t \times \tilde{c}_t \quad (3.6)$$

where, h_{t-1} stands for the output of the previous block, x_t represents the input vector, b_i and b_c are bias vectors, W_{ih} and W_{ix} are the weight matrices from the input gate to the hidden layer and the input layer, W_{ch} and W_{cx} are the weight matrices from the state of the current memory block to the hidden layer and the input layer, f_t and i_t are the vectors of forget and input gates at time t , $\tilde{c}_t$ and c_t denote the new memory and updated memory in the current block.

c. Output gate

As for the output gate, it makes decisions in the output in the current block and the memory to be exported as the input in the next memory block as Eqs. (3.7) and (3.8). More specifically, the sigmoid function provides the output information, while the multiplication of value from the sigmoid and $\tanh$ function determines the information taken by the hidden state. In general, these three gates collaborate to update memory iteratively, leading to a brief and clear training process. That is to say, the input gate and output gate both deal with gradient problems, while the forget gate provides an adaptive memory buffer to avoid infinite loop (Bengio, Boulanger-Lewandowski et al. 2013, Zazo, Lozano-Diez et al. 2016).

$$o_t = \sigma(W_{oh}h_{t-1} + W_{ox}x_t + b_o) \quad (3.7)$$

$$h_t = o_t \times \tanh(c_t) \quad (3.8)$$

where, h_{t-1} stands for the output of the previous and current block, x_t represents the input vector, b_o is the bias vector, W_{oh} and W_{ox} are the weight matrices from the output gate to the hidden layer and the input layer, o_t is the vector of the output gate at time t , c_t denotes the updated memory from the current block.

Figure 3.4. Memory block in LSTM NN.

3.2.3 Performance evaluation

Since various design commands will be divided into different classes for model training, the prediction problem in this research can be considered as a multi-class classification task. Thus, there is a need for criteria to understand and assess how a learned classifier performs on a test set. For the purpose of simply measuring the classification performance, the most commonly used metric is the prediction accuracy, referring to the overall classification ability expressed by the percentage of correct classification in Eq. (3.9).

$$Accuracy_i = \frac{tp_i + tn_i}{tp_i + tn_i + fp_i + fn_i} \quad (3.9)$$

where, tp_i is the true positive for class i , tn_i is the true negative for class i , fp_i is the false positive for class i , fn_i is the false negative for class i .

However, accuracy cannot always ensure robust evaluation of the model. Especially, accuracy has poor performance in the condition with quite a large quantity gap among different classes (Duan, Lin et al. 2018). Among other extensively used metrics, precision, recall, and F1 score can be adopted to make the evaluation more comprehensive for class-imbalanced datasets. Precision is derived from the ratio of correct classified data to the number of data labeled by the model as a member of the class in Eq. (3.10), while recall expressed in Eq. (3.11) is the proportion of the correctly classified data to the number of all class members in the data set (Wesoły and Ciosek 2018). In particular, the F1 score represents a trade-off between precision and recall for an overall evaluation of classifier performance, and the F1 score is expressed in Eq. (3.12) (Sokolova and Lapalme 2009). All these four metrics reach the best value at 1 and the worst result at 0.

$$Precision_i = \frac{tp_i}{tp_i + fp_i} \quad (3.10)$$

$$Recall_i = \frac{tp_i}{tp_i + fn_i} \quad (3.11)$$

$$F_1 \text{ score} = \frac{(1+\beta^2) \times Precision_i \times Recall_i}{\beta^2 \times (Precision_i + Recall_i)} \quad (3.12)$$

where, tp_i is the true positive for class i , tn_i is the true negative for class i , fp_i is the false positive for class i , fn_i is the false negative for class i , β represents the relative importance of recall and precision, which is usually set to 1.

3.3 Case study based on RNN

3.3.1 Data extraction from logs

The proposed RNN-based command prediction method is verified in a relatively small dataset of BIM event logs from an international design firm as a simple case study. I regard it as a relatively small database since it only contains design commands about the “Create” action in the *Revit* journal file. No other events, like delete, keyboard shortcut, and others, are taken into account. Therefore, the potential shortcoming of such a small dataset is that the data is not consolidated, which can not exactly reflect the actual design process. In fact, for numerical experiments, a small dataset is enough to validate the effectiveness of the RNN-based command prediction at a fast speed. It can also help to simplify the complex problem. Once the proposed prediction approach is proven useful, I can expand the volume and type of data. When various kinds of design commands that are not limited to the “Create” event are incorporated, it can be assumed that the dataset is sufficiently large for the more general analysis. I have deeply investigated a larger dataset in Section 3.4.

In this case, after log parsing and data preprocessing, the size of the cleaned dataset is 57,915 lines with 159 types of “Create” commands. To match the data requirements of a supervised multi-label problem, it is an important task to label data in a reasonable manner. Notably, logs provide a brief description of the executed commands. For instance, the descriptions “a wall”, “a floor”, “a ceiling”, and “a door” imply to create an object. As can be seen in Table 3.2, 159 kinds of commands will be labeled by number 1-6 in accordance with the description, which stand for creating dimensions, objects, view, elements, others, and edition, respectively. In each defined command class, Table 3.2 lists four commands as an example for a better understanding of the dataset. Therefore, the command examples shown in Table 3.2 are only a small part of the executed command types. For example, in the class about “Create object”, there are other detailed commands that are not outlined in Table 3.2, such as “a filled region”, “a staircase”, “a shaft opening”, “a railing”, and others. In total, 159 types of “Create” commands are incorporated in the prepared dataset. From the pie chart in Figure 3.5, commands labeled by 4 are executed more frequently than others, accounting for around 46.17% of the total recorded

commands. That is to say, commands associated with creating elements (class 4) are the most commonly performed command, while commands to create edition labeled as 6 are rarely conducted.

Table 3.2. Data labeling and examples.

Label	Description	Command Examples
1	Create Dimension	Aligned dimensions/Angular dimensions/ Vertical dimensions/Spot elevation
2	Create Object	A wall/ A floor/A ceiling/ A door
3	Create View	A section view/An elevation view/A floor plan view/ A default 3D orthographic view
4	Create Element	A point/ A line/ A circle/ A rectangle
5	Create Other	A text object/ A drawing sheet/ A new project/ A new family
6	Create Edition	A revision cloud/An array from the selected objects/Edit the path by sketching in a plane/ Edit the path by picking existing edges or lines

Figure 3.5. Pie chart of command number in each class. (The number outside the brackets is the command frequency and the number inside the brackets is the command percentage.)

3.3.2 RNN model development

As a preparation of the training set and the testing set, the cleaned dataset is split into an 80%-20% ratio (a common practice in data science). More specifically, the subset of 46,443 commands are utilized for RNN model training, while the rest of 11,583 commands, acting as a proxy of new data, serve for testing how the trained model can be

generalized on new data. Based on repeated experiments, I build an RNN model with 1 hidden layer, 64 hidden nodes, 10 timesteps, 32 batch size, 100 epochs, and 0.001 learning rate, which is compiled with the stochastic gradient descent (SGD) optimizer to minimize the cross-entropy loss. The activation function in the hidden layer is ReLu, and the Softmax function is applied in the output layer to shift the logits into probabilities. The next type of command will be predicted using the previous 10 design commands in sequence.

The performance of the RNN model in the training set and testing set during 100 epochs is displayed by two types of learning curves in Figure 3.6 (a) and (b), which are called the loss curve and the accuracy curve. From Figure 3.6 (a), the reliability of the RNN model can be preliminarily validated, since training and testing loss gradually decrease at a good learning rate, and the training loss is slightly smaller than the testing loss with a gap of 0.05. What's more, the training and testing accuracy give rise and converge with the number of epochs in Figure 3.6 (b). At the 100th epoch, there is no obvious discrepancy in the training accuracy (63.98%) and testing accuracy (63.86%). To compare it with a human behavior prediction case by the deep learning with only 47.4% accuracy (Almeida and Azkune 2018), there's a reason to believe that our developed RNN has a great classification ability. To this end, the next possible command is assigned probabilities of six classes. Table 3.3 provides an example of prediction results for five continuous design command classes ($3 \rightarrow 1 \rightarrow 4 \rightarrow 4 \rightarrow 2$), where the class can be identified based on the largest probability in bold fonts. These predicted commands can act as operation guidance in the modeling process. It means that designers no longer spend too much time thinking about the next possible command class, and then they can easily search the proper design command in a certain class. Although the behavior prediction performance of our RNN model has made a significant improvement compared to the existing studies, there are some potential methods to further raise the classification accuracy. For example, we can rely on k-fold cross-validation instead of an 80%-20% data split. We can try some optimization algorithms to better fine-tune parameters of deep learning models, such as the particle metaheuristic algorithm (PSO), genetic algorithm

(GA), and others. Also, we can carry out an oversampling technique named Synthetic Minority Oversampling Technique (SMOTE) to deal with an imbalanced dataset.

Figure 3.6. Learning curve of: (a) Loss; (b) Accuracy.

Table 3.3. Prediction results of five continuous command classes.

Label		Probability					
True	Predicted	Class 1	Class 2	Class 3	Class 4	Class 5	Class 6
3	3	0.057	0.023	0.721	0.178	0.017	0.004
1	1	0.584	0.158	0.037	0.155	0.063	0.003
4	4	0.047	0.057	0.056	0.800	0.036	0.004
4	4	0.214	0.078	0.062	0.332	0.291	0.023
2	2	0.033	0.692	0.024	0.224	0.026	0.001

3.3.3 Result analysis

To measure the RNN classification performance for each class, a 6×6 confusion matrix allowing a summary of the correct and incorrect prediction results on the set of the test data is presented in Figure 3.7, where the row corresponds to the true class and the column stands for the predicted class. The number along the major diagonal represents the data classified correctly, whose true label is equal to the predicted label. It is observed that totally 7397 data are predicted correctly herein, resulting in an overall accuracy of 63.86% (7397/11,583). Class 4 is more likely to acquire desired predictions, followed by class 2 and 5. Class 1, 2, 3, 5, and 6 all tend to be mislabeled as class 4, since the amount of data in class 4 is a little greater than other classes. On the contrary, the size of command 6, which only contributes to 1.24% of total data as illustrated in Figure 3.5, is too small to

be learned well and predicted accurately. From the view of Recall, commands with label 1-6 obtain the correct predictions in the probability of 37.27% (492/1320), 64.67% (1686/2607), 34.76% (308/886), 76.84% (3846/5005), 63.69% (1063/1669), and 2.08% (2/96), respectively, which also verify that the performance of the six classifiers largely depends on their data size. That is to say, design commands in class 4, 2, 5 can be predicted more easily.

Moreover, the receiver operating characteristic (ROC) curve and the Area under the ROC Curve (AUC) as shown in Figure 3.8 can also be considered, which graphically represent the trade-off between the true positive rate (TPR) and the false positive rate (FPR) at all classification thresholds in the range [0,1]. It can be seen that all ROC for the six classifiers 1-6 lies in the area of the left corner, which is far away from the blue line (a random classifier with AUC 0.5). Since the curve closer to the upper left corner in the graph implies a better classifier, it seems that the six classifiers work well. Moreover, a useful classifier can be determined when the AUC value is between 0.5 and 1. The AUC value of the six classifiers is all greater than 0.78, indicating reasonable discrimination and generalization ability. That is to say, the established RNN model is able to achieve satisfying classification performance. By comparison of the AUC value, the classifier for command class 5 is the best, which has the highest AUC 0.89.

Figure 3.7. Confusion matrix of prediction results in the testing set.

Figure 3.8. ROC and AUC of command class: (a) 1; (b) 2; (c) 3; (d) 4; (e) 5; (f) 6.

3.4 Case study based on LSTM NN

3.4.1 Data preparation

A more complicated case study is performed, which employs 4 GB real-world BIM design event log files documented in *Autodesk Revit* software from an international architecture design firm. To be specific, the large event logs concern the model evolution over 2,647 projects conducted by 97 designers jointly from Oct 2012 to Oct 2014. There are two main types of projects for different purposes recorded in these logs, which are the residential buildings (around 30%) and commercial buildings (around 70%). Designers model these projects according to their related design codes under similar steps to accomplish three important parts, namely the architecture, structure, and mechanical, electrical and plumbing (MEP). To deal with the great volume of indigestible text data, a journal file parse is employed to parse the design log file in an automatic manner. Accordingly, the relevant information is retrieved and imported into a CSV file owning 853,520 lines and 31,040 kinds of commands. Each line represents a detailed record of operation, which not only contains executed commands, but also documents corresponding information of the user, project, timestamp, and others. Nevertheless, some

incomplete, noisy, and inconsistent data will exist in the CSV file causing detrimental effects in analytical results and computation efficiency.

To ensure the data quality for intended data analysis, the step of data cleaning should be performed to detect and handle incomplete and useless data according to the following rules: (1) Null values are the most common issue to bring about problems in the data analysis, like poor statistical power, high bias, low representativeness of samples and high complexity in analysis, and others (Kang 2013). Thus, 12,544 rows containing null values will be firstly deleted as a quick solution. (2) Error records in the form of a single symbol, including “|”, “&” and “#”, represent unwanted noisy data with no effect on feature explanation, which can even add complexity and reduce result accuracy in the end. For noise elimination in this case, 485,467 rows with meaningless records are completely deleted. (3) Removing data with extremely low frequency can enhance predict accuracy. It is found that high-frequency data contribute a lot to make useful decisions, which could improve the prediction accuracy. Oppositely, data that appears in less frequency does not play a key role in the classification problem. For instance, Li et al. (2016) proved that the prediction results in the text classification would drop significantly when some high-frequency words are removed. Forman (2003) determined to get rid of words occurring fewer than two times in 299 binary text classification tasks, resulting in 98.2% accuracy. Similarly, it is also reasonable to take no account of non-dominant commands, which can be defined as commands executed less than 100 times with less than 1% occurrence probability. Herein, 3,453 rows with these least frequently used commands are deleted from the database.

Table 3.4 illustrates a comparison of the characteristics between original data and cleaned data. It is clear that the cleaned dataset owns 352,056 rows of commands, which is less than half of the rows in the original dataset. As the research objective, 377 projects have totally executed 289 kinds of commands 352,056 times. Figure 3.9 visualizes the command execution frequency within each project in descending order. Evidently, almost 80% of projects carry out a number of valid design commands between 100 and 1,000. Only three projects contain more than 10,000 times of valid command execution, with an exact frequency of 11,100, 10,475, and 10,207, respectively.

Table 3.4. Comparison of the original dataset and cleaned dataset.

Total Number	Original Dataset	Cleaned Dataset
Line	853,520	352,056
Project	2,647	377
Command Type	31,040	289
Journal Event Name	8	3

Figure 3.9. Design command execution frequency in each project.

3.4.2 Command classification

Before the processed data is fed into a deep learning model, data labeling is an indispensable step in the context of supervised learning, especially for classification problems. It is well-known that the quality of the labeled data exerts an important impact on the prediction performance, and thus how to classify and label the design commands in this research must be extremely attentive. According to the different roles of commands and their similarities, a series of independent design commands within the database can be sorted and categorized into distinct command classes. As can be seen in Table 3.5, all 352,056 commands are assigned into 14 manually predefined classes with numerical labels 1-14 based upon an integrative view concerning data itself in the parsed CSV file,

documents, and expert knowledge. In consequence, the rationale behind such 14 command classes can be summarized below.

Firstly, the column named “Event” in the parsed CSV file of Figure 3.2 has classified all design commands into three major events, namely “Create”, “Delete”, and “Other”. Secondly, the column “Command” in Figure 3.2 provides a specific description of the operation and its result. Based on the content in column “Command”, the semantically relevant commands can be recognized, which will then be summarized in the column “Description” in Table 3.5. For instance, “A line”, “An arc by specifying three points” and “A rectangle” in column “Command” all contribute to building elements, which can be assigned to the same class with the journal event “Create” and the description “Element” in Table 3.5. Clearly, this kind of content-based classification method is flexible and human-comprehensible. Thirdly, the *Revit* user interface is another source to determine the class. The “Ribbon” at the top of the user interface in *Revit* comprises a set of tools for creating a project or family, where tools with similar roles and features are arranged closer. For instance, in the *Revit* architecture tab, icons of building a wall, door, floor, roof, ceiling, and others, are in the same module, resulting in the classification results for command class 2 in Table 3.5. More detailed instructions of *Revit* can be found in (Tickoo 2013). Fourthly, experts with a great deal of specialized knowledge and experience in *Revit* modeling will check and modify the classification table to guarantee its logicity and rationality, and thus the wrong and counterintuitive results in Table 3.5 can be reduced if possible. In particular, a suitable choice of experts is project managers who are leaders in BIM-based modeling projects with a minimum of five years of modeling experience and high proficiency in *Revit* software. Moreover, professors, whose research area focuses on BIM applications especially in design, can also assist in checking command classification on the basis of their solid skills and theories.

In addition, Figure 3.10 visualizes the data grouping results in terms of the command class and the journal event, which are shown in the inner ring and outer ring, respectively. It is obvious that the frequency of executing a certain command class is varied significantly. In specificity, commands in class 12, 13, and 4 are performed most frequently, which take up 22.36%, 14.69%, and 11.32% of the total command records,

respectively. On the contrary, commands belonging to class 6, 10, and 14 are implemented less than 3% times. Commands in the journal event “Other” and “Create” can be considered as the most commonly performed actions, since their total percentage reaches up to almost 85%. In other words, commands related to the function of deleting something are less likely to occur than others.

Table 3.5. List of 14 command classes and related Top 5 commands.

Command Label	Journal Event	Description	Frequency of a command	Command Examples
1	Create	Dimensions	17,920	Aligned dimensions Spot elevation Angular dimensions Horizontal or vertical dimensions
2	Create	Objects	20,525	A wall An object similar to selected object A room A door A floor
3	Create	View	22,672	A default 3D orthographic view A 3D view by placing camera and focus A section view An elevation view A callout view
4	Create	Element	39,853	A line An arc by specifying three points A rectangle A circle
5	Create	Others	19,609	A spline by specifying control points A tag by category An instance of a component type A text object Associative group of objects A drawing sheet
6	Create	Edition	7,393	A revision cloud An array from the selected objects Edit the path by sketching in a plane Edit the path by picking existing edges or lines
7	Delete	Element	22,813	Site: <Sketch>: Model Lines Workset1: <Sketch>: Model Lines <Sketch>: Line <Sketch>: Model Lines Lines: Detail Lines
8	Delete	Furniture and elevator	10,808	2 items: Furniture 2 Furniture systems: Generic WS: Generic WS 3 items: Furniture 3 Workset1: Elevators and stair Furniture systems: Benching workstation
9	Delete	Object	13,308	AR_Interior: Walls: Basic Wall: 130mm

Command Label	Journal Event	Description	Frequency of a coammnd	Command Examples
10	Delete	Generic model and dimensions	7,534	<p>Workset1: Walls: Basic Wall: SD Generic-4.5 Partition</p> <p>Workset1: Walls: Basic Wall: SD Generic-6 Partition</p> <p>2 items: Wall 2</p> <p>IA_ Interior: Walls: Basic Wall: A4</p> <p>Units and Cores: Generic Models: UNT_NEF-A</p> <p>1+1_33: A1+1</p> <p>Units and Cores: Generic Models: UNT_NEF-B</p> <p>1+1_43-6: B 1+1_43-6</p> <p>View Floor Plan: LEVEL 29_Working: Dimensions:</p> <p>Linear Dimension style: PWLinear 3/32</p> <p>Dimensions: Linear Dimension Style: PWLinear 3/32</p> <p>Dimensions: Linear Dimension Style: Default linear style</p>
11	Other	Command "AccelKey"	28,904	<p>Copy the selection and put it on the clipboard</p> <p>Cut the selection and put it on the clipboard</p> <p>Save the active project</p> <p>Print the active window</p> <p>Close the print preview</p>
12	Other	Command "Internal"	78,720	<p>Finish sketch</p> <p>Save the active project back to the central model</p> <p>Align references</p> <p>Pick lines</p>
13	Other	Command "KeyboardShortcut"	51,717	<p>Activate this viewpoint</p> <p>Move selected objects or their copies</p> <p>Trim/Extend two lines or walls to make a corner</p> <p>Align references</p> <p>Control visibility and appearance of objects</p> <p>Move copies of selected objects</p>
14	Other	Other	10,280	<p>Command "SystemMenu": Quit the application; prompts to save projects</p> <p>Command "PrintPreviewUI": Print document</p> <p>Command "PrintPreviewUI": Close print preview</p> <p>Command "StartupPage": Open an existing project</p> <p>Command "StartupPage": Open this project</p>

Figure 3.10. Percentage of command number in 14 command classes and three journal events.

3.4.3 LSTM NN model development

For the LSTM NN training and testing, the labeled data is partitioned into a training set and a test set with an 80%-20% split. Model parameters are tuned in the training set to achieve optimal accuracy, while the test set is employed to evaluate the model performance. Notably, the data splitting process is not random. The entire data on the database is firstly sorted by the length of commands in projects from short to long. Then, the first four records are attributed to the training set, and the next one command is put into the test set, and so on. As a result, 300 projects containing various lengths of command sequences are in the training set to ensure the quality of training data. The proportion of data length in the test set is very alike to the training set. It means that the test set can also have more representative data to cover various lengths, which can enhance the generalization ability of the LSTM model. In sum, this systematic splitting method possesses a stronger ability to handle new data and is more likely to generate promising predictions. The size of the training set and test set is 285,292 and 66,764, respectively. All the training and testing procures are implemented in TensorFlow, Google's open-source machine learning framework.

In this case, the standard LSTM NN is established, which is configured by one input layer, one hidden layer with memory blocks, and one output layer with no innovation in structure. To feed the data into the same neural network multiple times, the preprocessed data is divided into several batches with a size of 32 and experiences 100 epochs. In charge of the training stage, two more hyper-parameters need to be taken into account, namely the learning rate and the number of memory cells, which are both closely related to the neural network performance. Since parameters are updated and optimized in each epoch of training by the method of SGD, the learning rate guarantees that the model can converge to the local minima with respect to the loss gradient at a proper speed. Figure 3.11 illustrates the variation trend of training and testing accuracy with the increase of epochs, when the learning rate and memory cells in the designed LSTM NN model are set to different values. The accuracy is derived from Eq. (3.9) in Section 3.3. From Figure 3.11 (a) and (b), the learning rate at 0.0001 gives rise to a stable training process, however, it will take a great deal of time to reach the minimum of the loss function. Besides, it is also inappropriate to take a too large learning rate, like 0.1 and 0.01, even though it is remarkably fast to reach high training accuracy within the first ten epochs. In the comparison between Figure 3.11 (a) and (b), the model under a learning rate of 0.01 experiences overfitting, and the corresponding testing accuracy gradually falls with no convergence. Then, Figure 3.11 (c) and (d) demonstrate the trends of accuracy when the number of memory cells gradually increases without dropout as 32, 64, 128, 256. Although more memory cells improve the training accuracy, the testing accuracy in the model with 128 or 256 cells will suddenly increase in the 80th epoch, resulting in larger accuracy than training without convergence. Consequently, the best performance can be reached with 64 memory cells without dropout, which are thus utilized here.

Ultimately, the developed LSTM NN with 1 input layer, 1 output layer, 1 hidden layer, 64 hidden nodes, and no dropout is trained at the learning rate 0.001 and optimized by the SGD optimizer, which finds the minimum of the cross-entropy loss function in the case. In addition, the previous 10 design commands are learned by the LSTM model to make an evidence-based prediction for the next potential command. Figure 3.12 demonstrates the loss and accuracy curves of the training set and test set with respect to

the number of epochs, which verifies the rationality of the developed model. Observably, both the training and testing loss gradually decrease during the training process, while training and testing accuracy tend to converge on around 70.7% and 70.5% at the end of 100 epochs. To further validate the LSTM model automatically, the simplest and time-efficient validation approach named the hold-out method (Yadav and Shukla 2016) is employed, which randomly splits available data into two non-overlapping parts for training and testing by different proportions. Thus, the part for testing called the hold-out dataset estimates the accuracy of the model. In this case, the accuracy is all greater than 60% when 10%, 20%, and 30% hold-out validation are carried out and repeated several times. It is worth noting that human behavior can be random and uncertain to some extent to increase the difficulty of behavior prediction. Based upon the research by Almeida et al. (2018), which predicted user's actions with only 47% accuracy by multiscale CNN, it is reasonable to affirm the validity of our developed LSTM model owning more than 60% accuracy in the validation process.

Figure 3.11. Accuracy curves at training and test sets: (a) training set at different learning rates; (b) test set at different learning rates; (c) training set with different numbers of memory cells; (d) test set with different numbers of memory cells.

Figure 3.12. Loss and accuracy curves at training and test sets: (a) Loss curve of training and test set; (b) Accuracy curve of training and test set.

3.4.4 Result analysis

In general, the training process of LSTM NN invokes a knowledge base of information from previous command sequences in different projects to catch the most relevant command at a category level. There are 77 projects owning 66,764 commands in the test set. Figure 3.13 displays the histogram of testing accuracy. Table 3.6 presents the results for precision, recall, and F1 score for each command class. The probability of the predicted command class based on actual commands belonging to class 12 is shown in Figure 3.14. To facilitate a better understanding of the prediction process, Figure 3.15 provides an example of a command sequence with 11 continuous commands. All results of the predicted command class are analyzed in detail as follows.

(1) The promising classification performance of the developed LSTM NN can be verified by four metrics mentioned in Section 3.3. Totally 47,096 records in the test set with the size of 66,764 are classified correctly, reaching an overall accuracy of approximately 70.5%. From the histogram shown in Figure 3.13, more than half of projects (totally 38 projects) have the test accuracy falling in the range of [0.65, 0.8]. Apart from the overall accuracy, another three metrics, namely precision, recall, and F1 score, are utilized to evaluate the model performance for each command class, separately.

Results of precision, recall, and F1 score associated with 14 kinds of design commands are demonstrated in Table 3.6. For each individual class, precision and recall are calculated by Eqs. (3.10) and (3.11) to manifest the significance in the retrieval of positive examples in design command classification. Besides, F1 score from Eq. (3.12) conveys the balance in precision and recall. Of particular interest is that the command class 12 owns the largest recall, which means that 87.65% data with the true label 12 can be predicted correctly as commands in class 12. The top three recall is in class 12, 13, and 4, which can be regarded as the majority class accounting for around 48.37% of the total commands. Nevertheless, the precision of these three majority classes 12, 13, and 4 is not the highest resulting from a great number of false positive, which is even lower than other classes. The relatively small precision in class 12, 13, and 4 is mainly due to the slight imbalance of data size in different classes, which adversely affects the reliability and precision of the predicted results to some extent. Since there is a high likelihood for predictions to be biased towards the majority class (12, 13, and 4), which are more used in this particular case, other command classes tend to be mistakenly categorized as class 12, 13, and 4. From F1 score, its maximum lies in class 12 (75%), while other classes are in the range of [58%, 70%]. It also indicates that the command class 12 could be predicted correctly easier than others. The overall F1 score can be calculated by the mean of the per-class F1 score, reaching an acceptable value of 64.36%.

(2) The developed LSTM-based intelligent command prediction approach generates a kind of specific knowledge in the form of probability, which is able to provide suggestions quantificationally about the next most possible command class to users through the whole designing process. In other words, different probabilities are assigned to 14 candidate command classes, and then predictions can be determined easily regarding the highest probability among all command classes. Figure 3.14 (a) is taken as an example to display the probabilistic predictions about the totally 14,928 commands actually belonging to class 12 in the test set. Intuitively, the largest probability in the range of [0.5, 0.85] is mainly represented by blue circles, implying that the predicted command class is most likely to be labeled as 12. To make Figure 3.14 (a) clearer, Figure 3.14 (b)-(o) present the frequency distribution and cumulative percent for each predicted command class. It is

obvious that Figure 3.14 (m) has a distinctive difference. Since 80% of records in Figure 3.14 (m) own the probability larger than 0.45, it implies that the predicted results have more than a 45% chance to be predicted correctly as class 12. In contrast, the probability to be other command classes except for 12 is substantially close to 0.1. Especially for class 6, 10, 14, 80% of commands are almost impossible to be predicted as 6, 10, 14 due to the probability smaller than 0.01. In brief, more than 80% of predicted results are consistent with the actual commands labeled 12, which proves that the developed LSTM NN will achieve reliable classification results in command 12. In addition, probabilistic information can act as the main idea in the creation of *Revit* plugin to realize a better user interface interaction in the future. For a better understanding of predicted probability, a continuous design sequence containing eleven commands is displayed in Figure 3.15. Accordingly, the command class sequence can be predicted as “12 → 13 → 11 → 12 → 1 → 2 → 4 → 4 → 7 → 9 → 12” based on the highest probability in each bar, which is exactly the same as the actual value.

(3) To minimize the negative influence from data imbalance, the top three most possible command classes with the top three highest probabilities are planned to be provided instead of only one single possible command class. It should be noted that desired prediction results sometimes cannot be directly obtained from the highest probability. A predicted result for a command belonging to class 4 is taken as an example. The correct prediction comes from the second-highest probability of 18.92% in class 4, rather than class 12 with the highest probability of 26.1%. Thus, a more convincing reference for the design process should be made up of the most possible command class along with two more potential command classes, whose probabilities are followed by the highest one. To further evaluate the model classification ability, a definition of top- k accuracy is adopted, which measures the probability of the top k prediction results matching the expected class (Lapin, Hein et al. 2015). Specifically, the top-1 accuracy (the conventional overall accuracy in Eq. (3.9)) in this case is 70.5%, while the top-3 accuracy under the same training and testing condition can even reach around 90.0%. In general, an accuracy greater than 90% is considered as a relatively high one representing the promising classification performance in most cases (Peter and Ying 2006). Besides,

the overall accuracy herein increases by 13% and 11% from top-1 to top-2 and from top-2 to top-3, respectively. When k is larger than 3, the rate of accuracy growth will keep very small below 5%, and the accuracy will display the indication of convergence. That is to say, $k=3$ can be an optimal choice here. The three most possible candidates are capable of raising the accuracy and reliability of the prediction method significantly, which meanwhile provide designers with more recommendations of possible commands to build models. With the comparison of LSTM-based human behavior prediction performance in (Almeida and Azkune 2018), where the highest top-1 and top-3 accuracy are only 47.4% and 72.6%, the performance of our developed design command prediction based on LSTM NN can be confirmed in learning the sequential data structures of designers' actions.

(4) The proposed data-driven approach has the potential to guide the design behaviors, which is possible to boost the disambiguation process of model evolution in both quality and efficiency. To be more specific, during the design process involving a great deal of subjectivity, randomness, and uncertainty, LSTM NN can learn a large number of command sequences and their dependencies, and then continuously provide the designer with the three most possible design commands in the next step. The superiority of LSTM NN-based methods mainly lies in two aspects. First, it is worth noting that these recommended commands can adapt to changeable conditions and the design behavior of different persons, and thus they are generally logical and meaningful. For example, a person is accustomed to using the keyboard shortcut for object copy after creating an object, like a wall, door, and others. If he executes the design command in class 2, LSTM NN may suggest him to do the next command from class 13. Second, since three potential commands can be offered, designers will have wider choices to smooth the complex design process. By directly following the three recommendations of the next command class from the probabilistic model, the designers can speed up their work. Only if all the three predicted command classes and their related commands are improper, the designer needs to spend time rethinking commands and come up with their own opinions. We have briefly introduced our idea to some unskilled and skilled designers and obtain their feedback. The discussion about the important role of the proposed prediction method is

presented as follows. For unskilled designers who are unfamiliar with the modeling software and process, they are convinced that this approach enables them to master the modeling method as quickly as possible. They need no more serious consideration about the next type of command at each step, which can accelerate the design significantly. As for the skilled designers, they expect that the LSTM model can explore characteristics of their design behavior to formulate customized command predictions in accordance with designers' habits from an individual level, which can even avoid some unwanted mistakes. Also, they desire a high-accuracy LSTM model, otherwise, they are afraid that some unnecessary hesitations and misleading may occur. To sum up, under the hypothetical experiments, all the designers believe that the proposed command recommendation method is beneficial to transform the tedious and time-consuming design process into a high degree of automation and reliability.

Figure 3.13. Histogram of test accuracy.

Table 3.6. Precision, recall, and F1 score for each class.

Class	Precision (%)	Recall (%)	F1 score	Class	Precision (%)	Recall (%)	F1 score
1	77.49	61.02	0.683	8	91.44	52.05	0.663
2	78.02	62.75	0.696	9	91.06	55.23	0.688
3	73.75	63.67	0.683	10	94.09	47.72	0.633
4	61.54	67.95	0.646	11	69.87	64.09	0.669
5	76.26	62.39	0.686	12	65.53	87.65	0.750
6	91.91	43.01	0.586	13	57.33	68.95	0.626
7	70.42	63.26	0.667	14	92.87	48.18	0.635

Figure 3.14. Probabilistic results to predict the actual command class 12 in (a); Probability distribution of the actual command class 12 to be predicted as command class (b) 1; (c) 2; (d) 3; (e) 4; (f) 5; (g) 6; (h) 7; (i) 8; (j) 9; (k) 10; (l) 11; (m) 12; (n) 13; (o) 14.

Figure 3.15. Example of a command sequence with 11 commands.

3.4.5 Discussions

In this section, I explore the impact of a parameter named the timesteps on the predicted accuracy from LSTM NN. To be more specific, the timestep indicates the number of lagged observations. It is believed that more lagged observations can pave a potential way to improve the predictive performance of the model. In this regard, the value of timestep n herein means LSTM NN will learn the previous n design commands to predict the next one. As for the physical time step, it is in the unit of second, since the execution of design commands is only to click the mouse, which is very fast. Also, I make comparisons of the prediction performance from LSTM NN and other three machine learning methods, namely k -nearest neighbors (KNN), random forest (RF), and support vector machine (SVM). Discussions are outlined as follows.

(1) Both the training accuracy and testing accuracy gradually rise along with the increase of timesteps number from 5 to 30. The timesteps representing the number of prior observations used for prediction is one of the most critical parameters to affect the performance of the LSTM NN, which is herein set up to be 5, 10, 15, 20, 25, and 30 for discussion. That is to say, LSTM NN will take into account the previous 5, 10, 15, 20, 25, and 30 design commands along with the current data point to make more accurate

predictions. To reduce the randomness in the predicted results, the training and testing process based on the developed LSTM NN is repeated ten times separately, and then the results from these ten experiments are shown by the bars in the curve in Figure 3.16. The length of the bars denotes the range of accuracy, which reflects the fluctuation of predicted results. As can be seen in Figure 3.16, accuracy has visible differences in terms of different timesteps. It is observable that a larger value of timesteps tends to reach higher training and testing accuracy. Nevertheless, the difference between accuracy from two large timesteps is much smaller than that from two small timesteps. To be specific, the distance between the curves of training accuracy under timesteps 5 and 10 is wider than that between timesteps of 25 and 30, and so does the testing accuracy. In Figure 3.16 (b), predictions under the consideration of the previous 25 and 30 commands hold very similar testing accuracy. That is to say, when the number of lagged observations goes up to some extent (such as the value of 25), there is no significant enhancement in prediction performance.

(2) It is unreasonable to blindly increase the number of previous commands to pursue high precision in prediction. To reveal the detailed training and testing accuracy after the ten experiments mentioned above, the box plot in Figure 3.17 is drawn to intuitively capture the characteristics of all the ten results under different timesteps at the end of 100 epochs. Clearly, the predicted results in the test set experience greater fluctuation than in the training set, which is reasonable. In Figure 3.17 (b), some outliers exist in the condition of 10 and 25 timesteps, and the length between the first and third quartile is quite long under the value of timesteps 5, 15, 25, and 30, indicating a greater deal of uncertainty in the testing phase. Additionally, after 100 epochs, the maximum and minimum of testing accuracy in timesteps 30 are 0.709 and 0.707, respectively, which are a rise of 0.748% and 0.754% in the higher and lower value under timesteps 5. In spite of the better performance in timesteps 30 than 5, the range of testing accuracy based on timesteps 25 between 0.707 and 0.709 is almost the same as timestep 30, and its mean value 0.708 is also nearly equal to that in timesteps 30. Hence, when timesteps increase from 25 to 30, no notable improvement occurs in accuracy. But the uncertainty from 25 to 30 even becomes greater, since overfitting is more serious in timesteps 30.

(3) LSTM NN is able to achieve the best prediction performance in both accuracy and training efficiency. Table 3.7 lists parameters of LSTM NN, KNN, RF, and SVM, and makes a comparative analysis of the four models with regard to predicted accuracy and training time. Observably, LSTM NN is superior to the three machine learning methods significantly with at least 7% accuracy improvement. It should also be underlined that SVM has the opportunity to gain relatively higher accuracy than KNN and RF. However, it will take a fairly long time to train SVM, which brings difficulty in optimizing its parameters promptly. Compared with the above-mentioned machine learning algorithms or a simple prediction just based on the frequency of use, the superiority of LSTM largely lies in its strong capability of modeling sequential data, which provides temporal memories to capture long-term dependencies from previous actions (Kumar, Goomer et al. 2018, Sagheer and Kotb 2019). Since the current step is greatly affected by the previous commands during the design, LSTM NN can be an ideal choice to realize a sequence prediction of command classes for different designers according to their design behavior and habits.

Figure 3.16. Accuracy at different timesteps based on (a) training set; (b) test set.

Figure 3.17. Accuracy about ten experiments after 100 epochs based on (a) training set; (b) test set.

Table 3.7. Comparison of predicted accuracy and training time by different methods.

Method	Parameters	Accuracy	Rank of training time (In descending order)
KNN	Number of neighbours = 3	0.612	3
RF	Number of trees = 100	0.614	1
SVM	Maximum depth of the tree = 2		
	Kernel = rbf	0.657	4
	Penalty parameter = 10		
	Gamma = 0.1		
LSTM	Batch size = 32	0.705	2
	Number of hidden layers = 1		
	Number of memory cells = 64		
	Learning rate = 0.001		
	Number of timesteps = 10		

3.5 Chapter Summary

This chapter develops a deep learning-based intelligent design command prediction approach towards the automation and intelligence of a design process. Thus, it presents the opportunity for accurately predicting design command sequence and then automating the design command execution during the design process. The main steps can be outlined as: data preprocessing, data mining underlying RNN or LSTM NN, and performance evaluation. More specifically, RNN and LSTM NN are powerful in learning sequence

data and modeling temporal dependency on the designer's sequential behavior, and thus they can provide suggestions about the next possible design command class to guide the design behavior of designers. Meanwhile, the top three most possible command classes can be offered to further improve prediction performance, contributing to reducing subjectivity, randomness, and uncertainty from designers. Compared with a previous study about the LSTM NN-based human behavior prediction under the top-1 accuracy of 47.4% and top-3 accuracy of 72.6% (Almeida and Azkune 2018), our proposed approach can significantly raise the accuracy to over 70% and 90%, respectively. As a result, these prediction results can serve as an operation reference to speed up modeling and avoid unnecessary operation mistakes, enabling a more automated, efficient, and reliable modeling process.

A point to be noted that I predict the design command at the command class level in the two case studies, aiming to ensure the simplicity and high accuracy of the multi-class classification problem. The detailed reason for class level prediction is given below. In order to make data to be understood by the deep learning model, it is necessary to group the cleaned data into several classes based on their effects and transformed it into numerical form. The concern of partitioning and labeling 352,056 design commands in this research must be extremely attentive for two reasons. For one thing, it has been proved that a smaller number of classes contribute to a reduction in the training time (Arnaiz-González, González-Rogel et al. 2017, Arnaiz-González, Díez-Pastor et al. 2018). To improve the model training efficiency, it is an ideal solution to categorize a series of independent design commands within the database into several command classes according to different roles of commands and their similarities. For another, due to the sparsity of labels arising from some rarely executed commands, to assign one label to each command is more likely to produce poor prediction performance. Bernardini et al. (2013) conducted experiments to compare the performance of the multi-class learning under different label number, which turned out that the complexity of learning a multi-classifier can be diminished and classification reliability can be raised with fewer classes and more data in one class.

In light of the simple case study based on RNN, it concerns 57,915 command records about the “Create” operation. Label 1-6 is assigned to different design commands in terms of their roles. Then, 80% of data as the training set is fed into the RNN model to tune its parameters for achieving optimal accuracy. The rest of 20% data are utilized as the testing set to predict sequential design commands based on the probability from the output layer. From evaluation of the confusion matrix, ROC, and AUC, it has been proved that the established RNN model has a strong ability to distinguish a certain command class from others with an overall accuracy of 63.86%. It is believed that the predicted command class can be helpful in improving the modeling process in both efficiency and quality.

Moreover, the LSTM NN is conducted in a more complex case study involving a 4GB real-world BIM design event log, which keeps all kinds of commands with different functions, including “Create”, “Delete”, and “Others”. As the preparation of LSTM NN inputs, commands from BIM design logs need to be firstly grouped into 14 classes according to their effects and encoded by numerical labels 1-14. To enhance the accuracy of design command classification, it is essential to properly tune parameters of LSTM NN, such as timesteps, number of memory cells, and others. In the end, the probability can be assigned to each command class as quantitative and convincing predictive results. Specifically, the overall accuracy in this case of a multi-class classification problem reaches 70.5% when the LSTM NN with 1 input layer, 1 output layer, 1 hidden layer, 64 memory cells, no dropout, and 10 timesteps is trained at the learning rate 0.001 and optimized by SGD optimizer. The performance of LSTM NN is greatly superior to KNN, RF, and SVM by at least 7% in terms of accuracy. Chances are more than 50% that all command classes are possible to obtain correct predictions except for class 6, 10, and 14. To sum up, the proposed approach performs well in learning the occurrence and dependencies of design command sequences from BIM event logs. Under the hypothetical example using the proposed approach, LSTM NN can learn features from designers’ subjective behavior effectively and predict the next possible design command class intelligently towards automation of the design process. As expected, the three most possible command classes can be offered as the recommendations under the assumption that the correct class tends to appear owning the top three highest probabilities. In

particular, the top-3 prediction accuracy can arrive at 90%. Due to the high reliability of the suggested command classes, it is believed that following recommendations during the BIM-based design phase can present a unique opportunity for designers to speed up their modeling process and prevent some unnecessary mistakes.

CHAPTER 4. EXPLORING CHARACTERISTICS OF DESIGN PERFORMANCE BY CLUSTERING METHODS

4.1 Introduction

This chapter addresses the Research Objective 2 of this thesis. The specific objective is to develop a clustering-based BIM event log mining approach to understand the characteristics of design performance from both the individual and team levels. Its ultimate goal is to support data-driven decision making for managers to strategically schedule personalized work for different designers, contributing to boosting design efficiency and smoothing the design process. The proposed framework consists of three major parts, including data preprocessing, data clustering, and cluster analysis. In the beginning, a set of features associated with designers' engagement and efficiency needs to be carefully pulled out from huge volumes of text-based event logs, which will inevitably raise challenges in uncovering latent and meaningful patterns. To deal with the non-deterministic and subjective design behaviors, two novel clustering algorithms incorporating neural networks and fuzzy clustering are proposed to proceed the prepared dataset. What's more, clustering validity indices (CVIs) are calculated to evaluate the goodness of clustering results numerically and decide the appropriate number of clusters. As expected, the hybrid clustering algorithm can retrieve inherent insights into the person's design behavioral patterns under satisfactory clustering quality and efficiency, allowing for information cohesion and smart BIM-enabled project management.

There are five major research questions, which are (1) How to preprocess huge multi-dimensional BIM log data with design and temporal information in text format, in order to make it understandable for the clustering algorithm; (2) How to conduct the two-level (individual and team) design efficiency analysis based on the EFKCN method with fewer iterations and more stable performance under noise, and thus alike design behaviors and designers with similar design efficiency can be divided into the same cluster; (3) How to

further improve the EFKCN method for a faster convergence rate and greater clustering performance; (4) How to define a new and improved CVI for the lower computational complexity, which no longer depends on cluster centers entirely; and (5) How to make reliable analysis and predictions from these partitioned clustering results, which can provide evidence for managers to customize workload and assess design performance for different designers accordingly. In the end, the in-depth analysis of the clustering results owns the potential to significantly distinguish the design efficiency at different time periods into the three different levels (i.e., high, medium, and low), which presents a unique opportunity in understanding and evaluating design performance objectively. Moreover, the proposed method in this chapter can serve as a powerful decision-making tool for managers to arrange schedules and workload reasonably towards a more effective and sustainable building design process.

The remainder of the paper is structured as follows: In Section 4.2, two hybrid clustering algorithms based on the interaction between the neural network and fuzzy logic, including EFKCN and AEFKCN, are presented with step-by-step procedures. They will be conducted to produce informative clusters of the designer's design efficiency for evaluating designer's performance and drawing up personalized work arrangements in the case study. Besides, a new CVI only associated with boundary points of a cluster is designed to reduce computational complexity. In Section 4.3, the EFKCN method is applied to cluster the real-world BIM design logs at individual and team views, and thus designer's efficiency can be automatically divided into the degree of high, medium, and low. In Section 4.4, the more advanced clustering algorithm named AEFKCN and the new CVI are tested by a series of experiments, where the novel AEFKCN with a modified learning rate can accelerate the convergence and the new CVI only associated with boundary points of a cluster can lower computational complexity. In section 4.5, conclusions are summaries.

4.2 Methodology

For the process of design behavior analysis and prediction, an EFKCN/AEFKCN-based clustering method is developed to explore the huge BIM design logs. A flowchart

of the proposed method is illustrated in Figure 4.1 to make it easy for practical applicability, containing three main stages: data preparation, EFKCN/AEFKCN clustering, and knowledge discovery. The relevant key concepts incorporated in the method are briefly presented below.

Figure 4.1. Flowchart of the proposed clustering method.

4.2.1 BIM log preprocessing

When *Autodesk Revit* software is employed as a model development tool, BIM design log data are generated automatically and saved in a considerably large number of *Revit* journal files. All the design events and designer-computer interactions are detailed in design logs, including the timestamp, designer, project, command, and others. From Figure 4.2, records in the log files are in a text format, which seems to be confusing. To make it well organized, a *Revit* journal file parser is utilized to retrieve useful information from the raw log data and store them into a CSV file. Table 4.1 lists the column name in the parsed CSV file and its relevant content from the first record in Figure 4.2. The prepared CSV is then fed into the clustering model for pattern discovery.

```

Tom 0022 2013-03-05 16:19:30 60.28 aboujaoudei.rvt 108 Level 02 north Create A line
\\Projects\185118.000_KSU_ENG_Phase4\DESIGN\BIM\REVIT\MODELS\KSU_ENGG_ARCH_INTERIOR.rvt

Tom 0022 2013-03-05 16:20:30 18.97 aboujaoudei.rvt 108 Level 02 north Other Command "AccelKey"
\\Projects\185118.000_KSU_ENG_Phase4\DESIGN\BIM\REVIT\MODELS\KSU_ENGG_ARCH_INTERIOR.rvt

Tom 0022 2013-03-05 16:20:55 12.477 aboujaoudei.rvt 108 Building section corridor Delete Basic wall
\\Projects\185118.000_KSU_ENG_Phase4\DESIGN\BIM\REVIT\MODELS\KSU_ENGG_ARCH_INTERIOR.rvt

```

Figure 4.2. Examples of three continuous records in BIM design log files.

Table 4.1. Column name and relevant content in the parsed CSV file.

Column Name	Examples of Column Content
User ID	Tom
Session	0022
Date	2013-03-05
Start Time	16:19:30
Duration	60.28
Project File Name	aboujaoudei.rvt
Project No.	108
View	Level 02 north
Journal Event	Create
Command	A line
File Path	\\Projects\185118.000_KSU_ENG_Phase4\DESIGN\BIM\REVI T\MODELS\KSU ENGG_ARCH_INTERIOR.rvt

4.2.2 Fuzzy Kohonen clustering

4.2.2.1 Preliminary

Various clustering methods are directly related to the quality of clustering results. In particular, Kohonen clustering network (KCN, also called self-organizing map SOM) and fuzzy C-means (FCM) are two significant clustering methods, which have been compared in (Mingoti and Lima 2006, Budayan, Dikmen et al. 2009). Indeed, no single method will typically outperform the other on different datasets (Kumar and Dhamija 2010). The KCN (Kohonen 1990) is fundamentally an unsupervised neural network with two layers of neurons, which has been developed into maturity in pattern extraction (Antonio, José D et al. 2008, Nohuddin, Coenen et al. 2012, Zhang, Chow et al. 2016). However, the KCN does not contain the optimized procedure and cannot guarantee a good convergence (Du 2010). Additionally, its results are sensitive to the number of clusters and initial parameters, including the learning rate, the neighborhood function, and the initialized weights (Su and Chang 2000). As for the FCM, it can assign a data point to more than one cluster under different probabilities, which stands out in fast convergence and high tolerance of ambiguity (Bezdek, Ehrlich et al. 1984). Due to such distinct advantages,

FCM has been combined with other concepts to obtain more desirable results for large data in multi-dimensional space and noisy environments (Zhang, Lu et al. 2016, Qian, Zhao et al. 2017).

In order to make the clustering results more satisfactory, it becomes a research focus on interfacing between neural networks and fuzzy clustering by incorporating fuzzy membership values into the learning rate in neural networks (De Almeida, De Souza et al. 2013). By merging KCN and FCM, a hybrid clustering method called fuzzy Kohonen clustering network (FKCN) is developed to inherit advantages from both KCN and FCM and make up for shortcomings of each method (Tsao, Bezdek et al. 1994). In other words, FKCN integrates the FCM into the learning rate and updating strategies of KCN. It should be noted that the superiority of FKCN is distinguished in three major ways: (1) It is capable of handling data with ambiguity and uncertainty; (2) It is not very susceptible to initial parameters; and (3) It can speed up the convergence rate with fewer training cycles. As reviewed, FKCN has been implemented well to process noisy data in real applications, such as in the field of image segmentation (Lu, Wei et al. 2009, Jabbar and Ahson 2010, Jabbar, Ahson et al. 2011), and automation control (Song and Huang 2004, Fan, Jia et al. 2013, Nurmaini, Tutuko et al. 2016).

4.2.2.2 EFKCN algorithm

It should be noted that FKCN has poor clustering performance in a tremendous volume of datasets, which is mostly caused by its learning rate in Eq. (4.1). Accordingly, the value of the learning rate α_{ij} for the winning neuron will increase to move weight vectors much closer to the winner, while the role of the non-winner neuron in weight updating will play smaller and smaller. Hence, it can be expected to limit the effect of low membership data in searching cluster centers, which can be realized through decreasing the learning rate of data with low membership value. From Eq. (4.1), the learning rate α_{ij} expressed in the form of $y = a^x$ has a decreasing property with x . That is to say, in order to diminish the impact of low membership data, the weight index $m(t)$ should be kept as small as possible to reduce its learning rate. However, this small $m(t)$ will simultaneously

generate a low learning rate in data with high membership, which will drive these important data away from cluster centers and slow down the convergence.

$$\alpha_{ij}(t) = (\mu_{ij}(t))^{m(t)} \quad (4.1)$$

where $m_t(t)$ is the weight index of the learning rate, and $\mu_{ij}(t)$ is the fuzzy membership value, which are defined in Eq. (4.2) and (4.3), respectively.

$$m_t(t) = m_0 - (m_0 - 1) \times \frac{t}{T_{max}} \quad (4.2)$$

$$\mu_{ij}(t) = \frac{1}{\sum_{k=1}^c \left(\frac{\|x_t - w_j\|}{\|x_t - w_k\|} \right)^{\frac{2}{m_t - 1}}} \quad (4.3)$$

where $m_0 > 1$ denotes the initial weight index, $t \in [0, T_{max}]$, and T_{max} represents the maximum number of iterations.

To alleviate the above-mentioned issue, a variation of FKCN named an efficient fuzzy Kohonen clustering network (EFKCN) algorithm is proposed by Yang et al to further reduce the computation of FKCN, which is adapted to extremely large datasets (Yang, Jia et al. 2008). To be more specific, EFKCN modifies the fuzzified learning rate of FKCN as presented in Eq. (4.4), which employs thresholds of membership value and fuzzy convergence operators. Based on three scenarios determined by the threshold of membership value, the optimal learning rate for high and low membership data can be calculated differently. In other words, the learning rate of data with high membership value can always increase to drive it closer to cluster centers continually. On the contrary, the low membership data can keep a relatively low learning rate with a small weight index, which can move away from centers in the end.

$$\alpha_{ij}(t) = \begin{cases} (\mu_{ij}(t))^{m_b}, & \mu_{ij}(t) > b \\ (\mu_{ij}(t))^{m(t)}, & a \leq \mu_{ij}(t) \leq b \\ (\mu_{ij}(t))^{m_a}, & \mu_{ij}(t) < a \end{cases} \quad (4.4)$$

where $a(a \in [0, 0.5])$ and $b(b \in (0.5, 1])$ are the lower and upper threshold of the membership value $\mu_{ij}(t)$, respectively. Two constants m_a and m_b ($m_a > m_b$) are two

fuzzy convergence operators. $m(t)$ is a time-varying weight index. Thus, the learning rate in the condition of $\mu_{ij}(t) \in [a, b]$ can be adjusted dynamically with time.

To sum up, the EFKCN algorithm is implemented with the following steps.

Step 1: The weight vector and fuzzy membership partition matrix are initialized. Parameters, including the number of clusters c , the weight index m_0 , the threshold of membership value a and b , fuzzy convergence operators m_a and m_b , the maximum iteration time T , and the minimum error threshold ε , are determined.

Step 2: The weight index of the learning rate is calculated using Eq. (4.2).

Step 3: The fuzzy membership value is updated by Eq. (4.3).

Step 4: The modified learning rate is determined by Eq. (4.4) according to three scenarios defined by the threshold of the membership value.

Step 5: All weight vectors are updated by Eq. (4.5).

$$w_i(t+1) = w_i(t) + \frac{\sum_{j=1}^n \alpha_{ij}(t)(x_j - w_i(t))}{\sum_{j=1}^n \alpha_{ij}(t)} \quad (4.5)$$

Step 6: The termination criteria is defined as $t > T$ or $\|w_i(t) - w_i(t-1)\| < \varepsilon$ in order to stop the iteration procedure.

4.2.2.3 Proposed AEFKCN algorithm

In EFKCN proposed by Yang, threshold values are introduced to distinguish the low and high membership values. Different constant values are set as weight indexes for conditions with low and high membership values. Nevertheless, there are two obvious weaknesses in EFKCN: (1) It will take a lot of time to determine the constants of weight indexes under several numerical experiments; and (2) The learning rate for low and high membership value data cannot be self-adaptive to iteration times, which tends to slow down the iteration procedure to some extent.

For these concerns, I develop a novel clustering algorithm named adaptive efficient fuzzy Kohonen clustering network (AEFKCN), which is a variation of EFKCN with three key components, including the fuzzy membership value in learning rates, the parallelism of FCM, and the updating strategy of KCN. Special attention should be paid on the

modified weight indexes of the learning rate (also called the fuzzy convergence operators), as shown in Eq. (4.6). That is to say, the weight index can be updated over time adaptively in accordance with three situations: (1) a membership value larger than the upper limit; (2) a membership value smaller than the lower limit; and (3) a membership value between the lower limit and upper limit. In turn, the learning rate α_{ij} closely related to the modified weight index $m(t)$ on each fuzzy membership value can also be adjusted adaptively.

$$m(t) = \begin{cases} Be^{-(m_0-1) \times \frac{t}{T_{max}}}, \mu_{ij} \geq b \\ m_0 - (m_0 - 1) \times \frac{t}{T_{max}}, a \leq \mu_{ij} \leq b \\ Ae^{-(m_0-1) \times \frac{t}{T_{max}}}, \mu_{ij} \leq a \end{cases} \quad (4.6)$$

where a ($a \in [0,0.5]$) is the lower limit of membership value, and b ($b \in (0.5,1]$) is the upper limit of membership value. A and B are two constants satisfying $Ae^{-(m_0-1) \times \frac{t}{T_{max}}} > m_0 - (m_0 - 1) \times \frac{t}{T_{max}} > 1 > Be^{-(m_0-1) \times \frac{t}{T_{max}}} > 0$.

The specific process of AEFKCN is outlined in Algorithm 1 below. It is clear that the small weight index will assign to a high membership value, aiming to make the learning rate fluctuate within a narrow range and accelerate convergence. Oppositely, a low membership value with a large weight index will play a minor role in the convergence of a network. In other words, data with low and high membership values will jointly update weight vectors. These different updating strategies for the weight index are helpful in improving the convergence speed globally. For one thing, low membership value data can be kept away from cluster centers. For another, high membership value data can be driven closer to cluster centers at a relatively fast pace.

Algorithm 1 AEFKCN

Input: data x_i , number of cluster prototypes c , initialized fuzzification parameter m_0 , minimum error threshold ε , maximum iteration T_{max} , lower and upper limit of fuzzy membership a and b , constant A and B

Output: fuzzy membership matrix U , weight vector W

1. Initialize randomly the weight vector $w_i(0) = (w_{i1}(0), w_{i2}(0), \dots, w_{ic}(0))$, and the fuzzy membership partition matrix $U(0)$.

2. For $t = 1, 2, \dots, T_{max}$:

2.1 Calculate the weight index of the learning rate by Eq. (4.2).

2.2 For $i = 1, 2, \dots, c, j = 1, 2, \dots, n$:

2.2.1 Calculate the fuzzy membership value μ_{ij} by Eq. (4.3).

2.2.2 Update the modified weight index of the learning rate (fuzzy convergence operators) $m(t)$ by Eq. (4.6).

2.2.3 Update the fuzzified learning rate $\alpha_{ij}(t)$ by Eq. (4.1).

2.2.4 Update the weight vectors $w_j(t)$ by Eq. (4.5).

2.2.5 If $\|w(t+1) - w(t)\|^2 < \varepsilon$ or $t > T_{max}$, then stop.

Else $t=t+1$, then return to 2.1.

End for

End for

4.2.3 Clustering performance analysis

4.2.3.1 Common clustering validity indexes

To assess the quality of clustering results, internal clustering validity indexes (CVIs), which only rely on data itself, are introduced as a measurement of the compactness within a cluster and separation between clusters. In the specification, compactness indicates how close data are concentrated in the same cluster, and separation means how far a cluster is away from one another. It is desirable to have a smaller within-class variance and greater inter-class distance. Besides, the most optimal number of clusters can be determined by maximizing/minimizing a certain CVI. In fact, no single CVI is superior to other CVIs in different datasets (Hämäläinen, Jauhiainen et al. 2017). In some complicated datasets, CVIs are prone to produce conflictive results (Qiu, Xu et al. 2016). Thus, it is of necessity to adopt more than one CVI to jointly assess the clustering performance. Arbelaiz et al. (2013) carried out an extensive comparative study of 30 CVIs in synthetic datasets, indicating that Silhouette index (SI), Calinski-Harabasz index (CHI), and Davies-Bouldin

index (DBI) were the three most recommended CVIs to achieve promising results in the experiments. For a comprehensive evaluation, we deploy these three common internal CVIs (SI, CHI, DBI) based on the internal criteria to compare the quality of clusters from different clustering algorithms. Besides, some CVIs associated with membership value, such as classification entropy (CE) and Xie and Beni's Index (XB), can effectively determine the optimum cluster number in the fuzzy clustering (Qiu, Xu et al. 2016). Herein, CE and XB can also be taken into account to detect the ideal cluster number. The five common CVIs used in this paper are presented as follows.

(1) Silhouette index (SI) (Rousseeuw and mathematics 1987) aims to quantify the ratio of the within-cluster cohesion to the cluster separation based on Eq. (4.7). A high value of SI closer to 1 will correspond to a well-defined partition.

$$SI(x) = \frac{b(x) - a(x)}{\max\{a(x), b(x)\}} \quad (4.7)$$

where $a(x)$ denotes the mean distance of data x_i to other points in the same cluster, and $b(x)$ represents the smallest average distance of data x_i to all points in each other cluster.

(2) Calinski-Harabasz index (CHI) (Caliński and Harabasz 1974) is the ratio of between-cluster variance and within-cluster variance, which is defined as Eq. (4.8). It will be better to obtain a higher value of CHI.

$$CHI(x) = \frac{\sum_{i=1}^c n_i \|v - v_i\|^2}{c-1} \times \frac{n-c}{\sum_{i=1}^c \sum_{x \in c_i} \|x - v_i\|^2} \quad (4.8)$$

where c is the number of clusters, c_i is the i th cluster, v is the overall mean of data points, v_i is the center of the i th cluster, n is the total number of data points, and n_i is the number of data points in the i th cluster. Particularly, $\sum_{i=1}^c n_i \|v - v_i\|^2$ stands for the overall between-cluster variance to measure the dissimilarity in different clusters, and $\sum_{i=1}^c \sum_{x \in c_i} \|x - v_i\|^2$ represents the overall within-cluster variance to demonstrate the dissimilarity in the same cluster.

(3) Davies-Bouldin index (DBI) (Davies, Bouldin et al. 1979) measures the ratio of the sum of within-cluster scatter to between-cluster separation, which is formulated in Eq.

(4.9). The value of DBI is expected to be smaller for better clustering results with the minimal within-class scatter and maximal between-cluster separation.

$$DBI(x) = \frac{1}{c} \sum_{i=1}^c \max_{j=1,2,\dots,c, i \neq j} \frac{diam(c_i) + diam(c_j)}{d(c_i, c_j)} \quad (4.9)$$

where c is the number of clusters, c_i and c_j represent the i th and j th cluster, respectively, and $d(c_i, c_j)$ denotes the distance of cluster centers in the i th and j th cluster. $diam(c_i)$ and $diam(c_j)$ are the diameter of the i th and j th cluster, respectively, which can be calculated by the distance between the data points and their corresponding cluster center in the same cluster.

(4) Classification entropy (CE) (Bezdek 2013) in Eq. (4.10) evaluates the fuzziness of the clustering partition. A smaller value of CE implies a more proper number of clusters.

$$CE(c) = -\frac{1}{n} \sum_{j=1}^c \sum_{i=1}^n \mu_{ij} \log(\mu_{ij}) \quad (4.10)$$

where μ_{ij} denotes the membership value of data point i in the cluster j .

(5) Xie and Beni's Index (XB) (Xie and Beni 1991) defines a ratio of intra-cluster compactness (the mean square distance between data and its related cluster center) to inter-cluster separation (the minimum squared distance between cluster centers), as expressed in Eq. (4.11). The optimal partition can be found with the smallest XB.

$$XB(c) = \frac{\sum_{j=1}^c \sum_{i=1}^n \mu_{ij}^m \|x_i - v_j\|^2}{n \min_{i,j} \|v_j - v_i\|^2} \quad (4.11)$$

where $i \neq j$.

4.2.3.2 A new cluster validity index

Clearly, five widely used indexes reported in Section 4.2.3.1 have their own limitations, which can be listed as: (1) These CVIs lack considerations of data size and distribution, which could be sensitive to arbitrary shapes of clusters (Song, Kim et al. 2018); (2) Since CHI, DBI, and XB are highly correlated with cluster centroids, they are

unable to ensure reliable evaluation in too-close centroid problems (Wu, Ouyang et al. 2015); (3) Although SI is irrelevant to cluster centers, all data points need to be involved in the calculation process to increase the computation cost inevitably. For the propose of both reducing the calculation complexity and assessing non-spherical clusters more efficiently, we consider developing an alternative CVI only relying on the extreme boundary of each cluster. Since the optimal clustering can be easily identified by the high closeness of data in the same cluster and great separation of data in different clusters, it suggests that our new index can also be defined based on two essential measures, namely intra-cluster property and inter-cluster distance. The new CVI is described as follows.

Take a dataset x with a set of n objects in a d -dimensional space as an example, which is given as:

$$x = \begin{matrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{matrix} \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1d} \\ x_{21} & x_{22} & \cdots & x_{2d} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{nd} \end{bmatrix} \quad (4.12)$$

If $x_{ij} = \max/\min\{x_{1j}, x_{2j}, \dots, x_{nj}\}$ ($j = 1, 2, \dots, d$) exists, x_i can be regarded as the data point in the extreme boundary. Let y with u objects be a new dataset to contain all boundary points in a cluster:

$$y = \begin{matrix} y_1 \\ y_2 \\ \vdots \\ y_u \end{matrix} \begin{bmatrix} y_{11} & y_{12} & \cdots & y_{1d} \\ y_{21} & y_{22} & \cdots & y_{2d} \\ \vdots & \vdots & \ddots & \vdots \\ y_{u1} & y_{u2} & \cdots & y_{ud} \end{bmatrix} \quad (4.13)$$

Moreover, when the dataset x is partitioned into c groups, the dataset z about the boundary points in c groups can be denoted as $\{z_1, z_2, \dots, z_c\}$. For the i th cluster prototype with d features, the boundary points can be expressed as:

$$z_i = \begin{matrix} y_{i1} \\ y_{i2} \\ \vdots \\ y_{i|C_i|} \end{matrix} \begin{bmatrix} y_{i1,1} & y_{i1,2} & \cdots & y_{i1,d} \\ y_{i2,1} & y_{i2,2} & \cdots & y_{i2,d} \\ \vdots & \vdots & \ddots & \vdots \\ y_{i|C_i|,1} & y_{i|C_i|,2} & \cdots & y_{i|C_i|,d} \end{bmatrix} \quad (4.14)$$

where $|C_i|$ denotes the number of data points in the i th cluster, and y_{ij} stands for the j th boundary point in the i th cluster.

(1) Compactness within a cluster

For intra cluster, the maximum distance between points can be just determined by boundary points. In general, data points that stay close together will result in a relatively small distance.

$$d_{max} = \max(\|y_p - y_q\|) = \max \sum_{k=1}^d w_k \sqrt{(y_{pk} - y_{qk})^2} \quad (4.15)$$

where w_k represents the weight for each dimension, which can measure the importance of data in each dimension, i.e.,

$$w_k = \frac{\sum_{i=1}^n x_{ik}}{\sum_{k=1}^d \sum_{i=1}^n x_{ik}} \quad (4.16)$$

where $\sum_{i=1}^n x_{ik}$ is the sum of value in the k th dimension, and $\sum_{k=1}^d \sum_{i=1}^n x_{ik}$ is the sum of all dimension values. In addition, the average distance between data points in the extreme boundary can be calculated by:

$$d_{avg} = \frac{\sum_{|C_i|(|C_i|-1)/2} \|y_p - y_q\|}{|C_i|(|C_i|-1)/2} = \frac{\sum_{|C_i|(|C_i|-1)/2} \sum_{k=1}^d w_k \sqrt{(y_{pk} - y_{qk})^2}}{|C_i|(|C_i|-1)/2} \quad (4.17)$$

where p and q are the p th and q th boundary points in the same cluster, respectively, and $|C_i|$ is the number of boundary points in the cluster i .

To quantify the compactness of data points within one cluster, we define a metric as $S_I = d_{max}/d_{avg}$. When $d_{max} \gg d_{avg}$, the value of S_I will become large, indicating an unbalanced distribution of data points. On the contrary, $S_I \rightarrow 1$ can be obtained in the condition $d_{max} \rightarrow d_{avg}$, which means that data points have high similarity. Observably, each cluster has its own value of S_I . To represent the overall intra-cluster property, it is reasonable to employ the maximum S_I as $(S_1)_{max} = \max_u S_1$. An ideal result of intra clustering will yield a small value approaching 1.

(2) Separation between clusters

The inter-cluster separation can be determined by the minimum distance between boundary points in pairs of clusters. The larger the minimum distance is, the more separate the two clusters are.

$$D_{min} = \min(\|y_{ip} - y_{jq}\|) = \min \sum_{k=1}^d w_k \sqrt{(y_{ip,k} - y_{jq,k})^2} \quad (4.18)$$

where $i, j = 1, 2, \dots, c, p = 1, 2, \dots, |C_i|, q = 1, 2, \dots, |C_j|$, c is the number of clusters, $|C_i|$ and $|C_j|$ are the number of boundary points in the i th and j th clusters, respectively. Also, the average distance between boundary points in two clusters i and j should be computed as:

$$D_{avg} = \frac{\sum_{|C_i| \times |C_j|} \|y_{ip} - y_{jq}\|}{|C_i| \times |C_j|} = \frac{\sum_{|C_i| \times |C_j|} \sum_{k=1}^d w_k \sqrt{(y_{ip,k} - y_{jq,k})^2}}{|C_i| \times |C_j|} \quad (4.19)$$

By dividing D_{avg} into D_{min} , a new metric $S_2 = D_{min}/D_{avg}$ can be obtained to quantify the degree of dispersion in different clusters. When D_{min} is close to D_{avg} , it can be concluded that two clusters are distinctly separated with $S_2 \rightarrow 1$. Similarly, I can gain various values of S_2 from different pairs of clusters. Aiming to assess isolation among clusters as a whole, the minimum value of S_2 represented as $(S_2)_{min} = \min_{c(c-1)/2} S_2$, is defined. It would seem that clustering results with a greater $(S_2)_{min}$ are better, implying a larger distance between clusters.

For comprehensively considering both the compactness and separation of clustering results, a new CVI termed S_{new} is designed with the combination of $(S_1)_{max}$ and $(S_2)_{min}$ mentioned above. In other words, S_{new} can perform as similar to other compactness-separation-based CVIs. The minimum value of S_{new} can indicate an optimal clustering result, since it is desirable to achieve the small within-cluster distance $(S_1)_{max}$ and large inter-cluster distance $(S_2)_{min}$ as much as possible.

$$S_{new} = (S_1)_{max} + (1 - (S_2)_{min}) \quad (4.20)$$

Moreover, the defined S_{new} is proven to reduce the computational complexity significantly. By assuming that a set of n input data in a d dimension space will be divided into c clusters, the data size in each cluster can be estimated as n/c averagely. The number of boundary points in each cluster is represented by $|C_i|$ ($|C_i| \leq 2d$), which can be approximated by $|C_i| \approx 2d$. The primary task is to search boundary points in all c clusters under $O(c \times n/c \times d) = O(nd)$. Then, computation in the term $(S_1)_{max}$ and $(S_2)_{min}$ will take

$O\left(c \times \frac{|C_i|(|C_i|-1)}{2} \times 2\right) = O(4cd^2 - 2cd) < O(4cd^2)$ and $O\left(|C_i| \times |C_i| \times \frac{c(c-1)}{2} \times 2\right) = O(4c^2d^2 - 4cd^2) < O(4c^2d^2)$, respectively. It should be noted that c and d can be regarded as constants since the condition $c \ll n$ and $d \ll n$ hold in general. That is to say, $O(4cd^2)$ and $O(4c^2d^2)$ need no consideration. In consequence, the computational complexity of our new CVI S_{new} is only $O(n)$, which has a linear relationship with the sample size. The computing complexity will reach $O(n^2)$ only under $d \approx n$, but it rarely occurs. Given classical CVIs based on all data points commonly in the complexity of $O(n^2)$ (Hämäläinen, Jauhiainen et al. 2017), it is clear that our new CVI S_{new} is less complicated with a lower $O(n)$.

4.3 Case study based on EFKCN

An illustrative application of EFKCN is provided in real BIM design logs from an international architecture design firm in a year span of 2013.10–2014.10, which has 853,520 records about 2,647 projects executed by 97 designers. A clustering algorithm EFKCN can be carried out from two aspects: individuals and teams, to provide new insights into the characteristics of design efficiency from log data. More specifically, the individual-level clustering can divide design behavior at different time into several clusters representing different design efficiency, while the team-level clustering can gather designers with similar design efficiency together. Therefore, it provides a valuable opportunity for managers to formulate reasonable design work arrangements, and make analyses and predictions about design performance in a data-driven manner.

4.3.1 Feature extraction

At the beginning stage, several false and useless records should be removed from the parsed CSV file, such as errors, null values, designers with less than 100 commands, and others. The process of data cleaning makes the searchable datasets more precise and meaningful. After data cleaning, only 53 designers are kept in the cleaned CSV file to perform modeling activities. In particular, Designer #1 executes the most commands (96,440), who is regarded as our research object in individual design behavior mining.

Useful features should be extracted from the cleaned CSV as the foundation of the clustering application, which could be varied in the individual dataset and the team dataset. In order to mine patterns of personal design behaviors, it is necessary to know the number of commands (x_3) and length of activation time in seconds (x_4) in each hour (x_2) at different day (x_1) about a certain designer. These types of information can be acquired from four columns, which are “User ID”, “Date”, “Start Time”, and “Duration” in Table 4.1. Columns “Session”, “Date” and “Command” are utilized to examine the similarity of design efficiency among designers, indicating that design efficiency will be evaluated in terms of finished sessions (x_5), activation days number (x_6), and executed command number (x_7), respectively. In the process of transforming the text into numerical information, Monday to Sunday in the feature x_1 are represented by the index 1–7 as shown in Table 4.2. For the feature x_2 , the index 0–23 refers to a one-hour time slot. For example, the value 8 indicates the time interval of 8:00–9:00. Besides, the number in features x_3 – x_7 quantifies the number of commands, sessions, days, and length of times (s). Table 4.2 and Table 4.3 are composed of the descriptive statistics of each feature in datasets for individual-level and group-level clustering, respectively.

To sum up, as an objective evaluation of design performance, I assess the design efficiency of a designer mainly relying on the number of executed commands per hour. In some degree, it is similar to the measurement of design productivity, but it is not exactly the same. According to the definition of design productivity in the book (Duffy 2012), it can be understood as “the efficiency of production of a design solution, within a business context, that is effective to the overall requirement”. Zhang et al. (2018) measured the number of commands and patterns are that were executed during a certain period to measure design productivity for simplicity. In this case, although the number of finished can be measured directly from the BIM event log data, it is still insufficient to reflect the actual productivity. The reason is that to gain a more reasonable calculation and explanation of a designer’s productivity, it is necessary to take into account additional factors, such as the characteristics of design projects, the complexity of design tasks, and others. Therefore, I use a more rigorous expression called “measure design efficiency” instead of “measure design productivity” herein. I will focus more on construction

productivity management in the future study by preparing a more reliable database for rational use of productivity measurement.

Table 4.2. Detail of dataset for Design #1 targeted in the individual-level clustering.

Dataset Size	Features	Statistic Characteristics of Features		
		Range	Mean	Median
757	Day of the week (x_1)	[1, 7]	3.819	4
	Time slot (x_2)	[0, 23]	15.151	15
	Number of commands (x_3)	[2, 600]	127.398	93
	Length of activation time (x_4)	[0.117s, 3597.686s]	2161.079s	2533.597s

Table 4.3. Detail of dataset for the design team targeted in the team-level clustering.

Dataset Size	Features	Statistic Characteristics of Features		
		Range	Mean	Median
53	Number of session (x_5)	[1, 157]	41.642	25
	Number of activation days (x_6)	[1, 137]	31.132	23
	Number of commands (x_7)	[147, 117,999]	15,768.755	7,529

4.3.2 Individual-level clustering

4.3.2.1 Dataset partitioning

At the level of the individual designer, the dataset about Designer #1 summarized in Table 4.2 is considered as an example. Since the quality of clustering results greatly depends on the initial value of parameters mentioned in step 1 of the EFKCN algorithm in Section 4.2.2.2, these parameters can be determined based upon several experiments, which repeat the EFKCN algorithm under different parameter values each time. By a brief comparison of the CHI value from Eq. (4.8) among these experiments, the better cluster can be easily recognized by the largest CHI. Accordingly, a set of rational parameters in this case can be defined as: $c=3$, $m_0=2.5$, $a=0.1$, $b=0.9$, $m_a=6$, $m_b=0.1$, $T=1000$, $\varepsilon=0.001$. Following the iteration process in the EFKCN algorithm, all 757 data points can be finally assigned to three clusters. To visualize the high-dimensional data in the three-dimensional (3D) space, the principal component analysis (PCA) (Abdi and Williams 2010) is run for

dimensionality reduction, which projects data into a new coordinate system with three principal components (PC1, PC2, and PC3). To be more specific, PC1, PC2, and PC3 are three main dimensions of variance measured by eigenvector and eigenvalue to hold most of the information in the dataset. For the first principle component (PC1), it contains the maximal possible information and can account for the most variance. From PC1 to PC3, the percentage of explained variance gradually decreases. As a result, Figure 4.3 provides a 3D scatterplot of PC1, PC2, and PC3, aiming to visualize the distribution of the clustering data with their corresponding cluster center. It is observed that three clusters represented by different colors and shapes are well separated, demonstrating the great capability of EFKCN in data partitioning.

To have an overview of the data points in three clusters, a graphical summary is provided in Figure 4.4 to illustrate the distribution of a single variable and the bivariate relationship in pairs of features, where three clusters are specified in red, green, and blue, respectively. In specificity, the histogram along the diagonal represents the distribution of the single feature itself, where the y-coordinate generally means the frequency counts. With regard to the scatter plots in the right upper corner, it emphasizes the relationship between two features in different clusters, which can be distinct from each cluster. To take two pairs of features: x_3 and x_1 , x_3 and x_2 , as an example, the scatter plot is able to determine the rank of the number of executed commands in each cluster as Cluster 1 > Cluster 2 > Cluster 3. In the same way, feature x_4 has similar characteristics as x_3 . It should be noted that the significant distinction within the three clusters is mainly due to feature x_3 and x_4 , which can be confirmed by the boxplot in Figure 4.5. Since both the number of commands and length of activation time gradually decrease from cluster 1 to cluster 3, the design efficiency level in cluster 1–3 can be simply evaluated as high, medium, and low, respectively. Additionally, the bivariate distributions can also be visualized by the 2D kernel density estimation (KDE) (Lampe and Hauser 2011) on the lower left triangle of Figure 4.4. In more specific terms, a sample elaboration of KDE is shown in Figure 4.6, which depicts a scatter plot and a contour plot of x_3 and x_4 along with the associated marginal distributions. The contour plot on behalf of the KDE is obtained from the summation of Gaussian kernels centered at each data point. That is to say, the KDE

approximates the probability density function (PDF) of the two variables by Gaussian kernels. From the KDE in Figure 4.4, there are obvious trends of clustering in pairs of x_1 and x_3 , x_1 and x_4 , x_2 and x_3 , x_2 and x_4 , x_3 and x_4 , which further testify the validity of the EFKCN clustering method.

Figure 4.3. Clustering results in 3D space.

Figure 4.4. Pair plots of four features in the dataset about Designer #1.

Figure 4.5. Boxplots of feature x_3 and x_4 .

Figure 4.6. An example of KDE for feature x_3 and x_4 .

4.3.2.2 Clustering results analysis

Observably, the EFKCN clustering algorithm has partitioned the dataset extracted from BIM design event logs into clusters 1–3 standing for the high, medium, and low level

of design efficiency, respectively. To facilitate a better understanding of the partitioned data, in-depth analysis is carried out in these three clusters from temporal perspectives, regression prediction, and comparison of different designers. Results are analyzed and discussed below, serving as quantitative evidence to assist managers in arranging design tasks in a more reasonable way.

(1) A valid cluster can be converted into a piece of personal design behavior. That is to say, a designer tends to exhibit different design efficiency at different time, according to hourly and daily data associated with features x_1 and x_2 . For Designer #1, the distribution shape of data in feature x_1 is depicted by a violin plot based on the KDE in Figure 4.7, where the wider part implies a higher frequency of the value. From the blue violin plot, it is more likely for Designer #1 to stay productive on Tuesday and Wednesday. On the other hand, the red plot indicates that the possibility of low design efficiency is quite high on Monday and Thursday. Thus, it is sound to allocate more tasks to Designer #1 on both Tuesday and Wednesday, whereas heavy tasks should be avoided on Monday and Thursday, if possible.

Additionally, if the manager is roughly aware of the working status of Designer #1, he can estimate the command number and working duration in a certain time period. The trend of design productivity can be discerned from Figure 4.8, which gives a general description of the data variation in feature x_3 and x_4 along with time for each cluster. For instance, when Designer #1 is considered to work overtime (19:00–2:00) with relatively high efficiency, the number of commands he will execute per hour is approximately 214.89–306.67 with the length of activation time in the range [3236.54s, 3584.10s]. Accordingly, proper workloads for Designer #1 could be set to take full advantage of his great working state. Besides, it is notable that the command number and activation time reach a high value during 14:00–16:00 in all three clusters, indicating that Designer #1 is prone to speed up his work at that time period. The minimum number of executed commands in clusters 1–3 will appear at 9:00–10:00 (60 commands), 12:00–13:00 (62.29 commands), 0:00–1:00 (8 commands), respectively, which means that working state of Designer #1 in each cluster tends to become inactive during the time period mentioned

above. In consequence, the analysis from the temporal perspective helps managers to conduct rational allocation of design tasks with less subjectivity and uncertainty.

(2) To examine the relationship between features x_3 and x_4 from the Designer #1 dataset, the regression analysis as a predictive technique is essentially conducted to quantify the design productivity in each cluster. From the data points in Figure 4.9, it seems to have a growing tendency in commands number (x_3) with the increase of activation time (x_4). Correspondingly, a linear equation $y = ax + b$ is adopted to fit data in clusters 1 and 2, while cluster 3 owns a non-linear relationship with an exponential fitting equation $y = ae^{bx}$. A 95% predictive interval (PI) accounting for uncertainty from both mean value and data scatter is also displayed in Figure 4.9, implying that the next observations are more likely to fall within the interval. Table 4.4 summarizes the fitting equation, p -value, and 95% confidence interval (CI) for parameters a and b in the fitting equation. Since the p -value of clusters 1–3 is all much less than 0.05, there is sufficient evidence to conclude the correlation of x_3 and x_4 formulated by the fitting function. Based upon the fitting functions and PIs, managers are able to make a rough estimate of the total number of executed commands in an hour quantificationally under three scenarios (clusters 1–3). For instance, if it is assumed that Designer #1 are in low efficiency working state (cluster 3) and his activation time will last only 600s in an hour, the command number can be calculated as $y = 4.958e^{0.002 \times 600} = 16.461$, which lies in the 95% PI [-4.890, 42.367]. Besides, the 95% CI can also be determined by the parameters a and b in Table 4.4, which is [12.862, 20.060]. In accordance with these more reasonable estimates about design productivity, a data-driven decision making can be therefore realized for managers to arrange justified design loads for each designer.

(3) Clustering results from the proposed EFKCN provide numerical evidence to reveal the distinctive characteristics of design behaviors from different designers. To make comparisons of different designers, another three datasets about Designers #2, #3, and #4 are extracted from BIM design logs in the size of 720, 383, and 271, respectively, which have the same features as Designer #1. Since the proper number of clusters depends on the dataset size, the dataset of Designer #4 will only be divided into two clusters: one for high efficiency and the other for low efficiency, in order to obtain optimal clustering

results. Table 4.5 lists the properties of clustering results for the datasets of Designers #1–#4 generated by EFKCN algorithm.

For instance, in regard to Designer #1, the high and medium efficiency most probably takes place in 14:00–17:00 and on Wednesday, and 17:00–20:00 on Thursday, respectively. Thus, it is better to keep Designer #1 working with heavier workloads during 14:00–20:00, especially on Wednesday and Thursday. In the meantime, the manager should try not to assign Designer #1 urgent tasks from 11:00 to 14:00 and on Monday. By contrasting the clustering results of Designer #1 with others, it is noticeable that Designer #1 has more records associated with the weekend (Saturday and Sunday) and evening time (17:00–2:00). In other words, Designer #1 is more used to working overtime under relatively high design efficiency, who can be the first choice to be arranged for more overtime work. For another, the records about the time slot 8:00–11:00 for Designer #1 is less than half of Designers #2–#4, indicating that Designer #1 is less active in the morning than Designers #2–#4. Thus, more morning's work ought to be assigned to Designers #2–#4. Moreover, Designer #2 can execute almost 1.5 times more commands within an hour in each cluster than Designers #1 and #3. Indeed, the length of activation time for Designers #1, #2, and #3 has no obvious difference. That is to say, Designer #2 tends to spend a similar length of time completing more commands than Designers #1 and #3. Under the condition that the due date of a design task is approaching, it is, therefore, a sensible arrangement to allocate this kind of urgent task to Designer #2.

Figure 4.7. Violin plots of feature x_1 .

Figure 4.8. Variation with time about (a) Number of commands (x_3); (b) Length of activation time (x_4).

Figure 4.9. Regression analysis about x_4 and x_3 in: (a) Cluster 1; (b) Cluster 2; (c) Cluster 3.

Table 4.4. Results of regression analysis in cluster 1–3.

Item	Cluster 1	Cluster 2	Cluster 3
Fitting Equation	$y = 0.116x - 174.280$	$y = 0.040x + 6.826$	$y = 4.958e^{0.002x}$
p-value	2.160×10^{-5}	1.500×10^{-4}	5.380×10^{-113}
95% CI for constant a	[0.057, 0.176]	[0.017, 0.062]	[3.874, 6.042]
95% CI for constant b	[-376.956, 28.406]	[-42.995, 56.648]	[0.002, 0.002]

Table 4.5. Clustering results and characteristics for datasets of Designer #1–#4.

Dataset	Character	Cluster 1	Cluster 2	Cluster 3
Designer #1 (size: 757)	Data Number	320	185	252
	x1	1 (22), 2 (56), 3	1 (32), 2 (16), 3	1 (48), 2 (39), 3
	(Frequency)	(64), 4 (46), 5	(34), 4 (38), 5	(41), 4 (47), 5 (28),
		(49), 6 (49), 7 (34)	(27), 6 (12), 7 (26)	6 (23), 7 (26)
	x2	8:00-11:00 (16)	8:00-11:00 (15)	8:00-11:00 (38)
	(Frequency)	11:00-14:00 (66)	11:00-14:00 (38)	11:00-14:00 (59)
		14:00-17:00 (83)	14:00-17:00 (46)	14:00-17:00 (48)
		17:00-20:00 (73)	17:00-20:00 (51)	17:00-20:00 (44)
		20:00-23:00 (53)	20:00-23:00 (24)	20:00-23:00 (41)
		23:00-2:00 (29)	23:00-2:00 (11)	23:00-2:00 (22)
	Average of x3	221.675	97.773	29.429
	Range of x3	[10, 600]	[3, 312]	[2, 198]
	Average of x4	3383.160	2231.587	557.468
	Range of x4	[2871.547, 3597.686]	[1513.2, 2880.564]	[0.117, 1500.297]
Designer #2 (size: 720)	Data Number	242	205	273
	x1	1 (50), 2 (51), 3	1 (30), 2 (42), 3	1 (48), 2 (55), 3
	(Frequency)	(49), 4 (42), 5	(51), 4 (42), 5	(56), 4 (50), 5 (58),
		(48), 6 (1), 7 (1)	(36), 6 (1), 7 (3)	6 (1), 7 (5)
	x2	8:00-11:00 (63)	8:00-11:00 (66)	8:00-11:00 (77)
	(Frequency)	11:00-14:00 (55)	11:00-14:00 (69)	11:00-14:00 (87)
		14:00-17:00 (122)	14:00-17:00 (65)	14:00-17:00 (66)
		17:00-20:00 (1)	17:00-20:00 (4)	17:00-20:00 (41)
		20:00-23:00 (1)	20:00-23:00 (1)	20:00-23:00 (2)
	Average of x3	317.001	143.810	44.703
	Range of x3	[8, 939]	[3, 809]	[2, 288]
	Average of x4	3298.982	2051.958	566.202
	Range of x4	[2697.360, 3599.623]	[1352.074, 2713.340]	[0.406, 3225.770]
Designer #3 (size: 383)	Data Number	135	103	144
	x1	1 (21), 2 (17), 3	1 (23), 2 (31), 3	1 (36), 2 (32), 3
	(Frequency)	(39), 4 (37), 5 (21)	(19), 4 (13), 5 (17)	(22), 4 (27), 5 (25),
				6 (1), 7 (1)
	x2	8:00-11:00 (30)	8:00-11:00 (39)	8:00-11:00 (42)
	(Frequency)	11:00-14:00 (37)	11:00-14:00 (20)	11:00-14:00 (42)

Dataset	Character	Cluster 1	Cluster 2	Cluster 3
		14:00-17:00 (50)	14:00-17:00 (30)	14:00-17:00 (37)
		17:00-20:00 (10)	17:00-20:00 (12)	17:00-20:00 (18)
		20:00-23:00 (7)	20:00-23:00 (2)	20:00-23:00 (5)
	Average of x3	209.081	105.699	26.229
	Range of x3	[14, 556]	[3, 377]	[2, 249]
	Average of x4	3253.000	2129.475	484.787
	Range of x4	[2767.693, 3595.867]	[1412.867, 2765.193]	[7.77, 1412.783]
Designer #4 (size: 271)	Data Number	177	94	—
	x1 (Frequency)	1 (20), 2 (35), 3 (41), 4 (48), 5 (32), 6 (1)	1 (28), 2 (14), 3 (22), 4 (15), 5 (14), 6 (1)	—
	x2 (Frequency)	8:00-11:00 (44) 11:00-14:00 (41) 14:00-17:00 (67) 17:00-20:00 (23) 20:00-23:00 (2)	8:00-11:00 (33) 11:00-14:00 (28) 14:00-17:00 (13) 17:00-20:00 (18) 20:00-23:00 (2)	—
	Average of x3	198	80.840	—
	Range of x3	[7,528]	[2, 304]	—
	Average of x4	3259.036	1187.094	—
	Range of x4	[2381.957, 3594.333]	[8.15, 2264.24]	—

Note: “—” refers to “Not Applicable”, as there are only clusters 1 and 2 for Designer #4. The value in bold indicates the maximum frequency.

4.3.3 Team-level clustering

From a team-level clustering, a dataset as illustrated in Table 4.3, which is about to modeling events conducted by a team of 53 designers. After feeding this dataset into the EFKCN clustering algorithm, 53 designers will be assigned to three clusters representing different levels of design productivity by a certain degree. To be more precise, a higher membership value indicates a stronger association between the data point and the cluster center. Since results from EFKCN are in the form of probability as seen in Figure 4.10, the largest probability helps in identifying the certain cluster which the data point is more likely to belong to. For instance, it can be seen in Figure 4.10 (b) that the length of the green bar representing cluster 2 is longer than clusters 1 and 3, which indicates that all the data points (Designers #10, #15, #21, #22, #28, #33, #38, #49, and #53) pertaining to

cluster 2 have the highest membership value in cluster 2 than others. Table 4.6 presents the clustering results and their characteristics, which can be analyzed as follows.

(1) Feature x_5 , x_6 , and x_7 are all significantly different among the three groups, enabling to jointly determine three clusters on behalf of high, medium, and low design productivity. Great concern can be focused on the cluster center due to its ability to represent the points grouped in the cluster and their numerical features. Known from Table 4.6, cluster centers are expressed by three numbers representing x_5 , x_6 , and x_7 , all of which reduce gradually from cluster 1 to 3. For instance, the center of session number in cluster 1 is at a value of 146.774, which is more than twice as that in cluster 2 and 11 times than cluster 3. Based upon the cluster center, the level of design efficiency can be preliminarily determined. To further validate the evaluation, statistic characteristics and data scatter of x_5 , x_6 , and x_7 are visualized in the boxplots of Figure 4.11. , which own an obvious downtrend from cluster 1 to 3. Thus, the design efficiency in clusters 1–3 can be reasonably deemed as high, medium, and low, respectively.

(2) The results from the group-level clustering can assist in recognizing groups of designers who own high, medium, and low design efficiency. Based on the y-coordinate in Figure 4.10. containing information of designer number, it can be known that Designer #1, #2, #3, #4, #9, #18, #24, #32, #40, #45, and #52 keep productive during 2013.10 – 2014.10, and thereby, managers can decide to give more rewards to them as incentives. Except as the reference for reward allocation, these 9 designers can be the best choice to handle urgent and heavy design tasks. As for the 33 designers in Figure 4.10. (c) who are inefficient during the modeling procedure, managers can help to find out the cause of the low efficiency, in order to improve their design efficiency. In addition, if there are records from new designers, they can also be put into the clustering model for design efficiency assessment. Once the efficiency level is determined, a general idea about the number of design sessions, activation days, and commands for the designers could be derived from the range of the three features in Table 4.6.

(3) Designers, who are grouped into the high efficiency cluster by the team-level clustering, will have more personal design behavior at high and medium efficiency levels.

For instance, the team-level clustering turns out that Designer #1–#4 are highly productive. From the clustering results of Designers #1–#3 in Table 4.5, a total number of records in cluster 1 (high efficiency) and 2 (medium efficiency) accounts for more than two-thirds of the total data points. For the Designer #4 dataset partitioned into two clusters in Table 4.5, around 65% data fall in cluster 1 (high efficiency). Similarly, it is more likely for designers in low efficiency groups to execute more commands under low efficiency. A new dataset of Designer #5 with a size of 10 is taken as an example. To conduct an individual-level clustering using this dataset, there are 7 records belonging to the low-efficiency group, in which it takes about 786.481s to perform 13 commands averagely. The rest 3 records are in another cluster denoting relatively high efficiency, which has the average value of commands and activation time 99 and 3345.118s, respectively. That is to say, 70% records of Designer #5 personal design behavior are carried out in low efficiency from the individual-level clustering. In fact, Designer #5 is grouped into the low-efficiency cluster with a high probability of 93.51% based on the team-level clustering. Thus, there is some consistency of clustering results between the individual-level clustering with the group-level clustering.

Figure 4.10. Membership value for data in: (a) Cluster 1; (b) Cluster 2; (c) Cluster 3.

Figure 4.11. Boxplots and data scatter of feature: (a) Number of sessions (x_5); (b) Number of activation days (x_6); (c) Number of commands (x_7).

Table 4.6. Clustering results and characteristics for the team-level dataset.

Item	Cluster 1	Cluster 2	Cluster 3
Center	(146.774, 99.412, 96806.073)	(71.688, 50.865, 11151.237)	(12.825, 11.701, 450.865)
Number	11	9	33
Range of x_5	[66, 157]	[27, 92]	[1, 54]
Mean/ Medium of x_5	105.727/ 86	55.222/ 50	16.576/ 12
Range of x_6	[33, 137]	[19, 55]	[1, 67]
Mean/ Medium of x_6	72.273/ 68	38.556/ 39	15.394/ 12
Range of x_7	[21064, 117999]	[10761, 17732]	[147, 10314]
Mean/ Medium of x_7	54088.091/ 38966	15569.444/ 15580	3050/ 1381

4.4 Case study based on AEFKCN

4.4.1 Experiment setup

To check the generalization performance of the proposed AEFKCN algorithm, an experimental dataset about the design behavior of Designer #2 is taken as an example with 720 data objects. After log parsing and data cleaning, four main features summarized in Table 4.7, whose meaning is similar as Table 4.2, can be obtained to directly reflect the designers' engagement in the modeling process. This processed dataset in the dimension of 720×4 is then fed into different types of clustering models for making comparative

experiments, including KCN, FCM, FKCN, EFKCN, and AEFKCN. Eventually, latent patterns and valuable knowledge about personal design behavior can be retrieved for design performance assessment. The initialized parameters of five algorithms are listed in Table 4.8. In particular, common parameters, like the number of clusters, maximum iterations, fuzziness index, minimum error thresholds, and others, are set to the same value for a fair comparison. Since each test is likely to produce different results, all algorithms will run 20 times repeatedly to reduce the uncertainty. All experiments are coded by Python 3.6 and run on a computer with 16.0GB RAM and Intel(R) Xeon(R) W-2123 CPU @3.60GHz.

Table 4.7. Description of dataset for Designer #2 (720 data points).

Statistic Characteristics	Four Features			
	Day of week (x_1)	Time slot (x_2)	Number of executed commands (x_3)	Activation time (x_4)
Minimum	1	7	2	0.406 seconds
Maximum	7	23	939	3599.623 seconds
Average	3.082	12.618	164.218	1902.048 seconds
Median	3	13	95	1939.774 seconds

Table 4.8. Parameters setting in five methods.

Algorithm	Parameters
KCN	$c=3, T=1000$
FCM	$c=3, T=1000, m=2.5, \delta=0.001$
FKCN	$c=3, T=1000, m=2.5, \delta=0.001$
EFKCN	$c=3, T=1000, m=2.5, \delta=0.001, a=0.9, b=0.1, ma=0.1, mb=6$
AEFKCN	$c=3, T=1000, m=2.5, \delta=0.001, a=0.9, b=0.1, A=0.1, B=6$

4.4.2 Comparison of results from different clustering algorithms

To understand the superiority of the proposed AEFKCN, its clustering performance is compared with other candidate algorithms, including KCN, FCM, FKCN, and EFKCN, mainly regarding the computation efficiency, partitions, and their quality. Comparisons of experimental results based on different clustering algorithms are summarized as follows.

- (1) AEFKCN is able to efficiently reduce iterations, leading to less running time than

the other four algorithms.

Table 4.9 shows that KCN computes the most slowly, since it will continue the clustering process until the predefined maximum iteration T_{max} is reached. The computational cost of FCM, FKCN, EFKCN, and AEFKCN is extremely smaller than KCN with a descending order $FCM > FKCN > EFKCN > AEFKCN$. In contrast to FCM, AEFKCN can reduce iterations by over 40% and cut short the running time from 5.883s to 4.233s. Evidently, EFKCN and AEFKCN can both converge at a faster speed than others. That is because they can update the learning rate by the threshold of the membership value, which offers the promise of driving weights in the network near cluster centers quickly. Moreover, AEFKCN can further increase the efficiency using its adaptive weight index of the learning rate for neural network updating.

(2) Clustering results obtained from AEFKCN have a high degree of similarity with three previous clustering models FCM, FKCN, and EFKCN, which can preliminarily confirm the reliability of the developed algorithm. With the help of PCA, Figure 4.12 visualizes the distribution of clustering data based on five candidate algorithms in a two-dimensional (2D) space. From Figure 4.12 (b)–(e) associated with FCM, FKCN, EFKCN, and AEFKCN, it is hard to observe differences in clusters directly, indicating the great consistency of results among these four methods. However, KCN in Figure 4.12 (a) partitions the dataset differently from others, which tends to assign fewer data points in cluster 2 and put cluster centroids at different locations. For checking the clusters in a quantitative angle, Figure 4.13 (a)–(d) employs a 3×3 confusion matrix to contrast the number of data points in pairs of the clustering algorithms. Value along the main diagonal implies the number of data points gathered in the same cluster by the two different methods. In a comparison of KCN and AEFKCN, only 78.47% (565 out of 720) of data points are grouped into the same cluster. Their significant discrepancy lies in cluster 2, where KCN only allocates 50 data points accounting for a quarter of AEFKCN (205). By contrast, FCM, FKCN, and EFKCN are more likely to produce similar clusters as AEFKCN under a relatively high probability of 96.1% (692 out of 720), 97.92% (705 out of 720), and 99.86% (719 out of 720), respectively.

(3) The quality of clustering results generated by FCM, EFKCN, and AEFKCN is similar, which further validates the effectiveness of the proposed algorithm. Figure 4.14 adopts three common CVIs (SI, CHI, DBI) to measure the average clustering performance of five algorithms under 20 experiments, where error bars are also provided to stand for the standard deviation of CVIs. Since the worse value of SI, CHI, DBI indicates large intra-cluster distance and low inter-cluster distance, KCN and FKCN can be recognized as the poorest solution in this case. Among the remaining three algorithms, the rank of the SI, CHI, and DBI values can be sorted as: FCM (0.609) > AEFKCN (0.601) > EFKCN (0.593), FCM (3443.292) > AEFKCN (3297.937) > EFKCN (3162.415), FCM (0.512) < AEFKCN (0.525) < EFKCN (0.536), respectively. In other words, FCM behaves a little better than AEFKCN in the extracted log dataset, and AEFKCN can improve EFKCN slightly. Regarding the error bars, it can be concluded that KCN is less stable than others. On the contrary, results of FCM remain the same in the repeated experiments, resulting in a zero-standard deviation.

(4) Evaluation based on the self-defined new index S_{new} shows that AEFKCN can return an excellent clustering structure, which will enlarge the distance between different clusters as far as possible. Apart from common CVIs, the new index S_{new} relying on the data points in the extreme boundary is also calculated to compare results from candidate algorithms, as summarized in Table 4.10. Since the larger S_{new} indicates the poorer clustering results, it is found that KCN is still regarded as the worst algorithm. Nevertheless, FCM is no longer the most desirable clustering algorithm due to the very small separation expressed by $(S_2)_{min}$. Although FCM can make data within clusters 4.59% more concentrated than AEFKCN in terms of $(S_1)_{max}$, it should be emphasized that the value of $(S_2)_{min}$ from AEFKCN is almost three times greater than FCM. That is to say, an outstanding advantage of AEFKCN is to drive dissimilar data points apart from each other, which plays a crucial role in dropping S_{new} from 2.610 (FCM) to the minimum of 2.597 (AEFKCN).

Figure 4.12. Visualization of clustering results by (1) KCN; (2) FCM; (3) FKCN; (4) EFKCN; (5) AEFKCN.

Figure 4.13. Comparison of clustering results in the pair of (1) KCN-AEFKCN; (2) FCM-AEFKCN; (3) FKCN-AEFKCN; (4) EFKCN-AEFKCN.

Figure 4.14. Evaluation of clustering results by three CVIs: (1) SI; (2) CHI; (3) DBI.

Table 4.9. Computational cost of five methods.

Algorithm	Average Time (seconds)	Average Iterations (times)
KCN	25.585	1000
FCM	5.883	37.85
FKCN	4.637	27.40
EFKCN	4.293	25.80
AEFKCN	4.233	22.15

Table 4.10. Clustering evaluation from new index.

Algorithm	$(S_1)_{max}$	$(S_2)_{min}$	S_{new}
KCN	3.122	0.026	4.096
FCM	1.656	0.046	2.610
FKCN	1.724	0.068	2.656
EFKCN	1.763	0.128	2.635
AEFKCN	1.732	0.135	2.597

4.4.3 Knowledge discovery from AEFKCN-based log mining

In this experiment of BIM design event log mining, the proposed AEFKCN algorithm is carried out to generate relevant clusters about design behavior, greatly contributing to the informed decision making in work arrangement and process optimization. Since fuzzy clustering is very sensitive to the number of clusters, an appropriate number of clusters should be determined in the first place to ensure that the fuzzy partitions can best fit the given data. That is to say, repetitions of the clustering process are conducted several times by changing the number of clusters different c ($c=2, 3, \dots, c_{max}$). Particularly, four benchmark CVIs (CE, XB, CHI, and DBI) are deployed herein to jointly determine the cluster number. Figure 4.15 depicts the variation of each CVI under cluster number c from 2 to 9. Obviously, CHI and DBI can reach the minimum to obtain a good partition when the number of clusters is set to 3. The value of CE descends abruptly at $c=3$, and then it tends to be stable. However, XB generates inconsistent results due to data complexity, giving the evidence that $c=2$ can lead to the better performance. Based on an overall consideration, $c=3$ is preferable to be the optimal cluster number. As a result, the data distribution of three clusters from the AEFKCN algorithm is visualized in a 3D space of Figure 4.16. Additionally, AEFKCN can produce a kind of specific

knowledge in membership value to quantify the probability of a data point belonging to a certain cluster category, which is presented by different colors and shapes in Figure 4.17. The cluster category can be therefore decided according to the highest membership value. For instance, blue circles representing cluster 1 are located at the top of Figure 4.17 (a), and thus, all data points in Figure 4.17 (a) can be grouped into cluster 1. Of particular note is that each cluster has its own distinct characteristics about design behavior, which deserves in-depth exploration as follows.

(1) The proposed clustering approach turns out to be an efficient tool to make a quick judgment about the design efficiency of a designer. In the light of information associated with features x_3 and x_4 in Table 4.11, the design efficiency in clusters 1–3 for Design #2 can be rated as three levels: high, medium, and low, respectively. For a detailed explanation of feature x_3 , cluster 1 can execute more than twice of commands than cluster 2, and command number in cluster 3 is decreased by around 68.95% against cluster 2. As for feature x_4 , its maximum value cannot exceed 3600 seconds (an hour). Since the average activation time in cluster 1 lasts for 3298.982 seconds, it can be inferred that Designer #2 keeps working through the whole time slot in x_2 almost without a break. Rather, cluster 3 only spends 566.202 seconds in modeling during time slots shown in x_2 , implying that more than 50 minutes within an hour is useless.

(2) Since designers' efficiency is highly relevant to work time, the temporal information (x_1 , x_2) in clustering results can guide managers to assign different workloads to designers at the appropriate time periods. In general, managers assess design efficiency and develop work plans based on their experiences, knowledge, and communications with designers, which could be subjective and unreasonable. To alleviate these weaknesses, historical records of design event logs can be deeply explored to reach its potential in hidden knowledge discovery, in order to assist managers to formulate more appropriate work arrangements in an objective manner. It is observed in Table 4.11 that Designer #2 tends to keep highly active during the time slot 14:00–17:00 and on Monday or Tuesday. Similarly, the medium design efficiency is more likely to occur in 11:00–14:00 and on Wednesday. Therefore, one of the possible recommendations is to allocate more tasks to Designer #2 in a period of time 11:00–17:00 from Monday to Wednesday. Besides, if

Designer #2 is identified in poor working conditions, the design manager should try to avoid arranging him to work from 11:00 to 14:00, especially on Friday. Moreover, the frequency of 17:00–20:00 in cluster 3 significantly outnumbers clusters 1 and 2, implying that Designer #2 is unable to concentrate entirely on design tasks in the evening. Thus, the design manager ought to take full advantage of Designer #2's daytime working hours, rather than making him work overtime. To sum up, a significant advantage of the clustering-based approach is to offer new insights into designers' behavior, which helps to create suggestions quickly and objectively according to the clustering result itself. However, a problem remains that this kind of recommendation takes no account of external factors, such as short meetings, phone calls, others. In the meantime, it could lack an in-depth understating of why the designer is active or less active. Therefore, such a recommendation is impossible to always be consistent with the actual situation, which can only serve as a supplementary of the expert judgment and assessment for reference in this case. For the purpose of drawing up a more convincing arrangement, comprehensive consideration of supervisory evaluations, clustering results, and important additional factors is suggested to reduce the bias and unreliability as far as possible, which will be a part of my future work.

(3) Differences of the executed command number (x_3) and activation time (x_4) within clusters 1, 2, and 3 are statistically significant, verifying the practicability of the proposed AEFKCN-based log mining in design efficiency assessment. For a more intuitive understanding of changes in design efficiency, Figure 4.18 describes features x_3 and x_4 in box plots along with scatters. It is observed that the mean, median, maximum, and minimum of x_3 (or x_4) from clusters 1 to cluster 3 descend gradually. In addition, a non-parametrical test called the Mann–Whitney U Test (also known as the Wilcoxon rank-sum test) (Weiner and Craighead 2010) is applied to examine differences in independent groups from a statistical perspective with no assumption of data distribution. From Table 4.12, the null hypothesis that data in clusters has no difference is rejected, since the p -value ($<2.2 \times 10^{-16}$) is far less than the level of significance $\alpha = 0.05$. Also, the range of Wilcoxon test statistic W for x_3 and x_4 cannot be contained in the corresponding intervals of W tail extreme value, which further confirms that the Wilcoxon test gives evidence

against the null hypothesis. That is, characteristics of x_3 (or x_4) among clusters 1–3 differ significantly from each other.

Figure 4.15. CVI for each cluster number: (a) CE; (b) XB; (c) CHI; (d) DBI.

Figure 4.16. Data distribution of clustering results from AEFKCN.

Figure 4.17. Membership value in three clusters: (1) Cluster 1; (2) Cluster 2; (3) Cluster 3.

Figure 4.18. Boxplots and scatters in cluster 1-3 for feature: (a) Number of executed commands x_3 ; (b) Activation time x_4 .

Table 4.11. Cluster properties of dataset for Design #2.

Item	Cluster 1	Cluster 2	Cluster 3
Data Number	242	205	273
Center	(2,842, 12.909, 293.273, 2966.372)	(3.164, 12.421, 159.407, 2169.020)	(3.198, 12.534, 53.859, 719.143)
x_1 (Frequency)	1 (50), 2 (51), 3 (49), 4 (42), 5 (48), 6 (1), 7 (1)	1 (30), 2 (42), 3 (51), 4 (42), 5 (36), 6 (1), 7 (3)	1 (48), 2 (55), 3 (56), 4 (50), 5 (58), 6 (1), 7 (5)
x_2 (Frequency)	8:00-11:00 (63), 11:00-14:00 (55), 14:00-17:00 (122)	8:00-11:00 (66), 11:00-14:00 (69), 14:00-17:00 (65)	8:00-11:00 (77), 11:00-14:00 (87), 14:00-17:00 (66)

Item	Cluster 1	Cluster 2	Cluster 3
	17:00-20:00 (1)	17:00-20:00 (4)	17:00-20:00 (41)
	20:00-23:00 (1)	20:00-23:00 (1)	20:00-23:00 (2)
Average x_3	317.001	143.810	44.703
Range x_3	[8, 939]	[3, 809]	[2, 288]
Average x_4	3298.982	2051.958	566.202
Range x_4	[2697.360, 3599.623]	[1352.074, 2713.340]	[0.406, 3225.770]

Table 4.12. Results of the Mann-Whitney U Test.

Item	Cluster 1, 2	Cluster 2, 3	Cluster 3, 1
p -value for x_3 (or x_4)		$< 2.2 \times 10^{-16}$	
Range of W for x_3	[12398, 37212]	[11277, 44688]	[4580, 61486]
Range of W for x_4	[5, 49605]	[288, 55678]	[88, 65987]
Range of W tail value for x_3 (or x_4)	[22138, 27472]	[25054, 30911]	[17905, 22509]

4.4.4 Experiments in additional datasets

To further verify the effectiveness of the proposed AEFKCN algorithm, a series of experiments are repeated in three public datasets from the public UCI repository (Asuncion and Newman 2007), which can be download from <http://archive.ics.uci.edu/ml/index.php>. Specifically, the Iris dataset is the most popular. Wine dataset owns more attributions. Ionosphere dataset is in a large size with many features. Meanwhile, three more new datasets about Designers #1, #3, and #4 are also extracted from the real BIM design log file to test the AEFKCN algorithm and new CVI S_{new} in mining and evaluating design behavioral patterns. The datasets of Designers #1, #3, and #4 are in the size of 757, 383, and 271, respectively, owning the same four features as Designer #2. The dataset of Designer #4 will be divided into only two clusters due to its small size (271), while the number of clusters for Designers #1 and #3 is still predefined as 3. Other parameters of the five clustering methods equal to the value in Table 4.8. Several conclusions can be derived from additional experiments as follows.

(1) The proposed AEFKCN is proven to outperform KCN, FCM, FKCN, and EFKN in the three public datasets, as tabulated in Table 4.13. In regard to the computation efficiency, AEFKCN converges at the fastest speed than others. Since the

ground truth is available in these three datasets, accuracy can be calculated by dividing the total number of correctly clustered data by dataset size. AEFKCN in Iris and Wine datasets has the highest accuracy, which means it can assign the fewest data points into wrong groups. In the Ionosphere dataset, the difference in the number of errors among FCM, FKCN, EFKCN, and AEFKCN is only one, indicating that these four algorithms can demonstrate almost the same clustering performance under approximately 71% accuracy. Based on the three internal CVIs (SI, CHI, DBI), results from AEFKCN always have the best compactness and separation in terms of cluster structure.

(2) The AEFKCN-based BIM event log mining exhibits superiority in both efficiency and effectiveness. From Table 4.14, AEFKCN runs the most rapidly with the fewest iterations in the new datasets of Designers #1 and #3. Although FKCN applied in the dataset of Designer #4 spends the shortest computational time, three internal CVIs (SI, CHI, DBI) experimentally show that clustering results from FKCN are far worse than others. According to SI, CHI, and DBI, AEFKCN always provides the second-best clustering results, which is almost as good as FCM. The advantage of AEFKCN over FCM is its fast convergence rate. To be more precise, AEFKCN cuts down around 50% of iterations in FCM when the experiment is carried on datasets of Designers #1 and #4, and the iteration reduction also occurs in datasets of Designer #3 by nearly 40%. Based upon our new CVI S_{new} , AEFKCN can always give back the lowest value of S_{new} , meaning that it yields more reliable clustering results in the three newly retrieved datasets. This is largely because the clusters from AEFKCN are farther apart from each other, resulting in the bigger separation $(S_2)_{min}$.

Table 4.13. Clustering results in three datasets from UCI repository.

Dataset	Description	Method	Time	Iteration	Error	Acc	SC	CHI	DBI
Iris	Size: 150 points Number of features: 150 Number of clusters: 3	KCN	3.648	1000	47	0.687	0.509	293.270	0.882
		FCM	0.871	32	19	0.873	0.554	581.907	0.654
		FKCN	0.743	30	24	0.840	0.391	116.105	1.101
		EFKCN	0.480	14	16	0.893	0.583	679.751	0.582
		AEFKCN	0.324	12	15	0.900	0.585	682.527	0.580
Wine	Size: 178 points	KCN	3.687	1000	57	0.680	0.224	248.063	0.558
		FCM	1.533	47	55	0.691	0.565	556.073	0.541

Dataset	Description	Method	Time	Iteration	Error	Acc	SC	CHI	DBI
	Number of features: 13	FKCN	1.881	70	54	0.697	0.302	91.479	1.293
		EFKCN	0.490	17	53	0.702	0.567	559.8189	0.536
	Number of clusters: 3	AEFKCN	0.423	14	53	0.702	0.567	559.8189	0.536
Ionosp here	Size: 351 points	KCN	5.619	1000	117	0.667	0.420	271.610	0.944
	Number of features: 34	FCM	0.508	15	103	0.707	0.461	341.107	0.892
		FKCN	0.325	11	102	0.709	0.442	299.273	0.948
	Number of clusters: 2	EFKCN	0.258	8	103	0.707	0.521	475.026	0.724
		AEFKCN	0.250	8	103	0.707	0.521	475.026	0.724

Note: Acc is the abbreviation of accuracy

Table 4.14. Clustering results of three new datasets.

Dataset	Algorithm	Time	Iterations	SC	CHI	DBI	$(S_1)_{max}$	$(S_2)_{min}$	S_{new}
Designer #1	KCN	26.146	1000	0.367	1759.551	0.864	1.822	0.092	2.730
	FCM	7.737	48.600	0.649	4465.464	0.470	1.737	0.059	2.678
	FKCN	5.760	34.550	0.476	1548.221	0.839	1.824	0.057	2.767
	EFKCN	5.318	31.300	0.632	4086.010	0.491	1.833	0.132	2.701
	AEFKCN	4.884	25.350	0.639	4257.235	0.488	1.863	0.195	2.668
Designer #3	KCN	13.359	1000	0.373	933.427	0.536	1.742	0.129	2.613
	FCM	5.408	42.900	0.639	2196.558	0.486	1.650	0.052	2.598
	FKCN	3.535	33.400	0.477	668.531	0.864	1.712	0.057	2.655
	EFKCN	3.393	31.350	0.591	1768.470	0.527	1.666	0.087	2.578
	AEFKCN	3.317	29.850	0.600	1845.806	0.517	1.759	0.265	2.494
Designer #4	KCN	7.269	1000	0.687	972.909	0.451	1.614	0.223	2.391
	FCM	1.433	17.200	0.709	1117.003	0.414	1.607	0.190	2.418
	FKCN	0.886	10.500	0.534	382.212	0.693	1.531	0.153	2.379
	EFKCN	1.301	12.500	0.708	1103.773	0.425	1.560	0.152	2.408
	AEFKCN	1.263	11.450	0.708	1113.928	0.418	1.685	0.312	2.373

4.5 Chapter Summary

In this chapter, a clustering-based BIM design log mining method is proposed for exploring characteristics of design behavior and efficiency from both the individual and team levels. Due to no need for labels in the training sets, cluster analysis is a promising tool to deeply mine log data without too many manual interactions. The extracted clusters can easily distinguish different levels of design efficiency (i.e., high, medium, low), which can guide managers to objectively assess designers' performance and strategically schedule personalized work for different designers. As reviewed, no previous studies have

employed the unsupervised clustering methods into BIM event logs to explore design efficiency. Only Zhang et al. (Zhang, Wen et al. 2018) put efforts in measuring design productivity by retrieving the frequent design sequence patterns from BIM event logs and comparing them among different designers, which still rested on the perspective of statistics and was hard to deal with the growing amounts of data. To address the limitations of existing work, I develop a framework of clustering-based design efficiency exploration under a hybrid clustering algorithm, which can well handle data overload and diversity and present the opportunity for automating the design performance evaluation with less individual bias. In the end, new knowledge from the automatic analysis of the extracted clusters can support data-driven decision making in drawing up a rational and personalized work arrangement to smooth the design process.

For the purpose of verifying the effectiveness and applicability of the proposed method, two case studies are performed in real-world BIM design log files from an international architecture firm using the EFKCN and AEFKCN algorithm, respectively. To be more specific, the findings can act decision-making tool for managers to arrange schedules and workload from the following two perspectives: (1) From the individual-level clustering, the cluster analysis can significantly distinguish the design efficiency of an individual designer at different time periods into high, medium, and low level, which presents a unique opportunity in understanding and assessing design efficiency objectively. Accordingly, it paves a new way for managers to figure out the design preference and efficiency of different designers, and then assign proper design tasks to the right designers at different time. For instance, the clustering-based analysis reveals that Designer #1, in this case, is more used to working overtime than others, and thus a feasible suggestion only depending on the clustering results is to treat him as an optional person to do overtime duties. Conversely, another finding from clustering is that Designer #2 tends to keep low efficiency after 17:00, implying that a potential solution is to assign this designer more tasks in the day rather than in the evening. To some extent, it can be argued that the personal design behavior hidden in different clusters is useful for designers to assign design work to the right designers rationally during the particular time period. Notably, although these data-driven recommendations are straightforward, they can only

reflect the characteristics of the collected data itself but fail to consider the subjective and objective reasons. Exploring the reasons behind a designer's work performance is an indispensable step, which can assist managers to more properly adjust recommendations about staff arrangement. That is to say, the complex process of decision-making involving both the clustering results and subjective explanation is bound to generate suggestions that are more grounded in reality.

(2) From the team-level clustering, it aims to distinguish designers in different levels of design efficiency. As a result, three distinct clusters representing high, medium, and low efficiency can be easily obtained. That is to say, the efficiency of designers in the three clusters can be automatically evaluated as high, medium, and low, respectively. It should be noted that these three clusters are used for objectively uncovering characteristics of designers' behavior and making performance evaluations, but they do not stand for the actual work allocation. Several clusters are discovered based on the intrinsic interaction behind large data, and thus unlabeled data sharing high similarity will be gathered. Their practical value lies in providing evidence to guide managers in defining proper staffing strategies. For instance, nine designers (Designers #1, #2, #3, #4, #9, #18, #24, #32, #40, #45 and #52) in the cluster 1 are the most productive and skillful, who will complete more sessions and commands and work longer time than designers from clusters representing medium or low efficiency. Therefore, it is reasonable to assign these nine designers exhibiting high efficiency to different design teams. During the design process, they will take a leading role to lead other senior designers in the team. Additionally, an alternative plan is to arrange them to deal with some heavy and important tasks. In the contrast, there are totally 33 designers in the cluster 3 representing low design efficiency. It means that these unskilled designers may need more additional training and practice. Managers should try to avoid arranging them to handle urgent tasks.

Another important thing to be noted is that the hybrid clustering algorithm EFKCN and the proposed novel algorithm AEFKCN are proved to be outstanding in both computing efficiency and clustering performance. From the view of clustering quality, EFKCN and AEFKCN are almost as good as FCM based upon three CVIs (SI, CHI, and DBI), and are always better than KCN and FKCN. As for the self-defined CVI termed

S_{new} , EFKCN and AEFKCN can generate clusters with larger inter-cluster distance than other alternative algorithms. Moreover, EFKCN can take only 60% time and 70% iterations of FCM to achieve a similar clustering performance. AEFKCN can further accelerate the convergence and cut down iterations of EFKCN using its adaptive weight index of the learning rate for neural network updating, which is especially helpful in the dataset with a complex structure and large size.

CHAPTER 5. DISCOVERING COLLABORATIVE PATTERNS BY SOCIAL NETWORK ANALYSIS

5.1 Introduction

This chapter addresses the Research Objective 3 of this thesis. The specific objective is to develop the SNA-related project management by mapping collaboration from massive BIM design log data into the network topology, aiming to explain designers' behavior and interdependence within the collaborative organization. Its ultimate goal is to mathematically reveal valuable knowledge, such as the detected communities of designers, the importance of designers, and the transmission of information, which can offer easy references for increasing cooperation chances among designers. There are two critical steps in the proposed framework. The first one is to build a social network based upon useful information extracted from logs for the graphical description of the collaborative design process, where nodes are designers with professional skills engaging in the collaborative design and ties are their interactions for information and knowledge sharing. The next is to fully explore the established network for knowledge discovery, such as community detection, node importance measurement, link prediction, and others, which is expected to help managers draw up more reasonable work arrangements to optimize the BIM-based collaborative design task. To achieve the objective, the social network is built in both the static and dynamic view, as introduced below.

For the static network, it aims to discover potential communities of designers and investigate each community in terms of node importance measurement and link prediction once they are perceived. The point of focus lies in developing a novel algorithm combining the graph embedding and clustering to discover and investigate potential clusters of designers. There are three main research questions remaining to be resolved: (1) How to extract useful data from a large amount of disordered and text-format BIM event logs to model the information exchange and communication in the design collaboration; (2) How

to generate feature representations to well preserve network structure, which can be readily understood and learned by a certain clustering algorithm for community detection; (3) How to explore characteristics of each community quantitatively (i.e., centrality metrics, web-page ranking, Adamic/Adar, SimRank), including the individual' role, potential work transmission among designers, in order to strategically increase the chance of cooperation for higher design productivity.

For the dynamic network, it aims to break down the static network into several sub-networks in a timely manner to capture the variation of structural and behavioral characteristics over the course of design. Moreover, a special emphasis can be put on the evaluation and prediction of designers' influence, such as to define a reasonable self-defined metric for comparatively low computational cost and high accurate ranking, to implement the proper machine learning by learning features from both network structure and human behavior. The following three research questions are expected to be addressed: (1) How to build dynamic networks relying on the logs with the notion of time to represent information and knowledge sharing among designers during the collaborative design; (2) How to discover collaboration patterns in terms of the network structure and operational behavior; (3) How to realize a more reliable and satisfactory evaluation of designers' engagement and their contribution in the collaboration.

The remainder of this chapter is organized as follows. Section 5.2 presents the key methods for collaboration exploration based on SNA. Apart from the common metrics for node importance measurement and link prediction, special emphasis is put on three novel methods, which are the hybrid algorithm for community detection termed *node2vec-GMM*, a new-defined metric called “impact score” for influence measurement, an emerging machine learning algorithm named CatBoost for influence prediction. Subsequently, two case studies are performed in the real BIM design event logs to validate the effectiveness of the proposed SNA approaches in monitoring and optimizing the collaborative design process. Section 5.3 carries out the developed *node2vec-GMM* algorithm to discover three possible communities with closely linked designers. Analysis of each community is performed from node importance measurement and link prediction to identify information spreading and designers' roles within the community. Section 5.4

focuses on dynamic SNA, which breaks the constant network into twelve sub-networks to capture the variation of structural and behavioral characteristics over the course of design. Finally, Section 5.5 draws up the conclusions.

5.2 Methodology

The goal of this chapter is to deeply mine the massive BIM design event logs from a social collaboration perspective. Figure 5.1 illustrates the flowchart of the developed network-enabled BIM design event log mining. To be more specific, a huge amount of BIM design event logs will be generated automatically as the rich data source to construct the collaborative networks. In subsequence, the network is explored from two aspects. One is to implement the node clustering algorithm for detecting potential communities of designers within the complex network. The other is the dynamic network analysis to discover the variation of collaboration patterns and characteristics in the execution of the project. According to results from SNA, managers can draw up more reasonable work arrangements to facilitate cooperation and speed up the design procedure.

Figure 5.1. Framework of the network-enabled BIM design event log mining.

5.2.1 Network development

A collaborative network is primarily developed based upon useful information extracted from BIM event logs. The cooperative relationship can be determined when a designer contributes to a part of the design task and then passes the task to other designers. Taking a simple network in Figure 5.2 as an example, three designers (Designer #1-3) are involved in the collaborative design process and are represented by nodes. The directed edge indicates the propagation direction of the task, and the weighted value on the edge means the frequency of collaboration between two designers. For instance, design tasks will be passed from #1 to #2, from #3 to #1, and from #3 to #2. Designer #3 transmits design tasks to #2 10 times, and thus the directed edge from #3 to #2 owns the largest weight than the other two edges.

Figure 5.2. Example of a simple collaborative network.

5.2.2 Proposed algorithm for node clustering

5.2.2.1 Preliminary

One of the important research issues in SNA is the node clustering for community detection, which groups vertices with more densely connections together to uncover the intrinsic structure of complex social networks (Papadopoulos, Kompatsiaris et al. 2012). In essence, the node clustering can be accomplished by two parts as introduced below.

(1) **Network feature representation:** As a solution of learning features, graph embedding can map each node into a low-dimensional vector to preserve network structure and properties. The necessity of graph embedding comes from two aspects: one is the high computational and space cost in the direct analysis of complex networks, and the other is the very limited algorithms on graph analytics for nodes and edges. Early work

of graph embedding mainly focuses on dimensionality reduction, which maps inputs into a desired low-dimensional space, such as IsoMap (Tenenbaum, De Silva et al. 2000), Laplacian Eigenmaps with locality-preserving character (Belkin and Niyogi 2002), locally linear embedding (LLE) (Roweis and Saul 2000). However, these approaches largely depend on the leading eigenvectors from the adjacency matrix containing neighborhood information, which will undergo great time complexity and poor statistical performance in large and diverse graphs. To make the graph embedding method more suitable for large-scale graphs, an alternative named graph factorization is developed with the small time complexity $O(|E|d)$ (E and d are the number of edges and dimensions, respectively), which factorizes the adjacency matrix to approximate the node proximity in the lower dimensions (Ahmed, Shervashidze et al. 2013). Tang et al. (2015) proposed a large-scale network embedding model named LINE to describe a node pair by two joint probabilities, whose objective function was carefully defined in the sampling method.

Moreover, the recently developed graph embedding methods are inspired by random walks and skip-gram models from natural language processing (NLP). The random walk is a stochastic process of graph traversal by moving from one node to one of its connected nodes. Given nodes from random walks, the skip-gram model can maximize the probability of the nodes' neighborhood within a window size. DeepWalk (Perozzi, Al-Rfou et al. 2014) is the most widely used one to sample node sequences by a series of random walks and feed nodes into skip-gram for learning latent network feature representation. Although DeepWalk is proved to effectively represent scalable networks in low computational cost $O(|V|d)$ (V is the number of nodes), it has no specific winning sampling strategy. To obtain more informative and reliable embeddings, *node2vec* (Grover and Leskovec 2016), as an extension and modification of DeepWalk, samples neighborhoods of nodes by a flexible biased random walk with two more hyperparameters, which explores neighborhoods in both the breadth-first sampling and depth-first sampling way.

(2) **Clustering method:** When features are obtained from graph embedding, they are input into clustering algorithms to partition nodes into several groups depending on the topology characteristics, and thus nodes in each cluster are more likely to connect with

each other. Among various partitional clustering algorithms, K -means is the most well-known, interpretable, and fastest one to divide data into k number of clusters by minimizing Euclidean distance between data in the same cluster. It is a kind of hard assignment assuming that a data point only belongs to one cluster, which is unable to measure the uncertainty in slightly overlapped clusters. Besides, it can only represent clusters in circle and sphere, which is inflexible to tackle non-circle data. Noticeably, K -means can be seen as a special case of the Gaussian mixture model (GMM), and thus these two methods are often compared with each other (Musumeci, Rottondi et al. 2018, Wang, Da Cunha et al. 2019). It is demonstrated that GMM tends to be more appropriate than K -means to achieve greater clustering performance on account of its probabilistic model and the flexible mixture modeling. To be specific, GMM offers a measure of uncertainty in the soft assignment, which models input data by seeking a mixture of multi-dimensional Gaussian probability distributions and estimating relative parameters from maximizing the posterior probability in an expectation-maximum (EM) approach (Dempster, Laird et al. 1977). Since GMM is successful in speech and image recognition, it can also be supposed to learn network features from graph embedding for node clustering, and thereby fit and visualize identified clusters by a multivariate Gaussian distribution in an ellipse (Cavallari, Zheng et al. 2017).

5.2.2.2 *node2vec*-GMM algorithm

As a definition of SNA problem, a given network is expressed as $G = (V, E, W)$, where $V = \{v_1, v_2, \dots, v_n\}$ stands for a set of n vertices, $E = \{e_{ij}\}_{i,j=1}^n$ denotes a set of edges, and $W = \{w_{ij}\}_{i,j=1}^n$ is the weight of edges. If two vertices v_i and v_j are linked, the edge will own a weight w_{ij} in a range of $(0, 1)$; otherwise $w_{ij} = 0$. Motivated by the graph embedding, each vertex is represented in a low-dimensional space by a mapping function: $f: V \rightarrow R^d$, where f is the size of $|V| \times d$ and the dimension of feature representation d is much less than $|V|$ ($d \ll |V|$). To group vertices with tight connections together, a new node clustering algorithm for community detection is developed as outlined in Algorithm 5.1 with a hybrid of the *node2vec* graph embedding method (Grover and Leskovec 2016)

and the GMM clustering approach (Shental, Bar-Hillel et al. 2004). The ultimate objective function can be expressed as a summation in Eq. (5.1). For the *node2vec*, its objective function is modified into a more understandable form, which can be more accessible to the clustering model. For GMM, I also revise the objective function to make it more suitable to data from the network structure. In brief, the novelties of the method consist in its combination and modified objective functions. Apart from clusters of nodes, the proposed method can give back the feature representation of clusters termed the cluster embedding simultaneously.

$$L = L_1(\Phi) + L_2(\Pi, \Phi, \mu, \Sigma) \quad (5.1)$$

where $L_1(\Phi)$ and $L_2(\Pi, \Phi, \mu, \Sigma)$ are the modified objective functions for the *node2vec* and *GMM*, respectively, which are explained concretely below.

(1) **The *node2vec* graph embedding:** Firstly, the *node2vec* performs a biased random walk, also known as a neighborhood sampling method, to intelligently guide the walk direction in the process of sampling vertex sequences $\{v_1, v_2, \dots, v_L\}$ with a fixed length L , ensuring to better capture network structure. These generated vertex sequences are then learned for network feature representation based on a method inspired by the skip-gram architecture (Mikolov, Chen et al. 2013), which is the neural network model for searching the most related neighborhoods of a given word. That is to say, the *node2vec* in (Grover and Leskovec 2016) actually optimizes a log-probability objective function: $\max_f \sum_{v \in V} \log p(N_s(v) | f(v))$, where $N_s(v)$ is a set of all network neighborhoods of node v obtained from a biased random walk method S , and $f(v)$ is the feature representation of node v .

However, this abstract objective function is too hard to be understood and solved. Therefore, I reformulate the function as a loss function in Eq. (5.2) based on two standard assumptions (namely the conditional independence and symmetry in feature space) and a negative sampling strategy (Mikolov, Sutskever et al. 2013). This rewritten formula is more comprehensible and computationally easier, which is also convenient for the implementation of the GMM algorithm.

$$L_1 = - \left(\log \sigma(\phi_{ni}^T E_v^T v_i) \right) + \sum_{t=1}^K E_{v_n \sim P_n(v)} [\log \sigma(-\sigma(\phi_{ni}^T E_v^T v_i))] \quad (5.2)$$

where $\sigma(x) = (1 + e^{-x})^{-1}$ is a sigmoid function, v_i is the node in V ($v_i \in V$), K is the number of sampling nodes, $\phi_{ni} \in R^d$ is the node embedding for the node v_{ni} itself (v_{ni} is the neighborhood of node v_i), $\phi'_{ni} \in R^d$ is the representation of “context” of other nodes, and $E_{v_n \sim P_n(v)}$ indicates the samples follow a noise probability distribution $P_n(v)$ ($P_n(v)$ is empirically set as $P_n(v) \propto d_v^{0.75}$, and d_v is nodes’ out-degree). In addition, a unit set $\Phi = \phi_{ni} \cup \phi'_{ni}$ is utilized to simplify Eq. (5.2) into Eq. (5.3), which means that the feature representation of the network can be learned by minimizing Eq. (5.3) through the stochastic gradient descent (SGD) on a single hidden-layer feedforward neural network.

$$L_1 = -\sum_{v_{ni}, v_i \in V} \log \sigma(\Phi^T E_v^T v_i) \quad (5.3)$$

(2) **GMM clustering method:** Afterwards, the network features are fed into the probabilistic clustering method GMM, in order to divide nodes within a network into K clusters following a multivariate Gaussian distribution $N(\mu_k, \Sigma_k)$ with mean μ_k and covariance Σ_k . To learn the network embedding more effectively, I redefine the objective function of GMM in Eq. (5.4) incorporating the node embedding and a log-likelihood function, which is in the consistent format as Eq. (5.2). An iterative optimization technique named Expectation-Maximization (EM) (Dempster, Laird et al. 1977) will then be performed to minimize Eq. (5.4) to continually estimate μ_k , Σ_k , and Π_k until the equation goes converge.

$$L_2 = -\sum_{k=1}^K \Pi_{ik} N(\phi_i | \mu_k, \Sigma_k) \quad (5.4)$$

where $\phi_i \in R^d$ is the node embedding for the node $v_i \in V$, $\mu_k \in R^d$ denotes the mean vector, $\Sigma_k \in R^{d \times d}$ stands for the covariance matrix, and Π_{ik} represents a mixing coefficient for the k th distribution. It is clear that $\Pi_{ik} = p(c_i = k)$ indicates the probability of the node v_i in the k th cluster satisfying two constraints: $0 \leq \Pi_{ik} \leq 1$ and $\sum_{k=1}^K \Pi_{ik} = 1$. Remarkably, μ_k and Σ_k are perceived as the cluster embedding, which donate the feature representation of the k th clusters.

Additionally, it remains an important question to determine the appropriate number of the mixture components K in GMM. For this purpose, two information criterion tests, named Akaike Information Criteria (AIC) (Akaike 1998) and Bayesian Information Criteria (BIC) (Schwarz 1978), are commonly carried out especially for the GMM model

(Li, Prasad et al. 2011, Cao, Fu et al. 2015). They add a penalized term to the negative log-likelihood function to penalize the complex model, which can effectively avoid overfitting in the model. The lowest AIC/BIC leads to the most optimal model.

Algorithm 5.1: The node clustering algorithm *node2vec-GMM*

Input: $G = (V, E, W)$, Embedding dimension d , Walks per node r , Walk length l , Window size m , Return p , In-out q , clusters number K determined by AIC/BIC, Maximum iteration in GMM T , Expected mean value in GMM U .

Output: Graph embedding Φ , Community embedding μ and Σ , Probability of nodes in each community Π

```

1: For iteration =1 to  $r$  Do
2:   For all nodes  $v_i \in V$  Do
3:     Sample node sequences by a biased
       random walk ( $G, v_i, l$ ) introduced in (Grover and Leskovec 2016)
4:   End For
5: End For
6: Perform SGD on Eq. (3) ( $m, d$ , sampled node
   sequences) to obtain the graph embedding  $\Phi$ 
7: Initialize parameters  $\mu_k$ ,  $\Sigma_k$ , and  $\Pi_{ik}$  randomly
8: While  $t < T$  or  $|\mu_k - U| > \varepsilon$  Do
9:   Perform EM to maximize Eq. (4) and update
        $\mu_k$ ,  $\Sigma_k$ , and  $\Pi_{ik}$ 
10:   $t = t + 1$ 
11: End While

```

5.2.3 Network analysis

5.2.3.1 Common metrics for node importance measurement

An essential task in SNA is to measure the node importance within the whole network. It helps to recognize the most influential nodes with powerful abilities of passing information to other nodes as quickly as possible. Two kinds of metrics are typically employed for node importance measurement, which are presented as follows.

(1) **Centrality:** Four centrality metrics, which are varied in their definition, are adopted to measure and identify the critical nodes in a complex network. Specifically, degree centrality in Eq. (5.5) counts the number of links attached to the given node under the assumption that important nodes will have more connections.. Closeness centrality in

Eq. (5.6) calculates the reciprocal of the sum of the shortest path distance between the given node and all other nodes, which assumes that the node closer to others is more important and can transmit information more efficiently. Betweenness centrality in Eq. (5.7) estimates the frequency of the given node falling on the shortest path between all pairs of nodes, and thus nodes bridging two disconnected groups are considered to be more important. Eigenvector centrality in Eq. (5.8) is a modification of the degree centrality, which takes into account both the link number and the importance of neighbors for a given node in a function of the centralities of its neighbors.

$$C_{degree}(v) = d(v) \times (|N| - 1)^{-1} \quad (5.5)$$

$$C_{close}(v) = (|N| - 1) \times (\sum_{u \in N(v)} d(u, v))^{-1} \quad (5.6)$$

$$C_{between}(v) = \left(\sum_{s,t \in N, s,t \neq v} \frac{\sigma_{s,t}(v)}{\sigma_{s,t}} \right) \times ((|N| - 1)(|N| - 2))^{-1} \quad (5.7)$$

$$C_{eigen}(v) = \lambda^{-1} \times \sum_{u \in N(v)} C_{eigen}(u) \quad (5.8)$$

where N is the set of nodes in the network, v is the given node, $d(v)$ is the degree of v , $N(v)$ is the set of neighbors of v , $d(u, v)$ is the shortest path between nodes u and v , $\sigma_{s,t}(v)$ is the number of shortest paths from nodes s to t passing through v , $\sigma_{s,t}$ is the number of all shortest paths from s to t , and λ is a constant.

(2) **Web-page ranking:** PageRank and Hypertext Induced Topic Search (HITS) are two web-page ranking algorithms to consider the influence from the neighboring nodes and even the neighbors of the neighboring nodes, which is outstanding in ranking nodes for the complex directed graph. The PageRank of node v is recursively defined by Eq. (5.9), which largely relies on the PageRank of nodes pointing to v (Page, Brin et al. 1999). For another algorithm named HITS, it iteratively updates an authority score and a hub score for a node v by Eq. (5.10). In specificity, an authority is a node which many hubs link to, while a hub is a node that links to many authorities in the root.

$$PR(v) = (1 - d) + d \sum_i \frac{PR(T_i)}{C(T_i)} \quad (5.9)$$

where $PR(T_i)$ is the PageRank of node T_i linking to the node v , $C(T_i)$ is the outgoing links of node T_i to allocate weight to $PR(T_i)$, and d is a damping parameter in the range of $[0,1]$ indicating the probability of choosing an outgoing link at a random walk.

$$a_v = \sum_{j \rightarrow v} h_j, \quad h_v = \sum_{v \rightarrow j} a_j \quad (5.10)$$

where $a = (a_1, a_2, \dots, a_n)$ and $h = (h_1, h_2, \dots, h_n)$ are the authority score matrix and hub score matrix for n nodes respectively, and $j \rightarrow v$ denotes a link from node j to v . The iterations will repeat until a and h converge.

5.2.3.2 A new defined metric for node importance measurement

It is known that the most common means of quantifying the node influence is through the basic centrality metrics to describe the network structure from the local or global level. Although these benchmark metrics are easy to implement, they have their own shortcomings. For instance, the degree centrality directly counts the number of neighbor nodes under low computation complexity, which neglects topological connections from neighbors (Gao, Ma et al. 2014). Although the closeness centrality and betweenness centrality consider the global structure, they only work with the whole topology information available, which are incapable in large-scale networks (Wei, Pan et al. 2018). Besides, there is a centrality metric called the k -shell to divides the network into ordered shells with a full hierarchy of nodes, but it is hard to distinguish node importance especially when most nodes are grouped in the same layer (Liu, Tang et al. 2015). To address these issues, I intend to define a new metric called “impact score”, as presented below. The core idea of the new metric is to combine the k -shell method and 1-step neighbors to achieve comparatively low computational cost and high accurate ranking.

An unweighted and undirected social network can be defined as $G = (V, E)$, where $V = \{v_i\}_{i=1}^n$ is the set of n nodes, and $E = \{e_{ij}\}_{i,j=1}^n$ is the set of ties. The degree centrality simply assumes that the most highly connected node must own the strongest influence. But for nodes locating at the network boundary rather than the core, they tend to exert less impact even if they have a large degree centrality. Hence, it is necessary to target the location of nodes. For this purpose, a decomposition analysis k -shell is developed to recursively take the peripheral nodes with a degree less than the current shell index k_s (an integer index) (Kitsak, Gallos et al. 2010). By grouping nodes with the same index k_s into the k_s -shell, the k -shell method partitions the network into ordered shells from a

hierarchical view, where a large and small value of k_s is the representative of node location in the innermost and outermost layer, respectively (Garas, Schweitzer et al. 2012). More specifically, the k -shell method begins from deleting nodes with the degree $d = 1$ and their related links. Thereafter, there may be nodes with only one connection in the updated network, which need to be removed iteratively until no such node remains. All these removed nodes are labeled as $k_s=1$ and gathered in the 1-shell. Similarly, this kind of pruning process can be repeated to remove nodes with an increased degree ($d = 2, 3, \dots$) until all nodes obtain a k_s index. That is to say, it can be assumed that a group of nodes at the same k_s -shell have a similar spreading capability, even though their degree could be varied ($k \geq k_s$). However, k -shell is a coarse analysis to assign more than one node in an identical layer, which fails to differentiate the importance of these nodes by precise ranking.

It is pointed out that the 1-step neighbors, which are nodes straight linking to their seed node, play a vital part in information propagation. Information originating from a node will firstly go through its neighboring nodes and then spread out to other nodes. Inspired by the 1-step neighbors, I improve the k -shell method in terms of the similarity of the neighboring nodes for pairs of seed nodes, in order to more accurately sort the node influence. A fact is that two nodes with the highly overlapped 1-step neighbors can only limit the information spreading range in their common neighboring nodes, whereas nodes with dissimilar neighborhoods will potentially exert more effects on a wider scope. Specifically speaking, the sphere of potential influence from two nodes largely depends on the dissimilarity level of their 1-step neighbors. I refer to the Jaccard distance given by Eq. (5.11) to quantify the influence.

$$D(i, j) = \frac{|d(i) \cup d(j)| - |d(i) \cap d(j)|}{|d(i) \cap d(j)|} \quad (5.11)$$

where $d(i)$ and $d(j)$ are the set of neighboring nodes adjacent to node i and j , respectively, $|d(i) \cap d(j)|$ denotes the number of neighbors the two nodes i and j have in common, and $|d(i) \cup d(j)|$ represents the total number of neighbors the two nodes i and j have.

A larger Jaccard distance $D(i, j)$ implies that nodes i and j have less similar 1-step neighbors, contributing to effectively promoting information dissemination in more nodes. Rather, the connection of two focal nodes with a low Jaccard distance will act less importantly, since information can expand to the same neighborhood with no need for the interaction between the two nodes. Therefore, it is sound to adopt the calculated Jaccard distance as the link weight to distinguish the function of ties in information spreading. Through considering synthetically with the node location by the k -shell method and the 1-step neighbors by the Jaccard distance, the definition of the new metric called impact score is given in Eq. (5.12) to measure the node influence of node i more reasonably.

$$k_{IS}(i) = k_s(i) \sum_{j=1, i \neq j}^n a(i, j) D(i, j) k_s(j) \quad (5.12)$$

where $k_s(i)$ and $k_s(j)$ is the k_s index for node i and j , respectively, $a(i, j)$ equals to 1 or 0 to indicate whether node i and j are adjacent or not, and $D(i, j)$ stands for the Jaccard distance in the 1-step neighbors of node i and j . A larger value of the impact score implies a more influential node. Besides, the low computational complexity of the k -shell method $O(|V| + |E|)$ ensures the efficiency of the proposed metric impact score, where $|V|$ and $|E|$ are the number of nodes and ties in the given graph, respectively.

5.2.3.3 CatBoost regression algorithm for node importance prediction

Apart from metrics to quantify node influence, proper machine learning algorithms can also be leveraged for predicting the numerical value of influence through learning relevant factors. Notably, another limitation of influence measurement metrics is that they ignore the impact of individual behavior. Since the influence of designers actually takes root in both the topological and behavioral changes, a series of features associated with time, design behavior, and network structure should be taken into account. Therefore, it can be defined as a regression task to explore the kind of dependencies in the target output and input features. Notably, the latest ensemble learning model termed CatBoost (Prokhorenkova, Gusev et al. 2018) is a modification of the gradient boosting decision tree (GBDT) with superiority in handling heterogeneous features, reducing overfitting,

and enhancing calculation efficiency, which has been successfully applied in social media popularity prediction (Kang, Lin et al. 2019), hydrology condition prediction (Huang, Wu et al. 2019), and others. Eq. (5.13) gives its objective function, where a dataset $D = \{X_i\}_{i=1,\dots,n}$ is split into the left subset $\{X_i^L\}_{i=1,\dots,n}$ and the right subset $\{X_i^R\}_{i=1,\dots,n}$. Especially for categorical features, the discrete set of values (such as the month in this case), CatBoost shows advantages in converting them into numerical ones by the means of the ordered target statistics (TS). Thus, the large dimension from one-hot codes in available boosting algorithms can be effectively avoided. That is to say, CatBoost generates multiple random permutations of datasets, which can be learned by the ordered and plain boosting mode and be predicted by oblivious trees.

For regressor training herein, four features about designers' engagement extracted from the huge BIM event logs, including the designer's active month, the number of his working days, finished tasks, and his degree in the social network, are fed into the CatBoost model, contributing to intelligently and accurately predicting designers' influence without calculating metrics of node influence. All the training and testing processes are fulfilled in Python 3.6 based on the CatBoostRegressor model from CatBoost package, a high-performance open-source library for gradient boosting on decision trees (<https://catboost.ai/>). I tune three important parameters, namely the iteration, learning rate, and maximum depth of the tree, according to the regression loss function Mean Square Error (MSE). More specifically, MSE is the average of square of the error (also called the residual) between the observed value and the predicted value, which can ensure the optimal capability of CatBoost in regression prediction. Due to the limited data, a 5-fold cross-validation is implemented to evaluate the predictive performance of the CatBoost model on new data. That is to say, the dataset is split into 5 folds and each fold can be utilized as a testing set once. Additionally, the standardized residual is calculated to measure the magnitude of error, which is a ratio of error to the standard deviation of the observed value in chi-square hypothesis testing. Outliers can be easily identified when the standardized residual is greater than 2 or smaller than -2.

$$\arg \min_r \{P(r, y, M)\} = \arg \min \frac{1}{\sum_{i=1}^n |x_i|} \left(\sum_{r=1}^n |X_i^L| \text{var} \left(y(X_i^L) \right) + |X_i^R| \text{var} \left(y(X_i^R) \right) \right) \quad (5.13)$$

where r is the decision rule, whose optimality is measured by the function M , y is the target function.

5.2.3.4 Link prediction

The link prediction problem is to predict the next potential links in the network, which has been successfully applied in the recommendation systems (e.g. LinkedIn, Facebook). Various metrics have been developed to predict prospective links, which focus on different aspects for similarity measurement, such as neighborhoods, path, node, and edge attributes. Two effective metrics are given below, namely, Adamic/Adar and SimRank, which are considered in this chapter to estimate the linkage likelihood among nodes. Based on the observed network-structured data, they serve as numerical evidence to foresee the possible information transmission and support useful inferences for future collaboration.

(1) **Adamic/Adar** (Adamic and Adar 2003): Generally, two nodes sharing more common neighbors are likely to connect in the future. As an extension of the common neighbors, Adamic/Adar measures the number of common neighbors for a pair of nodes u and v by weighting lower-connected neighbors more heavily, as expressed by Eq. (5.14).

$$AA(u, v) = \sum_{n \in N(u) \cap N(v)} \frac{1}{\log(|N(n)|)} \quad (5.14)$$

where $n \in N(u) \cap N(v)$ stands for a set of the common neighbors for the node u and v , and $|N(n)|$ is the degree of nodes adjacent to n .

(2) **SimRank** (Jeh and Widom 2002): SimRank can be computed by a recursive definition in Eq. (5.15), which captures the notion that two similar nodes will also have high similarity in their neighbors. From a point of random walk, SimRank indicates how two random walkers starting from a pair of nodes u and v will meet at the same node.

$$SimRank(u, v) = \begin{cases} 1 & , if u = v \\ \gamma \frac{\sum_{a \in N(u)} \sum_{b \in N(v)} SimRank(a, b)}{|N(u)||N(v)|} & , otherwise \end{cases} \quad (5.15)$$

where γ is a constant in $[0,1]$.

5.3 Case study for community detection

5.3.1 Construction of social network

As a case study, I adopt 4GB real BIM design event logs stored in the *Autodesk Revit* journal files to create and analyze the social network at the design phase, which are provided by an international architecture firm. It records 853,520 lines of design activities that occurred during Oct 2013-Oct 2014. The log data as shown in Figure 5.3 is in text, where each line represents one design operation with detailed information of the designer, project, timestamp, command, and others. Useful data is retrieved from the textual event logs by a developed Journal File Parser, which is then saved in an organized and comprehensive Comma Separated Values (CSV) format. Since noise will inevitably exist in the parsed CSV file, it is necessary to conduct data cleaning to ensure the data quality. In the end, there are a total of 667,156 lines of records performed by 68 designers remaining in the cleaned CSV for a more reliable analytical process.

The BIM-based design collaboration network will be built depending on two valuable attributes named “Designer ID” and “Session ID” in the cleaned CSV. As an explanation of “Session ID”, a large *Revit* project is often split into several sessions in around 200 MB to improve the modeling efficiency. On the one hand, information can be spread faster in small sessions. On the other hand, it can effectively avoid time-consuming rework arising from disordered task management and poor communication among designers. That is to say, these sessions can be transformed seamlessly among various designers in the BIM platform, which are the key source to present information dissemination among design groups. With the common goal of accomplishing the large project in the design team, design collaboration is generally realized when a designer is responsible for a part of the session and then passes it to another designer.

According to the data extracted from BIM event logs, Figure 5.4 visualizes a directional network with a total of 68 nodes and 436 ties to describe the complex collaborative design work, where an individual designer is considered as a node and the transmission of sessions from a designer to another is represented by an arrow. The weight of links is visualized by the width and color shade in an arrow, aiming to measure how frequent sessions will be transferred between two designers. For instance, since the largest number of sessions (50 sessions) was transferred from Designer #8 to Designer #1, Designer #8 and #1 were connected most strongly than others. No link between two designers means they would carry out different design sessions with no cooperation relationship. Moreover, the size and color in nodes stand for the node degree. The larger and deeper color the node is, the more interactions with others the designer owned. It is observed that Designer #31, #3, #37, #9, and #51 had the five-top value of the degree, who could be evaluated as the most critical designers during design with link numbers 39, 36, 35, 34, and 31, respectively. Table 5.1 summarizes the statistical analysis of the network structure. To explain the network density and diameter, 19.6% potential links actually appear in the network, and the longest length of all the shortest paths between node pairs is 8. This implies that information can flow easily through the network to realize a comparatively cohesive collaboration. The modularity value 0.623 is quite high to verify that the network is likely to be composed of some small groups, and thus the established network is worth detecting clusters.

Designer ID	Date	Design File	View	Specific Command
Sam 0414	2014-02-15 12:47:35.047	cent5ral_sam.rvt 212	LEVEL 01-working plan	Create A default 3D orthographic view
Sam 0414	2014-02-15 12:47:49.810	cent5ral_sam.rvt 212	3D View	Create A wall
Sam 0414	2014-02-15 12:47:35.047	cent5ral_sam.rvt 212	LEVEL 01-working plan	Other Jrn.Command "Internal" "Align references" "
Sam 0414	2014-02-15 12:47:49.810	cent5ral_sam.rvt 212	3D View	Create A new family
Sam 0415	2014-02-15 12:58:44.633	cent5ral_sam.rvt 212	Ref. Level	Create Edit the path by sketching in a plane
Sam 0415	2014-02-15 12:58:48.287	cent5ral_sam.rvt 212	Ref. Level	Create A line
Sam 0415	2014-02-15 12:59:13.860	cent5ral_sam.rvt 212	Ref. Level	Other Jrn.Command "Internal" "Pick Lines" "
Session ID	Time	Project ID	Event	

Figure 5.3. Example of six continuous records in BIM design logs.

Figure 5.4. Framework of the network-enabled BIM design event log mining.

Table 5.1. Characteristics of the BIM-based design collaboration.

Item	Description	Number
Nodes	Node number	68
Edges	Edge number	436
Average Degree	Average number of edges per node.	6.412
Average Weighted Degree	Average sum of weights of the edges per node	25.725
Network Density	Ratio of actual edges and the maximum possible edges	0.196
Network Diameter	Shortest length between the most distant nodes	8
Modularity	Tendency of nodes to be clustered	0.623

5.3.2 Implementation of *node2vec*-GMM

In this case, the key idea is to fully understand the interrelationship of designers and detect communities (clusters) containing close-connected nodes. At first, the graph embedding algorithm *node2vec* is implemented to learn appropriate graph features, in order to well keep the complicated network structure. It is known that an adjacency matrix is a straightforward representation to characterize the social network, which is a square node-by-node matrix comprised of only neighboring information expressed by $A=[a_{ij}]$ (i is the i th out-node in the row, and j is the j th in-node in the column). A 68×68 adjacency matrix can be built and visualized in the heatmap of Figure 5.5 (a), where the row and

column correspond to a designer sending the session and a designer receiving the session, respectively. The value $a_{ij}=1$ in the adjacency matrix is shown in blue, indicating there is a link from a designer to another. Otherwise, the white means no task transmission between two designers with $a_{ij}=0$. The matrix is asymmetric due to the directed network. However, a lot of zero values exist in the adjacency matrix, illustrating a sparse graph with few edges. In other words, if the network is fully connected, there will be $n(n-1)/2=68 \times 67/2=2278$ edges here (n is the total number of nodes). Since the number of actual edges is relatively small with only 436, only 19.14% matrix cells will take effect, leading to a waste of memory, high time complexity, and unreliable results in the subsequent machine learning applications.

A solution for better graph embedding is the *nodevec2* algorithm, which learns node feature representations in a biased random walk procedure to maximally reserve the network neighborhood of nodes. The parameters for the *node2vec* are set as: embedding dimension $d=128$, walks per node $r=10$, walk length $l=100$, window size $m=5$, return parameter $p=2$, and in-out parameter $q=0.5$. That is to say, a 100-length random walk will be repeated at each node 10 times with a neighborhood size 5. For the hyperparameters p and q , a high value of $p=2$ provides a low probability to revisit the starting nodes, which can avoid sampling redundancy. A small value of $q=0.5$ drives the walks away from the starting nodes to ensure global features. Following the biased random walks and model optimization mentioned in Section 5.2.2.2, a high-quality vector representation for all the 68 nodes is expressed in a 128-dimensional space. To graphically simplify the new graph embeddings in a 2D space, a non-linear dimensionality reduction technique named *t*-distributed Stochastic Neighbor Embedding (*t*-SNE) (Maaten and Hinton 2008) is carried out for node feature visualization, as depicted in Figure 5.5 (b).

After the graph embedding has been prepared via the *node2vec* algorithm, the unsupervised clustering task GMM can be, therefore, performed to learn these features and discover possible communities within the complex network. One of the major issues to be firstly resolved is how to choose the optimum number of GMM components. It should be noted that GMM is technically a generative probabilistic model to characterize data distribution, which is usually evaluated by the method of likelihood estimation AIC

and BIC. Figure 5.6 provides the variation of AIC and BIC values when the number of components is set as 1–5. The smallest AIC in the blue line appears at the points under three components, while BIC value from the orange line suggests that the ideal number of components is two, which is followed by three. It also proves that BIC usually yields a smaller cluster number than AIC. Since AIC and BIC do not agree on the preferred number, it makes sense to choose the value of two or three to be the component number. Herein, we define the optimal cluster number as three.

Afterward, the GMM model is conducted to iterate the EM steps until converge, aiming to ultimately assign each node to different communities with a certain probability. The covariance type in GMM is set as “Full”, allowing each cluster modeled by an ellipse to own independent shape and position. Consequently, the mean vector and covariance matrix for the three clusters are presented below, which are the cluster embedding to explain the position of the cluster center and the spread and orientation of the distribution, respectively. The results of GMM are displayed in Figure 5.7 (a) with three partitioned clusters modeled by three different Gaussian distributions. Figure 5.7 (a) also provides the contour plot from the probability density functions (pdf) of GMM, where the region with darker color closer to the center area has a higher probability. In a more intuitive way, Figure 5.7 (b) visualizes the three discovered communities directly in the design collaborative network, where 15, 26, and 27 designers are falling into three clusters, respectively. Table 5.2 lists all the likelihood of each designer in the three communities, where the sum of probabilities of three clusters for one designer is one. The highest probability in the bold font can determine a certain cluster, which a designer is more likely to belong to.

Figure 5.5. Node features from (a) Adjacency matrix visualized by a heatmap; (b) *node2vec* algorithm visualized by *t*-SNE.

Figure 5.6. AIC and BIC for each cluster number.

Figure 5.7. Results of community detection visualized in (a) Gaussian distribution; (b) BIM-based design collaboration network.

Table 5.2. Probability assignment for each designer in community #1– #3.

Community		Probability assignment in three clusters			
#1 (Size: 15)	(1, 0, 0)	(1, 0, 0)	(1, 0, 0)	(1, 0, 0)	
	(1, 0, 0)	(0.932 , 0.067, 0.001)	(0.997 , 0, 0.003)	(0.973 , 0, 0.027)	
	(0.975 , 0.003, 0.022)	(0.994 , 0.005, 0)	(0.975 , 0.022, 0.003)	(0.702 , 0.286, 0.013)	
	(0.713 , 0.228, 0.06)	(0.842 , 0.002, 0.156)	(0.997 , 0.002, 0.001)		
#2 (Size: 26)	(0, 0.925 , 0.075)	(0.019, 0.931 , 0.049)	(0.001, 0.994 , 0.005)	(0.003, 0.976 , 0.021)	
	(0, 0.978 , 0.022)	(0, 0.991 , 0.009)	(0.133, 0.866 , 0.001)	(0.001, 0.935 , 0.064)	
	(0.154, 0.842 , 0.004)	(0, 0.965 , 0.035)	(0, 0.799 , 0.201)	(0.003, 0.747 , 0.25)	
	(0, 0.993 , 0.007)	(0, 0.985 , 0.015)	(0.026, 0.963 , 0.011)	(0, 0.959 , 0.041)	
	(0.115, 0.860 , 0.025)	(0.018, 0.981 , 0.001)	(0.006, 0.989 , 0.005)	(0.005, 0.993 , 0.002)	
	(0, 0.632 , 0.368)	(0.081, 0.744 , 0.175)	(0.035, 0.963 , 0.002)	(0.031, 0.969 , 0)	
	(0.053, 0.947 , 0.001)	(0, 0.637 , 0.363)			
#3 (Size: 27)	(0.005, 0.02, 0.973)	(0.054, 0.135, 0.812)	(0, 0.484, 0.516)	(0, 0, 1)	
	(0.01, 0.001, 0.989)	(0, 0, 1)	(0, 0, 1)	(0, 0, 1)	
	(0, 0, 1)	(0, 0, 1)	(0.001, 0, 0.999)	(0, 0.025, 0.975)	
	(0, 0.005, 0.995)	(0, 0.013, 0.987)	(0, 0, 1)	(0, 0.003, 0.997)	
	(0.001, 0.311, 0.688)	(0, 0.152, 0.848)	(0, 0.095, 0.905)	(0, 0.013, 0.987)	
	(0, 0, 1)	(0, 0.001, 0.999)	(0, 0.004, 0.996)	(0, 0, 1)	
	(0, 0.361, 0.639)	(0, 0.093, 0.907)	(0, 0.125, 0.875)		

5.3.3 Analysis of detected communities

Based on the huge amount of BIM design event logs, three possible design groups are derived from a network with 68 designers and 436 design work transmissions by the developed node clustering algorithm *node2vec-GMM*. More investigations in cluster properties and cooperation evolutions among designers are in demand to provide a numerical basis for managers to better comprehend and optimize the collaborative design. The analysis results are outlined as follows.

(1) Three discovered communities can be distinguished by the measurement of node importance, implying each community has its unique structural characteristics. To quantify the community properties, Figure 5.8 and Figure 5.9 visualize the value of node importance for each node by group, and then fit a linear regression model with a 95% confidence interval. Except for the betweenness centrality in Figure 5.8 (c), there are obvious downtrends in the fitting lines from cluster #1 to #3. Overall, the importance of designers in cluster #1 ranks the highest compared with clusters #2 and #3 from different perspectives by Eqs. (5.5), (5.6), (5.8)-(5.10), which shows that designers grouped in cluster #1 tend to exert a much greater social influence than others in clusters #2 and #3 during the collaborative design. It also suggests that designers in cluster #1 will be more active with highly frequent interactions with others to contribute more in the collaboration. As for the betweenness centrality based on Eq. (5.7), its value of the fitting line in three clusters is roughly the same around 0.038 in Figure 5.8 (c). It means that all designers have probably an equal chance to lie on the shortest path, owning almost the same capabilities to control information flows over the network.

(2) Each community has several key designers (also known as leaders), who will organically affect the collaborative design work to a greater extent than other designers. I refer to two web-page ranking methods, namely PageRank and HITS, to ideally recognize the possible leaders in the directed graph from a quantitative view. A higher value of PageRank, Authority, and Hub helps in ranking the top five critical designers in each cluster, as tabulated in Table 5.3. Although these different measurements can generate some inconsistent results, there are some commonalities in the top five designers as emphasized in bold in Table 5.3. Specifically, clusters #1–#3 have four (Designer #31,

#51, #3, and #39), three (Designer #9, #37, and #23), and two (Designer #18 and #50) common leaders from PageRank and HITS. In the process of organizing the design work schedule, managers should concentrate more on these critical designers in each group. Herein, it can be simply assumed that these identified leaders are the most competent and suitable designers in their group. Design efficiency and collaboration are expectedly enhanced when leaders are allocated more complex and heavier tasks. Furthermore, the difference of PageRank, Authority, and Hub between leaders ranked 1st and 5th in cluster #3 is considerably greater than clusters #1 and #2. The value of the most critical designer in cluster #3 is approximately 50% larger than the fifth top designer. That is to say, Designer #18 and #50 capture the absolute leading position in cluster #3, whose impact is far greater than the top one leader in clusters #1 and #2.

(3) The cooperation pattern can be described that flows of design tasks through links are more likely to occur within the same community than between communities. In other words, a partitioned cluster consists of more densely connected designers than the rest of the whole network. Since the three explored clusters all have their own distinctive characteristics, the effectiveness of the developed node clustering algorithm can also be confirmed. From the Sankey diagram in Figure 5.10, it depicts the task transmission of inter-cluster and cross-cluster, where the width of the flow is proportional to its quantity. It is clear that a cluster will have a much thicker connection to itself than other clusters. More than 55% of design tasks sending from one certain cluster will be received by designers in the same cluster. For instance, 90 design tasks beginning with cluster #1 will be given to designers also in cluster #1, which take up about 62.5% of total sessions from cluster #1. That is because designers within the same cluster are more familiar and trust with each other. Thus, effective communication is prone to occur in the group rather than cross clusters, in order to make designers face fewer barriers in information exchange and schema discussion, which can also accelerate the modeling procedure. Accordingly, it is an idea to give more tasks to be accomplished within a cluster, which is expected to rationalize the design workflow.

(4) Managers are able to conduct the data-driven decision making for more ideal work arrangements, aiming to promote intensive cooperation and achieve great design

efficiency. In particular, the collaborative behaviors of designers will dynamically change during the modeling process. To understand the potential cooperative ways underlying the network evolution, link prediction can be carried out to quantitatively capture the next possible links between pairs of nodes, which mainly calculates the likelihood of links based on the intrinsic network structure, such as Adamic/Adar and SimRank. Since leaders in each cluster play a more significant role in the design work, they can be the key target to run the link prediction. Herein, I concentrate on the most key leaders (Designer #31, #9, and #18) in the three communities, respectively, who are identified by the web-page ranking. Figure 5.11 and Figure 5.12 illustrate the top twelve possible designers to receive design tasks from group leaders by the Adamic/Adar and SimRank, separately. Although predictions from these two methods are not exactly the same, nearly half of the possible associations are developed between the leader and designers in the same cluster. Taking Figure 5.12 (c) as an example, the leader Designer #18 in cluster #3 has a great tendency to forward the task to Designer #64 and #68 grouped in the same cluster, which presents the managers a valuable chance to better allocate the design sessions and develop workflows accordingly. In other words, managers can no longer formulate design plans totally depending on their personal ideas and experience, which are subjected to a lot of subjectivity and uncertainty.

Figure 5.8. Comparison of clusters measured by (a) Degree centrality; (b) Closeness centrality; (c) Betweenness centrality; (d) Eigenvector centrality.

Figure 5.9. Comparison of clusters ranked by (a) PageRank; (b) Authority; (c) Hub.

Figure 5.10. Sankey diagram about the design task flows among clusters.

Figure 5.11. Top 12 most possible links based on the value of Adamic/Adar index for (a) Designer #31 in cluster #1; (b) Designer #9 in cluster #2; (c) Designer #18 in cluster #3. (The number in brackets are the cluster label.)

Figure 5.12. Top 12 most possible links based on the value of SimRank for (a) Designer #31 in cluster #1; (b) Designer #9 in cluster #2; (c) Designer #18 in cluster #3. (The number in brackets are the cluster label.)

Table 5.3. Top five critical designers in cluster 1-3 by different web-page ranking.

Cluster	PageRank		Authority		Hub	
	Designer	Value	Designer	Value	Designer	Value
#1	#31	0.041	#3	0.351	#51	0.233
	#51	0.041	#31	0.278	#39	0.203
	#3	0.039	#51	0.266	#31	0.320
	#39	0.034	#39	0.238	#13	0.210
	#21	0.031	#25	0.185	#3	0.213
#2	#9	0.046	#9	0.285	#17	0.281
	#1	0.042	#37	0.274	#37	0.273
	#37	0.041	#17	0.257	#9	0.223
	#23	0.040	#23	0.240	#23	0.208
	#2	0.040	#52	0.205	#6	0.203
#3	#18	0.020	#18	0.112	#50	0.128
	#28	0.016	#20	0.089	#28	0.081
	#20	0.013	#66	0.070	#18	0.076
	#50	0.012	#26	0.057	#19	0.072
	#29	0.010	#50	0.052	#66	0.071

5.3.4 Validation of *node2vec-GMM*

In order to further validate the proposed node clustering algorithm *node2vec-GMM* in BIM event log mining, I also compare it against three state-of-the-art graph embedding methods: matrix factorization (MF) (Ahmed, Shervashidze et al. 2013), DeepWalk (Perozzi, Al-Rfou et al. 2014), and LINE (Tang, Qu et al. 2015), which are integrated

with two partitional clustering methods: GMM and *K*-means, respectively. All experiences are conducted on the same BIM log dataset from this case study. Indeed, 68 designers are from three teams in this real BIM design project, indicating that prior knowledge about ground truth clustering is available. Clustering quality can be, therefore, evaluated by two frequently used external CVIs, namely Adjusted Rand Index (ARI) (Hubert and Arabie 1985) and Adjusted Mutual Information (AMI) (Vinh, Epps et al. 2010), which assess that how the predicted clusters fit the true partitions in original data. A more promising clustering algorithm owns a larger external CVI, implying a higher similarity between the candidate partitions and ground truth.

Comparisons of eight different node clustering methods are demonstrated in Figure 5.13 and Table 5.4. From the network visualization on the 2D space, although all methods are capable of clustering nodes into three groups, it is a little difficult to distinguish the best method directly due to the ambiguous boundaries of each cluster. Besides, no direct judgment can be made to point out that whether designers assigned in a group are actually in the same team. Thus, I contrast the true and predicted cluster labels to verify the superiority of the proposed node clustering algorithm quantitatively. Firstly, the ARI and AMI in *node2vec-GMM* are at least 6.0% and 13.4% more than the other seven algorithms, which mean *node2vec-GMM* can predict clusters more alike to the truth. Secondly, since the top two highest values of ARI and AMI come from the *node2vec-GMM* and *node2vec-Kmeans*, signifying that the *node2vec* owns the comparatively powerful capability over other graph embedding methods in this case to learn and reserve the complicated network structure through exploring the various neighborhood in a more flexible way. In contrast, *LINE-GMM* evaluated by ARI and AMI is approximately 70.6% less than the best performance from the *node2vec-GMM*. It turns out that LINE is the worst method to learn the network representations here probably due to its incapability to reuse samples. Thirdly, under the condition of the same graph embedding algorithm, the node clustering method based on GMM can slightly improve the clustering quality than the popular *K*-means in terms of ARI and AMI. But the impact from the clustering method is smaller than the graph embedding method. It suggests that to choose the appropriate graph embedding method ought to have a higher priority. What's more,

GMM can directly offer cluster embedding by results of the mean vector and covariance matrix to numerically present the cluster structure.

As for the log mining approach based on the *node2vec-GMM* algorithm, there are still some limitations worthy of further improvement. For one thing, the *node2vec-GMM* algorithm does not have very high robustness to noise. When the uncleaned data with empty, errors, and unobserved collaboration inputs into the algorithm, it only returns ARI and AMI in the value of 0.319 and 0.342, which are nearly half of the value from the cleaned data. That is because noise has an inevitable effect on the network structure, making it deviated from actuality. In consequence, the algorithm will learn unreal network representation, and then group noisy data into clusters. For another, the work only depends on the network topology, but ignores features about the designers, such as their work experience and efficiency. To some extent, columns of “Designer ID” and “Session ID” are insufficient to offer promising features. In order to reach a sounder decision making for work arrangement, the algorithm is required to learn both the structural and behavioral features for making better use of BIM logs. Besides, more than one candidate method is employed to measure node importance and predict possible links from different perspectives, which sometimes could produce conflicting conclusions. To obtain more explicit results, the most appropriate one can be selected in response to the situation.

Figure 5.13. Visualization of designer clustering results in 2D by: (a) MF-GMM; (b) DeepWalk-GMM; (c) LINE (2nd)-GMM; (d) *Node2vec*-GMM; (e) MF-Kmeans; (f) DeepWalk-Kmeans; (g) LINE (2nd)-Kmeans; (h) *Node2vec*-Kmeans.

Table 5.4. Comparison of clustering performance from different node clustering methods.

Method	Adjusted Rand Index (ARI)	Adjusted Mutual Information (AMI)
<i>MF-GMM</i>	0.466	0.456
<i>MF-Kmeans</i>	0.444	0.444
<i>DeepWalk-GMM</i>	0.566	0.522
<i>DeepWalk-Kmeans</i>	0.484	0.486
<i>LINE-GMM</i>	0.180	0.189
<i>LINE-Kmeans</i>	0.076	0.085
<i>Node2vec-GMM</i>	0.614	0.643
<i>Node2vec-Kmeans</i>	0.579	0.567

5.4 Case study for dynamic network analysis

5.4.1 Discovery of dynamic social networks

As a case study, I investigate a real-world dataset of BIM design event logs over 4 GB provided by an international architectural design firm. These event logs captured the ordered model evolutionary dynamic over a large design project, which was collectively completed by 34 designers during a one-year period. Since collaborative groups/patterns evolved over time in the context of the one-year ongoing design project, the parsed logs with the notion of time allow for building time-based networks instead of a single static network in large size and complicated structure. As the project progressed, a series of dynamic networks could be built to better describe and understand the change of interrelationship among multiple designers with professional knowledge.

More specifically, I break down the year-long records from parsed logs into several parts with the maximum duration of a month to capture the structural changes in networks. The reason to chosen the monthly interval is briefly summarized below. For one thing, when the network is built on a weekly or bi-weekly basis, the number of nodes and edges within a network is limited, which is fewer than 9 and 20, respectively. Since the network structure is relatively simple, no deep investigation is required. For another, since it is a

year-long project, the original static network will be divided into only two parts based on the half-year interval. Although these two obtained networks incorporating a lot of nodes and edges are sufficiently complex, they are impossible to capture the dynamic characteristics of the project evolution. In reality, the selection of proper time intervals to create sub-networks for dynamic analysis largely depends on the project size and duration, which can be flexibly adjusted in different engineering projects to support the in-depth analysis and knowledge discovery. Meanwhile, the monthly analysis is one of the most common ways in construction project management. Generally, the project managers need to prepare a monthly progress report to track and analyze the last month's activities, which is helpful to make some timely adjustments and draw up plans for the next month.

According to necessary data items in the columns of event logs, including designer ID, session ID, date and start time, I construct a total of twelve networks in Figure 5.14 with varied size and density. These month-based networks are established and arranged from Jan 2014 to Dec 2014, which graphically display collaboration structures among multiple designers by the month. More specifically, interdependent designers are defined as nodes and their interactions are visualized as undirected and unweighted ties. The darkness of the color in nodes is proportional to the value of the node degree, which is the number of ties from a certain node to others. It is assumed that two designers can be connected on behalf of the cooperative relationships when they work together to build and modify the model as well as to share ideas at the same time period. For instance, Designer #1 and #2 were considered as the cooperation partners in the network about Jan due to the fact that there were frequent information transmission and knowledge exchange between them during the time interval 9:00 – 18:00 on Jan 1– 8. The darkest blue node for Designer #1 means that he was more engaged in the design work, since he was linked with the greatest number of collaborators than the other ten designers in Jan. The simultaneous working increases the opportunities of information and knowledge sharing among designers, which can not only put forward the modeling process effectively, but also promote communication and mutual understanding for detecting design errors and revising design scheme in time.

Figure 5.14. Structure of the monthly-based collaborative networks for design work.

5.4.2 Exploration of collaborative patterns

Changes in the form of cooperation are depicted in the twelve monthly-based networks, which can be clearly distinguished into a large or small group as two collaboration patterns in light of network size. It can be found that networks in the same group have both topological and behavioral similarities. In Figure 5.14, it is observed that the number of designers and their interactions in a network experienced a sudden increase at the beginning of the project from Jan 2014 to Feb 2014, which was then sustained at high value during Feb 2014 – Aug 2014 and ultimately dropped back to a low value in Sep 2014 – Dec 2014. This dynamic aspect of networks could be explained that the task in the first month was just to determine the model boundary and sketch the building frames in rough, which could be finished by very few designers. As the model progressed, the workload would grow heavily in the following six months, and thus more than 30 designers were involved in the design work to add more key entities and relevant details into the model cooperatively. By Aug, since more than 80% of the modeling project had been accomplished, it did not need too many designers participating in the design work simultaneously during the last four months of the year. These designers could, therefore,

involve in other new projects, which required lots of manpower. Based upon the network complexity, six networks as representative of collaboration in Feb 2014 – Aug 2014 with more than 27 designers and 46 links are categorized as the large collaborative group, while the remaining networks with designers fewer than 11 and links smaller than 15 are deemed as the small collaborative group.

As tabulated in Table 5.5, I shed light on the differences between these two collaborative patterns from two characteristics, namely the network structure and designers' behavior. The significant differences have been verified by the Wilcoxon rank-sum test, which returns the *P*-value less than 0.05. From features associated with network structures, although networks in the small group are relatively simple with a small effective size and average degree, they are prone to be more cohesive and highly connected according to the network density, a ratio of ties to the total possible number. To be specific, networks in the small collaborative group have approximately three times fewer designers and nine times fewer interactions than the large group's networks, causing designers in the small group to have only half of the potential collaborators. But the reduction in the degree cannot obscure the fact that the network density from the small group is more than twice the large one, which stands chances to raise the efficiency of data dissemination in the small group's networks.

For a better understanding of networks from both the macro and micro levels, I also summarize three network features and three centrality metrics for the one-year project in Figure 5.15. It is clear in Figure 5.15 (a) that network density and modularity display a strong negative correlation, indicating that a highly interconnected network is less likely to be divided into sub-groups. Twelve networks can be distinctly separated by a line $y=x$, which will also provide evidence for collaboration pattern discovery. Apart from the network standing for Jan, the grouping result from Figure 5.15 (a) is consistent with our previous partitions intuitively determined by the number of nodes and ties. In other words, small group's networks except Jan are gathered in the lower right corner, where the network density is greater than 0.3 and the modularity is less than 0.22. Based on the bubbles' size and color, large group's networks present a comparatively long average shortest path length greater than 2.42 due to their structural complexity. With regard to

the degree, closeness, and betweenness centrality metrics measuring the node importance in Figure 5.15 (b), the mean value for all three metrics in the small group's networks is significantly greater than the large group, since the maximum of the three metrics in the small group's networks is larger than those in the large group with a considerable difference over 0.2. The importance of the critical designers within the small group is noticeably higher, who tend to play a more decisive role in boosting the collaborative design in the current month than leaders in the large group.

From the features concerning the individual modeling behavior, designers in the large collaborative group are more physically active and productive compared to those in the small group. It is noticeable that a large *Revit* project is usually broken down into multi-pieces of sessions in around 200 MB, which can be more manageable and deliverable. For simplicity and efficiency, designers will carry out sequences of design commands in the sessions rather than the whole project. Therefore, it is reasonable that designers' contribution can be generally evaluated by the number of days, sessions, and commands. Since the average days of collaboration in the large group are twice longer than the small group, the large group's designers are probably more engaged in the collaborative design, resulting in more accomplished sessions and executed commands. In addition, designers in the large group have a wider interquartile range (IQR) of behavioral characteristics, meaning that design performance differs greatly in individual designers within the network belonging to the large collaborative group.

Table 5.5. Characteristics of two collaboration patterns (i.e., large and small groups).

Items	Features	Collaborative pattern 1: Large group	Collaborative pattern 2: Small group	<i>P</i> -value
Time	Month	Jan, Aug, Sep, Oct, Nov, Dec	Feb, March, Apr, May, Jun, Jul	--
Network structures (Mean [IQR])	Number of nodes	31.00 [30.20, 31.00]	6.50 [5.25, 9.25]	0.0037
	Number of ties	67.50 [65.20, 74.20]	7.50 [6.25, 10.20]	0.0022
	Network density	0.15 [0.14, 0.16]	0.39 [0.35, 0.45]	0.0022
	Network degree	4.36 [4.21, 4.91]	2.31 [2.07, 2.38]	0.0022
Designers' behaviors (Mean [IQR])	Number of days	23 [20.50, 24.80]	9.00 [7.50, 9.75]	0.0039
	Number of sessions	208.5 [138.00, 218.00]	24.50 [24.00, 25.00]	0.0038
	Number of commands	61480 [17837, 137490]	7728.5 [5198, 9662]	0.0411

Figure 5.15. Network structural characteristics: (a) Relationship in network density, modularity, and average shortest path length; (b) Mean value of three centrality metrics and the 95% confidence interval.

5.4.3 Measurement of designers' influence

A new metric termed the impact score is developed in Eq. (5.12) for measuring node influence by integrating the k -shell method and the 1-step neighbors, allowing to reliably rank and identify the influential designers in controlling information spreading within the BIM-based collaborative design process. Figure 5.16 (a) shows the histograms of designers' impact scores across two collaborative groups, where the mean and medium of the impact score are decreased by 34.31 and 24.93 from the large group to the small one. Meanwhile, designers with the impact score ranging in $[0, 80]$ account for 86.41% of designers in the large group, whereas 86.41% of small group's designers possess the impact score in the range of $[0, 15]$. The Wilcoxon rank-sum test is also adopted to validate the pronounced difference in the impact score between the two groups/patterns with the P -value smaller than 0.05. In other words, the final scale of information spreading by designers in large group's networks is possibly tripled wider than that in the small group's networks. That is because influential designers can potentially affect more designers as the network size grows.

Ranking results derived from the impact score and three standard metrics (i.e., degree centrality, closeness centrality, and betweenness centrality) have a lot in common, proving the correctness of the new metric. Herein, the Kendall's tau correlation coefficient (Kendall 1938) is adopted to quantify the similarity in two ranking lists from the impact score and benchmarks. When the Kendall's tau is closer to 1, it means that the two ranking lists tend to agree with each other perfectly. In Figure 5.16 (b), it is considered that the impact score has relatively strong consistency with the degree centrality and closeness centrality, since the Kendall's tau keeps high values greater than 0.7. There is an obvious decrease in the minimal Kendall's tau between impact score and betweenness centrality, especially in Jan (0.098) and Feb (0.488). Through comparison of the two collaborative groups, Kendall's tau in the large group is generally lower than the small one, signifying that a dissimilar rank between the impact score and three popular metrics will more easily appear in complicated networks. Besides, since the main point of interest lies in the most influential designers, I also describe the similarity under the Jaccard index only focusing on the top-5 and top-10 ranked critical designers in the small group and the top-5, top-10, and top-15 designers in the large group, as shown in Figure 5.16 (c). The Jaccard index all reaches 1 in the small group, except the pairs of the impact score and betweenness centrality in Jan. It indicates that these top- N nodes in rankings from the impact score can be considered basically equivalent to those from benchmarks. For the large group, the top-5 designers from the impact score and the other metrics have a great likelihood of suffering from discordance with the Jaccard index less than 0.67, while ranking lists for the top-10 and top-15 designers are almost identical based on the Jaccard index in the range of [0.67, 1]. In other words, the proposed impact score is more prone to shift the rankings of the most critical designers, leading to different results about the team leaders.

Since the efficient spreading of design information and knowledge has the potentiality to greatly improve the designers' interactions, the precise identification of the most influential designers becomes an essential step towards optimizing the design task allocation and boosting collaboration. According to the proposed impact score, Table 5.6 lists the top-5 most critical designers, who help in reaching the maximum scale of information propagation in the twelve dynamic networks. The bold emphasizes the same

value on behalf of inaccurate ranking. It turns out that the impact score performs better in providing more accurate rankings for both large and small groups by contrasting the rankings from the three common centrality metrics. To be specific, the degree centrality and closeness centrality suffer more from the inaccurate rankings. The betweenness centrality has a similar ranking ability as the impact score, but its heavy computational cost makes it hard to implement. As for the impact score taking the 1-step neighbors into account, it is able to easily distinguish designers' influence, which is particularly useful for large networks resulting in diverse ranks. Taking the network in Dec as an example, values of the degree centrality (0.75) and closeness centrality (0.8) imply that Designer #2, #1, and #8 tie for the second place. Meanwhile, there are two designers ranking at the second and two more designers placing at the bottom using the betweenness centrality. The superiority of the impact score shows that only two designers (#1 and #8) share the same rank.

Moreover, designers' roles vary month by month, but there are characteristics in common for networks belonging to the same group. Figure 5.17 depicts the variation of the role importance of designers per month, where the higher and redder the peak is, the more important the designer is. For instance, the top-1 most critical designer in Feb is Designer #13, whose role importance is shown by the highest peak with the deepest red. If designers do not participate in the collaboration, the importance of role will reach zero. It is observed the key designers are distinctively different between two collaborative groups, while they are generally similar within the same group. Although Designer #1, #3, and #4 have the powerful ability to spread information in all the six networks from the small group, they are no longer the most influential ones in the large group's networks. Instead, Designers #15, #16, and #18 always lead the collaborative design process in Feb 2014 – Jul 2014, who have more opportunities to exchange their tasks, ideas, and knowledge to others in more complex networks with a larger size. Thus, it cannot be simply assumed that critical designers from a small collaborative group can still keep active in the large group, since these designers may feel tense and overwhelmed in sharing their work and opinions with a great number of partners. They may also be lacking in the experience of working with big teams. Since leaders will change in different collaboration

patterns, managers need to arrange more proficient, communicative, and logical designers to smoothly promote the collaboration in the large networks. It is unreasonable to demand leaders from the small group to retain their impact in all situations. Apart from leaders, the influence of other designers in the same group's networks will also remain basically unchanged. For instance, Designers #22 – #28 in the large group are always ranked in the last indicating their less importance, which also means that they are less responsible for the six months' collaboration from Feb to Jul. In light of the role variation in Figure 5.17, managers are more accessible to designers' performance and influence hidden in the two discovered collaborative groups, who can, therefore, allocate appropriate work and prepare a rational cooperation plan evidently.

Figure 5.16. Results of the impact score and their validity: (a) Designers' impact score in two collaborative groups; (b) The Kendall's tau correlation coefficient between the impact score and three benchmark metrics; (c) Similarities for top-5, 10, and 15 designers between the impact score and three benchmark metrics. (Note: DC, CC, and BC are the

abbreviations of the degree centrality, closeness centrality, and betweenness centrality, respectively. IS represents the impact score.)

Figure 5.17. Variation in the role importance of designers based on the impact score for networks in: (a) the large collaborative group; (b) the small collaborative group.

Table 5.6. The top-5 most critical designers ranked by the impact score and three centrality metrics in per month.

	Month	Feb	Mar	Apr	May	Jun	Jul
	Large group	#13 (60.21)	#16 (74.63)	#16 (109.31)	#12 (105.01)	#14 (143.92)	#15 (137.22)
Rank from IS		#15 (49.15)	#15 (68.54)	#12 (108.54)	#18 (104.97)	#13 (129.11)	#14 (119.08)
		#16 (43.50)	#18 (57.40)	#15 (96.31)	#11 (91.50)	#15 (101.94)	#4 (117.90)
		#14 (42.00)	#12 (52.10)	#18 (93.59)	#15 (80.38)	#18 (93.76)	#16 (113.14)
		#4 (38.56)	#6 (47.96)	#14 (90.19)	#16 (77.29)	#16 (90.06)	#18 (99.82)
	Month	Jan	Aug	Sep	Oct	Nov	Dec
Small group		#1 (12.76)	#3 (36.42)	#1 (11.20)	#1 (12.53)	#1 (3.00)	#2 (8.40)
		#2 (8.42)	#1 (22.77)	#6 (11.20)	#6 (10.33)	#3 (2.00)	#1 (8.20)
		#3 (6.33)	#4 (22.43)	#4 (8.67)	#3 (8.00)	#2 (1.00)	#8 (8.20)
		#4 (4.00)	#6 (22.00)	#3 (6.40)	#11 (8.00)	#8 (1.00)	#3 (6.00)
		#5 (3.00)	#7 (15.00)	#2 (2.00)	#4 (6.20)	#9 (1.00)	#11 (2.00)
Rank	Month	Feb	Mar	Apr	May	Jun	Jul
	Large group	#13 (0.50)	#11 (0.51)	#16 (0.57)	#13 (0.50)	#14 (0.59)	#14 (0.58)
		#21 (0.47)	#15 (0.49)	#15 (0.55)	#18 (0.48)	#13 (0.54)	#11 (0.52)
		#14 (0.46)	#16 (0.49)	#12 (0.52)	#12 (0.47)	#12 (0.52)	#16 (0.51)
		#15 (0.46)	#6 (0.48)	#18 (0.51)	#4 (0.47)	#15 (0.52)	#15 (0.51)
		#17 (0.45)	#8 (0.47)	#4 (0.48)	#11 (0.47)	#9 (0.51)	#18 (0.49)

from	Month	Jan	Aug	Sep	Oct	Nov	Dec
DC	Small group	#2 (0.53)	#3 (0.82)	#1 (0.75)	#1 (0.83)	#1 (0.80)	#2 (0.75)
		#4 (0.50)	#1 (0.64)	#6 (0.75)	#6 (0.71)	#3 (0.67)	#1 (0.75)
		#1 (0.48)	#4 (0.6)	#4 (0.67)	#4 (0.63)	#2 (0.50)	#8 (0.75)
		#3 (0.42)	#7 (0.6)	#3 (0.55)	#11 (0.63)	#8 (0.50)	#3 (0.50)
		#5 (0.40)	#6 (0.6)	#2 (0.46)	#3 (0.56)	#9 (0.44)	#11 (0.25)
Rank from	Month	Feb	Mar	Apr	May	Jun	Jul
	Large group	#13 (0.50)	#11 (0.51)	#16 (0.57)	#13 (0.50)	#14 (0.59)	#14 (0.58)
		#21 (0.47)	#15 (0.49)	#15 (0.55)	#18 (0.48)	#13 (0.54)	#11 (0.52)
		#14 (0.46)	#16 (0.49)	#12 (0.52)	#12 (0.47)	#12 (0.52)	#16 (0.51)
		#15 (0.46)	#6 (0.48)	#18 (0.51)	#4 (0.47)	#15 (0.52)	#15 (0.51)
		#17 (0.45)	#8 (0.47)	#4 (0.48)	#11 (0.47)	#9 (0.51)	#18 (0.49)
CC	Month	Jan	Aug	Sep	Oct	Nov	Dec
	Small group	#2 (0.53)	#3 (0.82)	#1 (0.75)	#1 (0.83)	#1 (0.80)	#2 (0.80)
		#4 (0.50)	#1 (0.64)	#6 (0.75)	#6 (0.71)	#3 (0.67)	#1 (0.80)
		#1 (0.48)	#4 (0.6)	#4 (0.67)	#4 (0.63)	#2 (0.50)	#8 (0.80)
		#3 (0.42)	#7 (0.6)	#3 (0.55)	#11 (0.63)	#8 (0.50)	#3 (0.57)
		#5 (0.40)	#6 (0.6)	#2 (0.46)	#3 (0.56)	#9 (0.44)	#11 (0.50)
Rank from	Month	Feb	Mar	Apr	May	Jun	Jul
	Large group	#21 (0.35)	#16 (0.39)	#16 (0.35)	#18 (0.28)	#14 (0.26)	#15 (0.34)
		#13 (0.33)	#18 (0.27)	#7 (0.25)	#13 (0.25)	#7 (0.21)	#14 (0.32)
		#20 (0.21)	#15 (0.22)	#15 (0.15)	#4 (0.21)	#18 (0.19)	#11 (0.16)
		#15 (0.20)	#11 (0.16)	#4 (0.12)	#12 (0.19)	#13 (0.15)	#16 (0.14)
		#14 (0.18)	#6 (0.13)	#18 (0.11)	#29 (0.14)	#12 (0.13)	#4 (0.12)
BC	Month	Jan	Aug	Sep	Oct	Nov	Dec
	Small group	#11 (0.60)	#3 (0.53)	#1 (0.40)	#1 (0.55)	#1 (0.83)	#2 (0.50)
		#1 (0.56)	#7 (0.24)	#6 (0.40)	#6 (0.2)	#3 (0.50)	#1 (0.17)
		#4 (0.53)	#6 (0.22)	#4 (0.33)	#11 (0.1)	#2 (0.00)	#8 (0.17)
		#8 (0.38)	#1 (0.08)	#2 (0.00)	#3 (0.05)	#8 (0.00)	#3 (0.00)
		#3 (0.00)	#4 (0.04)	#3 (0.00)	#4 (0.00)	#9 (0.00)	#11 (0.00)

Note: DC, CC, and BC are the abbreviations of the degree centrality, closeness centrality, and betweenness centrality, respectively. IS represents the impact score.

5.4.4 Discussion of structural and behavioral effects on designers' influence

It is worth noting that features of network structure and individual design behavior may collectively have an effect on information spreading within the collaborative networks, leading to statistically significant correlation relationships with the proposed impact score. One structural feature (the node degree) and three behavioral features (the number of days, sessions, and commands) are considered as the key determinants, which

can substantially affect the designers' influence in both large and small groups. Figure 5.18 describes how the impact score changes with these determinants of interest, which is further checked in the statistic by regression analysis with a fitted line and 95% confidential interval. The univariate distribution is also available in the margins. For one thing, it is not surprising that the impact score is dependent on the node degree (Pearson correlation coefficient is 0.93, P -value is less than 0.05) due to the nature of the impact score encoding the 1-step neighbors. For another, the impact score is more linearly associated with behavioral features (Pearson correlation coefficient is greater than 0.5, P -value is less than 0.05), including the number of active days, finished sessions, and the executed design commands. On the contrary, the benchmark metrics in Figure 5.19 have a less significant linear correlation with the focused behavioral features (Pearson correlation coefficient is significantly smaller than 0.35, P -value is less than 0.05). That is to say, the impact score contains hidden knowledge about the designers' behavioral characteristics, which outperforms the degree, closeness, and betweenness centrality. Another prominent advantage of the impact score shows that it is particularly helpful in manifesting not only the network topological characteristics, but also the designers' engagement and productivity. Specifically speaking, the influential designer determined by the impact score is supposed to spend more days in modeling more sessions by carrying a series of commands, who can, therefore, share more information and knowledge with more designers. Based on the linear regression in Figure 5.18, managers can well predict the trend and value of the impact score (dependent variable) by the independent variable from structural and behavioral features.

To make designers' influence more predictive in a data-driven way, I train an emerging machine learning model named CatBoost with the sufficient capability to learn relevant features, and thus the model can serve as a useful alternative resolution for a multivariable regression problem. The impact score quantifying designers' influence can, therefore, be estimated intelligently, which no longer only depends on the network topology. As the preparation of the model training, four desirable features integrating the time, network structure and designer's behavior are input into the CatBoost model. Notably, since the collaboration characteristic is varied from month to month, the month

should be regarded as an important feature to dynamically capture the change of influence from the aforementioned three features, namely the node degree and the number of days, sessions, and commands. By learning these easily acquired features from logs and minimizing the objective function in Eq. (5.13), the CatBoost is able to illuminate how much the designer will affect the BIM-based collaboration within a month. In order to achieve higher predicting accuracy, I set main parameters including iterations, learning rate, and the maximum depth of the tree as 2000, 0.02, and 4, respectively, which can minimize the loss function MSE as much as possible. Figure 5.20 (a) provides an intuitive way to display and examine how well the predicted data are fitted with the actual value. Since the orange line of the predicted results is in good agreement with the blue line representing the ground truth, it preliminarily confirms the credibility of the trained model. Figure 5.20 (b) and (c) suggest that the standardized residual, a statistic term to estimate the strength of difference in predicted and actual data, is normally distributed, where only 9 of 228 samples are mistakenly estimated out of the confidence interval $[-2, 2]$. The standardized residual of nearly half of the predictions falls in the range $[-0.5, 0.5]$. It reveals that the model is suitable in both large and small groups reaching satisfactory performance. In addition, the CatBoost is further proven to be a reasonable choice in this case, which is superior to two leading machine learning algorithms, namely support vector regression and random forest in Table 5.7 in accordance with regression evaluation metrics, namely MSE, mean absolute error (MAE), and R^2 . To be specific, MSE and MAE quantify the prediction error in the predictive and actual data, and R^2 is a goodness-of-fit measurement. A better model can be confirmed when it owns the smaller MSE and MAE and the higher R^2 approaching to 1. The number in bold shows that the CatBoost model is the most ideal one among the three candidates.

In actual application, when the predicted impact score is larger than 40, the designer is more likely in a network belonging to the large group and acts as a potential leader within the network. For a new designer participating in the design work in a certain month, the CatBoost model can independently adapt to offer reliable and repeatable predictions in the new designer's influence. It mainly relies on the learning mechanism from the historical data. In other words, the model can realize a dynamic estimation for designers'

influence and role under the consideration of months. Meanwhile, since data about the designers' interactions will be updated in the BIM platform online as the project evolves, the CatBoost can learn these new data and incorporate them into the model continuously, enabling the predictions to be more authentic. What's more, the computing process for measuring the strength of designers' influence can be cheaper and more automatic, especially for networks with increasing size and complexity.

Table 5.7. Comparison of prediction performance from different machine learning algorithms.

Method	MSE	MAE	R ²
Support Vector Regression (SVR)	35.845	25.919	0.170
Random Forest (RF)	15.244	10.492	0.788
CatBoost	13.439	10.359	0.835

Figure 5.18. Relationship between the impact score and features of network structures (degree) and designers’ behaviors (number of days, tasks, and commands). (Note: The “pearsonr” is the Pearson correlation coefficient and the “p” is the P -value.)

Figure 5.19. Relationship between the centrality metrics and behavioral features.

Figure 5.20. Overall performance of the CatBoost model: (a) Predictive results and ground truth of designers' influence; (b) Scatter plots of the standardized residual of the predictions; (c) Distribution of the standardized residual with a kernel density estimate.

5.5 Chapter Summary

Motivated by SNA, this chapter presents network-enabled BIM log mining approaches to gain practical insights into hidden knowledge in the collaborative design task. It presents the opportunity for automatically understanding collaboration characteristics among designers from a new viewpoint of complex networks, which offers rich evidence to optimize work arrangements particularly for strengthening cooperation. As expected, new knowledge concerning the structure of a design team, roles of different designers, and regular patterns of workflow, and others, can be quickly and objectively discovered, which can guide managers to inform critical staffing decision in leader selection, design team setting up, process planning, workload distribution, and others. So

far, only Zhang et al. (Zhang and Ashuri 2018) focused on SNA to explore BIM event logs. But it only adopted some basic metrics to examine the network characteristics, which was unable to perform more advanced tasks in terms of community detection, link prediction, dynamic analysis, and others.

In order to further explore the topic of SNA in BIM-based design, a social network is constructed to describe the collaboration among designers during the modeling process based upon the meaningful information extracted from BIM logs in this chapter, where nodes are designers enrolled in and vertices are design tasks transmitted between two designers. The main contributions of this chapter can be summarized from two aspects. For one thing, a novel algorithm termed *node2vec-GMM* combining a graph embedding algorithm named *node2vec* and a clustering method named GMM to cluster designers within a network into several subgroups, and then makes cluster analysis. For another, I build networks on a monthly basis as the portrayal of dynamic design collaboration, and thus the information and knowledge sharing among designers can be graphically depicted in a new way. Special emphasis is put on measuring designers' influence by a defined new metric called "impact score", which combines the k-shell method and 1-step neighbors to achieve comparatively low computational cost and high accurate ranking. As for the novel findings, the *node2vec-GMM* algorithm is proven superior over other state-of-the-art methods in two perspectives: one is its efficient feature learning ability to preserve network structure, and the other is its powerful clustering ability to tackle uncertainty and visualize results. This hybrid algorithm can be executed with ease to promise high-quality network feature representation, creditable probabilistic results, explicit visualization, and the cluster embedding. Besides, extensive analytical results confirm that the dynamic social networks are worthy of full exploration for extracting collaboration patterns, assessing designers' behavior, and forecasting the network evolution in an objective manner, which can potentially serve as month-by-month feedback to monitor the ongoing modeling process and avoid unreliability and bias from the manual and burdensome subjective methods. Accordingly, managers can perform data-driven decision making to encourage a highly collaborative and efficient design process. For instance, the measurement of node importance helps managers determine the key designers, who can

be selected as the team leader. Link prediction provides managers with evidence to plan more logical workflows. To be more specific, the key conclusions from two case studies have been presented as follows.

In the case study about the proposed designer clustering approach, a collaborative network can be built based on BIM design event logs to describe information flows about 436 design tasks among 68 designers. Regarding the novel node clustering algorithm *node2vec-GMM*, a 128-dimensional feature vector is learned to preserve network structure and inherent properties, which is then fed into the GMM to infer the likelihood of a designer grouped into a certain community. Three possible clusters owning 15, 26, and 27 closely linked designers are discovered by *node2vec-GMM*. Several conclusions can be drawn from the cluster analysis: (1) Each community has its unique characteristics, which can be revealed by metrics of node important measurement. The most active and critical designers can also be determined, who have more influence than other designers in the same group and need the most concern. (2) More than half of the design tasks are transmitted within the community, implying that inter-cluster information exchange and sharing are more likely to occur than cross communities. Strategies to promote collaboration within the group can, therefore, be developed for more efficient communication and task transfer. (3) The future associations in pairs of designers can be mathematically predicted, providing managers with suggestions to schedule design plans in an evidence-based manner to pursuit a high-productivity modeling process. Additionally, to compare the *node2vec-GMM* against hybrid methods of three state-of-the-art graph embedding methods (MF, DeepWalk, LINE) with GMM or *K*-means, the performance of node clustering can be improved at least 6.0% and 13.4% by using the proposed *node2vec-GMM* method in terms of external CVIs (i.e., ARI and AMI).

In the case study about the dynamic network analysis, twelve networks on a monthly basis are developed instead of a constant network to consider the variation of inherent cooperation. Regarding the engineering signification, the proposed method has great potential to not only graphically understand the collaboration but also provide strong monthly-based evidence. As expected, it helps dynamically guide managers to develop changeable work arrangements and adjustments in a data-driven manner, which is

supposed to strengthen cooperation among groups of designers and boost project efficiency. Meaningful findings can be outlined as follows: (1) These month-based networks can be easily separated into two collaborative patterns (large and small groups) by network size. Two patterns have significant differences in characteristics of both network structure and designers' behaviors. Besides, the most influential designers are similar within the same group but varied from different groups. (2) It has been proved that the self-defined metric named the impact score is superior to the popular centrality metrics in lower computational cost and more accurate ranking. What's more, it can even yield a statistically strong correlation with behavioral features, meaning that it will not only directly show the topological features of the network, but also indirectly reflect the individual design performance. (3) The latest ensemble learning model termed the CatBoost enables computers to learn input data continuously for making optimal predictions about a designer's influence. Instead of only considering the structural characteristics in the centrality metrics, various features attributed most to the designers' influence are prepared, including time and designers' behavior. The experiment verifies that the developed model is suitable to perform the automatic estimation of designers' influence under satisfactory accuracy in both large and small collaboration patterns. In other words, the combination of SNA and machine learning can perform accurate prediction in an automatic and dynamic manner, which could be particularly useful in extremely complex and large networks. When the data size is sufficiently large, it can act as a powerful tool for time series analysis, allowing to identify the nature of the designer's modeling performance represented by the sequence of observations and predict future values of the time series variable named the designer's impact score.

CHAPTER 6. SIMULATING AND INVESTIGATING CONSTRUCTION ACTIVITIES BY PROCESS MINING

6.1 Introduction

This chapter addresses the Research Objective 4 of this thesis. The specific objective is to develop a novel framework of process mining-based BIM event log mining to simulate and optimize the activities of modeling a building containing dozens of tasks and behavioral interactions, which can then be reasonably integrated into BIM and IoT to construct a digital twin under a high degree of automation and intelligence. Its ultimate goal is to fully understand how a construction project actually proceeds, which can serve as evidence in process improvement through identifying deviations, inefficiencies, and collaboration features in the current process and predicting the variation trend of construction productivity in the next phase.

The motivation of this chapter is briefly presented below. The previous chapters have explored BIM event logs associated with the design phase. Nonetheless, the penetration of BIM has been expanded to large-size construction projects. Since more than 60% of BIM users from Germany rate very high value of BIM in supporting improved planning and tracking of schedule, labor, cost, and materials on the construction field (Analytics 2014), it also deserves facilitating more intelligent use of such event logs accumulated in the construction phase. In other words, the construction phase is also a data-rich environment, but BIM event log mining has not yet reached its full potentials in simulating a series of activities of modeling a building and producing strategic decisions for optimizing the complex construction process. Therefore, it is necessary to move forward by extending the application prospect of BIM event log mining from the design stage to the construction stage, aiming to improve the burdensome activities of modeling a building that are traditionally suffered from chronic productivity problems and task conflicts. Overall, the proposed conceive of BIM event log mining for smart project management can become more complete and practical. To actualize the objective and

narrow the gap between data science and BIM-based construction, two major research questions of this chapter can be summarized as: one is how to perform proper process mining techniques for the automated process discovery and analysis during the construction phase; the other is how to integrate process mining with BIM, IoT, and other popular data mining methods to design a data-driven digital for smart construction project management. In this regard, two case studies will be carried out to scientifically address these defined research questions. The detailed tasks in the two case studies are briefly presented as follows.

For the useful technique named process mining connecting process science with data science, it can be summarized into two main aspects. One is to automatically generate dynamic processes with concurrency, loops, logical counterpart, nodes, and others, as described in the available event logs. The other is to uncover causalities behind the process model by different levels of analysis, such as conformance checking, deviation detection, delay prediction, organizational exploration, and others. There are three research tasks to be performed: (1) To automatically discover a simplified and comprehensible process model as transparency of process knowledge (i.e., Petri net, BPMN, etc.), which is typically displayed by a direct follower graph containing key process-related information about the representative behavior and dependencies in the real process to describe the flow of activities in modeling a building; (2) To validate the established process model by proper evaluation metrics and check the conformance with the actual process recorded in the event log; and (3) To analyze the process model systematically from different views to reasonably instruct the task assignment, workflow optimization, and performance evaluation. In short, process mining can take advantage of the prepared event log data about a BIM-enabled construction project to diagnose possible problems in terms of events, people, and social network, which allows for extra suggestions to reduce the unwanted bottlenecks and prioritize actions towards great efficiency and reliability in the upcoming construction process.

For the digital twin under the combination of BIM, IoT, and data mining, it can facilitate data communication and exploration, and thus the complex workflow can become more understandable, controllable, and predictable. To be more specific, IoT

connects the physical and cyber world to capture real-time data for modeling and analyzing, and advanced process mining techniques incorporated in the virtual model aim to discover hidden knowledge in collected data by process modeling, bottleneck diagnoses, and productivity prediction. It leaves three main research tasks: (1) To design a rational architecture of digital twin with the help of BIM, IoT, and data mining (DM) to support intelligent process control and project management; (2) To automatically construct the high-fidelity virtual model as a digital replica of the physical object, which can simulate the as-happened construction process; and (3) To fully mine the large amount of BIM event logs delivered from the physical to the virtual side in both the current and future perspective, aiming to detect possible risk and predict the construction progress. In return, the usage of process mining techniques gives continual feedback about developing and adjusting the project planning and staffing, which can adapt to the changeable construction condition in the real world. This data-driven practice loop efficiently reduces the dependency of decision making in project management on expert knowledge and subjective judgment.

The rest of this chapter is structured as follows. Section 6.2 presents two important process analysis methods, namely process mining and time series analysis, to fully understand the actual construction project proceeds and identify underlying trends for future event prediction, which can eliminate the great dependency on expert judgment. Then, a closed-loop digital twin architecture is designed under the integration of BIM, IoT, and the important DM techniques mentioned above to better control and optimize the complex construction process. Section 6.3 performs a case study to manage and optimize the complex construction process towards the ultimate goal of narrowing the gap between BIM and process mining. Section 6.4 establishes a digital twin under the integration of BIM, IoT, and DM for a practical construction project to demonstrate its practicability. Section 6.5 summarizes the conclusions

6.2 Methodology

Figure 6.1 illustrates the process mining-based framework about automated process discovery and analysis for smart BIM-enabled construction management. Its goal is to capture an objective and holistic view of the procedure of modeling a building from the BIM as-planned event log with the opportunity to delve into possible defects, work efficiency, and collaboration patterns. It helps high-level managers to quickly diagnose the root causes of poor performance and predict the variation of productivity. In return, relevant responses for continual process improvement can be realized. In brief, the process mining-based method begins from the BIM server to parse event logs from BIM software automatically. Then, process discovery refines and displays meaningful behavior in proper process models with visibility and reliability. Lastly, in-depth analysis in the discovered process model can be run from the current and future perspectives.

Figure 6.1. Process mining-based framework for BIM event log mining.

6.2.1 Current perspective: Process discovery and diagnosis

6.2.1.1 Algorithms of process discovery

The first task in process mining is process discovery for constructing rational process models from the event log. That is to say, the key information extracted from event logs will be translated into the desired notations, like the terminator, activity, decision, arrows, and others, resulting in the data-based visualization of a process. As a view on reality, the discovered model demonstrates a holistic and deep insight into the current process to examine sequences of activities taken by actors, which is taken as the basis of further

process analysis and optimization. Thus, a process model benefits in graphically depicting the executing processes of complicated work for easier understanding and knowledge exploration. The automated discovery of the process model depends on proper process mining algorithms, which only take event logs with no prior information as input and then return process models in a visually structured and comprehensive process graph. It is noteworthy that the early process discovery method α -algorithm tends to inefficiently generate useless spaghetti-like models containing complete processes with all details. That is to say, it is incapable of distinguishing important and non-important information in noisy and less-structured logs. To deal with the challenges, two more advanced process mining algorithms are deployed, as introduced below.

(1) Fuzzy mining: Fuzzy mining (Günther and Van Der Aalst 2007, Günther 2009) is proposed to display suitable abstractions or aggregations of the observed process graphically using a map metaphor. That is to say, it mainly concentrates on subsets of the most significant behavior within the process to make process models simpler and more interpretable. The fundamental idea of fuzzy mining in model simplification and visualization lies in configuring two metrics named significance and correlation, where significance is commonly quantified by frequency of events and routings, and correlation estimates the closeness degree between two events. For the purpose of retaining high-level information, undesirable events and relations with both low significance and correlation need to be removed, while less significant but highly correlated behavior should be aggregated into clusters. From the map-like view of abstract process models, primitive and cluster nodes are linked by edges in different width and color representing relative significance and correlation after conflict resolution and edge filtering. Besides, fuzzy mining has taken effect especially in interactively simplifying models and investigating frequency and time duration in some practical applications (Jaisook and Premchaiswadi 2015, Premchaiswadi and Porouhan 2015, Gurgun Erdogan and Tarhan 2018). However, fuzzy mining is prone to suffer from unfitness and unsoundness due to its deliberately imprecise model.

In regard to the process analysis, the superiority of the fuzzy miner lies in its diagnose ability, which can intuitively project the bottlenecks into the current process map under

the consideration of frequency and duration attached in each event. It has been proved useful for bottleneck detection in practice (Jans, Van Der Werf et al. 2011, Premchaiswadi and Porouhan 2015, Gurgun Erdogan and Tarhan 2018). What's more, animation movie based on the fuzzy miner provides a powerful tool in visualizing the bottlenecks, which assists to better explain and resolve possible delays for flow time reduction in the actual process.

(2) Inductive mining: Inductive mining (Leemans, Fahland et al. 2013) is an improvement over α -algorithm and fuzzy mining. It is developed to tackle infrequent behavior and huge models, resulting in a block-structured process with high fidelity. The method starts from splitting original event logs into sub logs according to four operators, namely the exclusive-choice operator ($\times$), sequence operator ($\rightarrow$), parallel operator ($\wedge$), and redo-loop operator ($\cup$). Then, directly-follows graphs can be built for each sub log, which defines a set of activities by nodes and their execution sequences by directed edges. The splitting procedure will repeat until every subset is only comprised of one node (activity). In the end, the output of inductive mining is a process tree with no duplicated activities, which can be fit and sound to the observed behaviors in the event log. It can be regarded as an abstract representation of a sound block-structured workflow net with a leaf node referring to a single event and a non-leaf node denoting an operator (Hwang and Jang 2017). For instance, the inductive miner can produce a process model expressed as $Q \Rightarrow (a, \times (\wedge (b, c), e), d)$ to replay process in an event log $L = [< a, b, c, d >^3, < a, c, b, d >^2, < a, e, d >]$ recording 6 cases and 23 events (Van der Aalst 2016). Also, the process tree can be easily converted into an equivalent Petri net and business process modeling notation (BPMN).

It should be emphasized that inductive mining is flexible in creating process models with executable semantics and fitness guarantees. Due to the quality, flexibility, and scalability of the process model from the inductive miner, its important application is the conformance checking to identify undesirable deviations between the discovered process model and the corresponding observations in the event log. Therefore, the captured discrepancies can take effect in not only judging the great alignment of activity sequences, but also suggesting proper adjustments of the virtual model to make it closer to reality.

6.2.1.2 Representations of process models

A process model serves as an abstraction of the complicated process recorded in event logs, which can be visualized in different forms to better describe and understand execution sequences and dependencies in a series of activities. Herein, I refer to three common types of process models to convert the discovered results into desired notations.

(1) Petri net: Petri net (Petri 1962) originally developed in the late 1960s is one of the most prominent process modeling languages. It combines the mathematical formalism with a graphical representation, which shows superiority in exhibiting both the concurrency and asynchrony nature of processes. From a simple example in Figure 6.2 (a), the Petri net is typically a bipartite graph, where places in circles and transitions in squares are connected by a collection of directed arcs on behalf of various relationships.

(2) BPMN: The flow chart named BPMN is commonly utilized in business process management. It contains two critical kinds of notations, namely activity nodes and control nodes to represent the detailed execution of business activities. More specifically, the activity nodes stand for business events, while the control nodes indicate the flows and logic between activities. Compared to Petri net, BPMN can offer a more comprehensive set of elements to express the flow behavior. As a high-level notation for representing complicated processes, it has been proved that BPMN is easier to understand even for people with no professional knowledge. The BPMN in Figure 6.2 (b) has a similar meaning as the petri net in Figure 6.2 (a).

(3) Process tree: Process trees is another optional graph notation to ensure the soundness of representations. It has a hierarchical structure consisting of nodes and children, where the inner nodes stand for operators and the leaves are labeled with activities. In particular, the process tree is good at addressing the problem of Petri nets that they are prone to experience deadlocks and some anomalies. Besides, process tree benefits a lot in inductive process discovery. As an example, the Petri net in Figure 6.2 (a) is convertible to the process tree in Figure 6.2 (c).

Figure 6.2. Examples of: (a) Petri nets; (b) BPMN; and (c) Process tree (AND means parallel composition, XOR means exclusive choice, and SEQ means sequential composition).

6.2.1.3 Validation of discovered process models

Since the reliability of the process analysis heavily relies on the model quality, it is of necessity to evaluate how well the established model from process discovery algorithms can describe the observed behaviors (including cases and events) in the event log. In this regard, three quality dimensions called fitness, precision, and generalization are taken into account (Buijs, Van Dongen et al. 2012). They have been detailedly introduced in (Buijs, Van Dongen et al. 2012). Generally speaking, the lack of great fitness or precision leads to an oversimplified process model, while the lack of generalization causes overfitting

(1) Fitness: The role of fitness is to measure the model's competence in replaying the event log, which is defined by an alignment-based calculation in Eq. (6.1). During the process of aligning events to the process model, cost should be given when events are skipped or activities are inserted with no expectation. If all cases from logs are fully reproduced, we can obtain the perfect fitness closer to 1. Oppositely, the fitness of 0 signifies that the process model fails to replay traces in the log. Although an effective mean of raising fitness is to add more parts into the process model, it may simultaneously increase the probability of overfitting. Thus, behaviors, which are unobserved in logs, should be avoided if possible to appear in the process model.

$$Q_f = 1 - \frac{fcost(L,M)}{move_L(L) + |L| \times move_M(M)} \quad (6.1)$$

where $fcost(L,M)$ represents the total alignment cost for event log L and model M . For example, if $fcost(L,M) = 0$, it means that the model M can perfectly replay the log L . For the denominator, it stands for the maximal possible cost, where $move_L(L)$ is the cost of moving through logs rather than the model, and $move_M(M)$ is the cost only in the model. It is applied to normalize the total alignment cost.

(2) Precision: As defined in Eq. (6.2), precision is associated with underfitting. It calculates the fraction of behavior allowed in the process model, which is not observed in the event log. It is clear that a poor precision approaching 0 can be caused by $|en_L(e)| \ll |en_M(e)|$, which is a notion of underfitting. This would imply that behaviors in the process model are quite different from the event log. When almost all of the behavior in the process model can be actually seen in the log, it returns a high precision reaching the value of 1.

$$Q_p = \frac{1}{|E|} \sum_{e \in E} \frac{|en_L(e)|}{|en_M(e)|} \quad (6.2)$$

where $|en_M(e)|$ represents the number of activities enabled in the model M , $|en_L(e)|$ refers to the number of actual activities executed in event logs L under the similar context, $e \in E$ is events, and $|E|$ is the number of events in logs L .

(3) Generalization: Generalization given in Eq. (6.3) is related to overfitting. It estimates how generic the process model is able to describe the unknown behavior, which

is not limited in the event logs. Greater generalization ability is confirmed when more parts of the discovered process model can be frequently visited. Inversely, when some parts of the process model rarely work, it implies that the model requires more behavior to depict the actual process. It should be noted that fuzzy mining is typically a generalizing algorithm.

$$g = 1 - \frac{\sum_n (\sqrt{|executions|})^{-1}}{|n|} \quad (6.3)$$

where $|execution|$ is the number of executions of certain parts of the process tree, and $|n|$ is the number of nodes in the process tree.

6.2.1.4 Analysis of discovered process models

There are two major kinds of analysis in process performance evaluation: one is based on the process model itself, and the other focuses on individual interactions, which are presented below.

For the model-based analysis, the deviation about behavior between the extracted process model and log data can be easily checked, which needs more discussion and elaboration for process optimization. Besides, information about duration and frequency can also be projected into the process model, which highlights the place to spend more time or be executed more often. Relying on these discovered frequently taken paths and significant bottlenecks, reasonable suggestions are generated accordingly to shorten the overall flow time.

For the interaction-based analysis, relationships among participants in the defined process model can be expressed in network topologies with nodes and edges for quantitative analysis. SNA can also be performed to shed light on the network structure configuration at three levels for the examination of individual roles, possible communities, and cooperative characteristics. It can support the important staffing strategy in strengthening cooperation and enhancing efficiency. Apart from exploring the network by some basic metrics, like density, diameter, average path length, Modularity, centrality, web-page ranking, and others, three novel indicators (Durugbo, Hutabarat et al. 2011) in Eqs. (6.4)-(6.6) are employed to analyze the extent of the intra-organizational

collaboration in the scale of teamwork, decision making, and coordination, respectively. Therefore, three significant abilities of the node, including engaging in teamwork for a common goal, making decisions based on interconnectedness, and harmonizing activities with others, are measurable.

$$\tau = \frac{\sum_{i=1}^N (C_i + DC_i) \gamma_i}{N} \quad (6.4)$$

$$\delta = \frac{\sum_{i=1}^N (C_i + CC_i) \beta_i}{N} \quad (6.5)$$

$$\chi = \frac{\sum_{i=1}^N (CC_i + DC_i) \alpha_i}{N} \quad (6.6)$$

where C_i , CC_i , and DC_i are the clustering coefficient, closeness centrality, and degree centrality of node i , γ_i , β_i , and α_i are the teamwork, decision, and coordination constant based on node i 's capability of pooling resources, making choices, and harmonizing interactions.

6.2.2 Future perspective: Process prediction and analysis

6.2.2.1 Time series prediction

Noticeably, the sequence data send from the physical model to the virtual one is ordered in clearly defined time components, which is regarded as the time series data to carefully track the evolution of construction work. It is common to carry out proper algorithms for pattern discovery in the time series data, which are likely to persist in the future. That is to say, data can be explored from a future perspective through examining characteristics of changes and predicting coming construction progress and workload, which can potentially guide the construction schedule and optimize the workflow in turn. Particularly, the Autoregressive Integrated Moving Average (ARIMA) model (Box, Jenkins et al. 2015) is one of the most popular statistical methods to understand and forecast time series data. Eq. (6.7) defines the ARIMA model to specify the current observation in terms of the linear relationship with past values, which can be decomposed into three components: autoregressive part (AR), integrated part (I), and moving average part (MA) with three non-negative parameters p , d , and q , respectively. To be specific,

AR (p) describes a regression involving dependencies between the current observation and the observations over a prior period, which means the variable of interest is regressed on its own lagged values. I (d) identifies the times in differencing the observations to ensure a stationary time series with constant mean and variance over time. MA (q) provides a regression error in a linear combination of error terms, which take into consideration dependencies between an observation and a residual term from the moving average to the lagged observations.

$$(1 - \sum_{i=1}^p \phi_i L^i)(1 - L)^d X_t = (1 + \sum_{i=1}^q \theta_i L^i) \varepsilon_t \quad (6.7)$$

$$L^k X_t = X_{t-k} \quad (6.8)$$

where t is the index, L is the lag operator provided in Eq. (6.8), X_t represents the time series data, ε_t refers to the residual, ϕ_i and θ_i are the numerical coefficient for the value associated with the i th lag in the AR and MA mode, respectively. Besides, p and q are the order of the AR and MA model, respectively, and i denotes the degree of difference.

It should be noted that ARIMA is primarily proven useful in analyzing univariate stochastic time series. Indeed, values for every period are possibly influenced by not only past periods but also one or more outside factors associated with each time period. Therefore, it is convinced that the model forecasting performance can be raised in view of some extra explanatory variables in categorical or numerical form. In this regard, the multivariate ARIMA termed the ARIMAX model is developed to integrate covariates into the ARIMA model using Eqs. (6.8)-(6.10), which is the variation of Eq. (6.7) (Broniatowski, Dredze et al. 2015). Specifically speaking, the ARIMAX model is fully capable of handling the time series of interest and its orders along with additional inputs called the exogenous variables (augmentations).

$$(1 - \sum_{i=1}^p \phi_i L^i)(1 - L)^d (X_t - m_t) = (1 + \sum_{i=1}^q \theta_i L^i) \varepsilon_t \quad (6.9)$$

$$m_t = c + \sum_{i=0}^b \eta_i y_{t,i} \quad (6.10)$$

where L is the lag operator from Eq. (6.8), $y_{t,i}$ is a set of exogenous variables affecting the time series, η_i is the weight of exogenous variables fitted based on the model selection, and b is the size about the set of exogenous variables.

6.2.2.2 Model selection and evaluation

In the pursuit of promising model performance, how to figure out proper order parameters of the ARIMAX model becomes the main priority. The most intuitive method is to read the correlogram plot of the autocorrelation function (ACF) and partial autocorrelation (PACF) by Eqs. (6.11) and (6.12), respectively. Specifically, ACF calculates the autocorrelation between an observation X_t and the lagged observation X_{t-k} , while PACF is the correlation in X_t and X_{t-k} conditioned on observations between these two observations. However, when the data is in high complexity, it could be a little confusing to determine parameters directly by viewing the decay from plots. Thus, a more effective method called the grid research can be utilized to iteratively run the developed ARIMAX model on multiple combinations of p , d , and q , and then make the comparison of model performance based on the criteria of goodness-of-fit, namely the log-likelihood, Akaike information criteria (AIC), Bayesian information criterion (BIC). Especially for AIC and BIC in Eqs. (6.13) and (6.14), they have penalized likelihood with similar expressions, and the major difference is that BIC penalizes the model complexity more heavily. Regarding model selection, we prefer the fitting model with higher log-likelihood and lower AIC and BIC.

$$\varphi_k = \text{corr}(X_t, X_{t-k}) \quad (6.11)$$

$$\varphi_{kk} = \text{corr}(X_t, X_{t-k} | X_{t-1}, \dots, X_{t-k+1}) \quad (6.12)$$

where $k=0, 1, 2, \dots$ represents the lag.

$$AIC = -2\log L + 2(p + q + k + 1) \quad (6.13)$$

$$BIC = -2\log L + (p + q + k + 1)\log(n) \quad (6.14)$$

where p and q are the parameters of AR and MA model, respectively, L denotes the likelihood function, k represents the number of parameters in the model, and n stands for the number of data points.

Besides, the fitting model determined from the training set need to perform a forecast on the test set to return continuous values. For comprehensively assessing the quality of predictions, two basic evaluation metrics named Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) are adopted in comparison of the paired true and predicted value on the test set. MAE presented in Eq. (6.15) is an arithmetic average of the absolute errors in a set of predictions. Although MAE is easy to understand where individual differences have equal weight, it fails to alert very large errors. To deal with the issue, RMSE in Eq. (6.16) is expressed in a quadratic scoring rule to measure the average magnitude of errors. RMSE can make large errors more noted through assigning a higher weight to them. The minimum value of both MAE and RMSE is 0, and the smaller value indicates better prediction performance of the fitting model.

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (15)$$

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (16)$$

where n is the number of data points, y_i is the predicted value, and $\hat{y}_i$ is the true value.

6.2.3 Digital twin architecture

Based upon the great amounts of IoT data from the BIM-enabled construction project, a data-driven digital twin framework is put forward to build a closed-loop between the physical and digital world. Figure 6.3 presents the conceptual architecture of the digital twin, which can come into play throughout the project life cycle for smart construction monitoring and management. Noticeably, it is an integration of BIM with real-time data collected by IoT devices and knowledge extraction from data analytics, which is comparatively a new development. The workflow of the proposed digital twin incorporating BIM, IoT, and DM can be briefly presented below.

To begin with, the unmanned aerial vehicle (UAV) equipped with the 3D Light Detection and Ranging (LiDAR) can deliver IoT services from great heights over the construction site (Lagkas, Argyriou et al. 2018). It takes 3D point clouds to sense and act upon the actual (as-built) environment for real-time operational monitoring. Subsequently, this inspection data is sent to the BIM cloud system for storing. Cloud storage offers a large resource pool to address the problem of information overload (Ding and Xu 2014). It can be seen from Figure 6.3 that the BIM cloud performs as a bridge of the physical-cyber system to continuously collect the comprehensive set of information from the physical entity and send data to the virtual part. To make full use of the point cloud, it is compared with the as-planned IFC by a tool named “Real-Time and Automated Monitoring and Control (RAAMAC)” in BIMserver (<https://bimserver.org/>). The developed tool is responsible for identifying and communicating discrepancies between actual and planned performance, resulting in as-built IFC for the purpose of automated construction progress monitoring (Golparvar-Fard, Peña-Mora et al. 2009, Dimitrov and Golparvar-Fard 2014). However, IFC saving the digital building description is in a plain text file, which is unreadable by DM algorithms. As a solution, another existing tool named “IFC Logger” (Kouhestani and Nik-Bakht 2020) is employed to automatically parse useful data from IFC, such as construction tasks, workers, time, and others, which can output event logs in a comprehensible form for computers. To further ensure the data quality, data cleaning methods are conducted to remove noise. Lastly, the latest and prepared data gathered via IoT devices offer opportunities to pair physical entity into the high-fidelity virtual models along with vivid simulation, such as the 4D model and refined process model. Various DM techniques are applied in the virtue of digital twin integrating large data to realize process modeling, bottleneck diagnoses, and progress prediction automatically, which can return positive and timely feedback to managers. For instance, the 4D model in the combination of the 3D model and construction schedule owns a strong capacity in information visualization. As for the process model, it provides a concise and graphical representation of the complicated process, which demonstrates the practical implication for comprehending and managing the workflows and collaboration in the construction phase.

As elaborated in Figure 6.3, the knowledge discovery and reasoning in the virtual part are mainly conducted from two views. On the one hand, process mining is adopted to provide a current perspective of the construction project implementation. A better understanding of workflow and collaboration can be realized from the discovered process model. Moreover, possible bottlenecks arising in the actual process can be detected easily, and thus response measures can be taken to avoid these unnecessary delays before occurring. On the other hand, time series analysis is performed to intelligently measure and predict the successive construction progress from the future perspective. Managers can keep abreast of the workers' current performance and the related trend. Since these predictions from the updated information provide directions for controlling and improving the construction work, they should be fully utilized to draw up reasonable plans and adjustments at an early stage. In other words, the prominent advantage of data analysis in the digital model is that it helps in exploring observed data timely and automate strategic decisions for process optimization, and thus managers no longer depend too much on expert experience and domain knowledge. The feedback can be delivered back to the physical side in time to dynamically regulate the construction scheduling and worker arrangement. In short, the developed digital twin architecture under the inclusion of BIM, IoT, and DM techniques realizes the remote and efficient interaction between physical and virtual objects, allowing for smart construction process management and assessment.

Figure 6.3. Architecture of the proposed digital twin for a BIM-enabled construction project.

6.3 Case study on automated process discovery and analysis

6.3.1 Data preparation and description

Since BIM event logs are a premise for the success at process mining analytics, they should be prepared carefully from raw data to meet requirements of high reliability. Before the practical construction, 4D BIM tools are used to simulate the entire workflow in a virtual environment by linking the planned 3D model with a new dimension of temporal information. The semantics, relations, and properties in this planned model are commonly captured by IFC, a standardized, digital, and open data source. However, IFC is not a suitable data structure for process mining techniques, where information associated with the cases and events of the construction project is implicitly available. Therefore, the required process-related information needs to be extracted from the source data of IFC

and organized in the desired BIM event logs. For this purpose, the model-driven architecture approach named BIMserver (<https://bimserver.org/>) is utilized to centralize IFC. The tool “Eventlog Service” in BIMserver automatically analyzes these IFC and exports them in the BIM event log data format (Beetz, van Berlo et al. 2010).

To be more specific, the BIMserver takes the as-planned IFC model as the input. Subsequently, a number of query mechanisms are performed to retrieve important information about building products and processes with the help of *IfcEntity*, *IfcProcess*, *IfcControl*, *IfcActor*, and others (Kouhestani and Nik-Bakht 2020). For example, *IfcProcess* describes the process of an activity/event/task related to the construction project, and *IfcActor* presents parsons or organizations that take part in the project execution (Lu, Xie et al. 2020). In subsequence, these captured process-related data, such as task ID, task name, start time, finish time, and others, can act as attributes and be converted into flat event logs to describe the steps of process execution (Andrews, van Dun et al. 2020). It is known that the reliability of process mining is dependent on the quality of inputs from event logs, and thus another necessary step is to check the prepared event log manually for data quality assurance. For instance, since this research only explores tasks associated with physical objects, some other tasks irrelevant to build objects are not taken into account in this research. Such information unaffiliated with the research target needs to be deleted from the prepared event logs, in order to eliminate redundant information, decrease the size of the relational dataset, and even simplify the complicated problem. As a result, we can more easily focus on critical processes and relationships to identify where the problems and opportunities lie, and thus priority measures taken for improvement are expected to be determined more efficiently. Besides, if there is noise from missing values, we can refill them according to the original IFC files. For the relatively simple case study in this research, no missing value exists due to the high-quality IFC files and reliable “Eventlog Service” tool, and thus the step of addressing null values can be skipped. Lastly, an essential step for the creation of event logs is performed through saving the extracted data from IFC into a readable and understandable data format, such as the frequently-used CSV or XES (eXtensible Event Stream) supported by IEEE. That is to say, the as-planned event log defines a set of scheduled tasks in specific

sequences, each line of which possesses both the general properties of the IFC model (i.e., *IfcClass*) and the process properties (i.e., name, start and end time of the task, and participants). Based on the well-prepared event log, techniques of process mining can be then carried out to support process analysis and diagnosis in a systematic manner towards a specific engineering goal.

In order to verify the effectiveness and practicability of the developed process mining-based method, a case study is performed in a 3-story building construction project in the Netherlands under 39 kinds of activities and 11 constructors during Feb 2015 – Oct 2015. Since process mining is aimed at extracting hidden knowledge from event logs, I turn to the “Eventlog Service” tool in BIMserver for event log preparation, which helps to parse IFC files available in the Synchro software. As a result, a collection of cases and associated events about the scheduled flow of construction can be extracted and then stored in the appropriate data structure named the as-planned event log. The obtained event log is most readily understood and digested by algorithms for process discovery and improvement. The obtained event log combining multiple information from IFC is most readily understood and digested by process mining algorithms to bring series of benefits, like to easily focus on the crucial paths, to quickly search the problem causes, to strategically arrange work and allocate resources for boosting process efficiency and effectiveness, and others. Additionally, the integration of IoT and BIM opens a novel way of monitoring and controlling the ongoing construction operation, which can bring in large volumes of real-time data. point clouds by drone scanner can track the as-built construction status, which are a kind of progress data. When this acquired progress information is automatically compared and incorporated into the 4D BIM model, the executing state of the certain activities can expectedly be assessed. Herein, the as-built event logs can be produced based on the automatic comparison of the expected progress from BIM and real-time data from point clouds. One noticeable characteristic of the as-built event logs lies in one additional column compared to the as-planned logs, holding information to judge whether the event is executed on time or not. In short, as-built event logs are specifically leveraged for identifying the discrepancies between the plan and the actual operation over time, owning the same attributes as the as-planned event log and one

more attribute about the punctuality. Noteworthy, the prerequisite for process mining is the high-quality BIM event log, which has been prepared by Schaijk from Eindhoven University of Technology (van Schaijk 2016). Thus, this case study needs no tedious effort in extracting the right event log from the BIM platform. Table 6.1 summarizes six main attributes in the exiting as-planned event logs. This event log is saved in a CSV file with 3,661 lines, where one line indicates a specific event (activity). To make the data suitable for process analysis tools, the event log can also be converted into XES formats with semantics for attributes.

In this case study, the top priority is to fully explore the prepared event log using process mining. Through intelligent analysis of such an end-to-end process, lessons can be learned to optimize the activity procedure of modeling a building and make better plans for other projects. To satisfy the requirement of process discovery, certain attributes should be necessarily defined as case and event. It is notable that different ways of definition will generate process models for different purposes. For instance, construction tasks represented by the attribute “TaskName” can be defined as events to play central roles in a task-specific process model, while events can come from the attribute “Participant” to build a participant-specific process model. As for the case, it is related to a sequential list of ordered events, which is helpful in distinguishing patterns of activities. I identify the attribute named “IfcClass” as the case, which is a representation of entities in the IFC standard. To be more specific, entities are the information agent to symbolize abstract objects with the same properties in nature due to the hierarchy and modularity of the IFC standard (Zhiliang, Zhenhua et al. 2011). For instance, IfcSlab/IfcBeam/IfcWall is to describe components in the group of constructing slabs/beams/walls. In this targeted event log, attribute “IfcClass” has 13 unique names to constitute 13 cases, whose characters are displayed in Figure 6.4. It is observed that the case duration will last longer when the case comprises more construction tasks in more types. More attention can be paid on the three major cases with the top three most frequent cases, namely IfcCovering (1,015), IfcWall (789), and IfcSlab (560), which are responsible for comparatively the most execution time (27.23, 25.23, 29 days) and the most task types (9, 14, 20). Besides, in order to study how participants execute various tasks, a participant-specific process

model can be built by setting 11 participants as the event. Its characteristics are briefly described in Figure 6.5. It can be seen that participants in different roles focus on different cases at construction. In particular, Roofer2 and Carpenter1 are more likely in charge of IfcCovering, while IfcSlab is principally finished by Installer1, Carpenter1, and Structuer1. Carpenter1 is more active and all-around than others, who keeps working over the life of the project and can even involve in a greater variety of cases.

Table 6.1. Six attributes in the BIM as-planned event logs.

Attribute	Description	Example
IfcClass	Groups of objects for particular purposes	IfcSlab/IfcBeam/IfcWall
TaskID	Serial number for a certain construction task	ST00060/ST00070/ST00080
TaskName	Name of a certain construction task	External facade levelling work/Installation/Masonry work
TaskStart	Start time of a certain construction task	26/2/2015 — 9/10/2015
TaskFinish	Finish time of a certain construction task	27/2/2015 — 15/10/2015
Participant	Person to perform a certain construction task	Carpenter1/Installer1/Roofer1

Figure 6.4. Bubble chart about the relationship in frequency, duration, and task types of cases.

Figure 6.5. Dotted chart about cases, events, and the corresponding timestamp in a participant-specific process model.

6.3.2 Process discovery

To facilitate the automatic creation of a fitting process model, the prepared as-planned event log containing 13 unique cases and 11 unique events is fed into a powerful inductive mining algorithm, which can reproduce all observed behavior. In terms of readability, the discovered process to describe the planned construction progress is depicted by two desired notations named a Petri net and a process tree. Both of the model representations are dedicated to giving a holistic glance of the actual execution order in the process, which are explained briefly as follows.

Figure 6.6 (a) shows the well-structured Petri net about the participant-specific process model, which is made up of 74 arcs, 23 places, and 36 transitions in total. It allows for visualizing the sequence, concurrency, and duplication of workflows among participants. Clearly, transitions standing for participants are interconnected by places, which are devoted to model the possible process status. The transition will be active to execute tasks once tokens are input into the place.

The process tree in Figure 6.6 (b) adopts four operators (“xor loop”, “xor”, “seq”, “and”) to straightforwardly translate connections in participants, making the Petri net more comprehensible. For instance, Carpenter1 and Roofer2 are more likely to work in parallel according to the “and” operation. Based upon “seq”, Roofer1 often executes tasks prior to Carpenter3, and then tasks are passed to other participants. From the “xor loop”, it can be inferred that Structuer1 and Mason1 are prone to redo tasks multiple times. Moreover, the tree structure can roughly divide participants into three major groups, in which participants tend to be more closely interrelated. The first group consists of Roofer1 and Carpenter3 and the second group contains Mason2, Installer1, Carpenter1, Roofer2, Structurer2, both of which demonstrate the sequential relationship among participants. The remaining two people Structuer1 and Mason1 under the “xop loop” can be categorized into the third group.

Figure 6.6. Representation of the process model by: (a) Petri net; (b) Process tree.

6.3.3 Conformance checking

It is known that a process model with obvious overfitting can lead to unreliable or even wrong results. To address this issue, an effective solution is to minimize redundancy in terms of the infrequent participants and paths using the variation of inductive mining. This is implemented by an inductive miner available in the tool ProM (<http://www.promtools.org>), a commonly-used process mining framework. The remarkable advantage of such an easy-to-use process mining tool is that it can both automatically discover process models and compare them with the actual processes in event logs (Leemans, Fahland et al. 2014). In this case, since Installer2 and Carpenter2 only execute construction tasks 4 times accounting for 0.11% (4 out of 3661) of total records, they have no additional effect on the process. They can be reasonably removed from the discovered model for better abstraction and exploration. Meanwhile, 20% of noise filtering is applied to filter paths with less frequency. After a few iterations, the new targeted flow with 9 major participants is produced as displayed in Figure 6.7. To be more specific, the process starts from the green point on the left and ends at the red point on the right. Arcs show the directly-follows relations (i.e., XOR split/join, AND split/join) in connected people. It should be noted that frequency is taken into consideration to obtain a semantic model, where the number in a box denotes frequencies the participant performs

tasks, and the number above arcs is the number of times the process traverses between participants.

To better understand the discovered process visualized in Figure 6.7, some typical mode concepts are highlighted in Figure 6.8. More specifically, Figure 6.8 (a) shows the common paths in the model represented by edges and activities. It indicates that Installer1 performs activities 58 times, which is the same as the incoming edges to its left. Figure 6.8 (b) explains the concurrency sign, where the path is split at the “AND split” to make Carpenter1 and Roofer2 work together, and then these paths are merged at the “AND join”. However, the collaboration opportunity for Carpenter1 and Roofer2 is not high in reality due to the 2083/2509 arcs bypassing Carpenter1/Roofer2, implying that 68.95% and 83.05% of work cannot be handled by Carpenter1 and Roofer2, respectively. Moreover, some deviations inevitably appear in the process model defined in Figure 6.7 due to the simplification. To facilitate the detection of deviations, the conformance checking technique is performed by comparing behavior in the discovered model and event logs. Overall, there are two main types of deviations (Leemans, Fahland et al. 2014): one is the log move demonstrated in Figure 6.8 (c) (an event recorded in the log does not truly reflect in the model), and the other is the model move given in Figure 6.8 (d) (an event required by the model does not present in the log). The red dash arcs in Fig. 6 clarify where the deviations probably occur during the process. Specifically, an arc circumventing a node is the model move, while a self-arc is a log move. It is obvious that the total number of deviations (37) is quite small to guarantee the quality of the abstracted process model. The only deviation about the log move is reflected in the path above Carpenter3 in Figure 6.8 (c), meaning that Carpenter3 will not conduct 8 out of 563 expected tasks. As a result, these diagnosed discrepancies support to improve the alignment of construction tasks for better work instruction and management. Moreover, metrics of fitness and precision in Eqs. (1) and (2) are calculated to assess the process model from inductive mining numerically. The evaluation results are listed in Table 2. Since fitness is the most closely relevant to conformance, the defined model with fitness greater than 0.8 indicates a great re-discoverability property. All the precision is above 0.85 to ensure no underfitting in the

model. In other words, the effectiveness of the discovered process model in Figure 6.7 is verified.

Figure 6.7. Process model from the inductive miner.

Figure 6.8. Mode concepts of the discovered process model from the inductive miner: (a) edge and activity; (b) concurrency activities; (c) model move deviation; and (d) log move deviation.

Table 6.2. Evaluation of the discovered process model based on the inductive miner.

Metric	Value
Log-move Fitness	1.0
Model-move Fitness	0.799
Precision	0.855
Backwards Precision	0.868
Balanced Precision	0.862

6.3.4 Frequency and bottleneck analysis

In order to easily recognize the important facts from the reconstructed process model, fuzzy mining is a proper choice to deliberately discard and aggregate some information, which strives for higher simplicity and understandability instead of precision. In view of time, the insightful process maps about frequency and duration in Figure 6.9 are generated by the tool Disco Fluxicon based on the fuzzy miner (<https://fluxicon.com/disco/>), where boxes stand for participants and arrows visualize the main process flow. In other words,

the map is able to reflect the critical workflows among all the 11 participants along with the casual dependencies between them. To validate the reliability of the fuzzy model in Figure 6.9, the fitness of each case is calculated according to Eq. (6.1) and outlined in Table 6.3. Except for cases “IfcColumn” and “IfcMember” with fitness less than 60%, the other 9 cases can be well fitted to verify the discovered model. It also turns out that cases composed of more construction tasks tend to reach higher fitness.

From the view of frequency in Figure 6.9 (a), the absolute frequency referring to the total number of times that a particular process is executed is visually by the thickness of arrows and the coloring of participants. The higher the frequency is, the more significant and remarkable the process is. It is clear that Manson2 (1056), Carpenter1 (938), and Carpenter3 (563) can be regarded as the top three participants playing the central role in the construction process, who are more active to finish about 28.84%, 25.62%, and 15.38% of total construction tasks, respectively. Besides, the three core process paths are Carpenter1 to Carpenter1 (658), Manson2 to Manson1 (497), and Carpenter1 to Roofer2 (257), which are performed the most frequently. It is found that the top two critical participants named Manson2 and Carpenter1 are in charge of these three core paths. Furthermore, dominant rework loops are prone to appear at Carpenter1, the second most important participants. For instance, 658 tasks finished by Carpenter1 is then sent back to himself, and only 157 activities are given to Carpenter1 again after having been conducted by Roofer2.

From the point of duration in Figure 6.9 (b), the average execution time for different parts of the process known as mean duration is adopted as the performance metric, which is calculated by the presence of timestamps with millisecond precision in the historical data. It is observed from the redder boxes that Roofer1, Mason2, and Roofer2 take the longer service time on average to complete their tasks. Although it seems that Roofer1 and Roofer2 involve more in the construction process, they are actually assigned less heavy workloads than Carpenter1 and Carpenter3 (the top three participants). That is to say, the productivity of Roofer1 and Roofer2 is lower than others. As the identification of bottlenecks, the thicker and redder arrows in Figure 6.9 (b) highlight the place where the longer waiting time is spent on task transmission between two participants. Clearly, the

three most problematic sequences are in the transition from Carpenter1 to Roofer2 (4.8d), from Carpenter1 to Carpenter1 (61.4hrs), and from Carpenter1 to Manson1 (58.2hrs). It implies that these sequences cost comparatively a longer time than others, leading to a greater likelihood of severe bottlenecks. Since arrows going in or out Carpenter1 are more likely to represent a longer time, the path related to Carpenter1 can be regarded as the higher impact area for delays. Also, it can be assumed that the root cause of bottlenecks is raised by the key participants who are expected to accomplish more construction tasks. Indeed, these participants cannot always execute all processes smoothly as desired. They may feel disorganized and sluggish to handle such burdensome and collaborative work. Hence, once the main reason for delays is found out, project managers can make fast responses in fixing causes and removing bottlenecks, such as to keep participants' work organized and on track, to enhance participants' efficiency, to eliminate unnecessary repetitions, and so on.

Figure 6.9. Process model from the fuzzy miner focusing on: (a) Absolute frequency; (b) Mean duration.

Table 6.3. Evaluation of the discovered process model based on the fuzzy miner.

Case	Fitness	Case	Fitness	Case	Fitness
IfcSlab	91.70%	IfcStair	94.74%	IfcBuildingElement Part	98.49%
IfcWall	96.83%	IfcRailing	88.89%	IfcMember	57.14%
IfcBeam	91.68%	IfcCovering	92.42%	IfcBuildingElement Proxy	89.66%
IfcWallStand ardCase	93.40%	IfcDoor	97.04%		
IfcColumn	39.13%	IfcWindow	98.79%		

6.3.5 Social network analysis

From an organizational perspective, social networks in the form of sociograms are built to delineate the complex process flowing through individuals, based on which SNA is then performed to examine patterns of interactivity and evaluate the roles of individuals quantitatively. As shown in Figure 6.10, three kinds of metrics are applied to generate different social networks (Van Der Aalst, Reijers et al. 2005), where nodes refer to all 11 participants involved, and the directed links correspond to relations between participants. The size of each node is proportional to its degree. Specifically speaking, the metric of “Handover of Work” defines a causal dependency between two participants. As an example, the direct succession by the arrow from Carpenter1 to Carpenter3 in Figure 6.10 (a) displays a task completed firstly by Carpenter1 and secondly by Carpenter3. The metric of “Subcontracting” used in Figure 6.10 (b) aims to determine whether an individual can work between two tasks executed by another individual, and thus the start/end point of the link denotes a contractor/subcontractor, respectively. Figure 6.10 (c) is derived from the metric of “Working Together”, which connects two participants working for the same case with no consideration of causal dependencies. Table 6.4 summarizes the characteristics of three network structures at the network level. In particular, the subcontracting network in a density of 0.1 is much sparser than others. To better understand the network structure, the metric called modularity allows detecting clusters (subgroups) embedded within the organization. From Table 6.5, the handover-of-work network and the subcontracting network can be further divided into three and six clusters, respectively. Since a partitioned cluster comprised of participants with denser

connections can transfer tasks and share knowledge with ease, a promising way of enhancing efficiency is to arrange participants in the same group to jointly undertake a task. On the contrary, the working-together network under a more cohesive structure exists no detectable subgroup, which is mostly due to its large density to make participants work as a whole.

Since participants will exert different impacts on the collaboration, it is of necessity to measure and rank their importance at the node level by PageRank and HITS, as illustrated in Figure 6.11. Thereby, more attention can be paid to the critical participants, who are in the leadership position with stronger influences in controlling the deep exchange of tasks, information, and opinions during the construction process. In the handover-of-work network, Carpenter1, Roofer2, and Structurer1 have the largest PageRank and Authority, who are the three most active participants to interact with others more frequently. These three leaders are assigned to three clusters in Table 6.5, respectively, which can possibly balance the influence of different subgroups and facilitate the handover process. The third-placed participant derived from Hub is Manson2 instead of Structurer1, since Manson2 sends out relatively more tasks. The top three key participants in the subcontracting network determined by PageRank, Authority, and Hub are the same, who are Carpenter3, Roofer2, and Roofer1 pertaining to the same cluster (cluster 2). In other words, the subcontracting process tends to be most affected by cluster 2. Obviously, there is no major difference in metrics of participants in the work-together network, implying that all the 11 participants play important roles and make similar contributions when working toward the common goal.

Motivated by the collaboration-level metrics, the network structure can be further assessed regarding the scales of teamwork, decision making, and coordination, which quantify the ease of nodes in pooling resources, making choices, and harmonizing interactions during cooperation. Above all, the constant value is set to 0.7 for nodes serving as the most important hub, which is decreased by 0.02 as the ranking of the hub drops. Then, values from Eqs. (6.4)-(6.6) are divided by its corresponding maximum to obtain a percentage, which is outlined in Figure 6.12. A larger percentage indicates a higher potential in collaboration for a specific purpose. The average value of the

coordination-scale indicator in the handover-of-work network (0.568) is taken as an example. It is derived from the expression $1.007/1.774$, where 1.007 is the average value and 1.774 is the maximum value. Since there is only a 56.8% chance of the maximum value to be 1.774, it can be inferred that this network has poor coordination ease. From Figure 6.12, the three-defined networks certainly have their respective characteristics. Observably, the leading feature in the handover-of-work network is decision making, while the subcontracting network is superior in coordinating work. In particular, no discrepancy among three scales exists in the work-together network, which has a more than 95% chance of achieving efficient teamwork, decision making, and coordination.

Figure 6.10. Three different social networks based on metrics: (a) Handover of Work; (b) Subcontracting; and (c) Working Together. (Note: Number in brackets are the node degree.)

Table 6.4. Characteristics of the three social networks based on different metrics.

Items	Networks based on three metrics		
	Handover of Work	Subcontracting	Working Together
Number of nodes	11	11	11
Number of edges	53	11	104
Average Degree	4.818	1	9.455
Network Density	0.482	0.1	0.945
Network Diameter	3	3	2
Average Path Length	1.5545455	1.682	1.055
Modularity	0.103	0.316	0

Table 6.5. Cluster detection in the discovered social network based on modularity.

Network	Cluster	Participants in each cluster
Handover of Work	Cluster1	Carpenter3, Mason1, Mason2, Roofer1, Roofer2
	Cluster2	Carpenter1, Carpenter2, Installer2
	Cluster3	Installer1, Structuer1, Structurer2
Subcontracting	Cluster1	Carpenter1, Installer1, Mason2
	Cluster2	Carpenter3, Roofer1, Roofer2, Structurer2
	Cluster3	Mason1
	Cluster4	Installer2
	Cluster5	Structuer1
	Cluster6	Carpenter2
Working Together	Cluster1	All the 11 participants

Figure 6.11. Importance of participants measured by the PageRank and HITS.

Figure 6.12. Comparison of collaboration metrics in three networks.

6.4 Case study on digital twin implementation

6.4.1 Data description

The proposed architecture of digital twin is implemented in a dataset about an actual BIM-enabled construction work of a three-story house in the Netherlands, which has already been prepared by Schaijk from Eindhoven University of Technology (van Schaijk 2016). That is to say, data acquisition based on the IoT-based process has been finished by the previous study. My work is to perform the developed digital twin framework in this existing dataset about a project carried out as a joint effort of 11 workers from Feb 2015 to Dec 2015. To make the process of data acquisition clearer, a brief introduction about it is given below. A UAV carrying the LiDAR scanner is taken as the IoT device. That is because the laser scanning is less susceptible to the effects of the outdoor environment, which gains dominance over the traditional photo scanning. The UAV flies above the construction site covering most parts of the building surface and surrounding space during the project, in order to efficiently capture scanned-surface models and the current operation status represented by high-quality point clouds in real-time. It is important to emphasize that the BIM cloud storage system is essentially used to store and manage these IoT data in great volumes. The tool “RAAMAC” in the BIMserver helps to parse the

information in point clouds and convert them into the desired IFC, while the tool “IFC Logger” further translates the IFC file into the event log as a collection of cases. That is to say, point clouds are automatically uploaded, saved, and maintained in the BIM cloud to create a real-time database, which can be accessed by different users and shared between the physical and virtual sides. In the meantime, real-time information regarding cases and events can be extracted from the IFC and organized in the event log. As is known to all, the event log is the properly formatted time series data with multiple attributes concerning events, ordered cases, and their associated properties to trace detailed flows of construction. All the crucial preliminary work in data acquisition has been done. Based on these prepared data, I intend to build a data-driven digital twin and mainly focus on one of the most important layers in the system called data analytics.

It is noteworthy that event logs are the output to track the as-happened construction process in machine-interpretable formats, including CSV and eXtensible Event Stream (XES). Process mining is especially used to discover knowledge from such data, which provides a new way of monitoring and improving the process. To be more specific, one event log describes a process made up of several cases, while one case occurs based on a sequence of ordered events (tasks). In this case study, the extracted CSV file contains 26,970 lines and 5 columns, where each line corresponds to a specific construction event and each column stands for an attribute. Table 6.6 shows an example of the event log data, where “IfcClass” is regarded as the case identifier. Events with the same name in the attribute “IfcClass” belong to the same case and have the same properties. For instance, “IfcSlab” can donate occurrences of slabs. In total, the case owns 13 unique types of “IfcClass”, among which “IfcSlab”, “IfcWall”, and “IfcColumn” are the three key cases comprising the largest number of tasks (>3000). “TaskName” stands for a well-defined event in the construction process. In terms of “Worker”, it refers to a certain worker to execute an event. There are 11 different workers participating and collaborating in this project, and workers 7, 1, and 3 are the top three most hard-working ones to carry out the most tasks. The last two attributes named “TaskStart” and “TaskFinish” are the timestamp to state the sequences of events related to a case. In short, this prepared event log in the size of 26970×5 is the data basis for constructing a digital twin, which needs to be deeply

explored using advanced DM techniques. Relying on the high level of bidirectional coordination between the physical and virtual structures, it is expected to bring potential benefits in the timely service of knowledge discovery and reasoning for process optimization purposes.

Table 6.6. Example of continuous records from construction event logs in the CSV format.

IfcClass	TaskName	Worker	TaskStart	TaskFinish
IfcSlab	Casting channel plate	Worker11	4/3/2015	5/3/2015
IfcSlab	Casting channel plate	Worker2	5/3/2015	6/3/2015
IfcWall	Framing lift walls	Worker1	6/3/2015	7/3/2015
IfcBeam	Steel beams	Worker1	6/3/2015	8/3/2015
IfcBeam	Steel beams	Worker1	6/3/2015	8/3/2015

6.4.2 Modeling of construction process

The prepared IFC and event log associated with day-to-day operations in the construction phase are accessible in the cloud database, which can be employed to recreate and simulate the progress in a virtual environment. In the context of cyber-physical synchronicity, digital entities can be built as a reflection of the actual activity sequences under ideal accuracy and update them through dynamic reconfiguration. The virtual model plays crucial roles in better simulating and understanding the construction logistics, which can then communicate closely with the physical system based upon their comprehensive data analysis. Herein, I perform two ways of building the virtual counterparts incorporating temporal information, namely the 4D model and process model, which are introduced below.

For one thing, the data-rich 4D model can synchronize with IoT data, which links the traditional 3D geometrical model with timelines to produce a digital description of the current project status. The clear visual context is established by importing IFC files generated based on point clouds. Moreover, animations with great visibility and transparency can also be performed to effectively imitate the execution of physical activities over the notion of space and time, particularly targeting at a continuous process monitoring and simulation for further investigation. In consequence, some schedule

problems can be disclosed at an early stage to reduce unwanted conflicts and failures of the project before it occurs. Figure 6.13 takes the constructed as-built models at the end of Feb, May, Aug, and Dec as examples to reveal how the construction work proceeds as time passes. Especially for Figure 6.13 (d), it can be observed that the virtual model and its corresponding point clouds demonstrate a pretty good match, which simply validates the correctness of the virtual visual expression.

For another, process mining relying on the inductive miner is performed to realize the automation of process discovery. As a view on reality, the as-happened construction work can be mapped into a process model on a monthly basis using the tool of ProM (<http://www.promtools.org>). Figure 6.14 and Figure 6.15 show what the process looks like in May from views of the task and worker, separately. The process models are expressed as BPMN and Petri nets with causal relationships of sequence, concurrency, loop, choice, and others. To overcome the complexity in construction, the discovered model is abstracted from noise (i.e. infrequent/exceptional events), and thus only representative behavior covering 99% of records in event logs is taken into account. As a result of model simplicity, the task-centered model in Figure 6.14 preserves 7 core tasks (out of 11 in total), which are executed by 2296 times (out of 2325 total records). Similarly, 7 productive workers remain in the worker-centered model in Figure 6.15, who are responsible for 98.41% of tasks. To be more specific, Figure 6.14 starts with an XOR split to create four clusters of tasks, which are “prefabricated stairs and land” (Cluster 1), “masonry work” and “external facade work” (Cluster 2), “placing window frames” (Cluster 3), and “deposit” (Cluster 4). Tasks in the four clusters can be executed parallelly. Figure 6.15 provides a clear insight into collaboration among workers. In the beginning, Worker 1 involves in the process execution together with Worker 10, or Worker 3 and 4, or Worker 3 and 6. Then, either Worker 7 or 8 takes over the work and finishes it. Moreover, the virtual part in the process model format can be animated to dynamically display sequences of construction work and track the progress over time.

In terms of evaluating the discovered virtual model, the Petri nets in Figure 6.14 (b) and Figure 6.15 (b) directly integrate with the conformance checking, where the first number in the bracket is the number of records aligned correctly with event logs and the

second number represents undesirable deviations between the modeled and observed behavior. Only the part of “external facade work” has the deviation, which is highlighted by the red border frame in Figure 6.14 (b). More precisely, 1.81% of this certain task (12 out of 660) cannot correspond to the event log correctly. It can be seen from Figure 6.14 (b) and Figure 6.15 (b) that there is a relatively high degree of agreement to well match the discovered and actual process. To further measure the quality of discovered models in reflecting the actual behavior from log data, evaluation metrics in Eqs. (6.1) – (6.3) are calculated. As listed in Table 6.7, precision is approximately 0.3 lower than the reply fitness, which means that there is a trade-off between underfitting and overfitting. Fitness and generalization are guaranteed with a value closer to 1, indicating that both the task-centered and worker-centered process models are generalized enough to replay the most executed sequences of events observed in the logs. Precision larger than 0.7 is also acceptable to characterize the process credibility.

Figure 6.13. 4D snapshots for the virtual model at the end of (a) Feb; (b) May; (c) Aug; and (d) Dec. (Note: Point clouds are also provided in (d).)

Figure 6.14. Task-centered process model represented by (a) BPMN; and (b) Petri nets.

Figure 6.15. Worker-centered process model represented by (a) BPMN; and (b) Petri nets.

Table 6.7. Evaluation of the discovered process model.

Model	Reply fitness	Precision	Generalization
Task-centered	0.997	0.698	0.999
Worker-centered	1	0.727	0.981

6.4.3 Diagnosis of construction process

With an understanding of the frequent activities and paths during construction, the discovered process model based on a fuzzy miner can diagnose and foresee the most frequently occurring bottlenecks, which are not visible via observation. Feedback from the diagnosis is expected to strengthen operations and collaboration, bringing an inherent

benefit in construction efficiency enhancement. By Disco Fluxicon software (<https://fluxicon.com/disco/>), the fuzzy model can be generated and simplified into the desired level to be easily comprehended, as shown in Figure 6.16. The average duration spent in the process is projected into the model by the coloring of boxes and the thickness of the arrows. The diagnostic results from process mining and their comparisons with physical processes are summarized below.

(1) Regarding the task-centered model in Figure 6.16 (a), the most significant bottleneck highlighted by the software was the construction path between “Deposit” and “Adhesive work sand-lime brick elements”, which took up 4 days. It is worth noting that the long path “Edge processing – Reinforcement – Deposit – Adhesive work sand-lime brick element” was prone to be slower than others. It can be inferred that the delay in a certain task could pass to negatively influence another, resulting in a chain reaction. Except for the process diagnosis and interpretation from the chart, the actual construction record is also explored to verify the identified bottleneck. During the real construction, there was a lag after the task “Deposit” concerned with casting concrete. After workers finished the activities for curing the concrete on objects, they just waited with nothing to do, leading to a great waste of manpower and time. For this concern, managers can arrange these workers to do other tasks once they complete the “Deposit”. As for a single task, the chart presents that the task named “Masonry work” spent the longest time. It is rational since the real case shows that the task number of “Masonry work” was the largest accounting for 42.80% of workloads in May. In contrast, the task named “External facade work” constituted less than 1% of the total work in the actual process, but it took the second-longest days (10 days) to complete. That is to say, this task should be underlined as a root cause of delays in May.

(2) For the worker-centered model in Figure 6.16 (b), there were two big red arrows in the path of “Worker 3-Worker 1” and “Worker 1-Worker 3”. As an interpretation from the chart, a possible bottleneck between Worker 1 and Worker 3 was recognized by the software, which needed to be investigated at first. One of the reasons causing the particular bottleneck may be a lack of proper cooperation and communication between the two workers. Managers can therefore target Worker 1 and 3 to adjust their inappropriate

workflows and promote greater cooperation. Then, we go back to the actual construction process to check whether the bottleneck shown in the chart has occurred. It could be found that there was actually a real observed record of conflicts between Worker 1 and 3, which was consistent with the process diagnosis from the chart to validate the practicability of the process mining results. To explain the bottleneck in the terms of factuality, that is because both of the workers were carpenters with the same duties. If the work arrangement was unreasonable or communication between them was poor, they tended to take tasks with significant overlapping to slow down the progress. The fact also suggests that it is necessary to optimize the workflow in workers with the same occupations for minimizing duplication in efforts. Besides, although Worker 8 kept active in the highest number of days (19 days), he only completed 3.26% amount of work within the month in the real case. In other words, Worker 8 was more likely to generate delays than other workers participating in May due to his poor efficiency. More instructions should be given to Worker 8, aiming to facilitate him to carry out construction more skillfully and quickly.

(3) Self-loops in “Pedestal sand-lime brick”, “Prefabricated stairs and land” and “Worker1” from Figure 6.16 stood for unnecessary reworks, which should not be ignored. These recognized reworks from process maps were supposedly problematic to cause additional time and costs, which also deserved careful checks and serious consideration. In comparison to the real case, the more amount of reworks actually appeared in the two tasks “Pedestal sand-lime brick” and “Prefabricated stairs and land”, since these finished works were more likely to fail in meeting the acceptable quality criterion. Besides, Worker 1 was an unskilled carpenter without much work experience, who was unable to perform construction tasks in a reliable and efficient manner. The physical truth has proven that the undesired reworks could negatively impact project period and cost, and thus managers should strive to decrease reworks in the pursue of a more linear and branching process.

Apart from the process model, the 4D model provides another intuitive way to visually highlight unwanted bottlenecks. When the possible delays are detected, color schemes can be given to the specific components of the 4D model causing the bottlenecks as a visual representation. For example, Figure 6.17 assigns magenta to the important cause of delay named “External facade work”, and thus this noteworthy part can be easily

distinguished from others. It offers an opportunity in triggering warnings on the possible delays before they emerge in physical conditions. Based on the early warning, managers can provide guidance and adjustment to construction workers ahead of time. In return, workers can take more notice of the inefficient parts, who can then implement corresponding actions to effectively reduce or even eliminate negative effects from potential bottlenecks if possible.

Figure 6.16. Fuzzy process model about May for bottleneck detection: (a) Task-centered model; and (b) Worker-centered model.

Figure 6.17. 4D model visualization of the certain bottleneck in task “External facade work”.

6.4.4 Prediction of construction process

Since the event logs cover 11 months of the construction process, it can be organized into a new dataset with 230 lines and 3 features for time series analysis. As outlined in Table 6.8, each line of the dataset describes daily work using three attributes, including the date, number of finished tasks, and active workers. Remarkably, the number of finished tasks is worthy of being forecasted to describe its variation tendency in a quantitative manner. That is to say, predictions based on time series data are possible to provide an overview of the construction progress in advance, which can instruct real-time decision making in optimizing the work arrangement to ensure satisfactory performance. Since the size of the prepared dataset herein is relatively small, a classical model named ARIMAX is sufficient to capture temporal structures in time series data and achieve promising prediction results. In other words, if we take more effort to build and train a more complex deep learning model, its prediction performance may not exceed the classical ARIMAX model but the calculation cost will undoubtedly increase. In this regard, the ARIMAX model is integrated into the data-driven virtual system for the prediction from a future perspective. It serves to fit the temporal evolution of the construction phase by learning historical data of task numbers along with the outside factor termed worker number.

From the beginning, the Ljung-Box test is performed in the time series data to test its randomness on a series of lags. It returns a p -value smaller than 0.05 to reject the null hypothesis that the original data is white noise. In other words, the time series data embedding patterns deserves in-depth exploration. Then, the meaningful dataset is partitioned into a training set and a test set under an 80%-20% split, where the test set is the most recent end of data (16/10/2015 – 18/12/2015) accounting for typically 20% of the total sample. It can be seen in Figure 6.18 (a) that the original data of task number is non-stationary in nature, which is also checked statistically by the augmented Dickey-Fuller test to accept the null hypothesis that the time series sample has a unit root (p -value >0.05). Since stationary processes with constant mean and variance over time can make reliable predictions with ease, the time series scale is necessarily transformed into the stationarity with a p -value below 0.05 using the first-order difference ($d=1$), as

displayed in Figure 6.18 (b). Thirdly, two important orders q and p in ARIMAX can be roughly identified from ACF and PACF plots, which visualize the correlation of present with lags and the correlation of residuals with the next lags, respectively. It is observed that the second points in Figure 6.19 (a) and (b) fall on the lower edge of the blue area, indicating the levels at which the autocorrelation is significant. Meanwhile, a too complex model with many lags is not required due to its risk of overfitting. Therefore, the value of p and q can be primarily set as 2. To further verify the determined orders, six ARIMAX models under different combinations of p and q have been built in Table 6.9. The examination of the goodness of fit turns out that ARIMAX (2, 1, 2) with the maximal log-likelihood and the minimal AIC and BIC is the best-fitted one for producing dependable forecasts of future points in the time series.

For developing a predictive model, the training set is used to estimate coefficients of the ARIMAX (2, 1, 2) model associated with the lagged worker number as the covariate. Table 6.10 summarizes the optimal coefficients as the weights of each term derived from the maximum likelihood estimation. Notably, a p -value less than 0.05 indicates the statistical significance of all coefficients. Based on the fitted ARIMAX (2, 1, 2) model, we can predict the number of tasks on a certain day relying on the full history up to the day. In Figure 6.20, the predicted value (red line) denoting the number of tasks thought to be executed in the following days is plotted against the true value (blue line), which appears to be in the correct trend and scale. That is to say, the developed model in a satisfactory fit is able to make promising forecasts aligned with the truth well, contributing to evaluating the next construction workload numerically. Also, the red line with the mean value 124.894 is averagely below the blue line with mean 126.348, implying that our predictions are relatively conservative. To better understand the accuracy of prediction, Figure 6.21 (a) visualizes the residual error, which oscillates near zero to demonstrate the great quality of the forecasts. Clearly, Figure 6.21 (b) and (c) reveal that residual errors in both the training set and the test set have approximately normal distributions, which are centered on 0.085 and -1.454, respectively. Although there exists a bias in the prediction, the value of the residual seems acceptable. The negative sign in the average residual error

of the test set also proves that the prediction of construction efficiency is slightly lower than the actual value.

In sum, the developed ARIMAX model allows for learning time series data in the virtual model, which possesses a strong predictive ability in estimating the trend of construction progress in the next few months. It can give back pieces of numerical evidence to managers for schedule design, task allocation, and workflow optimization. For one thing, it seems that the number of finished tasks is on the rising trend as the construction process runs. Hence, managers can reasonably arrange more workers and tasks after Jun. For another, if the manager hopes to fulfill the project ahead of schedule, he'd better focus on the work during Feb – Jun at slow construction speed through optimization of the relevant construction process and worker arrangement. Moreover, since the number of finished tasks estimated by the developed ARIMAX model tends to be slightly smaller than observations, the project duration in the proposed scheduling could be a little longer than the reality. When workers proceed to work as planned, the rate of progress in the physical part is likely to exceed managers' expectations through speedy actions.

Table 6.8. Summary of time series data.

Characteristic	Date	Number of finished tasks	Number of workers
Range	2/2/2015 – 18/12/2015	[98, 131]	[8, 11]
Mean (Std)	—	117.261 (9.572)	9.543 (0.631)
Median	—	120.500	10

Figure 6.18. Plots and the augmented Dickey-Fuller test for: (a) Original time series data; and (b) Stationary data after the first-order difference.

Figure 6.19. ACF and PACF plots for stationary data after the first-order difference.

Table 6.9. Goodness of fit for six candidate ARIMAX models.

Model	Log-likelihood	AIC	BIC
ARIMAX (1, 1, 1)	-284.941	579.881	595.929
ARIMAX (1, 1, 2)	-282.400	576.799	596.056
ARIMAX (2, 1, 1)	-285.019	582.038	601.295
ARIMAX (3, 1, 3)	-282.586	579.173	601.639
ARIMAX (2, 1, 2)	-273.855	565.711	594.596
ARIMAX (4, 1, 4)	-281.653	585.306	620.611

Table 6.10. Coefficient estimation of ARIMAX (2, 1, 2) model.

Item	Coefficient	Std error	<i>p</i> -value	97.5% confidence interval
Constant	-5.641	0.111	0.000	[-5.858, -5.423]
Workers	0.013	0.002	0.000	[0.009, 0.017]
AR. ϕ_1 . Tasks	1.802	0.000	0.000	[1.802, 1.802]
AR. ϕ_2 . Tasks	-0.802	0.000	0.000	[-0.802, -0.802]
MR. θ_1 . Tasks	-0.999	0.078	0.000	[-1.152, -0.845]
MR. θ_2 . Tasks	0.141	0.075	0.000	[-0.007, 0.288]

Figure 6.20. Plots of the forecast line and corresponding true value in: (a) Whole dataset; and (b) Test set.

Figure 6.21. Residual errors in: (a) Whole dataset; (b) Training set; and (c) Test set.

6.4.5 Discussion

Remarkably, the time series data contains lots of hidden knowledge about tasks and workers, which can shed light on the nature of project evolution. Besides, the superiority of ARIMAX in forecasting the construction progress can be further validated based on the comparison against four common time series algorithms. The discussions are summarized as follows.

(1) Characteristics of finished tasks and involved workers can be observed directly from time series data, which can serve as direct evidence for managers in project management. Linear regression and a variation of linear regression in the form of $y \sim \log(x)$ are fitted well along with a 95% confidence interval in Figure 6.22 (a) and (b), respectively, which manifest a growing tendency in the number of both tasks and workers

over the month. That is to say, as a building rises through its floors, more trades can perform work. More workers involved especially after Jun is entirely expected to increase the task number. It has been proved in Figure 6.22 (c) that there is a positive correlation between the number of finished tasks and workers. In particular, 10 or more workers can averagely execute more than 9 tasks each day than workers fewer than 9. Apart from more workers, it can be assumed that the more skilled techniques and closer collaboration can be another method to accelerate the project process. As the construction proceeds, workers will gradually be more and more familiar with the tasks and their co-workers. Accordingly, managers can consider assigning more than 10 skilled workers every weekday in the intermedia-late course of the project.

(2) The developed ARIMAX model is compared with other popular time series algorithms to exhibit its outstanding predictive ability. Specifically, SARIMA and SARIMAX stand for the seasonal ARIMA and ARIMAX model incorporating the seasonal order argument. It is found in Figure 6.23 that predictions from the ARIMAX model (green line) and SARIMAX model (red dash line) show the consistent trend as the true value (blue line), verifying the necessity of exogenous variables in achieving precise forecasting of the task number. Meanwhile, the green line gets much closer to the blue line, indicating that the ARIMAX model is prone to ensure the prediction quality. Although the two lines from AR and ARIMA model taking no account of outside factors can also be near the blue line, both of them have an obvious downward trend, which is just the opposite of the reality. According to evaluation metrics MAE and RMSE in Eqs. (6.15) and (6.16), the performance of five candidate models are measured quantifiably in Table 6.11, resulting in the rank as: $ARIMAX > ARIMA > AR > SARIMA > SARIMAX$. It suggests that our model choice in ARIMAX (2, 1, 2) associated with the number of workers turns out to be the best one under the smallest RMSE (2.635) and MAE (2.204). Noteworthily, SARIMA and SARIMAX considering seasonality are the two most inaccurate models, whose RMSE and MAE are at least 62.24% and 33.89% lower than the most appropriate ARIMAX. That is to say, construction performance does not experience obvious seasonal variation. Besides, a more complex time series model does not always mean better.

Figure 6.22. (a) and (b) Variation of task number and worker month by month; and (c) Relationship between the number of tasks and workers.

Figure 6.23. Comparisons of predictions from different time series algorithms visualized in: (a) Whole dataset; and (b) Test set.

Table 6.11. Evaluation of predictions from different time series algorithms.

Model	RMSE	MAE
AR (1, 0)	3.681	2.930
ARIMA (1, 1, 1)	3.678	2.901
ARIMAX (2, 1, 2) with number of workers	2.635	2.204
SARIMA (1, 1, 0) (2, 0, 1, 5)	4.275	3.334
SARIMAX (2, 1, 0) (2, 0, 1, 5) with number of workers	5.545	6.076

6.5 Chapter Summary

A novel framework of process mining in the BIM-based construction project is proposed to capture and study the nature of the complicated workflow and collaboration during the construction progress. The process mining-based approaches present unprecedented opportunities in automating the simulation and analysis for a series of

construction activities about modeling a building, which is distinguished from the traditional method heavily relying on the expert subjective opinions to be less susceptible to human cognitive errors. What's more, a detailed framework of the digital twin containing a physical model, a virtual model, and connection data is developed based upon the integration of BIM, IoT, and process mining, which has been highlighted as a prime candidate for facilitating the automation and intelligence of construction project management. To be specific, IoT devices are deployed to collect real-time data about the actual status of the construction operation with little manual interaction. The rich data source from IoT data serves as the foundation of the cyber-physical synchronicity, which needs to be mapped into the IFC scheme for model interoperability and then saved as event logs. In other words, these logs are passive data sources embedding a lot of valuable knowledge about what actually happens. For in-depth analysis and smart reasoning, process mining is conducted in log data to keep track of operations and uncover behavior aspects.

In the case study on automated process discovery and analysis, two advanced process discovery algorithms named inductive mining and fuzzy mining are implemented to map the planned flows of construction activities executed by 11 participants from the as-planned event logs into a concise process model. Some meaningful conclusions can be summarized as: (1) The discovered process models are sufficient to replay observed behaviors recorded in logs, since their fitness and precision are larger than 0.8. (2) The model-based process analysis can identify potential deviations, inefficiencies, and collaboration patterns during the construction from three viewpoints, including process, time, and organization, instead of specialist experience and judgment. Accordingly, project managers can promptly adjust the construction timetables and strategies in a data-driven manner, aiming to avoid reworks, bottlenecks, and poor collaboration in processes as much as possible. (3) Based on the site survey, truthful information about certain persons causing the actual delay can be gained, serving as a valuable supplementary data source for further understanding the SNA-related results from process mining. It has been found that participants, who are identified as the group leaders with relatively high centrality value in the three established social networks, are more likely to cause unwanted

delays and discrepancies to slow down the process. A possible explanation is that the critical participants in the network often handle more tasks and connect to more people, who could sometimes be disorientated and stressed at work. Therefore, project managers should focus more on these group leaders to regularly check whether their work goes smoothly.

In the case study on digital twin implementation, a semantic construction digital twin is created in a constant loop between the physical and virtual parts for continuous process analysis, prediction, and optimization, which largely relies on the point clouds taken by IoT devices during the real-time operational monitoring. The key points of the established digital twin can be outlined as: (1) The BIMserver on the cloud act as the data repository to continuously synchronize with IoT data, interpret IoT data into proper formats, share and communicate data. These updated data can be passed to the cyber world as the source for automatically building the virtual model paired with physical features and conducting an in-deep analysis for tactical decision making. (2) The virtual model can be built in two formats with identical fidelity, namely 4D visualization and process models, both of which emphasize the nature of task execution and worker collaboration through process simulation. Especially for the process model, it is established in the view of tasks and workers, which can well reply to the event log with the value of fitness and generalization around 1 and precision larger than 0.7. (3) The virtual model encompasses three critical process mining algorithms, which are inductive mining, fuzzy mining, and ARIMAX (2, 1, 2) model associated with the lagged worker number in the minimal RMSE (2.635) and MAE (2.204). Besides, since construction is an ongoing process, continual data influx can be collected from the construction site and send to the cyber world in a machine-interpretable way. These new data will undoubtedly facilitate the updating of the existing virtual model and algorithms in real-time. Therefore, the virtual counterpart is able to grasp the changeable situation to support automated progress monitoring and timely services in terms of problem diagnosis and process prediction. It is worth noting that the potential benefit of the updated model with high fidelity is to make evaluation, prediction, and decisions dynamically, driving the digital twin to be more adaptive and intelligent. On the one hand, bottlenecks causing delays can be constantly detected to issue immediate

warnings. On the other hand, predictions about future problems and progress can be gained over time to realize performance assessment for optimization purposes. When more and more data are fed into the time-series forecasting model, the RMSE and MAE are expected to be lower than the present results. Although the digital twin has promised remarkable potentials in IoT and AI integration, it still owns an associated uncertainty from the data connection between the cyber-physical world. It is known that the data transmission efficiency and quality will exert great impacts on real-time analysis. In short, both the IoT and process mining contribute to making these digital replicas far more useful. The virtual part is expected to output suggestions dynamically to guide the physical process, which can even respond to changes in the real construction site. Eventually, managers can formulate more rational construction scheduling with well-arranged workloads and workers, aiming to promptly improve operational efficiency and strengthen cooperation in the physical construction process.

CHAPTER 7. CONCLUSIONS AND FUTURE WORKS

7.1 Conclusions

Acting as a promising and emerging technology, BIM has been utilized more and more in AECO to speed up the digitalizing process in the old construction industry, which can provide information solutions in the life-cycle management for infrastructure systems. BIM can be seen as a data repository to store massive data gathered from data-rich objects, inputs, documents, sensors, building management tools, and others during project execution (Eastman, Eastman et al. 2011, Peng, Lin et al. 2017). As the adoption of BIM grows, the amount of BIM data will increase exponentially, resulting in some characteristics of “big data” (Pan and Zhang 2020). It is easy for BIM data files to reach a large size in dozens or hundreds of gigabytes (Ding and Xu 2014). For instance, the BIM project for an airport terminal with 548,300 m² can reach approximately 50 GB, which is saved within a scalable NoSQL database in a cloud environment (Lin, Hu et al. 2016). This kind of heavily accumulated data captures details of the parametric model and executing process to offer affluent evidence for decision making, which are worthy of deep exploration to seek hidden knowledge and further enhance the value of BIM (Pan and Zhang 2020).

It should be noted that a kind of BIM data named event logs can be automatically generated and heavily accumulated during the BIM implementation. These vast sources of process-specific data record details of model evolutionary and task execution in chronological order, which is believed to contain a wealth of hidden knowledge. However, the previous study in the topic of BIM event log mining is still rare. Since the adoption of AI has gained significant attention, I also intend to perform several AI-related methods to reveal meaningful insights into the available BIM event log data in great volumes. It has been found that various AI techniques have created tremendous value in the digital revolution, leading to a more reliable, automated, self-modifying, time-saving, and cost-effective process of construction project management. In contrast to traditional

computational methods and expert judgments, the promising AI is superior in dealing with complex and dynamic problems under great uncertainty and intensive data. To sum up, the significance of the proposed data-driven BIM event log mining lies in facilitating the automation, digitalization, and intelligence of advanced project management, which could be less susceptible to human cognitive errors. From the level of knowledge, experiments based on several AI approaches have been done in event logs from the real-world projects, contributing to converting data into the strategic value of information for process understanding, pattern extraction, and trend prediction in the complex construction project. From the level of application, the usage of AI techniques, in turn, gives the objective evaluation of design/construction performance and provides continual feedback about developing and adjusting project planning and staffing to maximize efficiency, reliability, and sustainability, which can greatly reduce the dependency of decision making in project management on expert knowledge and subjective judgment.

7.1.1 Key methods

In general, the steps of AI-based approaches include data acquisition and preprocessing, data mining based on appropriate models, and knowledge discovery and analysis. Figure 7.1 summarizes the methods utilized in each research objective to maximize the BIM benefits from the data layer. These methods can be grouped into four major categories. To be more specific, statistical models employ mathematical equations to inference the relationship between variables, which is a simplified method to approximate reality. Machine learning aims to teach machines how to discover patterns hidden in large data and realize data-driven predictions on future tasks. As machine learning evolves, deep learning has been developed at a higher level to be a new trend. Deep learning inspired by the neural networks of human brains is made up of multiple processing layers to process information, represent features, and gain knowledge. Besides, process mining acts as a young discipline between machine learning and process modeling, in order to support tasks of discovering, monitoring, and improving the physical processes under high complexity.

Figure 7.1. Summary of adopted methods

7.1.2 Key contributions

Research objective 1 presents the deep learning-based approach to learn sequential data from logs and predicts the next possible design commands at the categorical level towards automation of the design process, which has the potential to improve the modeling efficiency and quality. Its contributions can be summarized as: (1) The state of knowledge is to build a deep learning model with optimal parameters to learn features of temporal data from the large BIM design event log files, which is able to intelligently and accurately predict the next type of design command during the execution phase of modeling. (2) The state of practice is to provide the three most possible command classes to minimize the randomness and uncertainty in the prediction results, which can act as data-driven command recommendations to instruct the modeling process. With the help of the predicted results, designers can simply follow the suggested command to enhance the design efficiency and reduce the likelihood of possible wrong commands.

Research objective 2 performs hybrid clustering algorithms with high-quality clustering results and rapid convergence rates to reveal hidden patterns of designer's performance. This clustering-based approach is helpful in understanding work habits and

measuring design productivity objectively. Its contributions can be summarized as: (1) From the state of knowledge, it develops a novel clustering method named AEFKCN based on EFKCN and a self-defined CVI S_{new} . To be more specific, AEFKCN owns a self-adaptive learning rate to speed up the clustering process in determining cluster centers and taking clusters apart. Besides, AEFKCN incorporating the merits of the neural network and fuzzy theory can provide a more feasible way to handle a large amount of log data with great complexity, uncertainty, and randomness, resulting in high-quality clusters. Experiments in public datasets and real logs all verify the great competitiveness of AEFKCN in computational efficiency and cluster quality. As for another important task of cluster validation, a new CVI S_{new} based on boundary points is defined to work together with common CVIs (i.e., SI, CHI, and DBI). Emphatically, S_{new} owns inherent advantages in reducing computational complexity and dependency on cluster centroids, which is no longer restricted in spherical clusters. (2) From the state of practice, it seeks similarities among BIM design event logs rapidly and effectively to group design productivity into the high, medium, and low level. In other words, these extracted meaningful patterns can serve as concrete evidence to assess a designer's performance without unnecessary individual bias. Accordingly, managers can inform data-driven decisions to strategically make personalized work arrangements for different designers, thereby allowing a more efficient modeling process.

Research objective 3 explores the mass of BIM design logs based on a novel viewpoint of the social network. Its contributions can be summarized as: (1) It proposes a novel community detection approach named *node2vec-GMM* with the combination of the graph embedding algorithm *node2vec* and the probabilistic clustering algorithm GMM, aiming to output several possible clusters with densely linked designers; (2) It quantifies and predicts designers' influence from a self-defined metric (the impact score) and a newly-developed machine learning algorithm (CatBoost model). More specifically, I define a new metric named the impact score under the combination of k-shell and node's 1-step neighbor for measuring the influence power of designers, which assumes that the node with more neighbors and these neighbors have fewer overlapped neighbors can facilitate information to flow more broadly across the given network. The new metric is

proven superior over conventional centrality measures that tend to suffer from inaccurate ranking. Meanwhile, there is a moderate correlation between the impact score and features concerning the designer's behavior, which can be utilized to roughly estimate how the designer's operation will affect his influence power within the collaboration network. Moreover, I deploy the newly-developed machine learning model called CatBoost to predict the designer's impact score based on his structural and behavioral effects, driving the process of project monitoring and management more intelligent. Since it needs no local information on the network structure, it could be an effective way to relieve the computational burden in measuring the strength of the designer's influence; (3) For the practical value, it quantitatively understand the information transmission, individual roles, and possible links between pairs of designers, which can be an effective tool to not only monitor the BIM-based collaborative design process, but also support managers to better evaluate designers' performance, allocate design tasks, and formulate collaboration strategies with low uncertainty and subjectivity towards a sustainable modeling process. In short, the SNA-based methods for BIM log mining hold the promise of promoting design collaboration and raising design efficiency through better leadership and work arrangements formulated by managers in a data-driven manner.

Research objective 4 proposes a process mining-based framework to simulate and analyze activities of modeling a building during the construction process, aiming to discover potential problems and evaluate the performance of workflows and participants objectively. Furthermore, this idea can be employed in developing a closed-loop digital twin integrating a physical model, a virtual model, and a database to tie them. This mathematical digital twin under the integration of BIM, IoT, and DM facilitates data communication and exploration to make the complex workflow more understandable and predictable. Its contributions can be summarized as: (a) From the point of knowledge, it deploys process mining techniques to easily discover and visualize the participant-specific process based on the BIM construction event log and then make in-depth analysis in an efficient and objective way; (b) From the point of practicability, process mining helps in detecting the potential deviations, delays, and collaboration patterns based on data instead of specialist experience and judgment, which can serve as strong evidence to propose

solutions for process improvement in the early stage and make quantitative evaluations on participants' performance; (3) As for the digital twin, advanced DM techniques gain deep insights into massive IoT data gathered from the physical side and stored in the cloud BIM, which offers a comprehensive view of the entire process and realizes process simulation, conformance checking, bottleneck diagnosis, and productivity prediction objectively in the virtual space. The analytical results serve as evidence to not only support fast and cost-effective troubleshooting, but also inform strategic decisions to improve the workflows and staffing in the physical world at an early stage.

From a bigger and bolder view, the positive impacts of the proposed innovative technology on the state of design management in practice can be highlighted as high efficiency, risk mitigation, objectivity, and digitalization. The four critical opportunities of BIM event log mining in handling construction projects with inherent complexity and uncertainty have been outlined as follows: (1) High efficiency: The use of AI can make the design and construction phase run more smoothly and efficiently. For example, deep learning can capture the temporal dynamics of design commands to reliably predict sequential design commands, and thus the personalized command predictions can serve as operation reference to speed up modeling and avoid unnecessary operation mistakes, enabling an easier modeling procedure. Process mining can generate valuable insights into the complicated construction procedure, such as tracking key workflows, predicting deviations, detecting invisible bottlenecks, extracting collaboration patterns, and others. Tactical decisions can therefore be informed to guide the optimization of the construction execution process for improvement of operational efficiency, contributing to reducing reworks and conflicts, potential delays, and poor cooperation. (2) Risk mitigation: AI-related methods can be applied to learn data collected from BIM-enable projects to foresee the possible problems. Therefore, assistive and predictive insights on critical issues can be revealed to help project managers quickly prioritize possible risks and determine proactive actions instead of reactions for risk mitigation, such as to streamline operations on the job site, adjust staff arrangement, and keep projects on time and budget. In other words, AI presents valuable opportunities to realize early troubleshooting to prevent undesirable failure and accidents in the complex workflow. (3) Objectivity: The design

performance can be assessed in an objective manner, which no longer heavily relies on the traditional method by managers' subjective judgment and experience that could be unreliable and biased. The objective measure of performance by clustering-based or SNA-based methods can return valuable feedback across weekly, monthly, quarterly, and yearly timescales, which can help managers more reasonably plan and schedule personnel to maximize the work performance. (4) Digitalization: The integration of BIM and various data mining methods is playing a crucial role in digitalizing the construction industry, which has gone far more than the 3D modeling to provide a pool of information concerning the full project lifecycle. For one thing, BIM provides a platform for not only collecting large data about all aspects of the project, but also sharing, exchanging, and analyzing data in real-time to achieve in-time communication and collaboration among various participants. For another, the rich BIM data can be fully explored, and thus immediate reactions can be performed to streamline the complicated workflow, shorten operation time, cut costs, reduce risk, optimize staff arrangement, and others. Remarkably, since the digital twin has shown superiority in easily transforming massive data into useful knowledge, it will be the next digital frontier of the construction industry for pursuing a higher degree of digitalization. Overall, the practical value of the hybrid framework based on BIM event log mining lies in addressing challenges arising from characteristics of construction project management, including uniqueness, labor-intensive, dynamics, complexity, and uncertainty. It will deliver promises on prediction, optimization, and decision making, aiming to assist the traditional construction industry to catch up with the fast pace of automation and digitalization.

7.2 Future works

For research objective 1, the future works can be performed as follows: (1) I plan to implement the proposed command prediction approach as an *Autodesk Revit* plugin for a better user experience. It is supposed that users can quickly and easily click the three recommended command classes along with their relevant commands on the screen provided by the *Revit* plugin to complete modeling, leading to a simpler, more reliable and efficient design phase. In particular, the “skip” option should be designed in the plugin,

and thus designers can simply click it to minimize unnecessary misleading when no correct classes appear in the recommendation list. The *Revit* plugin will be used in a design company to test its effectiveness in improving design efficiency and reliability. (2) A potential pitfall in implementing LSTM NN is that it has difficulty in providing correct predictions for non-dominant commands. It is advisable to optimize the LSTM NN by incorporating useful algorithms for learning from imbalanced data streams with concept drift (Wang, Minku et al. 2018), which can achieve a more balanced degree of model performance in predictions for every class. Another way is to add more non-dominant command records in the dataset to sufficiently large numbers, which can increase the likelihood of making correct predictions for them. (3) I will continuously expand the dataset by adding more commands executed from different designers and projects. It is notable that when the data size for an individual designer or a single project grows large enough, the LSTM NN can return the more accurate prediction. Therefore, LSTM NN is able to offer personalized suggestions about design command classes with the strong capability of studying design preferences for the particular designer. Similarly, LSTM NN can also learn a huge amount of data about one project to make predictions to meet the characteristics of the project. (4) When the size of the dataset grows large enough and each command can get enough records, I can try to predict the next command instead of the next command class. It is assumed that providing the specific command to designers is more instructive in practice, which can potentially bring about greater improvement in the modeling process.

For research objective 2, the future works can be performed as follows: (1) The parameters of the EFKCN/AEFKCN algorithm are sensitive to the clustering results. Since it is a hard and time-consuming task to set the appropriate value of these parameters in the clustering model, a more efficient method for parameter initialization should be considered to avoid subjectivity and enhance efficiency in parameter determination. Since some researchers have made attempts to more efficiently initialize parameters in *K*-means (Celebi, Kingravi et al. 2013) and FCM (Zou, Wang et al. 2008, Tan, Lim et al. 2013), I can refer to them to put forward a reliable initialization scheme for EFKCN/AEFKCN. (2) The quality of the model in *Revit* established by a designer is another great concern.

Although a designer can be productive during the modeling process, it is an issue that he could possibly build models in very poor quality, which are useless. Therefore, the idea of model evaluation need to be combined with the design productivity analysis based on data associated with model quality and design behavior, aiming to improve both quality and efficiency in the modeling procedure. (3) The clustering-based approach offers new insights into the designer's working productivity, resulting in potential recommendations of work arrangements to accelerate modeling. Although these data-informed decisions can be made in a fast and objective manner, they take no account of additional factors in terms of environment, psychology, and others, and ignore to discuss reasons of high or low productivity within a duration of time. In this regard, I plan to synthetically consider clustering results and additional factors to make the recommendations more sensible and practical. Such recommendations from the comprehensive analysis can potentially adapt and respond to the participants, local conditions, and dynamic changing processes towards a more smooth design procedure.

For research objective 3, the future works can be performed as follows: (1) The proposed *node2vec-GMM* algorithm can be further modified for better clustering performance. For instance, GMM can be integrated into the deep neural network as a softmax layer (Tüske, Tahir et al. 2015). The algorithm can be adjusted to the combination of network structure and node attributes, which can definitely generate more trustworthy clusters and work assignments. (2) More potentially relevant features, such as the designers' seniority, educational background, and others, should be taken into account, in order to make the evaluation of designer's influence more reliable. In particular, the ranking of designers, like the cluster leader, senior, junior, etc, inside the company will exert an impact on the social and leadership behavior. It is necessary to take the actual ranking as an important feature in future analysis. (3) Since the research points out that the concept of 2-step neighborhood stands a chance to enhance the ranking performance for the node influence (Liu, Tang et al. 2016), I can consider incorporating both the 1-step and 2-step neighbor into the new metric for better measurement of designers' influence in the dynamic information propagation, which is particularly beneficial for networks in a large size. (4) The proposed network analysis framework can be applied to the BIM-

enabled full-cycle project management to explore collaboration among participants, including architects and MEP/HVAC/Structure engineers, which can possibly reduce errors, failure, time, and cost during the whole life of the project. (4) The current validation relies on evaluating clustering and prediction performance by some popular metrics, like ARI, AMI, MSE, MAE, and R^2 . However, such the validation part is still not so strong since it takes no account of the real-world experiences of actual designers. For this concern, a potential practice in evaluating the results of SNA is to compare the analysis results against the actual situation. The agreement between the established network and the observed behavior in the design process can be carefully examined by experts. For example, we can check whether the key designers recognized by the SNA are the real leaders to exert more impacts on information transmission and work control. We can measure how much the suggested work arrangement formulated by SNA can improve the design efficiency and cooperation degree. If the experts think that the satisfactory agreement has been achieved, the confidence level of SNA results can be proved, otherwise, I need to adjust the established network and analytical methods to pursue a closer agreement. Besides, I can conduct some surveys among the designers to receive their ideas and feedback in this regard. To connect the design industry with the feedback from data analysis in real-time, I consider embedding the proposed network-enabled approach into a cyber-physical system, a computer system with mechanisms controlled or monitored by intelligent algorithms. In other words, the cyber-physical system as a prototype of the digital twin can bring advances in monitoring and controlling the design procedure under a feedback loop, which provides a basis for delivering smart construction services with increased information cohesion through integrated physics and logic.

For research objective 4, the future works can be performed as follows: (1) The construction supply chain, which allows for transparency and logical alignment of information and coordination (Deng, Gan et al. 2019), can be considered to incorporate with the process mining for tracking the material logistics and construction activities. This hybrid method in BIM event log mining is capable of supporting project managers to not only draw up high-efficiency construction plans but also achieve on-time and cost-saving deliveries of material. (2) To better emphasize the practicability and superiority of the

process-mining-based method for intelligent project management, I intend to quantify how much the risk of failure is reduced and the efficiency is raised after the proposed approach is implemented. (3) The original dataset in this research only contains construction tasks associated with physical objects, which is insufficient in evaluating construction productivity. The fact is that more than half of works on buildings, like material preparation, pre-assembly, tool delivery, erection of temporary facilities, and others, are not in proximity to objects. It is necessary to take into account all kinds of actual behavior beyond object modeling, in order to prepare a more reliable database for rational use of productivity measurement. (4) As the project goes on, more and more data will be accumulated in the BIM platform. To deal with the huge amount of data for time-series forecasting, we can refer to the more complex algorithm, like RNN and LSTM NN. These deep learning models are powerful in catching the nonlinear relationships between variables, which can yield better results for long-term modeling. (5) I only rely on a single monitoring source that is the point clouds from a UAV in this case for simplicity. Although a single data stream is easy to obtain and explore, it is inadequate to reveal the complex nature of a large-scale project in reality. Therefore, multiple sources of monitoring data should be collected and merged in future studies. Besides, it is expected to provide detailed information about the occupations of workers to better explain the construction logic from the stem of data. In the end, more meaningful interpretations of what people were actually doing and what the meaning of the patterns is can be generated.

Apart from each research objective, the direction of future work can be determined based upon the full text of this thesis. First of all, the generalization of results from all case studies is worthy of exploration, since addressing design problems with generic suggestions is still challenging. To my point of view, the extension of obtained research findings to other similar projects paves an easy way in developing more effective and more collaborative project environments. The following key points may help to yield potentially generalizable results to drive the rapid digital transformation in construction project management. It will help to better understand results from the previous clustering-based investigation for intelligent decision-making. Firstly, if some patterns/processes occur frequently, there are reasons to believe that they will continue under similar

circumstances in the future. This assumption is based on statistical probability. For example, it has been found from Chapter 4 that nine designers (Designer #1, #2, #3, #4, #9, #18, #24, #32, #40, #45, and #52) are more likely to keep relatively high design efficiency. Therefore, to plan a new design project, managers can assign these nine designers to different design teams and hope them to lead other senior designers within a team for fast modeling. Another example is that Chapter 5 has discovered three potential communities in each of which designers show stronger cooperation and more frequent information exchange. Since more efficient information exchange and communication tend to occur in a community instead of cross groups, it's better to make these designers from a community work together in the future work. Secondly, machine learning in Chapters 3 – 6 can iteratively sense and learn data from the previous construction project to automate analytical model building for perception, knowledge representation, reasoning, problem-solving, and planning. That is to say, when new data from other new construction projects is fed into the machine learning model, the model performance in terms of accuracy and efficiency can be improved over time. In return, better decisions that adapt to the changeable environments can be informed. Due to the advantages of machine learning in continuous improvement, no need of human intervention, and ease of pattern and trend identification, various machine learning-related algorithms have become more and more popular to handle complicated and ill-defined problems in different construction projects in an intentional, intelligent, and adaptive manner, contributing to a smarter decision-making process on the physical asset under less dependency on human experience and knowledge. Thirdly, similar ideas and methods about BIM event log mining can be broadly applicable to other construction projects. Since the code and framework for the topic of behavior prediction and evaluation, process modeling and mining have been well prepared and tested, they can be simply used to analyze new projects. That is to say, it is unnecessary to spend a lot of time developing new data mining approaches. Only small adjustments need to be made on the existed method to make it usable for the decision-making process in a new condition.

In the second place, it is known that beyond design production metrics, there are several other factors in the building design process, such as design quality, design

excellence, energy efficiency, sustainability, and resilience. How to take into account various design dimensions to study building design as a whole system is another unsolved problem. In my opinion, a possible solution is to add these additional design dimensions and various data mining methods into a virtual-data-physical integration paradigm, which can boost fast information retrieval and analysis across the full lifecycle of construction projects. For example, in the design stage, if the dimension of design quality, schedule, and cost are taken into account, AI-related techniques are helpful to realize not only the automatic design but also the automatic model checking and planning. In the construction stage, various techniques of Internet of Things (IoT), such as unmanned aerial vehicles (UAVs), augmented reality (AR), location tracking, and others, can be combined with BIM for site monitoring, construction simulation, and safety management, aiming to ensure a smooth construction process. In the O&M stage, the building operational performance needs to be carefully evaluated to discover problems about building energy consumption early. For long-term sustainability, multiple criteria, variables, and constraints can also be synthetically considered to guide the energy renovation interventions and building upgrading. All the envisions mentioned above can be realized through the deployment of a comparatively integrated digital twin that can sync information between the actual work part and data analysis part to make strategic decisions dynamically. In particular, different data mining approaches can be equipped in different stages for achieving various goals. Hence, future work can concentrate on creating a whole system under the concept of the digital twin to facilitate full service throughout the whole lifecycle of the BIM-enabled projects.

Thirdly, the data analytics in this thesis has been conducted at the lowest level of executing the commands, I can expand the research to a higher level of design tasks. Typically, three desirable tools are able to produce an unbiased appraisal of the design process and explain the data analysis results at a macro level, including the pass-fail evaluation, evaluation matrix, and SWOT analysis refers to the strengths, weaknesses, opportunities, and threats. They are useful in discovering problems inherent in the ongoing design procedure and finding solutions for problem-solving. To be more specific, a simple pass-fail evaluation preliminarily checks whether the design fulfills its purpose and meets

defined criteria through raising some evaluation questions. It aims to drive the design task to pool down to manageable levels. Then, with the help of the evaluation matrix in a simple array, experts can carefully compare the design tasks with a set of prioritized criteria and give them a score. They will also provide comments on their ratings and advise on some improvements on events with a low score. In order to reach a final decision, a stronger tool named SWOT analysis can be carried out to provide a more rounded review of the entire design process from different perspectives. Through evaluating potential strengths, weaknesses, opportunities, and threats in the design tasks, we can gain a deep understanding of all positive and negative factors inside and outside the project. As a result, it benefits in forecasting the changing trends and formulating strategic plannings to improve the design process. All three useful methods under the combination of qualitative and quantitative information can be properly added into the developed BIM event log mining framework in future work to explain and assess the design tasks from a macroscopic aspect, contributing to determining the best way forward during the design.

7.3 Future research trends

It is believed that more and more advanced technologies inspired by AI will be implemented and spread to the entire lifecycle of the BIM-based construction project management, driving the digital transformation in the domain of civil engineering. That is to say, BIM has evolved to be the backbone of digital strategies to deliver streamlined workflows, achieving great improvement in efficiency, reliability, and collaboration during the whole lifecycle of a project. Various types of emerging techniques can be coupled with BIM to accelerate digital progress. Herein, I list five hotspots in the near future as the key technological innovators to further embrace innovation in construction. The tremendous potential of these future directions lies in paving a more affordable and effective way to relieve the burden on manual labor and facilitate smart construction management, as presented below.

(1) Smart robotics: Smart robotics have been progressing rapidly to drive a wide range of semi- or fully-autonomous construction applications. There are two broad types of robotics, namely the ground robots and aerial robots (Ardiny, Witwicki et al. 2015).

For instance, construction robots in different functions have developed based on human requirements, which can automate some manual processes and take over repeatable tasks, such as brick-laying, masonry, prefabrication, model creation, rebar tying, demolition, and others. In other words, robots make it easy to transform low-level components (i.e., steel, wood, concrete, etc.) into high-level building blocks. Also, robots can be in charge of some high-risk tasks to protect workers from work-related injuries and accidents. Thus, there are several foreseeable benefits of such robots, including to address the labor shortage, to lower operation costs, to ensure overall quality, productivity, and safety. Regarding the aerial robots, UAV carrying image acquisition systems (i.e., camera, laser scanner, go-pros) are typical representatives. They are the rising trend in land survey, site monitoring, and structure health monitoring, since they can make the procedure easier, safer, more efficient and affordable. Instead of the manual inspecting, UAVs fly over the construction site or even fly into the building structure to take high-resolution images, capture real-time videos, conduct laser scanning remotely, in order to maintain the safety of employees and detect structure defects (i.e., cracks, erosion, blister, spall, etc.). Moreover, machine learning can be deployed to train robots, and thus robots with talent can act more intelligently by learning from a simulation. An issue in the current state is that the adoption of smart robotics has not reached a large scale and the approaches of construction automation still remain at the seed phase (Bock 2015). Therefore, continued effort needs to be put to enhance robot usage by equipping the robot systems with more powerful abilities and merging them into the built environment. As the robot technology becomes increasingly ubiquitous, robots will be used for performing more professional tasks in unstructured environments, which is expected to bring opportunities for future construction automation.

(2) Cloud virtual and augmented reality (VR/AR): The evolutionary path of VR/AR is towards the cloud. Based on the fifth-generation (5G) networks and edge cloud technologies, cloud VR/AR solutions have appeared to speed up VR/AR applications and improve users' experience. For one thing, VR/AR performs as the information visualization technology to realize more interactions between the physical and cyber worlds, where VR simulates the entire situation and AR integrates the information about

the real entities with computer-generated images. Due to the merit of providing an engaging and immersive environment, VR/AR has been tentatively applied to simulate hazardous construction scenarios, which helps managers to easily recognize underlying dangers and issues in the working environment, and then formulate reasonable plans and measures ahead of accidents in a visual and interactive way (Li, Yi et al. 2018). Another common adoption of VR/AR that emerged in recent years is construction engineering education and training (Wang, Wu et al. 2018). Instead of courses taught by professionals, VR/AR technologies can well train workers on the basis of both visualization and experience in real time, aiming to strengthen workers' cognitive learning and safety consciousness and even raise overall productivity. For another, the 5G evolution is fast enough to stream VR and AR data from the cloud. That is the say, the significant advances of cloud VR/AR root in cloud computing and interactive quality networking, which can effectively strengthen the data processing capability from the local computer to the cloud and then make real-time perception along with responsive interactive feedback. As for the future work about construction safety instruction and evaluation, it is desired to design a cloud architecture of VR/AR under the integrated applications of virtualization, cloud computing, edge computing, AI techniques, network slicing, and others. As expected, it can rapidly process imagery data from different cloud VR/AR services for supporting a rapid and automatic process of as-built model generation, and thus the immersive and intuitive scene information can be revealed for risk evaluation. Moreover, another potential topic is to configure cloud VR/AR with BIM to further maximize the value of BIM. The integration of cloud VR/AR and BIM can visualize and immerse the physical context of the construction activities into the real environments, which is expected to bring various benefits, such as to make the complex interdependencies between tasks more explicit, to make people literally walk into buildings for a better understanding of the project, facilitate onsite assembly with fewer unnecessary mistakes, and others (Wang, Love et al. 2013, Wang, Truijens et al. 2014).

(3) Artificial Intelligence of Things (AIoT): AIoT is the new generation of IoT, which incorporates AI techniques into IoT infrastructure for more efficient IoT operation and data analysis. To be more specific, IoT can be defined as a network of interconnected

physical devices, like sensors, drones, 3D laser scanner, wearable and mobile devices, radio frequency identification devices (RFID), which is attached to construction resources to collect real-time data about the operational status of the project. Many studies have focused on developing some smart IoT-based sensing systems to feasibly track the progress, monitor the worksite, which are expected to support continuous project improvement and accident prevention (Kanan, Elhassan et al. 2018). In the meantime, the huge amount of recorded data can be shared over a network, and then be analyzed deeply by various AI methods to offer actional insights for better supervision and decision making. In other words, AIoT solutions for the construction industry rely on real-time data transformation and instantaneous data analysis. Since AIoT is empowered by AI, its superiority over the traditional IoT lies in providing analysis and control functions for intelligent decision making. Through synthesizing and analyzing data collected via IoT infrastructure in unprecedented volumes and rates, it can automate the real-time decision making at an operational level to remotely control the construction worksite, optimize the project performance, and predict future conditions for the maintenance planning (Louis and Dunston 2018, Cheng, Chen et al. 2020). However, the practical use of AIoT is still in the startup phase, since this new technology still has some wrinkles to work out, like the edge computing issue, security issue, and others. Besides, a literature review reveals that the BIM-IoT integration is increasingly beneficial in several prevalent domains, like construction operation and monitoring, health and safety management, construction logistics and management, facility management (Tang, Sheldon et al. 2019). That is to say, BIM offers an information delivery and management platform, while IoT provides a steady flow of time-series data. Accordingly, it can be envisioned that the synergy between AIoT and BIM under 5G wireless communication will become the hot spot in future works, which can considerably promote the efficiency of the data collection, data transmission, data processing based on cloud computing towards smart home, smart city, and smart construction industry (Mo, Zhao et al. 2020).

(4) Digital twin: The digital twin is a realization of the cyber-physical system for visualization, modeling, simulation, analyzing, predicting, and optimizing. It incorporates three key components, namely the physical entity, virtual entity, and connection of data,

to form a practical loop (Min, Lu et al. 2019). Typically, there are two ways of dynamic mapping in the digital twin (Qi and Tao 2018). On the one hand, inspection data is collected in the physical world, which is then transferred to the virtual world for further analysis. On the other hand, simulation, prediction, and optimization are performed in the virtual model by learning data from multiple sources, which can provide immediate solutions to guide the realistic process and make it adapt to the changeable environment. As evidence from literature (Boje, Guerriero et al. 2020), more attention has been paid to the inclusion of BIM, IoT, and data mining techniques into the digital twin, aiming to deliver smarter construction services. More specifically, BIM as a digital representation can be the start point of the digital twin, and the web-based integration of IoT gathers a large amount of data to enrich BIM. Both the as-built and as-designed models can be accessible in the digital twin, where information from these two parts can continuously exchange and synchronized. To maximize the strength of data, various data mining and AI techniques are leveraged to make digital twins generic across the board domains for automated monitoring of site progress, early detection of potential problems, optimization of construction logistics and scheduling, value chain management of the construction company, evaluation of structural health, and others. Due to industry trends, the research attempts on the development of digital twins will continue to increase. Except for the buildings and other infrastructure assets, the next point can focus on the practical use of digital twins under cloud computing and IoT-based services at the city level integrating heterogeneous sub-assets, like buildings, utilities, transportation infrastructure, and people (Lu, Parlikad et al. 2020). Besides, VR simulation can be paired with the human-centered digital twin to model, monitor, and predict a person's cognitive status, which is expected to become a key component of the future infrastructure equipped with smart information and communication technology in smart cities (Du, Zhu et al. 2020).

(5) Blockchain: A nascent technology called blockchain is a powerful shared global infrastructure, which is originally utilized for simplifying and securing transactions among parties (Turk and Kline 2017). Basically, the concept of blockchain can be explained as a verified chain with blocks of information, and each block embodies data associated with processes in a trusted environment. That is to say, history data along with modifications

can be saved across a network and protected by cryptographic technology. Since the blockchain builds a distributed ledger, all users of the network can access the stored digital information concurrently. Once a block is entered and verified, no modification is allowed in the information. In the same way, blockchain in construction can aggregate the adaptable and scalable knowledge into a shared dashboard, and thus the project management systems can be converted into a more transparent and secure practice. As literature shows, the key opportunities of blockchain in CEM lie in the built environment for smart energy, cities, government, homes, transportations, and others, which are still insufficiently developed (Li, Greenwood et al. 2019). For example, blockchain can be served as a decentralized, transparent, and comprehensive database for the improvement of built asset sustainability, resulting in a more inclusive and reliable process for the project lifecycle assessment (Shojaei, Wang et al. 2019). It can also be combined with BIM to collect large data from various stages of the project and share data securely among stakeholders, aiming to support life-cycle project management (Wang, Wu et al. 2017). The BIM model can be updated timely when it receives the next block of information. Therefore, project delivery can become automated and streamlined, achieving improved productivity, trustworthiness, and cost. In addition, the creation of a smart contract written into code is another critical application of blockchain to enforce the expected behavior by itself and reduce payment fraud (Ahmadisheykhsarmast and Sonmez 2018). The process will only be executed when the corresponding criteria are satisfied, resulting in high accuracy, compliance, transparency, cost-effectiveness, and collaboration in activities, like payment, contract administration, and others.

(6) Synthesis of human-machine intelligence: Although BIM and AI attempt to boost the high degree of automation and digitalization in construction, human intervention and communication are still an indispensable part across the lifecycle of a project. Therefore, it is necessary to incorporate human factors, such as behavior and psychology, into the BIM-enabled project to form a complicated socio-technical system and realize human-automation interactive decision making. This can also be a future direction to better automate the production of engineering designs and the execution of complex and interdependent tasks in digital environments, resulting in more reasonable decisions. To

be more specific, exploration of human influence paves a new way to empower human performance, which will take combined action with AI to facilitate more reliable and efficient construction. It is suggested to adopt more advanced sensing technologies, such as natural language processing (NLP), computer-vision-based human tracking, wearable devices, and others, to monitor human activities from both the physical and cognitive aspects (Zhang, Tang et al. 2017). These collected data in large volumes offer a basis for understanding the uncertainty in human factors, which can be tightly integrated with BIM and data mining methods towards a human-in-the-loop cyber-physical system (Schirner, Erdogmus et al. 2013). Such a close loop containing human, cyber parts, and physical parts can be reasonably regarded as the knowledge fusion from civil engineering, computer science, and psychology. It can well support the human-in-the-loop simulation, analysis, and decision making by dynamically considering the complex interaction in human, tasks, and environments, contributing to extracting important insights into the ongoing projects for reliable diagnosis, prediction, and optimization and achieving the proactive improvement for quality, safety, and efficiency assurance. With the advent of human-machine intelligence, the social-technical-based project management can be implemented, resulting in more promising decisions that feasibly adapt and respond to the participants, local conditions, and dynamic changing processes in real time.

REFERENCE

- Abdi, H. and Williams, L. J. J. W. i. r. c. s. (2010). "Principal component analysis." **2**(4): 433-459.
- Adamic, L. A. and Adar, E. (2003). "Friends and neighbors on the Web." Social Networks **25**(3): 211-230.
- Ahmadisheykhsarmast, S. and Sonmez, R. (2018). Smart contracts in construction industry. 5th International Project & Construction Management Conference.
- Ahmed, A., Shervashidze, N., Narayanamurthy, S., Josifovski, V. and Smola, A. J. (2013). Distributed large-scale natural graph factorization. Proceedings of the 22nd international conference on World Wide Web, ACM.
- Ailenei, I., Rozinat, A., Eckert, A. and van der Aalst, W. M. (2011). Definition and validation of process mining use cases. International Conference on Business Process Management, Springer.
- Akaike, H. (1998). Information theory and an extension of the maximum likelihood principle. Selected papers of hirotugu akaike, Springer: 199-213.
- Al Hattab, M. and Hamzeh, F. (2015). "Using social network theory and simulation to compare traditional versus BIM-lean practice for design error management." Automation in Construction **52**: 59-69.
- Al Hattab, M. and Hamzeh, F. (2018). "Simulating the dynamics of social agents and information flows in BIM-based design." Automation in Construction **92**: 1-22.
- Alahi, A., Ramanathan, V., Goel, K., Robicquet, A., Sadeghian, A. A., Fei-Fei, L. and Savarese, S. (2017). Learning to predict human behavior in crowded scenes. Group and Crowd Behavior for Computer Vision, Elsevier: 183-207.
- Alizadehsalehi, S., Yitmen, I., Celik, T. and Arditi, D. (2018). "The effectiveness of an integrated BIM/UAV model in managing safety on construction sites." International journal of occupational safety and ergonomics: 1-16.
- Almeida, A. and Azkune, G. (2018). "Predicting human behaviour with recurrent neural networks." Applied Sciences **8**(2): 305.
- Almeida, A., Azkune, G. and Bilbao, A. (2018). Embedding-level attention and multi-scale convolutional neural networks for behaviour modelling. 2018 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation IEEE.
- Analytics, D. D. (2014). "The business value of BIM for construction for infrastructure 2017." Smart Market Report: 1-68
<https://www62.deloitte.com/content/dam/Deloitte/us/Documents/finance/us-fas-bim-infrastructure.pdf>.
- Andrews, R., van Dun, C. G., Wynn, M. T., Kratsch, W., Röglinger, M. and ter Hofstede, A. H. (2020). "Quality-informed semi-automated event log generation for process mining." Decision Support Systems: 113265.
- Antonio, S.-A., José D, M. n.-G., Emilio, S.-O., Alberto, P., Rafael, M.-B. and Antonio J, S.-L. (2008). "Web mining based on Growing Hierarchical Self-Organizing Maps:

- Analysis of a real citizen web portal." Expert Systems with Applications **34**(4): 2988–2994.
- Antwi-Afari, M., Li, H., Pärn, E. and Edwards, D. (2018). "Critical success factors for implementing building information modelling (BIM): A longitudinal review." Automation in construction **91**: 100-110.
- Arayici, Y., Coates, P., Koskela, L., Kagioglou, M., Usher, C. and O'Reilly, K. (2011). "Technology adoption in the BIM implementation for lean architectural practice." Automation in construction **20**(2): 189-195.
- Arbelaitz, O., Gurrutxaga, I., Muguerza, J., Pérez, J. M. and Perona, I. (2013). "An extensive comparative study of cluster validity indices." Pattern Recognition **46**(1): 243-256.
- Ardiny, H., Witwicki, S. and Mondada, F. (2015). Construction automation with autonomous mobile robots: A review. 2015 3rd RSI International Conference on Robotics and Mechatronics (ICROM), IEEE.
- Arnaiz-González, Á., Díez-Pastor, J.-F., Rodríguez, J. J. and García-Osorio, C. (2018). "Local sets for multi-label instance selection." Applied Soft Computing **68**: 651-666.
- Arnaiz-González, Á., González-Rogel, A., Díez-Pastor, J.-F. and López-Nozal, C. (2017). "MR-DIS: democratic instance selection for big data by MapReduce." Progress in Artificial Intelligence **6**(3): 211-219.
- Asuncion, A. and Newman, D. (2007). UCI machine learning repository, <http://archive.ics.uci.edu/ml/index.php>.
- Azhar, S. J. L. (2011). "Building information modeling (BIM): Trends, benefits, risks, and challenges for the AEC industry." Leadership and management in engineering **11**(3): 241-252.
- Badi, S. and Diamantidou, D. (2017). "A social network perspective of building information modelling in Greek construction projects." Architectural engineering and design management **13**(6): 406-422.
- Barda, N., Riesel, D., Akriv, A., Levy, J., Finkel, U., Yona, G., Greenfeld, D., Sheiba, S., Somer, J. and Bachmat, E. (2020). "Developing a COVID-19 mortality risk prediction model when individual-level data are not available." Nature communications **11**(1): 1-9.
- Basole, R. C., Bellamy, M. A., Park, H. and Putrevu, J. (2016). "Computational analysis and visualization of global supply network risks." IEEE Transactions on Industrial Informatics **12**(3): 1206-1213.
- Beetz, J., van Berlo, L., de Laat, R. and van den Helm, P. (2010). BIMserver.org—An open source IFC model server. Proceedings of the CIP W78 conference.
- Belkin, M. and Niyogi, P. (2002). Laplacian eigenmaps and spectral techniques for embedding and clustering. Advances in neural information processing systems.
- Belsky, M., Sacks, R., Brilakis, I. J. C. A. C. and Engineering, I. (2016). "Semantic enrichment for building information modeling." Computer - Aided Civil and Infrastructure Engineering **31**(4): 261-274.
- Bengio, Y., Boulanger-Lewandowski, N. and Pascanu, R. (2013). Advances in optimizing recurrent networks. 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, IEEE.

- Bernardini, F. C., da Silva, R. B., Meza, E. and das Ostras–RJ–Brazil, R. (2013). "Analyzing the influence of cardinality and density characteristics on multi-label learning." Proc. X Encontro Nacional de Inteligencia Artificial e Computacional-ENIAC.
- Bezdek, J. C. (2013). Pattern recognition with fuzzy objective function algorithms, Springer Science & Business Media.
- Bezdek, J. C., Ehrlich, R., Full, W. J. C. and Geosciences (1984). "FCM: The fuzzy c-means clustering algorithm." Computers & Geosciences **10**(2-3): 191-203.
- Bilal, M., Oyedele, L. O., Qadir, J., Munir, K., Ajayi, S. O., Akinade, O. O., Owolabi, H. A., Alaka, H. A. and Pasha, M. (2016). "Big Data in the construction industry: A review of present status, opportunities, and future trends." Advanced engineering informatics **30**(3): 500-521.
- Block, P., Hoffman, M., Raabe, I. J., Dowd, J. B., Rahal, C., Kashyap, R. and Mills, M. C. (2020). "Social network-based distancing strategies to flatten the COVID-19 curve in a post-lockdown world." Nature Human Behaviour: 1-9.
- Bock, T. (2015). "The future of construction automation: Technological disruption and the upcoming ubiquity of robotics." Automation in Construction **59**: 113-121.
- Bogarín, A., Cerezo, R. and Romero, C. (2018). "A survey on educational process mining." Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery **8**(1): e1230.
- Boje, C., Guerriero, A., Kubicki, S. and Rezgui, Y. (2020). "Towards a semantic Construction Digital Twin: Directions for future research." Automation in Construction **114**: 103179.
- Bonchi, F., Castillo, C., Gionis, A., Jaimes, A. J. A. T. o. I. S. and Technology (2011). "Social network analysis and mining for business applications." ACM Transactions on Intelligent Systems and Technology **2**(3): 22.
- Bortolini, R., Formoso, C. T. and Viana, D. D. (2019). "Site logistics planning and control for engineer-to-order prefabricated building systems using BIM 4D modeling." Automation in Construction **98**: 248-264.
- Box, G. E., Jenkins, G. M., Reinsel, G. C. and Ljung, G. M. (2015). Time series analysis: forecasting and control, John Wiley & Sons.
- Bradley, A., Li, H., Lark, R. and Dunn, S. (2016). "BIM for infrastructure: An overall review and constructor perspective." Automation in Construction **71**: 139-152.
- Broniatowski, D. A., Dredze, M., Paul, M. J. and Dugas, A. (2015). "Using social media to perform local influenza surveillance in an inner-city hospital: a retrospective observational study." JMIR public health and surveillance **1**(1): e5.
- Budayan, C., Dikmen, I. and Birgonul, M. T. (2009). "Comparing the performance of traditional cluster analysis, self-organizing maps and fuzzy C-means method for strategic grouping." Expert Systems with Applications **36**(9): 11772-11781.
- Buijs, J. C., Van Dongen, B. F. and van Der Aalst, W. M. (2012). On the role of fitness, precision, generalization and simplicity in process discovery. OTM Confederated International Conferences" On the Move to Meaningful Internet Systems", Springer.
- Caliński, T. and Harabasz, J. (1974). "A dendrite method for cluster analysis." Communications in Statistics-theory and Methods **3**(1): 1-27.

- Campbell, J. P., McHenry, J. J. and Wise, L. L. (1990). "Modeling job performance in a population of jobs." Personnel psychology **43**(2): 313-575.
- Cao, B., Fu, K., Tao, J. and Wang, S. (2015). "GMM-based research on environmental pollution and population migration in Anhui province, China." Ecological Indicators **51**: 159-164.
- Cao, D., Li, H., Wang, G., Luo, X. and Tan, D. (2018). "Relationship network structure and organizational competitiveness: Evidence from BIM implementation practices in the construction industry." Journal of management in engineering **34**(3): 04018005.
- Cavallari, S., Zheng, V. W., Cai, H., Chang, K. C.-C. and Cambria, E. (2017). Learning community embedding with community detection and node embedding on graphs. Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, ACM.
- Celebi, M. E., Kingravi, H. A. and Vela, P. A. (2013). "A comparative study of efficient initialization methods for the k-means clustering algorithm." Expert Systems with Applications **40**(1): 200–210.
- Champa, H. and AnandaKumar, K. (2010). "Artificial neural network for human behavior prediction through handwriting analysis." International Journal of Computer Applications **2**(2): 36-41.
- Chen, C. and Tang, L. (2019). "BIM-based integrated management workflow design for schedule and cost planning of building fabric maintenance." Automation in Construction **107**: 102944.
- Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K. and Yuille, A. L. (2017). "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs." IEEE transactions on pattern analysis and machine intelligence **40**(4): 834-848.
- Chen, L. and Luo, H. (2014). "A BIM-based construction quality management model and its applications." Automation in construction **46**: 64-73.
- Cheng, J. C., Chen, W., Chen, K. and Wang, Q. (2020). "Data-driven predictive maintenance planning framework for MEP components based on BIM and IoT using machine learning algorithms." Automation in Construction **112**: 103087.
- Choi, E., Bahadori, M. T., Schuetz, A., Stewart, W. F. and Sun, J. (2016). Doctor ai: Predicting clinical events via recurrent neural networks. Machine Learning for Healthcare Conference.
- Choi, S., Kim, E. and Oh, S. (2013). Human behavior prediction for smart homes using deep learning. 2013 IEEE RO-MAN, IEEE.
- Chua, D. and Hossain, M. A. (2011). "A simulation model to study the impact of early information on design duration and redesign." International journal of project management **29**(3): 246-257.
- Construction, M.-H. (2012). "The business value of BIM in North America: multi-year trend analysis and user ratings (2007-2012)." Smart Market Report: 1-72 <https://bimforum.org/wp-content/uploads/2012/2012/MHC-Business-Value-of-BIM-in-North-America-2007-2012-SMR.pdf>.

- Construction, M. H. J. S. M. (2014). "The business value of BIM for construction in major global markets: How contractors around the world are driving innovation with building information modeling." 1-60.
- Cortez, B., Carrera, B., Kim, Y.-J. and Jung, J.-Y. (2018). "An architecture for emergency event prediction using LSTM recurrent neural networks." Expert Systems with Applications **97**: 315-324.
- Davies, D. L., Bouldin, D. W. J. I. t. o. p. a. and intelligence, m. (1979). "A cluster separation measure." IEEE Transactions on Pattern Analysis and Machine Intelligence(2): 224-227.
- De Almeida, C. W., De Souza, R. M. and Candeias, A. L. (2013). "Fuzzy Kohonen clustering networks for interval data." Neurocomputing **99**: 65-75.
- Dempster, A. P., Laird, N. M. and Rubin, D. B. (1977). "Maximum likelihood from incomplete data via the EM algorithm." Journal of the Royal Statistical Society: Series B **39**(1): 1-22.
- Deng, Y., Gan, V. J., Das, M., Cheng, J. C. and Anumba, C. (2019). "Integrating 4D BIM and GIS for Construction Supply Chain Management." Journal of construction engineering and management **145**(4): 04019016.
- Dhand, A., White, C. C., Johnson, C., Xia, Z. and De Jager, P. L. (2018). "A scalable online tool for quantitative social network assessment reveals potentially modifiable social environmental risks." Nature communications **9**.
- Dimitrov, A. and Golparvar-Fard, M. (2014). "Vision-based material recognition for automated monitoring of construction progress and generating building information modeling from unordered site image collections." Advanced Engineering Informatics **28**(1): 37-49.
- Ding, L. and Xu, X. (2014). "Application of cloud storage on BIM life-cycle management." International Journal of Advanced Robotic Systems **11**(8): 129.
- Ding, L., Zhou, Y. and Akinci, B. (2014). "Building Information Modeling (BIM) application framework: The process of expanding from 3D to computable nD." Automation in construction **46**: 82-93.
- dos Santos Garcia, C., Meinheim, A., Junior, E. R. F., Dallagassa, M. R., Sato, D. M. V., Carvalho, D. R., Santos, E. A. P. and Scalabrin, E. E. (2019). "Process mining techniques and applications—a systematic mapping study." Expert Systems with Applications **133**: 260-295.
- Du, J., Zhu, Q., Shi, Y., Wang, Q., Lin, Y. and Zhao, D. (2020). "Cognition digital twins for personalized information systems of smart cities: Proof of concept." Journal of Management in Engineering **36**(2): 04019052.
- Du, K.-L. (2010). "Clustering: A neural network approach." Neural networks **23**(1): 89–107.
- Du, Y., Wang, W. and Wang, L. (2015). Hierarchical recurrent neural network for skeleton based action recognition. Proceedings of the IEEE conference on computer vision and pattern recognition.
- Duan, R., Lin, Y. and Hu, L. (2018). "Reliability evaluation for complex systems based on interval-valued triangular fuzzy weighted mean and evidence network." Journal of Advanced Mechanical Design, Systems, and Manufacturing **12**(4): JAMDSM0087-JAMDSM0087.

- Duffy, A. H. (2012). The design productivity debate, Springer Science & Business Media.
- Durugbo, C., Hutabarat, W., Tiwari, A. and Alcock, J. R. (2011). "Modelling collaboration using complex networks." Information Sciences **181**(15): 3143-3161.
- Dymora, P., Koryl, M. and Mazurek, M. (2019). "Process Discovery in Business Process Management Optimization." Information **10**(9): 270.
- Eadie, R., Browne, M., Odeyinka, H., McKeown, C. and McNiff, S. (2013). "BIM implementation throughout the UK construction project lifecycle: An analysis." Automation in construction **36**: 145-151.
- Eastman, C. M., Eastman, C., Teicholz, P., Sacks, R. and Liston, K. (2011). BIM handbook: A guide to building information modeling for owners, managers, designers, engineers and contractors, John Wiley & Sons.
- El-Diraby, T., Krijnen, T. and Papagelis, M. (2017). "BIM-based collaborative design and socio-technical analytics of green buildings." Automation in Construction **82**: 59-74.
- Elman, J. L. (1990). "Finding structure in time." Cognitive science **14**(2): 179-211.
- Evermann, J., Rehse, J.-R. and Fettke, P. (2017). "Predicting process behaviour using deep learning." Decision Support Systems **100**: 129-140.
- Fan, J., Jia, S. and Li, X. (2013). The application of fuzzy Kohonen clustering network for intelligent wheelchair motion control. 2013 IEEE International Conference on Robotics and Biomimetics (ROBIO), IEEE.
- Fan, J., Li, Q., Hou, J., Feng, X., Karimian, H. and Lin, S. (2017). "A spatiotemporal prediction framework for air pollution based on deep RNN." ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences **4**: 15.
- Fan, J. and Li, R. (2006). "Statistical challenges with high dimensionality: Feature selection in knowledge discovery." arXiv preprint math/0602133.
- Forman, G. (2003). "An extensive empirical study of feature selection metrics for text classification." Journal of machine learning research **3**(Mar): 1289-1305.
- Fransen, K., Van Puyenbroeck, S., Loughhead, T. M., Vanbeselaere, N., De Cuyper, B., Broek, G. V. and Boen, F. (2015). "Who takes the lead? Social network analysis as a pioneering tool to investigate shared leadership within sports teams." Social networks **43**: 28-38.
- Fu, J., Chai, J., Sun, D. and Wang, S. (2012). Multi-factor analysis of terrorist activities based on social network. 2012 Fifth International Conference on Business Intelligence and Financial Engineering, IEEE.
- Gao, S., Ma, J., Chen, Z., Wang, G., Xing, C. J. P. A. S. M. and Applications, i. (2014). "Ranking the spreading ability of nodes in complex networks based on local structure." Physica A: Statistical Mechanics and its Applications **403**: 130-147.
- Gao, X. and Pishdad-Bozorgi, P. (2019). "BIM-enabled facilities operation and maintenance: A review." Advanced Engineering Informatics **39**: 227-247.
- Garas, A., Schweitzer, F. and Havlin, S. (2012). "A k-shell decomposition method for weighted networks." New Journal of Physics **14**(8): 083030.
- Géry, M. and Haddad, H. (2003). Evaluation of web usage mining approaches for user's next request prediction. Proceedings of the 5th ACM international workshop on Web information and data management, ACM.

- Ghaffarianhoseini, A., Tookey, J., Ghaffarianhoseini, A., Naismith, N., Azhar, S., Efimova, O. and Raahemifar, K. (2017). "Building Information Modelling (BIM) uptake: Clear benefits, understanding its implementation, risks and challenges." Renewable and Sustainable Energy Reviews **75**: 1046-1053.
- Glaessgen, E. and Stargel, D. (2012). The digital twin paradigm for future NASA and US Air Force vehicles. 53rd AIAA/ASME/ASCE/AHS/ASC structures, structural dynamics and materials conference 20th AIAA/ASME/AHS adaptive structures conference 14th AIAA.
- Golparvar-Fard, M., Peña-Mora, F. and Savarese, S. (2009). "D4AR—a 4-dimensional augmented reality model for automating construction progress monitoring data collection, processing and communication." Journal of information technology in construction **14**(13): 129-153.
- Graves, A., Mohamed, A.-r. and Hinton, G. (2013). Speech recognition with deep recurrent neural networks. 2013 IEEE international conference on acoustics, speech and signal processing, IEEE.
- Grover, A. and Leskovec, J. (2016). node2vec: Scalable feature learning for networks. Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining, ACM.
- Gu, N. and London, K. (2010). "Understanding and facilitating BIM adoption in the AEC industry." Automation in construction **19**(8): 988-999.
- Guerbas, A., Addam, O., Zaarour, O., Nagi, M., Elhadj, A., Ridley, M. and Alhadj, R. (2013). "Effective web log mining and online navigational pattern prediction." knowledge-based systems **49**: 50-62.
- Günther, C. W. (2009). "Process mining in flexible environments."
- Günther, C. W. and Van Der Aalst, W. M. (2007). Fuzzy mining—adaptive process simplification based on multi-perspective metrics. International conference on business process management, Springer.
- Gupta, M., Sureka, A. and Padmanabhuni, S. (2014). Process mining multiple repositories for software defect resolution from control and organizational perspective. Proceedings of the 11th Working Conference on Mining Software Repositories.
- Gurgen Erdogan, T. and Tarhan, A. (2018). "A goal-driven evaluation method based on process mining for healthcare processes." Applied Sciences **8**(6): 894.
- Hämäläinen, J., Jauhiainen, S. and Kärkkäinen, T. J. A. (2017). "Comparison of internal clustering validation indices for prototype-based clustering." Algorithms **10**(3): 105.
- Hamma-adama, M. and Kouider, T. (2019). "Comparative analysis of BIM adoption efforts by developed countries as precedent for new adopter countries." Current Journal of Applied Science and Technology: 1-15.
- Harari, G. M., Wang, W., Müller, S. R., Wang, R. and Campbell, A. T. (2017). Participants' compliance and experiences with self-tracking using a smartphone sensing app. Proceedings of the 2017 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2017 ACM International Symposium on Wearable Computers, ACM.

- Hochreiter, S. (1998). "The vanishing gradient problem during learning recurrent neural nets and problem solutions." International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems **6**(02): 107-116.
- Hochreiter, S. and Schmidhuber, J. (1997). "Long short-term memory." Neural computation **9**(8): 1735-1780.
- Hu, X., Lu, M. and AbouRizk, S. (2014). BIM-based data mining approach to estimating job man-hour requirements in structural steel fabrication. Proceedings of the 2014 Winter Simulation Conference, IEEE Press.
- Hu, Z.-Z., Tian, P.-L., Li, S.-W. and Zhang, J.-P. (2018). "BIM-based integrated delivery technologies for intelligent MEP management in the operation and maintenance phase." Advances in Engineering Software **115**: 1-16.
- Huang, G., Wu, L., Ma, X., Zhang, W., Fan, J., Yu, X., Zeng, W. and Zhou, H. (2019). "Evaluation of CatBoost method for prediction of reference evapotranspiration in humid regions." Journal of Hydrology **574**: 1029-1041.
- Hubert, L. and Arabie, P. (1985). "Comparing partitions." Journal of classification **2**(1): 193-218.
- Hung, M., Lauren, E., Hon, E. S., Birmingham, W. C., Xu, J., Su, S., Hon, S. D., Park, J., Dang, P. and Lipsky, M. S. (2020). "Social network analysis of COVID-19 Sentiments: Application of artificial intelligence." Journal of medical Internet research **22**(8): e22590.
- Hwang, I. and Jang, Y. J. (2017). "Process mining to discover shoppers' pathways at a fashion retail store using a WiFi-base indoor positioning system." IEEE Transactions on Automation Science and Engineering **14**(4): 1786-1792.
- Inoue, M., Yamashita, T. and Nishida, T. (2019). Robot path planning by LSTM network under changing environment. Advances in Computer Communication and Computational Sciences, Springer: 317-329.
- ISO, B. (2019). "19650–1: 2018: Organization and digitization of information about buildings and civil engineering works, including building information modelling (BIM)–Information management using building information modelling–Part 1: Delivery phase of the assets." BSI Standards Limited.
- Jabbar, N., Ahson, S. and Mehrotra, M. (2011). Fuzzy Kohonen Clustering Network for Color Image Segmentation. 2009 International Conference on Machine Learning and Computing, Australia.
- Jabbar, N. I. and Ahson, S. (2010). Modified fuzzy Kohonen clustering network for image segmentation. 2010 International Conference on Financial Theory and Engineering, IEEE.
- Jaisook, P. and Premchaiswadi, W. (2015). Time performance analysis of medical treatment processes by using disco. 2015 13th International Conference on ICT and Knowledge Engineering (ICT & Knowledge Engineering 2015), IEEE.
- Jans, M., Van Der Werf, J. M., Lybaert, N. and Vanhoof, K. (2011). "A business process mining application for internal transaction fraud mitigation." Expert Systems with Applications **38**(10): 13351-13359.
- Jeh, G. and Widom, J. (2002). SimRank: A Measure of Structural-Context Similarity. Eighth Acm Sigkdd International Conference on Knowledge Discovery & Data Mining.

- Jin, R., Zou, Y., Gidado, K., Ashton, P. and Painting, N. (2019). "Scientometric analysis of BIM-based research in construction engineering and management." Engineering, Construction and Architectural Management.
- Jordan, M. (1986). Attractor dynamics and parallelism in a connectionist sequential machine. Proc. of the Eighth Annual Conference of the Cognitive Science Society (Erlbaum, Hillsdale, NJ), 1986.
- Kanan, R., Elhassan, O. and Bensalem, R. (2018). "An IoT-based autonomous system for workers' safety in construction sites with real-time alarming, monitoring, and positioning strategies." Automation in Construction **88**: 73-86.
- Kang, H. (2013). "The prevention and handling of the missing data." Korean journal of anesthesiology **64**(5): 402.
- Kang, P., Lin, Z., Teng, S., Zhang, G., Guo, L. and Zhang, W. (2019). Catboost-based Framework with Additional User Information for Social Media Popularity Prediction. Proceedings of the 27th ACM International Conference on Multimedia, ACM.
- Kang, T. W. and Choi, H. S. (2018). "BIM-based data mining method considering data integration and function extension." KSCE Journal of Civil Engineering **22**(5): 1523-1534.
- Kang, T. W. and Hong, C. H. (2015). "A study on software architecture for effective BIM/GIS-based facility management data integration." Automation in construction **54**: 25-38.
- Kanter, J. M. and Veeramachaneni, K. (2015). Deep feature synthesis: Towards automating data science endeavors. 2015 IEEE International Conference on Data Science and Advanced Analytics (DSAA), IEEE.
- Kendall, M. G. (1938). "A new measure of rank correlation." Biometrika **30**(1/2): 81-93.
- Kitsak, M., Gallos, L. K., Havlin, S., Liljeros, F., Muchnik, L., Stanley, H. E. and Makse, H. A. (2010). "Identification of influential spreaders in complex networks." Nature physics **6**(11): 888.
- Kohonen, T. (1990). "The self-organizing map." Proceedings of the IEEE **78**(9): 1464-1480.
- Kouhestani, S. and Nik-Bakht, M. (2020). "IFC-based process mining for design authoring." Automation in Construction **112**: 103069.
- Kovács, I. A., Luck, K., Spirohn, K., Wang, Y., Pollis, C., Schlabach, S., Bian, W., Kim, D.-K., Kishore, N. and Hao, T. (2019). "Network-based prediction of protein interactions." Nature communications **10**(1): 1240.
- Kumar, J., Goomer, R. and Singh, A. K. (2018). "Long short term memory recurrent neural network (lstm-rnn) based workload forecasting model for cloud datacenters." Procedia Computer Science **125**: 676-682.
- Kumar, U. A. and Dhamija, Y. (2010). Comparative analysis of SOM neural network with K-means clustering algorithm. Proc. 2010 IEEE International Conference on Management of Innovation & Technology, IEEE.
- La Rosa, M., Wohed, P., Mendling, J., Ter Hofstede, A. H., Reijers, H. A. and van der Aalst, W. M. (2011). "Managing process model complexity via abstract syntax modifications." IEEE Transactions on Industrial Informatics **7**(4): 614-629.

- Lagkas, T., Argyriou, V., Bibi, S. and Sarigiannidis, P. (2018). "UAV IoT framework views and challenges: towards protecting drones as "things"." *Sensors* **18**(11): 4015.
- Lampe, O. D. and Hauser, H. (2011). Interactive visualization of streaming data with kernel density estimation. 2011 IEEE pacific visualization symposium, IEEE.
- Lapin, M., Hein, M. and Schiele, B. (2015). Top-k multiclass SVM. Advances in Neural Information Processing Systems.
- Leemans, S. J., Fahland, D. and van der Aalst, W. M. (2013). Discovering block-structured process models from event logs-a constructive approach. International conference on applications and theory of Petri nets and concurrency, Springer.
- Leemans, S. J., Fahland, D. and Van Der Aalst, W. M. (2014). "Process and Deviation Exploration with Inductive Visual Miner." *BPM (Demos)* **1295**(46): 8.
- Li, J., Fong, S., Zhuang, Y. and Khoury, R. (2016). "Hierarchical classification in text mining for sentiment analysis of online news." *Soft Computing* **20**(9): 3411-3420.
- Li, J., Greenwood, D. and Kassem, M. (2019). "Blockchain in the built environment and construction industry: A systematic review, conceptual models and practical use cases." *Automation in Construction* **102**: 288-307.
- Li, W., Prasad, S., Fowler, J. E. and Bruce, L. M. (2011). "Locality-preserving dimensionality reduction and classification for hyperspectral image analysis." *IEEE Transactions on Geoscience and Remote Sensing* **50**(4): 1185-1198.
- Li, X., Wu, P., Shen, G. Q., Wang, X. and Teng, Y. (2017). "Mapping the knowledge domains of Building Information Modeling (BIM): A bibliometric approach." *Automation in Construction* **84**: 195-206.
- Li, X., Yi, W., Chi, H.-L., Wang, X. and Chan, A. P. (2018). "A critical review of virtual and augmented reality (VR/AR) applications in construction safety." *Automation in Construction* **86**: 150-162.
- Li, Y., Cao, B., Xu, L., Yin, J., Deng, S., Yin, Y. and Wu, Z. (2013). "An efficient recommendation method for improving business process modeling." *IEEE Transactions on Industrial Informatics* **10**(1): 502-513.
- Liebich, T. (2010). Unveiling IFC2x4-The next generation of OPENBIM. Proceedings of the 2010 CIB W78 Conference.
- Liebich, T. (2013). IFC4—The new buildingSMART standard. IC Meeting, bSI Publications Helsinki, Finland.
- Lin, J. R., Hu, Z. Z., Zhang, J. P. and Yu, F. Q. (2016). "A natural-language-based approach to intelligent data retrieval and representation for cloud BIM." *Computer-Aided Civil and Infrastructure Engineering* **31**(1): 18-33.
- Lin, J. R., Hu, Z. Z., Zhang, J. P., Yu, F. Q. J. C. A. C. and Engineering, I. (2016). "A Natural - Language - Based Approach to Intelligent Data Retrieval and Representation for Cloud BIM." *Computer - Aided Civil and Infrastructure Engineering* **31**(1): 18-33.
- Lin, S.-C. (2014). "An analysis for construction engineering networks." *Journal of construction engineering and management* **141**(5): 04014096.
- Linares, D. A., Anumba, C. and Roofigari-Esfahan, N. (2019). "Overview of Supporting Technologies for Cyber-Physical Systems Implementation in the AEC Industry." *Computing in Civil Engineering*.

- Lipton, Z. C., Kale, D. C., Elkan, C. and Wetzel, R. (2015). "Learning to diagnose with LSTM recurrent neural networks." arXiv preprint arXiv:1503.03677.
- Liu, A.-A., Shao, Z., Wong, Y., Li, J., Su, Y.-T. and Kankanhalli, M. (2019). "LSTM-based multi-label video event detection." Multimedia Tools and Applications **78**(1): 677-695.
- Liu, B., Wang, M., Zhang, Y., Liu, R. and Wang, A. (2017). Review and prospect of BIM policy in China. IOP Conference Series: Materials Science and Engineering, IOP Publishing.
- Liu, H., Singh, G., Lu, M., Bouferguene, A. and Al-Hussein, M. (2018). "BIM-based automated design and planning for boarding of light-frame residential buildings." Automation in Construction **89**: 235-249.
- Liu, Y., Tang, M., Zhou, T. and Do, Y. (2015). "Improving the accuracy of the k-shell method by removing redundant links: From a perspective of spreading dynamics." Scientific reports **5**: 13172.
- Liu, Y., Tang, M., Zhou, T. and Do, Y. (2016). "Identify influential spreaders in complex networks, the role of neighborhood." Physica A: Statistical Mechanics and its Applications **452**: 289-298.
- Liu, Y., Van Nederveen, S. and Hertogh, M. (2017). "Understanding effects of BIM on collaborative design and construction: An empirical study in China." International Journal of Project Management **35**(4): 686-698.
- Lopes, P. and Roy, B. (2015). "Dynamic recommendation system using web usage mining for e-commerce users." Procedia Computer Science **45**: 60-69.
- Louis, J. and Dunston, P. S. (2018). "Integrating IoT into operational workflows for real-time and automated decision-making in repetitive construction operations." Automation in Construction **94**: 317-327.
- Love, P. E., Edwards, D. J., Han, S. and Goh, Y. M. (2011). "Design error reduction: toward the effective utilization of building information modeling." Research in Engineering Design **22**(3): 173-187.
- Lu, B., Wei, Y. and Li, J. (2009). A noise-resistant fuzzy kohonen clustering network algorithm for color image segmentation. 2009 4th International Conference on Computer Science & Education, IEEE.
- Lu, Q., Parlikad, A. K., Woodall, P., Don Ranasinghe, G., Xie, X., Liang, Z., Konstantinou, E., Heaton, J. and Schooling, J. (2020). "Developing a Digital Twin at Building and City Levels: Case Study of West Cambridge Campus." Journal of Management in Engineering **36**(3): 05020004.
- Lu, Q., Xie, X., Parlikad, A. K. and Schooling, J. M. (2020). "Digital twin-enabled anomaly detection for built asset monitoring in operation and maintenance." Automation in Construction **118**: 103277.
- Lu, R. and Brilakis, I. (2019). "Digital twinning of existing reinforced concrete bridges from labelled point clusters." Automation in Construction **105**: 102837.
- Ma, X., Tao, Z., Wang, Y., Yu, H. and Wang, Y. (2015). "Long short-term memory neural network for traffic speed prediction using remote microwave sensor data." Transportation Research Part C: Emerging Technologies **54**: 187-197.
- Ma, Z., Ren, Y., Xiang, X. and Turk, Z. (2020). "Data-driven decision-making for equipment maintenance." Automation in Construction **112**: 103103.

- Maaten, L. v. d. and Hinton, G. (2008). "Visualizing data using t-SNE." Journal of machine learning research **9**(Nov): 2579-2605.
- Makarenyov, V., Rokach, L. and Shapira, B. (2019). "Choosing the right word: Using bidirectional LSTM tagger for writing support systems." Engineering Applications of Artificial Intelligence **84**: 1-10.
- Mannino, A., Dejacco, M. C. and Re Cecconi, F. (2021). "Building Information Modelling and Internet of Things Integration for Facility Management—Literature Review and Future Needs." Applied Sciences **11**(7): 3062.
- Marzouk, M. and Abdelaty, A. (2014). "Monitoring thermal comfort in subways using building information modeling." Energy and buildings **84**: 252-257.
- Matic, A., Osmani, V. and Mayora-Ibarra, O. (2014). Mobile monitoring of formal and informal social interactions at workplace. Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication, ACM.
- Merschbrock, C. (2012). "Unorchestrated symphony: The case of inter-organizational collaboration in digital construction design." Journal of Information Technology in Construction **17**(22): 333-350.
- Mikolov, T., Chen, K., Corrado, G. and Dean, J. (2013). "Efficient estimation of word representations in vector space." ICLR Workshop.
- Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S. and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. Advances in neural information processing systems.
- Min, Q., Lu, Y., Liu, Z., Su, C. and Wang, B. (2019). "Machine learning based digital twin framework for production optimization in petrochemical industry." International Journal of Information Management **49**: 502-519.
- Mingoti, S. A. and Lima, J. O. (2006). "Comparing SOM neural network with Fuzzy c-means, K-means and traditional hierarchical clustering algorithms." European Journal of Operational Research **174**(3): 1742–1759.
- Mirakhorli, M., Chen, H.-M. and Kazman, R. (2015). Mining big data for detecting, extracting and recommending architectural design concepts. 2015 IEEE/ACM 1st International Workshop on Big Data Software Engineering, IEEE.
- Mirjafari, S., Masaba, K., Grover, T., Wang, W., Audia, P., Campbell, A. T., Chawla, N. V., Swain, V. D., Choudhury, M. D. and Dey, A. K. (2019). "Differentiating Higher and Lower Job Performers in the Workplace Using Mobile Sensing." Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies **3**(2): 37.
- Mo, Y., Zhao, D., Du, J., Syal, M., Aziz, A. and Li, H. (2020). "Automated staff assignment for building maintenance using natural language processing." Automation in Construction **113**: 103150.
- Musumeci, F., Rottondi, C., Nag, A., Macaluso, I., Zibar, D., Ruffini, M., Tornatore, M. J. I. C. S. and Tutorials (2018). "An overview on application of machine learning techniques in optical networks." IEEE Communications Surveys & Tutorials **21**(2): 1383-1408.

- Neumeyer, X. and Santos, S. C. (2018). "Sustainable business models, venture typologies, and entrepreneurial ecosystems: A social network perspective." Journal of cleaner production **172**: 4565-4579.
- Nohuddin, P. N., Coenen, F., Christley, R., Setzkorn, C., Patel, Y. and Williams, S. (2012). "Finding "interesting" trends in social networks using frequent pattern mining and self organizing maps." Knowledge-Based Systems **29**: 104-113.
- Nurmaini, S., Tutuko, B. and Putra, A. (2016). "Pattern recognition approach for swarm robots reactive control with fuzzy-kohonen networks and particle swarm optimization algorithm." Journal of Telecommunication, Electronic and Computer Engineering **8**(3): 155-160.
- Oh, M., Lee, J., Hong, S. W. and Jeong, Y. (2015). "Integrated system for BIM-based collaborative design." Automation in Construction **58**: 196-206.
- Oraee, M., Hosseini, M. R., Papadonikolaki, E., Palliyaguru, R. and Arashpour, M. (2017). "Collaboration in BIM-based construction networks: A bibliometric-qualitative literature review." International Journal of Project Management **35**(7): 1288-1301.
- Page, L., Brin, S., Motwani, R. and Winograd, T. (1999). The pagerank citation ranking: Bringing order to the web, Stanford InfoLab.
- Palau, J., Montaner, M., López, B. and De La Rosa, J. L. (2004). Collaboration analysis in recommender systems using social networks. International Workshop on Cooperative Information Agents, Springer.
- Pan, Y. and Zhang, L. (2020). "BIM log mining: Exploring design productivity characteristics." Automation in Construction **109**: 102997.
- Pan, Y. and Zhang, L. (2020). "BIM log mining: Learning and predicting design commands." Automation in Construction **112**: 103107.
- Pan, Y., Zhang, L. and Skibniewski, M. J. (2020). "Clustering of designers based on building information modeling event logs." Computer - Aided Civil and Infrastructure Engineering **35**(7): 701-718.
- Papadopoulos, S., Kompatsiaris, Y., Vakali, A. and Spyridonos, P. (2012). "Community detection in social media." Data Mining and Knowledge Discovery **24**(3): 515-554.
- Park, C.-S. and Kim, H.-J. (2013). "A framework for construction safety management and visualization system." Automation in Construction **33**: 95-103.
- Peng, Y., Lin, J.-R., Zhang, J.-P. and Hu, Z.-Z. (2017). "A hybrid data mining approach on BIM-based building operation and maintenance." Building and Environment **126**: 483-495.
- Perozzi, B., Al-Rfou, R. and Skiena, S. (2014). Deepwalk: Online learning of social representations. Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM.
- Peter, M. and Ying, X. (2006). Computational Systems Bioinformatics-Proceedings Of The Conference Csb 2006, World Scientific.
- Petri, C. (1962). "Kommunikation mit Automaten//Ph. D. thesis. Universitat Bonn, Schriften des Instituts fur Instrumentelle Mathematik, Germany. 1962 (in German)."
- Petrova, E., Pauwels, P., Svidt, K. and Jensen, R. L. (2019). In search of sustainable design patterns: Combining data mining and semantic data modelling on disparate

- building data. Advances in Informatics and Computing in Civil and Construction Engineering, Springer: 19-26.
- Petrova, E., Pauwels, P., Svidt, K., Jensen, R. L. J. A. E. and Management, D. (2019). "Towards data-driven sustainable design: decision support based on knowledge discovery in disparate building data." Architectural Engineering and Design Management **15**(5): 334-356.
- Phan, N., Dou, D., Wang, H., Kil, D. and Piniewski, B. (2017). "Ontology-based deep learning for human behavior prediction with explanations in health social networks." Information sciences **384**: 298-313.
- Pika, A., Wynn, M. T., Budiono, S., ter Hofstede, A. H., van der Aalst, W. M. and Reijers, H. A. (2019). Towards privacy-preserving process mining in healthcare. International Conference on Business Process Management, Springer.
- Premchaiswadi, W. and Porouhan, P. (2015). "Process modeling and bottleneck mining in online peer-review systems." SpringerPlus **4**(1): 1-18.
- Prokhorenkova, L., Gusev, G., Vorobev, A., Dorogush, A. V. and Gulin, A. (2018). CatBoost: unbiased boosting with categorical features. Advances in Neural Information Processing Systems.
- Qi, Q. and Tao, F. (2018). "Digital twin and big data towards smart manufacturing and industry 4.0: 360 degree comparison." IEEE Access **6**: 3585-3593.
- Qian, P., Zhao, K., Jiang, Y., Su, K.-H., Deng, Z., Wang, S. and Muzic Jr, R. F. (2017). "Knowledge-leveraged transfer fuzzy c-means for texture image segmentation with self-adaptive cluster prototype matching." Knowledge-based systems **130**: 33-50.
- Qiu, H., Xu, Y., Gao, L., Li, X. and Chi, L. (2016). "Multi-stage design space reduction and metamodeling optimization method based on self-organizing maps and fuzzy clustering." Expert Systems with Applications **46**: 180-195.
- Qiu, J., Wu, Q., Ding, G., Xu, Y. and Feng, S. (2016). "A survey of machine learning for big data processing." EURASIP Journal on Advances in Signal Processing **2016**(1): 67.
- Ramaji, I. J. and Memari, A. M. (2016). "Product architecture model for multistory modular buildings." Journal of construction engineering and management **142**(10): 04016047.
- Rebuge, Á. and Ferreira, D. R. (2012). "Business process analysis in healthcare environments: A methodology based on process mining." Information systems **37**(2): 99-116.
- Revit, A. (2011). Journal file parser, https://revitclinic.typepad.com/my_weblog/2011/11/journal-file-parser.html.
- Revit, A. (2017). About journal files, <https://knowledge.autodesk.com/support/revit-products/getting-started/caas/CloudHelp/cloudhelp/2019/ENU/Revit-GetStarted/files/GUID-477C6854-2724-4B5D-8B95-9657B636C48D-htm.html>.
- Rojas, E., Munoz-Gama, J., Sepúlveda, M. and Capurro, D. (2016). "Process mining in healthcare: A literature review." Journal of biomedical informatics **61**: 224-236.
- Rousseeuw, P. J. J. o. c. and mathematics, a. (1987). "Silhouettes: a graphical aid to the interpretation and validation of cluster analysis." Journal of Computational and Applied Mathematics **20**: 53-65.

- Roweis, S. T. and Saul, L. K. (2000). "Nonlinear dimensionality reduction by locally linear embedding." science **290**(5500): 2323-2326.
- Saeb, S., Zhang, M., Karr, C. J., Schueller, S. M., Corden, M. E., Kording, K. P. and Mohr, D. C. (2015). "Mobile phone sensor correlates of depressive symptom severity in daily-life behavior: an exploratory study." Journal of medical Internet research **17**(7): e175.
- Sagheer, A. and Kotb, M. J. N. (2019). "Time series forecasting of petroleum production using deep LSTM recurrent networks." **323**: 203-213.
- Sansone, C., Morf, C. C. and Panter, A. T. (2003). The Sage handbook of methods in social psychology, Sage Publications.
- Schirner, G., Erdogmus, D., Chowdhury, K. and Padir, T. (2013). "The future of human-in-the-loop cyber-physical systems." Computer **46**(1): 36-45.
- Schleich, B., Anwer, N., Mathieu, L. and Wartzack, S. (2017). "Shaping the digital twin for design and production engineering." CIRP Annals **66**(1): 141-144.
- Schwarz, G. (1978). "Estimating the dimension of a model." The annals of statistics **6**(2): 461-464.
- Shaikh, A. A., Raju, R. and Malim, N. L. (2016). "Global status of Building Information Modeling (BIM)-A Review." International Journal on Recent and Innovation Trends in Computing and Communication **4**(3): 300-303.
- Sharan, R., Ulitsky, I. and Shamir, R. (2007). "Network -based prediction of protein function." Molecular systems biology **3**(1).
- Shental, N., Bar-Hillel, A., Hertz, T. and Weinshall, D. (2004). Computing Gaussian mixture models with EM using equivalence constraints. Advances in neural information processing systems.
- Shi, X. and Yang, W. (2013). "Performance-driven architectural design and optimization technique from a perspective of architects." Automation in Construction **32**: 125–135.
- Shim, C.-S., Dang, N.-S., Lon, S. and Jeon, C.-H. (2019). "Development of a bridge maintenance system for prestressed concrete bridges using 3D digital twin model." Structure and Infrastructure Engineering **15**(10): 1319-1332.
- Shojaei, A., Wang, J. and Fenner, A. (2019). "Exploring the feasibility of blockchain technology as an infrastructure for improving built asset sustainability." Built Environment Project and Asset Management.
- Slanzi, G., Balazs, J. A. and Velásquez, J. D. (2017). "Combining eye tracking, pupil dilation and EEG analysis for predicting web users click intention." Information Fusion **35**: 51-57.
- Slanzi, G., Pizarro, G. and Velásquez, J. D. (2017). "Biometric information fusion for web user navigation and preferences analysis: An overview." Information Fusion **38**: 12-21.
- Šmite, D., Moe, N. B., Šāblis, A. and Wohlin, C. (2017). "Software teams and their knowledge networks in large-scale software development." Information and Software Technology **86**: 71-86.
- So, M. K., Tiwari, A., Chu, A. M., Tsang, J. T. and Chan, J. N. (2020). "Visualising COVID-19 pandemic risk through network connectedness." International Journal of Infectious Diseases.

- Sokolova, M. and Lapalme, G. (2009). "A systematic analysis of performance measures for classification tasks." Information Processing & Management **45**(4): 427-437.
- Son, H., Lee, S. and Kim, C. (2015). "What drives the adoption of building information modeling in design organizations? An empirical investigation of the antecedents affecting architects' behavioral intentions." Automation in construction **49**: 92-99.
- Song, J., Kim, J. and Lee, J.-K. (2018). NLP and deep learning-based analysis of building regulations to support automated rule checking system. ISARC. Proceedings of the International Symposium on Automation and Robotics in Construction, IAARC Publications.
- Song, K.-T. and Huang, S.-Y. (2004). Mobile robot navigation using sonar direction weights. Proceedings of the 2004 IEEE International Conference on Control Applications, 2004., IEEE.
- Srewil, Y. and Scherer, R. J. (2013). Effective construction process monitoring and control through a collaborative Cyber-Physical approach. Working Conference on Virtual Enterprises, Springer.
- Srivastava, J., Cooley, R., Deshpande, M. and Tan, P.-N. (2000). "Web usage mining: Discovery and applications of usage patterns from web data." Acm Sigkdd Explorations Newsletter **1**(2): 12-23.
- Stojanovic, V., Trapp, M., Richter, R., Hagedorn, B. and Döllner, J. (2018). Towards The Generation of Digital Twins for Facility Management Based on 3D Point Clouds. Proceeding of the 34th Annual ARCOM Conference.
- Su, M.-C. and Chang, H.-T. (2000). "Fast self-organizing feature map algorithm." IEEE Transactions on Neural Networks **11**(3): 721-733.
- Subrahmanian, V. and Kumar, S. (2017). "Predicting human behavior: The next frontiers." Science **355**(6324): 489-489.
- Sun, J., Liu, Y.-S., Gao, G. and Han, X.-G. (2015). "IFCCCompressor: A content-based compression algorithm for optimizing Industry Foundation Classes files." Automation in Construction **50**: 1-15.
- Sun, Z., Han, L., Huang, W., Wang, X., Zeng, X., Wang, M. and Yan, H. (2015). "Recommender systems based on social networks." Journal of Systems and Software **99**: 109-119.
- Swain, V. D., Saha, K., Rajvanshy, H., Sirigiri, A., Gregg, J. M., Lin, S., Martinez, G. J., Mattingly, S. M., Mirjafari, S. and Mulukutla, R. (2019). "A Multisensor Person-Centered Approach to Understand the Role of Daily Activities in Job Performance with Organizational Personas." Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies **3**(4): 130.
- Tan, K. S., Lim, W. H. and Isa, N. A. M. (2013). "Novel initialization scheme for Fuzzy C-Means algorithm on color image segmentation." Applied Soft Computing **13**(4): 1832-1852.
- Tang, J., Qu, M., Wang, M., Zhang, M., Yan, J. and Mei, Q. (2015). Line: Large-scale information network embedding. Proceedings of the 24th international conference on world wide web, International World Wide Web Conferences Steering Committee.
- Tang, S., Shelden, D. R., Eastman, C. M., Pishdad-Bozorgi, P. and Gao, X. (2019). "A review of building information modeling (BIM) and the internet of things (IoT)

- devices integration: Present status and future trends." Automation in Construction **101**: 127-139.
- Tao, F., Sui, F., Liu, A., Qi, Q., Zhang, M., Song, B., Guo, Z., Lu, S. C.-Y. and Nee, A. (2019). "Digital twin-driven product design framework." International Journal of Production Research **57**(12): 3935-3953.
- Tao, F. and Zhang, M. J. I. A. (2017). "Digital twin shop-floor: a new shop-floor paradigm towards smart manufacturing." IEEE Access **5**: 20418-20427.
- Tenenbaum, J. B., De Silva, V. and Langford, J. C. (2000). "A global geometric framework for nonlinear dimensionality reduction." science **290**(5500): 2319-2323.
- Tickoo, S. (2013). Autodesk Revit Architecture 2014 for Architects and Designers, CADCIM Technologies.
- Travaglini, A., Radujković, M. and Mancini, M. (2014). "Building information Modelling (BIM) and project management: A Stakeholders perspective." Organization, technology & management in construction: an international journal **6**(2): 1001-1008.
- Tsao, E. C.-K., Bezdek, J. C. and Pal, N. R. (1994). "Fuzzy Kohonen clustering networks." Pattern recognition **27**(5): 757-764.
- Turk, Ž. and Klinc, R. (2017). "Potentials of blockchain technology for construction management." Procedia engineering **196**: 638-645.
- Tüske, Z., Tahir, M. A., Schlüter, R. and Ney, H. (2015). Integrating Gaussian mixtures into deep neural networks: Softmax layer with hidden variables. 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE.
- Vachálek, J., Bartalský, L., Rovný, O., Šišmišová, D., Morháč, M. and Lokšík, M. (2017). The digital twin of an industrial production line within the industry 4.0 concept. 2017 21st International Conference on Process Control (PC), IEEE.
- Valle, A. M., Santos, E. A. and Loures, E. R. (2017). "Applying process mining techniques in software process appraisals." Information and software technology **87**: 19-31.
- Van der Aalst, W. (2016). Data science in action. Process Mining, Springer, Heidelberg.
- Van Der Aalst, W. M., Reijers, H. A. and Song, M. (2005). "Discovering social networks from event logs." Computer Supported Cooperative Work **14**(6): 549-593.
- van Schaijk, S. (2016). "Building Information Model (BIM) based process mining enabling knowledge reassurance and fact-based problem discovery within the Architecture, Engineering, Construction and Facility Management Industry."
- Vinh, N. X., Epps, J. and Bailey, J. (2010). "Information theoretic measures for clusterings comparison: Variants, properties, normalization and correction for chance." Journal of Machine Learning Research **11**(Oct): 2837-2854.
- Volk, R., Stengel, J. and Schultmann, F. (2014). "Building Information Modeling (BIM) for existing buildings—Literature review and future needs." Automation in construction **38**: 109-127.
- Wang, J., Wu, P., Wang, X. and Shou, W. (2017). "The outlook of blockchain technology for construction engineering management." Frontiers of engineering management: 67-75.

- Wang, P., Wu, P., Wang, J., Chi, H.-L. and Wang, X. (2018). "A critical review of the use of virtual reality in construction engineering education and training." International journal of environmental research and public health **15**(6): 1204.
- Wang, S., Minku, L. L. and Yao, X. (2018). "A systematic study of online class imbalance learning with concept drift." IEEE transactions on neural networks and learning systems(99): 1-20.
- Wang, W., Harari, G. M., Wang, R., Müller, S. R., Mirjafari, S., Masaba, K., Campbell, A. T. J. P. o. t. A. o. I., Mobile, Wearable and Technologies, U. (2018). "Sensing behavioral change over time: Using within-person variability features from mobile sensing to predict personality traits." Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies **2**(3): 141.
- Wang, X., Love, P. E., Kim, M. J., Park, C.-S., Sing, C.-P. and Hou, L. (2013). "A conceptual framework for integrating building information modeling with augmented reality." Automation in construction **34**: 37-44.
- Wang, X., Truijens, M., Hou, L., Wang, Y. and Zhou, Y. (2014). "Integrating Augmented Reality with Building Information Modeling: Onsite construction process controlling for liquefied natural gas industry." Automation in Construction **40**: 96-105.
- Wang, Y., Sun, H., Zhao, Y., Zhou, W. and Zhu, S. (2019). "A Heterogeneous Graph Embedding Framework for Location-Based Social Network Analysis in Smart Cities." IEEE Transactions on Industrial Informatics.
- Wang, Z., Da Cunha, C., Ritou, M. and Furet, B. (2019). "Comparison of K-means and GMM methods for contextual clustering in HSM." Procedia Manufacturing **28**: 154-159.
- Wäsche, H., Dickson, G., Woll, A., Brandes, U. J. E. J. f. S. and Society (2017). "Social network analysis in sport research: an emerging paradigm." European Journal for Sport and Society **14**(2): 138-165.
- Wei, D., Wang, B., Lin, G., Liu, D., Dong, Z., Liu, H. and Liu, Y. (2017). "Research on unstructured text data mining and fault classification based on RNN-LSTM with malfunction inspection report." Energies **10**(3): 406.
- Wei, H., Pan, Z., Hu, G., Zhang, L., Yang, H., Li, X. and Zhou, X. (2018). "Identifying influential nodes based on network representation learning in complex networks." PloS one **13**(7): e0200091.
- Weiner, I. B. and Craighead, W. E. (2010). The Corsini encyclopedia of psychology. New Jersey, United States, John Wiley & Sons.
- Wesoły, M. and Ciosek, P. (2018). "Comparison of various data analysis techniques applied for the classification of pharmaceutical samples by electronic tongue." Sensors Actuators B: Chemical **267**: 570-580.
- Whitlock, K., Abanda, F., Manjia, M., Pettang, C. and Nkeng, G. (2018). "BIM for construction site logistics management." Journal of Engineering, Project, and Production Management **8**(1): 47.
- Wu, C.-H., Ouyang, C.-S., Chen, L.-W. and Lu, L.-W. (2015). "A new fuzzy clustering validity index with a median factor for centroid-based clustering." IEEE Transactions on Fuzzy Systems **23**(3): 701–718.

- Wu, D. (2013). Building knowledge modeling: Integrating knowledge in BIM. Proceedings of the 30th International Conference of CIB W078, Beijing, 9-12 October.
- Xie, X. L. and Beni, G. (1991). "A validity measure for fuzzy clustering." IEEE Transactions on Pattern Analysis & Machine Intelligence(8): 841–847.
- Yadav, S. and Shukla, S. (2016). Analysis of k-fold cross-validation over hold-out validation on colossal datasets for quality classification. 2016 IEEE 6th International Conference on Advanced Computing (IACC), IEEE.
- Yang, X., Li, H., Yu, Y., Luo, X., Huang, T. and Yang, X. (2018). "Automatic pixel-level crack detection and measurement using fully convolutional network." Computer-Aided Civil and Infrastructure Engineering **33**(12): 1090-1109.
- Yang, Y., Jia, Z., Chang, C., Qin, X., Li, T., Wang, H. and Zhao, J. (2008). An efficient fuzzy kohonen clustering network algorithm. 2008 Fifth International Conference on Fuzzy Systems and Knowledge Discovery, IEEE.
- Yao, J., Raghavan, V. V. and Wu, Z. (2008). "Web information fusion: A review of the state of the art." Information Fusion **9**(4): 446-449.
- Yarmohammadi, S., Pourabolghasem, R. and Castro-Lacouture, D. (2017). "Mining implicit 3D modeling patterns from unstructured temporal BIM log text data." Automation in Construction **81**: 17-24.
- Yin, X., Liu, H., Chen, Y. and Al-Hussein, M. (2019). "Building information modelling for off-site construction: Review and future directions." Automation in Construction **101**: 72-91.
- Yin, X., Liu, H., Chen, Y., Wang, Y. and Al-Hussein, M. (2020). "A BIM-based framework for operation and maintenance of utility tunnels." Tunnelling and Underground Space Technology **97**: 103252.
- Yu, L., Huang, W., Wang, S. and Lai, K. K. (2008). "Web warehouse – a new web information fusion tool for web mining." Information Fusion **9**(4): 501-511.
- Yuan, X., Anumba, C. J. and Parfitt, M. K. (2016). "Cyber-physical systems for temporary structure monitoring." Automation in Construction **66**: 1-14.
- Yum, S. (2020). "Social Network Analysis for Coronavirus (COVID-19) in the United States." Social Science Quarterly.
- Zazo, R., Lozano-Diez, A., Gonzalez-Dominguez, J., Toledano, D. T. and Gonzalez-Rodriguez, J. (2016). "Language identification in short utterances using long short-term memory (LSTM) recurrent neural networks." PloS one **11**(1): e0146917.
- Zhang, C., Tang, P., Cooke, N., Buchanan, V., Yilmaz, A., Germain, S. W. S., Boring, R. L., Akca-Hobbins, S. and Gupta, A. (2017). "Human-centered automation for resilient nuclear power plant outage control." Automation in Construction **82**: 179-192.
- Zhang, H., Chow, T. W. and Wu, Q. J. (2016). "Organizing books and authors by multilayer SOM." IEEE Transactions on Neural Networks and Learning Systems **27**(12): 2537–2550.
- Zhang, L. and Ashuri, B. (2018). "BIM log mining: discovering social networks." Automation in Construction **91**: 31-43.
- Zhang, L. and Issa, R. R. (2013). "Ontology-based partial building information model extraction." Journal of Computing in Civil Engineering **27**(6): 576-584.

- Zhang, L., Lu, W., Liu, X., Pedrycz, W. and Zhong, C. (2016). "Fuzzy c-means clustering of incomplete data based on probabilistic information granules of missing values." Knowledge-Based Systems **99**: 51-70.
- Zhang, L., Wen, M. and Ashuri, B. (2017). "BIM log mining: measuring design productivity." Journal of Computing in Civil Engineering **32**(1): 04017071.
- Zhang, L., Wen, M. and Ashuri, B. (2018). "BIM log mining: measuring design productivity." Journal of Computing in Civil Engineering **32**(1): 04017071.
- Zhang, S., Sulankivi, K., Kiviniemi, M., Romo, I., Eastman, C. M. and Teizer, J. (2015). "BIM-based fall hazard identification and prevention in construction safety planning." Safety science **72**: 31-45.
- Zhang, Y., Dai, H., Xu, C., Feng, J., Wang, T., Bian, J., Wang, B. and Liu, T.-Y. (2014). Sequential click prediction for sponsored search with recurrent neural networks. Twenty-Eighth AAAI Conference on Artificial Intelligence.
- Zhao, X. (2017). "A scientometric review of global BIM research: Analysis and visualization." Automation in Construction **80**: 37-47.
- Zhao, Z., Chen, W., Wu, X., Chen, P. C. and Liu, J. (2017). "LSTM network: a deep learning approach for short-term traffic forecast." IET Intelligent Transport Systems **11**(2): 68-75.
- Zhiliang, M., Zhenhua, W., Wu, S. and Zhe, L. (2011). "Application and extension of the IFC standard in construction cost estimating for tendering in China." Automation in Construction **20**(2): 196-204.
- Zhou, Y., Yang, Y. and Yang, J.-B. (2019). "Barriers to BIM implementation strategies in China." Engineering, Construction and Architectural Management.
- Zou, K., Wang, Z. and Hu, M. (2008). "An new initialization method for fuzzy c-means algorithm." Fuzzy Optimization and Decision Making **7**(4): 409–416.