Prediction of learning outcomes via clickstream data using machine learning

Ng, Kelvin Hongrui

2020

Ng, K. H. (2020). Prediction of learning outcomes via clickstream data using machine learning. Doctoral thesis, Nanyang Technological University, Singapore.
<https://hdl.handle.net/10356/147631>

<https://hdl.handle.net/10356/147631>

<https://doi.org/10.32657/10356/147631>

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

Downloaded on 26 Apr 2023 18:12:28 SGT

PREDICTION OF LEARNING OUTCOMES VIA CLICKSTREAM DATA USING MACHINE LEARNING

NG HONGRUI KELVIN

SCHOOL OF ELECTRICAL AND ELECTRONIC ENGINEERING

A thesis submitted to the Nanyang Technological University

in partial fulfillment of the requirement for the degree of

Doctor of Philosophy

2020

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.

28/06/2020

Date

Kelvin Ng

Ng Hongrui Kelvin

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 and that the research data are presented honestly and without prejudice.

28/06/2020

Date

Andy Khong

Andy Khong Wai Hoong

Authorship Attribution Statement

This thesis contains material from 4 papers published in the following peer-reviewed journal(s) / from papers accepted at conferences in which I am listed as an author.

Chapter 3 is published as Kelvin H. R. Ng, K. Hartman, N. Goodkin, Andy W. H. Khong, “Mining for preparatory processes of transfer learning in a blended course,” in *Proc. Am. Geophys. Union Fall Meet., AGU Fall Meeting*, New Orleans, LA, USA, Dec 11 - 15 2017. The contributions of the co-authors are as follows:

- Dr K. Hartman provided the initial project direction.
- A/Prof N. Goodkin and Dr Kevin Hartman co-designed the course curriculum for data collection.
- I prepared the data for analysis and designed the proposed algorithm.
- Dr K. Hartman assisted in the analysis and interpretation of the results.
- I prepared the manuscript drafts. The manuscript was revised by Dr K. Hartman, A/Prof N. Goodkin, and A/Prof Andy W. H. Khong.

Chapter 4 is published as Kelvin H. R. Ng, K. Hartman, K. Liu, and Andy W. H. Khong, “Modelling the way: Using action sequence archetypes to differentiate learning pathways from learning outcomes,” in *Proc. 9th Int. Conf. Educ. Data Min., EDM, 2016*, Raleigh, NC, USA, Jun 29 - Jul 2 2016, pp. 167-174. The contributions of the co-authors are as follows:

- I co-designed the experiment with Mr K. Liu and A/Prof Andy W. H. Khong.
- I prepared the data for analysis, designed the proposed algorithm, and analyzed the results.
- Dr K. Hartman provided guidance in the interpretation of the results.
- I prepared the manuscript drafts. The manuscript was revised by Dr K. Hartman and A/Prof Andy W. H. Khong.

Chapter 6 is published as Kelvin H. R. Ng, S. Tatinati, and Andy W. H. Khong, “On-line education evaluation for signal processing course through student learning pathways,”

in *Proc. IEEE Int. Conf. Acoust., Speech and Signal Process., ICASSP, 2018*, Calgary, AB, Canada, Apr 15-20 2018, pp. 6458-6462. The contributions of the co-authors are as follows:

- I designed the experiment and collected the data. A/Prof Andy W. H. Khong assisted in the collection of the data.
- I prepared the data for analysis, designed the proposed algorithm, and analyzed the results.
- I prepared the manuscript drafts. The manuscript was revised by Dr S. Tatinati and A/Prof Andy W. H. Khong.

Chapter 7 has been published as Kelvin H. R. Ng, S. Tatinati, and Andy W. H. Khong, “Characterization of fuzziness for grade prediction using deep neural networks,” in *Companion Proc. 9th Int. Learn. Anal. Knowl. Conf., LAK 2019*, Tempe, AZ, USA, May 4-8 2019. and Kelvin H. R. Ng, S. Tatinati and Andy W. H. Khong, “Grade Prediction From Multi-Valued Click-Stream Traces via Bayesian-Regularized Deep Neural Networks,” in *IEEE Trans. Signal Process.*, vol. 69, pp. 1477-1491, 2021. The contributions of the co-authors are as follows:

- I designed the experiment and collected the data. A/Prof Andy W. H. Khong assisted in the collection of the data.
- I prepared the data for analysis. Dr S. Tatinati provided guidance to analyze the data.
- I designed the proposed algorithm, and analyzed the results.
- I prepared the manuscript drafts. The manuscript was revised by Dr S. Tatinati and A/Prof Andy W. H. Khong.

28/06/2020

Date

Kelvin Ng

Ng Hongrui Kelvin

Acknowledgments

First and foremost, I would like to sincerely express my gratitude to my advisor Dr Andy W. H. Khong for his continual support during my PhD study. His guidance and encouragement, together with his willingness to unrelentingly improve the paper quality, have tremendously benefited myself. His enthusiasm and guidance have made my postgraduate education a very rewarding experience.

In addition, I would like to thank Dr Kevin Hartman and Ms Jaime Koh for their kind mentorship and unrelenting patience throughout my postgraduate education that eventually led to many interesting discussions. I would also like to thank my wonderful teammates, Mr Liu Kai, Mr Nguyen Hai Trieu Anh, Mr Ho Mun Kit, Mr Cao Zhen, Ms S Supraja, Ms Qiu Wei, Mr Tan Zhi Wei for their companionship, valuable discussions and feedback throughout the journey. I am also grateful for colleagues from the Centre for Research and Development in Learning (CRADLE@NTU) for their support to make the projects possible.

Finally, I would like to express my gratitude to my family for their continual understanding during this arduous journey.

Kelvin

Table of Contents

Statement of Originality	I
Supervisor Declaration Statement	I
Authorship Attribution Statement	I
Acknowledgments	III
Table of Contents	IV
Summary	I
List of Abbreviations	III
List of Notations	V
List of Figures	IX
List of Tables	XI
1 Introduction	1
1.1 Motivation	1
1.2 Organization of the Thesis	4
1.3 Contribution of the Thesis	6
I Identification of Learning Behavior via Data Mining Approaches	10
2 Review of Data Mining Approaches for Identifying Learning Behaviors	11
2.1 Stakeholders for learning analytics	12
2.2 The role of VLEs and the available information within these platforms .	13
2.3 Theoretical basis for existing learning analytics studies	15
2.4 Existing studies identifying various learning behaviors	18
2.5 Chapter summary	23

TABLE OF CONTENTS

3	Identifying Lesson Preparation Preferences via Clustering	24
3.1	Introduction	24
3.2	Dataset	25
3.2.1	Data Pre-Processing	27
3.3	Proposed Clustering Framework	28
3.4	Results and Discussion	32
3.4.1	Lesson Preparation Preferences	32
3.4.2	Likelihood Analysis	34
3.5	Chapter summary	38
4	Problem-Solving Action Sequence Archetypes and the Relationship to Learner Drop-out	39
4.1	Introduction	40
4.2	Experiment Design	42
4.3	Dataset	44
4.4	Proposed Action Sequence Archetype Mining Algorithm	45
4.4.1	Proposed Similarity Measure	50
4.4.2	Action Sequence Archetype Clustering	50
4.4.3	Descriptive cluster labels via sequential pattern mining	52
4.5	Results and discussion	55
4.5.1	Impact of various similarity measures to identify clusters	55
4.5.2	Identification of action sequence archetypes	57
4.5.3	Drop-out prediction performance of action sequence archetypes	59
4.6	Chapter Summary	64
II	Deep Neural Modeling of Educational Click-stream Data	65
5	Review of Grade Prediction Models and Deep Neural Models	66
5.1	Types of grade prediction	67
5.2	Grade prediction task	67
5.2.1	Components of deep neural networks	70
5.2.2	Linear regression paradigm	73
5.3	Existing algorithms and motivation of proposed algorithms	74

TABLE OF CONTENTS

5.4	Other applications of deep neural networks in educational data mining and learning analytics.	78
5.5	Chapter summary	79
6	Regularizing Sequential Clickstream Grade Prediction Model using Non-Cognitive traits	82
6.1	Introduction	83
6.2	Proposed joint-space sequential grade prediction model	85
6.3	Evaluation framework	89
6.3.1	Dataset	89
6.3.2	Performance comparison	90
6.3.3	Model implementation	93
6.4	Results and discussion	94
6.4.1	Frequency-based versus sequence-based grade prediction model . .	94
6.4.2	Use of non-cognitive variables as prediction regulators	95
6.5	Conclusion	96
7	Characterization of Fuzziness in Education Data using Deep Neural Networks	97
7.1	Introduction	98
7.2	Multi-valued nature of sequential clickstream grade prediction	99
7.2.1	Data perspective of the multi-valued problem	100
7.2.2	Model perspective of the multi-valued problem	103
7.3	The proposed MV-sensitive grade prediction model	106
7.3.1	Problem formulation	106
7.3.2	The MV-sensitive grade prediction model	108
7.3.3	MV-sensitive Grade prediction model	112
7.4	Evaluation framework	113
7.4.1	Datasets	113
7.4.2	Performance comparison	113
7.4.3	Performance measures	116
7.5	Results and discussion	116
7.5.1	Organization of feature space	116
7.5.2	Classification performance	119

TABLE OF CONTENTS

7.5.3	Grade prediction performance	121
7.5.4	External validation for grade prediction performances	123
7.6	Conclusions	126
8	Conclusions and Future Research Directions	128
8.1	Conclusions	128
8.2	Future research directions	130
	Publication list	I
	References	II

Summary

This thesis addresses the identification of learning behaviors and the prediction of learning outcomes via interaction clickstream data. The challenges of analyzing these complex clickstream data include the balance between automaticity and interpretability and the effectiveness of prediction models to model learning outcomes; both of which are necessary to inform instructional and policy changes.

For the identification of learning behaviors, the thesis focuses on the use of clustering frameworks to identify learner groups with different study habits. To identify lesson preparation preferences in a blended course, a clustering framework that employs Gaussian mixture model and a heuristic rule-based labeler is proposed. As will be shown in this thesis, this proposed algorithm is able to identify and differentiate normal and alternative preparation preferences. Also, as will be shown in the experiment, learners who preferred less common lesson preparation preferences were associated with higher cognitive abilities to transfer their knowledge onto unseen domains. The focus is then shifted to the identification of click-stream problem solving approaches. A sequential clustering framework that employs string metrics to differentiate pairs of sequences and a sequential pattern mining algorithm to summarize cluster members is proposed. The proposed sequential clustering framework is able to identify action sequence archetypes that can predict student drop-out with an average F1 of 0.712 and AUC of 0.715. While the use of string metric to determine the similarity between pairs of sequences allows for accurate identification of problem-solving approaches, nuances between sequences that belong to the same cluster is summarized via the sequential pattern mining algorithm, thereby retaining the interpretability of its outcomes. Since these interactions were logged in an online elementary Mathematics course that taught students to solve arithmetic word problems via a series of steps, script-like and schematic variants of problem-solving approaches were analyzed. Learners who deviated away from the taught steps, which were expected of them, were also associated with higher persistence in their learning. The two experiments suggest that the greater autonomy in one's learning can be associated with better learning.

For the prediction of learning outcomes via interaction clickstream data, the use of deep learning techniques is proposed. To incorporate the balance between transient learning behaviors and persistent learner characteristics, a joint-space grade prediction model is proposed. As will be shown in the thesis, the use of persistent learner characteristics regularizes grade prediction performance. Performance of the proposed grade prediction performance is evaluated on a digital signal processing course. The proposed method is

able to improve the RMSE score by up to 12% over baseline methods. Furthermore, the notion that interaction clickstream data is multi-valued is raised. Since the clickstream sequences are ordered lists of discrete actions, conventional data mining algorithms are unable to identify the multi-valued inputs. The proposed MV identification module that employs the proposed Bayesian-regularized layer within a deep neural architecture is employed to determine the multi-valued characteristic of interaction clickstream data. The identification, and subsequent removal, of multi-valued inputs results in an improvement in grade prediction performance. The performance of the proposed MV identification module is verified on two interaction clickstream datasets. The proposed MV identification module is able to identify 32.3% more MV instances over brute force methods and, consequently, achieve improvement in grade prediction performance by up to 38.95%.

List of Abbreviations

Abbreviations	Full name
AP	Affinity propagation
ASA	Action sequence archetype
AUC	Area under the operating characteristic curve
BP	Backpropagation
CART	Classification and regression tree
CNN	Convolutional neural network
CWO	Common word order
DNN	Deep neural network
DSP	Digital signal processing
EDM	Educational data mining
EM	Expectation-maximization
FC	Fully-connected
FSLSM	Felder-Silverman learning style model
GMM	Gaussian mixture model
GMS	Global minimum support
GRU	Gated recurrent unit
HAC	Hierarchical agglomerative clustering
HMM	Hidden Markov model

Abbreviations	Full name
IQ	Intelligence quotient
ITS	Intelligent tutoring systems
KNN	k -nearest neighbors
LA	Learning analytics
LMS	Learning management systems
LSTM	Long short-term memory
MAE	Mean Absolute Error
MAS	Minimum action support
MCQ	Multiple-choice questions
MCWO	Modified common word order
MLP	Multilayer perceptron
MMS	Multiple minimum support
MOOC	Massive open online courses
MPS	Minimum possible support
MV	Multi-valued
PLMS	PLWAP algorithm with multiple minimum support
PLWAP	Position-coded pre-order linked web access pattern sequential pattern mining algorithm
SPM	Sequential pattern mining
SPOC	Small private online courses
STEM	Science, technology, engineering and mathematics
TF-IDF	Term-frequency inverse-document frequency
VLE	Virtual learning environment

List of Notations

Symbol	Name
$\cup$	Union operator
$\cap$	Intersect operator
$\otimes$	Hadamard product
$[\cdot, \cdot]$	Vector concatenation
$\wedge$	logical AND operator
$\langle \cdot, \cdot \rangle$	Distance measure between two vectors
i, j, k	Indices for iterations
$\mathcal{D}$	Dataset
N	Total number of instances in the dataset $\mathcal{D}$
$\mathbf{x}$	Input vector
$\mathbf{y}$	Target grade output
$\hat{\mathbf{y}}$	Predicted grade outcome
L_x	Input sequence length
L_y	Vector length for the target grade output

μ	Mean component
Σ	Covariance matrix
ϕ	Mixture model weights
$\gamma(i, k)$	Cluster assignment of the i th data point to the k th cluster
$\mathcal{N}(\mathbf{x} \boldsymbol{\mu}, \boldsymbol{\Sigma})$	Conditional probability of $\mathbf{x}$ given Gaussian distribution parameterized by $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$
$\mathcal{R}$	Lesson preparation preference
$\mathcal{Q}$	Grade quintile
$N(\mathcal{R}, \mathcal{Q})$	Number of learners who performed preferred lesson preparation $\mathcal{R}$ and obtained grade quantile $\mathcal{Q}$
$P(\mathcal{R})$	Probability of learner performing preferred lesson preparation $\mathcal{R}$
$P(\mathcal{R}, \mathcal{Q})$	Joint probability of learner performing preferred lesson preparation $\mathcal{R}$ and obtaining grade quantile $\mathcal{Q}$
$\mathcal{L}(\mathcal{R}, \mathcal{Q})$	Likelihood ratio of learners who performed preferred lesson preparation $\mathcal{R}$ and obtained grade quantile $\mathcal{Q}$
$\mathbb{D}_{\text{Jaccard}}$	Jaccard distance measure
$\mathbb{D}_{\text{CWO}}$	Common word order measure
$\mathbb{D}_{\text{idf}}$	Inverse document frequency measure
$\mathbb{D}_{\text{AR}}$	Term rarity measure
$\mathbb{D}_{\text{Aggr}}$	Aggregated similarity measure
$\mathbb{D}_{\text{MCWO}}$	Modified common word order measure
$\mathbb{D}_{\text{L}}$	Sequence length difference measure

w_{Jaccard}	Mixture weight for Jaccard distance measure
w_{MCWO}	Mixture weight for modified common word order measure
w_{AR}	Mixture weight for term rarity measure
w_{L}	Mixture weight for sequence length difference measure
Ξ	Pairwise similarity matrix
ξ_{ij}	Pairwise similarity measure
ρ_{ij}	Responsibility message for Affinity propagation algorithm
α_{ij}	Available message for Affinity propagation algorithm
q^k	Response for the k th question within the self-reported questionnaire
p^k	k latent factor from the self-reported questionnaire
$\mathbf{p}$	Psychometric covariate vector
f	Prediction model
ψ	Activation function for neural network layers
$\mathbf{W}$	Neural network weight
b	Neural network bias
κ	Convolutional neural network kernel
θ	Aggregated set of model parameter
ϵ_i	Residual term

$\mathbf{v}$	Embedded input sequence
$\mathbf{z}_i^{seq}$	Hidden feature vector for input sequence $\mathbf{x}_i$
$\mathbf{z}_i^{psych}$	Hidden feature vector for the psychometric variable
$\mathbf{z}_i^{joint}$	Joint embedding vector
L_{emb}	Vector length for embedding vector
L_{seq}	Vector length for $\mathbf{z}_i^{seq}$
L_{psych}	Vector length of $\mathbf{z}_i^{psych}$
σ	Standard deviation component of the proposed MV identification model
ϵ_h	Representation threshold for the multi-valued criterion
ϵ_y	Output threshold for the multi-valued criterion
$\mathcal{L}_{AE}$	Reconstruction error for an autoencoder
$\mathcal{L}_{NLL}$	Negative log-likelihood loss function

List of Figures

3.1	Average online activities for different categories of learning resources across the teaching weeks.	26
3.2	Overview of the proposed clustering framework for identifying learning preparation preferences.	29
3.3	BIC scores for clusters identified during a regular teaching week.	31
3.4	Likelihood ratio of learners obtaining grade-quintiles after undertaking various lesson preparation preferences for the first examination.	35
3.5	Likelihood ratio of lesson preparation preferences during the recess week on grades obtained for the first examination (top) and second examination (bottom).	37
4.1	The structured activity of the online elementary mathematics module. . .	42
4.2	The unstructured activity of the online elementary mathematics module.	43
4.3	Overview of the proposed action sequence archetype mining algorithm. .	46
4.4	Pre-processing process for the modified common word order measure. . .	48
4.5	An illustrative example of a PLWAP tree.	54
4.6	Pairwise similarity of action sequences measured using individual similarity measure and the proposed aggregated similarity measure for the video activity.	56
4.7	Illustrative example of clusters identified via the proposed aggregated similarity measure for the video activity.	57
4.8	Decision tree for the video activity.	61
4.9	Decision tree for the structured activity.	62
4.10	Decision tree for the unstructured activity.	63
5.1	Overview of an autoencoder.	70
6.1	The 2x2 Hierarchical Motivation Framework.	83

LIST OF FIGURES

6.2	Overview of the proposed joint-space sequential grade prediction model.	86
6.3	Overview of the baseline frequency-based models (left) and baseline deep neural models (right).	92
6.4	Average model loss at each learning rate for the proposed Bayesian-regularized sequence encoder at input sequence length $L_x = 20$. The optimal learning rate is chosen from the region with the highest rate of decrease of model loss.	93
7.1	Overview of the standard grade prediction model.	99
7.2	Identifying identical input sequences.	100
7.3	Grade pair distribution of identical input sequences in an undergraduate digital signal processing course. Each location on this plot represents the grades achieved from two identical action sequences (each of length 50). The shade denotes the number of instances for each grade pair.	101
7.4	Proportion of clickstream sequences with $>5\%$ grade differences within the truncated (solid) and augmented (dotted) dataset at various L_x .	102
7.5	Sliding window pre-processing data augmentation where the original action sequence is segmented into sequence each of length L_x .	102
7.6	Training and testing performances of existing grade prediction models trained using the complete dataset, WOID, and WOSIM at various sequence lengths L_x .	104
7.7	Overview of the proposed MV-sensitive grade prediction model.	105
7.8	Visualisation of feature space representing (a) the predicted grades, (b) target grades, (c) identical input sequences with $>5\%$ grade differences, (d) absolute errors, and (e) SD measures for BL (left) and BR (right). Decision boundaries of the MV classifiers derived from SVM are illustrated in Figs. (c)-(e).	117
7.9	Regression performance in terms of MAE and RMSE for DSP for input sequence length of (i) 10, (ii) 20 and (iii) 35.	122
7.10	Regression performance in terms of MAE and RMSE for LAKDC dataset for input sequence length of (i) 10, (ii) 20 and (iii) 35.	123

List of Tables

2.1	Summary of established learning style framework and related works identifying learning styles within VLEs.	19
3.1	Composition of learning preparation preferences obtained using the GMM clustering algorithm for each activity category.	32
4.1	List of logged actions for the elementary Mathematics online course. . . .	45
4.2	Header table example for the illustrative example depicted in Fig. 4.5 . .	53
4.3	Action sequence profile.	57
4.4	Action sequence archetypes for the video activity.	58
4.5	Action sequence archetypes for the structured activity.	58
4.6	Action sequence archetypes for the unstructured activity.	59
4.7	Mean activity scores for learners who stop during Phase 1 and persist to Phase 2.	60
4.8	Drop-out prediction performance via 2x3 fold cross-validated logistic regression.	60
5.1	Summary of prediction models for detecting at-risk learners and their corresponding feature sets.	81
6.1	List of logged actions for the DSP dataset.	90
6.2	Grade prediction performance using weekly clickstream sequences and complete historical clickstream data	95
6.3	Grade prediction performance using weekly clickstream sequences and non-cognitive covariates	95
7.1	List of logged actions for the LAKDC dataset.	114
7.2	Model configurations based on which performance was evaluated.	115
7.3	MV classification performances for the DSP dataset	120

LIST OF TABLES

7.4	Percentage improvement of MV classification performances for BR over BL (%).	121
7.5	Proportion of MV inputs identified from the DSP dataset (%).	121
7.6	Grade Prediction Performances of GP1-GP5.	124
7.7	Improvement in Grade Prediction Performances with GP2 to GP5 over GP1 in terms of MAE and RMSE.	125

Chapter 1

Introduction

Learning analytics (LA) and educational data mining (EDM) refer to the analysis of educational data; the former aims to understand and optimize learning and the environment in which it occurs as a whole [1] while the latter develops methods for exploring educational data and understanding individual stakeholders within the system [2, 3]. With recent advancement in technology, new opportunities have been presented to allow for the validation of existing theories and to gain a better understanding of learning experiences within virtual learning environments (VLEs). In this thesis, the development of machine learning algorithms to analyze fine-grain clickstream learning data for the purpose of understanding online learning behavior and its impact on learning outcomes will be discussed.

1.1 Motivation

The advent of technology in education resulted in a paradigm shift. Hitherto, studies that analyze learners and their learning had relied primarily on empirical classroom observations, self-reported questionnaires and speak-aloud interviews to probe into learning experiences. This type of data collection is costly, time-consuming and may introduce bias to the data [4]. When coding classroom observations, researchers have the dual task of discerning learner behavior while ensuring timely documentation of these obser-

vations. Furthermore, to study a particular set of behaviors, researchers also had to prioritize these behaviors over other similar behaviors [5]. As analyses demanded higher fidelity in the observations, their scalability suffered significantly [6]. The use of technology can mitigate such limitation, albeit a double-edged sword that introduces its own set of challenges.

This technological revolution in education gifts a trove of education data. Compared to human-coded observations, these data are logged consistently on a larger scale. This is advantageous since existing learning theories can now be validated over larger populations. In addition, information from multiple sources such as video and audio recording, physiological sensors and human-machine interactions with computer peripherals (clickstream data) can be synchronized for multi-modal analyses that may reveal deeper insights about learning [7]. Furthermore, since these sensors are constantly acquiring data associated with learning behavior, finer clickstream data that were previously not feasible are now available for analysis [7]. The fields of EDM and LA ensued from this paradigm shift. There are many factors differentiating EDM and LA. Generally, EDM is concerned with developing methods for exploring the unique types of data and using those methods to better understand students and the setting which they learn [2]. LA research, on the other hand, focuses on the measurement, collection, analysis and reporting of data about learners and their contexts, for purposes of understanding and optimizing learning and the environment in which it occurs [8]. Unlike conventional data mining research, EDM has both applied research objectives (such as improving the learning process and guiding students' learning) and pure research objectives (such as achieving a deeper understanding of educational phenomena [3]). While EDM and LA share a set of common goals [8], EDM and LA can be differentiated in terms of the type of discovery, reduction and holism, origin, techniques and methods, and adaptation and personalization [8]. More importantly, although both fields leverage on human judgment, EDM research leverages human judgment as a tool while leveraging human judgment is the primary aim of LA research.

Even though the use of technology in education has brought about many advantages, there are several notable issues within VLEs. As these VLEs offer relatively new vessels to deliver learning resources, learners are required to juggle between their goals to assim-

ilate knowledge from new content modes and the need to learn and navigate the novel platform [9]. As such, analysis involving clickstream data is challenging since learning behaviors have to be differentiated from navigational patterns. When this is not achieved, analysis results may be plagued with uncertainties [10].

While analysis methodologies have changed, core objectives of educational research, i.e., uncovering insights about the learning process, remained the same. In order to ensure model interpretability, initial studies employed rule-based algorithms such as decision trees and clustering algorithms [11]. However, as analysis complexity increases significantly over the years, these algorithms are incapable of modeling the complex and convoluted behaviors. Although deep learning techniques have developed significantly in the same period, these algorithms often focused on achieving stellar prediction performances at the expense of model interpretability, making these deep learning techniques less ideal for many education settings. There is, hence, a need for the development of bespoke algorithms that are tailored for LA and EDM .

This thesis addresses the above challenges from two lenses —a macro- and micro-perspective. While both perspectives ultimately share a common goal of identifying learners who are at risk, each perspective is driven by a different motive. The macro perspective accepts tolerances between individual differences and analyzes the relationship between different learner groups (based on various aspects such as their learning preferences, demographic, historical academic performances) and their learning outcomes. This has mainly been driven from a business perspective to reduce cost, improve pedagogy, and inform policy changes [1] since the cost of re-educating learners who are unsuccessful in completing the course is typically high [12,13]. On the other hand, the micro perspective focuses on differences within an individual and its effect on learning outcomes. This is subsumed under the umbrella of efforts that have been made to enable personalization within learning.

Studies conducted within this thesis are constrained to clickstream data for several reasons. First, complete information of a learner is often unavailable at the onset of a course. Often, only their prior academic achievements are available and, even that, is often subjected to data privacy policies. Subsequently, learners' preparation and re-

source usage during teaching weeks as well as their physiological and physical logs are made available as they progress through the course before their formative and summative assessment grades can be retrieved. Given the general consensus that at-risk learners respond best when interventions are provided at the earliest opportunity, such a timely process requires the predictors to be available in time for any interventions to be administered [14]. These clickstream data that are available across different courses, therefore, provide a perfect opportunity to detect unproductive learning behaviors. Furthermore, in some education settings, e.g., massive open online courses (MOOCs), clickstream data may be the only source of information about the learners.

1.2 Organization of the Thesis

This thesis addresses the analysis of education clickstream data via data mining and machine learning approaches to gain a better understanding of learning behaviors. The scope of this thesis is divided into two parts: Part I focuses on the use of data mining methods to identify learner sub-populations with similar learning behavior to establish the relationships between the identified learning behaviors, existing learning theories and learning outcomes. The focus is shifted towards modeling fine-grain clickstream data via deep learning techniques in Part II. Details of each chapter are summarized as follows:

In Part I, Chapter 2 reviews existing literature on data mining techniques for the identification of different learning behaviors and theories that drive these studies to understand how learning behaviors may impact learning outcomes. Given the vast diversity in research objectives, the discussion is presented independently according to the objectives, the type of data available, theoretical bases for these studies and the algorithms that were employed to achieve these objectives.

In Chapter 3, lesson preparation preferences in blended courses will be analyzed. Given the relatively rigid curriculum structures, approaches undertaken by learners to navigate these curricula are studied in relation to their ability to transfer their knowledge onto new domains. To identify and differentiate lesson preparation preferences, a data-driven clustering approach is proposed to identify learner sub-populations who had

demonstrated similar lesson preparation preferences. This is achieved via the Gaussian mixture model (GMM) clustering algorithm and a heuristic rule-based classifier. Finally, the relationship between lesson preparation preferences and learners' summative assessment performances is discussed.

In Chapter 4, the focus is shifted onto the identification of problem-solving approaches for worded arithmetic problems. Scripts and schemas are first reviewed to allow readers to appreciate the cognitive process undertaken by a learner when solving complex problems. The proposed sequential clustering algorithm that identifies different problem-solving approaches is then presented. Termed as the action sequence archetypes (ASAs), the impact of these problem-solving approaches on the persistence of learners to progress further in the online course is evaluated.

In Part II, Chapter 5 begins by formalizing the grade prediction task. The use of deep neural networks (DNNs) for modeling clickstream sequences is then presented, followed by a review of various DNN components. Subsequently, the relationship between DNN-based grade prediction model and the linear regression paradigm is established. This serves as the foundation for the proposed algorithm in Chapters 6 and 7. The chapter closes with a review of existing grade prediction literature.

In Chapter 6, the focus on the modeling of learning outcomes via clickstream sequences and non-cognitive traits. To incorporate both features for forecasting learner grades, a joint-space sequential grade prediction is presented. The proposed algorithm is evaluated on a dataset collected from a digital signal processing (DSP) course.

In Chapter 7, the inherent multi-valued (MV) characteristic of the education dataset and its adverse impact on grade prediction performance is first highlighted. The proposed MV identification module is then presented. The chapter concludes by evaluating the proposed MV identification module on its performance to identify MV inputs and its ability to predict learners' examination grades.

1.3 Contribution of the Thesis

Contributions made by the author are mainly described in Chapters 3 and 4 in Part I and Chapter 6 and 7 in Part II.

In Chapters 3 and 4, the author’s contributions are made towards the identification of online learning behaviors that are analyzed in conjunction with existing learning theories. In these chapters, resource usage for lesson preparation and problem-solving approaches extracted from the collected data are studied in tandem with concepts of cognitive persistence [15] and scripts [16], respectively.

For the proposed approach in Chapter 3, the underlying objective is to provide a data-driven definition of behavioral normality for lesson preparation within VLEs and study the impact of lesson preparation preferences on learning outcomes. This is important since while usage of digital learning resources is the main form of learning in these environments, study habits can, however, vary across different platforms as a result of its underlying pedagogies, platform features and layouts. Furthermore, given that many small private online courses (SPOCs) have rigid curriculum structures, this can limit the flexibility given to each learner to manage and strategize their learning. Since hand-crafted features are often not platform-agnostic, prediction models based on expert rule-based systems are, therefore, unable to generalize well on datasets collected under a different educational setting. The proposed clustering approach is centered on the GMM clustering to identify learner sub-populations with similar lesson preparation preferences. These learner sub-populations are then differentiated by the amount of effort they had placed into different categories of learning resources that have varying relevance to the current teaching week. A set of heuristics that defines the relative measure of normalcy and amount of effort is proposed and employed via a rule-based system to characterize the clusters. Finally, the identified weekly lesson preparation preferences are evaluated to determine its impact on examination grade outcomes for assessment items targeted at various cognitive complexities. The study found that learners who demonstrated cognitive persistence in their lesson preparation were associated with higher cognitive abilities to transfer their knowledge onto novel problems. It is believed that their persistence in learning stems from innately higher motivation, which, in turn, impact their study habits

and, ultimately, learning outcomes. This contribution has been published as Kelvin H. R. Ng, K. Hartman, N. Goodkin, Andy W. H. Khong, “Mining for preparatory processes of transfer learning in a blended course,” in *Proc. Amer. Geophys. Union Fall Meet., AGU Fall Meeting*, New Orleans, LA, USA, Dec 11 - 15 2017.

For the proposed approach in Chapter 4, the underlying objective is to identify different problem-solving approaches that learners have undertaken when faced with complex multi-step worded arithmetic problems. This is achieved via the proposed clustering approach that aims at reducing cognitive load of human annotators to differentiate these problem-solving approach. In this study, learners were taught in conventional classrooms to solve these problems using a series of steps known as the RIGHT method [17]. The RIGHT method decomposes a complex problem into smaller sub-problems for learners to focus on the application of specific skills at each step. They are then introduced to a VLE where the concept is first reinforced via video lectures and then practices were made available for them to execute the RIGHT method. To identify problem-solving approaches within the VLE, the proposed sequence clustering algorithm employs a combination of pairwise sequence distance measures to differentiate clickstream sequences. Independently, each of these distance measures designed to capture different aspects of the sequences is not effective in differentiating the clickstream sequences. By considering the four different aspects, nuances between clickstream sequences can be identified. A parametric clustering algorithm is then applied to identify groups of similar clickstream sequences before they are summarized via a sequential pattern mining (SPM) algorithm. Based on frequent sequence patterns identified within each cluster, the identified clusters are aggregated under a greater label termed as the action sequence archetypes. The proposed sequence clustering algorithm is, therefore, able to detect if learners followed the RIGHT method or made conscientious efforts to deviate away from it. An analysis of their preferred problem-solving approaches revealed that learners who deviated away from the scripted process have the tendency to progress further in their learning. This contribution has been published as Kelvin H. R. Ng, K. Hartman, K. Liu, and Andy W. H. Khong, “Modelling the way: Using action sequence archetypes to differentiate learning pathways from learning outcomes,” in *Proc. 9th Int. Conf. Educ. Data Min., EDM 2016*, Raleigh, NC, USA, Jun 29 - Jul 2 2016, pp. 167-174.

In Chapters 6 and 7, the focus shifts to the use of DNNs for grade prediction via clickstream sequences. It is well-known that the feature engineering process limits the generalizability of the grade prediction models. While similar applications that have seen successes with DNNs, the ability of DNNs to predict learning outcomes via clickstream sequences is not well-studied. Furthermore, since interaction clickstream sequences collected within VLEs are manifestation of the cognitive process, it is expected that the relationship between clickstream sequences and learning outcomes are plagued with uncertainties. The joint-space grade prediction model and the MV identification model are proposed to overcome some of these limitations and improve grade prediction performance.

For the proposed joint-space grade prediction model in Chapter 6, the objective is to improve grade prediction through the use of DNNs and non-cognitive traits as covariates to model clickstream sequences. Earlier studies have focused on the use of hand-crafted features with machine learning algorithms to ensure that prediction results remain interpretable. Given the complexity of clickstream sequences, it has, however, become increasingly difficult to design comprehensive sets of features that can encapsulate sequential patterns within the data. Furthermore, at the same time, it is well-known that non-cognitive traits are known to regulate learning behaviors. Learners with different non-cognitive traits performing the same clickstream sequences may, therefore, achieve different learning outcomes. To capture nuances between non-cognitive traits and clickstream sequences, a deep neural architecture that employs a joint-space embedding to fuse the two variables is proposed. The results highlighted that sequential-based modeling of interaction sequences is advantageous to discern clickstream learning behaviors. In addition, when the analysis period for frequency-based models is changed, performance of frequency-based models deteriorates rapidly. Furthermore, the results also support the hypothesis that non-cognitive traits can be employed to regulate grade prediction models that analyze clickstream sequences. This contribution has been published as Kelvin H. R. Ng, S. Tatinati, and Andy W. H. Khong, "Online education evaluation for signal processing course through student learning pathways," in *Proc. IEEE Int. Conf. Acoust., Speech and Signal Process., ICASSP*, 2018, Calgary, AB, Canada, Apr 15-20, pp. 6458-6462.

For the proposed MV identification model in Chapter 7, the underlying objective is to identify the MV characteristic of the dataset to improve grade prediction performances. Evidence pertaining to the MV characteristic of interaction clickstream sequences presented in this preliminary study reveals that similar clickstream sequences can be associated with a wide range of grade outcomes. This contradicts the underlying assumption of grade prediction models that assumes a surjective relationship between its inputs and outputs, i.e., the relationship follows a many-to-one mapping. As a consequence, grade prediction performance is severely impacted. Due to the complexity of clickstream sequential data, existing data mining approaches are not effective in identifying inputs that demonstrate this MV characteristic. The proposed MV identification module is, therefore, proposed to characterize the MV characteristic of the education clickstream dataset. This is achieved via the proposed Bayesian-regularized deep neural network. The Bayesian-regularized deep neural network exploits evidences from the preliminary study showing that MV inputs are associated with high error variances due to their incompatibility with machine learning methods that estimate the conditional averages given the inputs. By embedding the error variance on top of the target grade outputs within its feature space, the Bayesian-regularized deep neural network is able to differentiate MV input sequences from non-MV ones. This contribution has been published as Kelvin H. R. Ng, S. Tatinati, and Andy W. H. Khong, “Characterization of fuzziness for grade prediction using deep neural networks,” in *Companion Proc. 9th Int. Learn. Anal. Knowl. Conf., LAK 2019*, Tempe, AZ, USA, May 4-8 2019 and Kelvin. H. R. Ng, S. Tatinati and A. W. H. Khong, “Grade Prediction From Multi-Valued Click-Stream Traces via Bayesian-Regularized Deep Neural Networks,” in *IEEE Trans. Signal Process.*, vol. 69, pp. 1477-1491, 2021.

Chapter 8 finally summarizes the thesis and proposes several directions for future research.

Part I

Identification of Learning Behavior via Data Mining Approaches

Chapter 2

Review of Data Mining Approaches for Identifying Learning Behaviors

In Part I, the focus is on the identification of learning behavior profiles. This is important since generalizing learner populations into different profiles enable some degree of personalization without overwhelming instructors and platform designers, and allow for diagnoses into pedagogical limitations. Consequently, learners can derive learning experiences that are more pleasurable despite individual differences in preferences while policymakers and administrators can employ such information to improve the curriculum. Given numerous stakeholders and their diverse needs, learning behaviors may be differentiated based on different criteria. This chapter first begins by identifying the stakeholders of virtual learning environments (VLEs) and the type of information that can be collected from VLEs. Existing studies that analyze different types of learning behavior and identify various learner groups are then discussed. Given the vast factors that can affect learning, the discussion is limited to the identification of learning behaviors within VLEs, problem-solving approaches and various off-task behaviors that are common within VLEs.

2.1 Stakeholders for learning analytics

With majority of existing studies conducted in the past twenty years, learning analytics and educational data mining are considered budding research fields. These studies are primarily driven by the need of their users and their associated (and often diverse) set of objectives [1, 18, 19]. To appreciate the motivation behind existing studies, it is important to establish an understanding of the stakeholders and their respective roles. Stakeholders of VLEs are broadly categorized into institutions, instructors, researchers and learners; their respective objectives can be facilitated using methodologies developed in the two research fields.

Institutions, as with all business entities, are primarily concerned with providing quality education while keeping the cost low [1]. The advent of technology in education allow widespread distribution of content to the masses, thereby effectively lowering the cost of producing learning materials. However, the profit is only maximized when learners achieve the desired learning outcomes and complete the course successfully since opportunity costs are incurred when learners retake courses in order to fulfill program requirements [20, 21]. As representatives of these institutions and primary agents to deliver course content, instructors are tasked to ensure learners are able to complete the courses. Undertaking each course with different levels of prior knowledge, the rate at which each learner assimilate new information may differ. Instructors must, therefore, take individual efficacy into account when teaching and alter their teaching pace accordingly to avoid learners being left behind [1]. This can be daunting since instructors often have to evaluate learners perceptively (apart from occasional summative assessments), and this is further exacerbated by large cohort sizes. Furthermore, apart from the rate of assimilation, learners can be oriented (achievement- or mastery-oriented) differently [22]. This difference is reflected in their learning experiences [23–25]. Within a course, achievement-oriented learners can often be seen maximizing assessment grades while mastery-oriented learners may undertake more practices and review activities. As learners can make use of the flexibility granted in tertiary education programmes to achieve these goals, models developed for learning analytics can analyze historical records to help learners understand whether they would successfully navigate the programme after deviating from the rec-

ommended curriculum [26]. While there are many benefits to employ learning analytics to aid learners, it remains debatable if learners should have access to such information as learners may also be discouraged from pursuing their learning if these models constantly provide negative feedback [27, 28].

In addition to aiding various stakeholders achieve their objectives, researchers are also motivated by the pursuit of gaining a better understanding of learning and the cognitive process [11]. Naturally, they are also characterized by the most diverse set of objectives [11]. Central to these objectives lies the characterization of learning behavior and the evaluation of learning outcomes. Characterization of learning behaviors such as video-watching behavior, forum discussion, course pacing and resource release schedule (spacing between assessment) is beneficial for improving learning resources [29]. This would also allow researchers to understand how learners navigate the platforms, consume learning resources and manage their learning. Through the use of technology, student learning can also be evaluated differently. One such example is the use of knowledge tracing to assess learners' proficiency through their masteries of various knowledge components as opposed to the conventional summative and formative assessments [30, 31]. Like cognition, knowledge tracing is able to account for the probability of a learner guessing correctly or accidentally submitting wrong answers at their current proficiency. In order for machines to aid stakeholders in achieving their objectives, the availability of suitable data as well as the development of new algorithms must occur simultaneously.

2.2 The role of VLEs and the available information within these platforms

Virtual learning environments (VLEs) constitute the main source of learning activity data and can be categorized into learning management systems (LMSs), online course sites and intelligent tutoring systems (ITSs) and interactive learning environments that employ gamification, simulation and first-person experiences as their pedagogies. While nuanced in their purposes, each type of VLE collects different information pertaining to their learners.

LMSs, such as Blackboard [32] and Moodle [33], serve a dual role in documenting and tracking learner progress, and hosting learning resources for learners to download and consume offline. Since these systems are typically linked to institution databases, they often have access to a comprehensive set of students' demographic information (such as their age, nationality, highest attained academic achievements, social economic status indicators, and their historical assessment and course outcomes) about their learners. Furthermore, given that resource requests are automatically logged as network traffic on the worldwide web, information associated with the downloading of learning resources by learners onto their personal devices are also recorded. This also applies when learners undertake practice assessments and submit assignments within the LMS.

The role of online course sites, on the other hand, is primarily to host learning resources for direct consumption within these platforms. These platforms include massive open online courses (MOOCs), e.g., edX [34], Coursera [35], Khan academy [36], Udemy [37] and Udacity [38], and small private online courses (SPOCs); the former cater to the general masses while the audience for the latter comprises mainly students who are enrolled in the hosting institutions. While both LMSs and these online course sites host learning resources such as video lecture content, practice assessments and discussion forums, those found on LMSs are mainly for distribution purposes whereas that on online course sites are actively incorporated as part of their curriculum. Although they typically do not have access to demographic information as with LMSs, these platforms may occasionally request that learners supplement these information during the registration process. Apart from resource accesses, learner-activity interactions that are logged may include the amount of time and effort placed into each learning resource and the order in which they were utilized in their learning. More recently, there has been increased interest by these platforms to acquire high-resolution data such as video-watching behavior [39–42] (playing, pausing, and fast-forwarding) and step-wise problem-solving steps.

ITSs, such as the Cognitive tutor [43] and AutoTutor [44], are different from the above-mentioned environments; these systems focus on the development of problem-solving skills through immediate and customized instructions and feedback [45]. Learners' proficiency to apply the problem-solving skills are evaluated as mastery of concepts as opposed to grade outcomes of summative assessments. Similar to MOOCs, ITSs log fine-

grain clickstream interactions, i.e., step-wise problem-solving steps, within the platforms in order to tailor intervention prognosis to enhance their learning experiences. However, due to a much higher amount of effort involved to design an ITS compared to LMSs, MOOCs and SPOCs, most ITSs consequently only cover limited content.

While additional information such as physiological signals and personality traits are currently not widely adopted within digital spaces, they are collected in controlled laboratory settings to be studied in conjunction with learning behavior and outcomes [24, 46]. Physiological sensors are deployed to collect learners' electrocardiograms [47, 48], electroencephalograms [48, 49], electrodermal activity [48, 50]. Furthermore, visual and audio cues to infer their gross body movements [51], gaze [47, 49, 50, 52], facial expressions [47, 51, 53, 54] and speech [49] may be collected to understand learners' affective states such as boredom, engagement, curiosity, interest, surprise, delight, anxiety, and frustration [54]. Occasionally, personality traits, such as their motivation and e-learning readiness, may be quantified through self-reported questionnaires.

2.3 Theoretical basis for existing learning analytics studies

It is well-known that learning, being a personalized process, is confounded by the aforementioned factors [23, 24, 55–59]. For example, learners with different personalities may demonstrate significant differences in their choice of learning strategy, time management, engagement, and participation methods [60]. To better understand learning, existing studies, therefore, built their narratives around existing theories about learning.

Engagement is considered an important pre-requisite for online learning. Prior studies have found that time spent engaging in learning activities is generally positively correlated with performance and attrition [61]. In formal classroom settings where instructors determine the time allocated for learning activities, good classroom practices aimed at promoting effective use of time is also correlated with better achievements [62]. On the other hand, Carroll's time-on-task hypothesis states that a learner who spent longer duration engaging with materials would have more opportunities to learn [63]. That is,

when this time is spent off-task, this reduces the number of learning opportunities and, in turn, affect their learning. It is, however, arguable whether all off-task behaviors are counter-productive.

One of the most widely studied phenomenon in conventional classrooms is preferences in learning modality, which may include video lectures, reading materials, hands-on activities [64–66]. This is often referred to as their preferred learning styles. Learning styles play an important role since learners are most comfortable and efficient to process the information and construct new knowledge when the modality of these resources are aligned with their personal preferences [67–69]. While instructors may experience inertia to design the same content across different modalities (due to the additional efforts), the fact that their content can be distributed to wider audiences through the use of technology makes it more gratifying for them to do so.

Many well-established frameworks associated with learning styles have been proposed for conventional classroom settings to differentiate learning styles. They include the VARK (visual, auditory, kinesthetic, read/write) model [70], Kolb model [71], Felder-Silverman Learning style model (FSLSM) [72], Keefe’s learning styles [65], and Honey and Mumford model [73]. While there exist some nuances across these frameworks, they share a common theme of describing an individual’s preferences to the type of content and their cognitive strategies to assimilate these learning resources. Since these frameworks were mostly established for conventional classroom settings, they were mainly identified through the use of self-reported questionnaires. However, self-reported questionnaires can be highly biased, error-prone and costly [4] and, with the advent of education technology, automatic detection of these learning styles is preferred [74, 75]. Only limited studies in learning styles have, however, demonstrated a positive effect [76] while many studies have reported negative results. Nevertheless, there are merits for identifying learning styles, particularly for learning-styles hypotheses, since they can have action-relevant conclusions. The reason for its wide adoption by the research community and general public in the past stems from successes in fostering learning and instruction and that people generally would believe that they are different and preferred personalized experiences [77].

The type of behaviors demonstrated by learners may also vary based on non-cognitive traits such as motivation and self-regulation of learning. Motivation is described as a conceptual construct that directs and improves learner’s behavior towards a goal [78]. There are three types of motivation—*intrinsic motivation*, *extrinsic motivation* and *amotivation* [79]. A learner who is high on *intrinsic motivation* derives pleasure and enjoyment by perceiving a task as interesting or challenging. In contrast, *extrinsically motivated* learners expect rewards for their participation in the task that are not naturally associated with the activity. *Amotivation*, on the other hand, is the lack of motivation. Naturally, learners who are motivated differently also approach learning tasks differently. Prior studies have demonstrated that motivation significantly impact learner participation and the most successful participants are naturally *intrinsically motivated* [80].

Given that contemporary VLEs do not provide learners with much support and guidance throughout their learning journeys, a learner’s degree of self-regulation is, therefore, crucial in determining his/her success. Self-regulation of learning is characterized by the ability to initiate meta-cognitive, cognitive, affective, motivational and behavioral processes to undertake tasks in order to achieve their learning goals and persist when faced with challenges [81]. Learners who fail to regulate their learning are often feel frustrated and disengage from the task [82], and they, in turn, become less successful [83–85]. While learners who reported higher in motivation, commitment to learn and formal education tend to have higher self-regulation in learning [86, 87], it has been argued that self-regulation is a skill, rather than a trait, that can be developed through experiences and practices strategies [88–90]. Prior studies have found that VLEs that incorporate self-regulation strategies such as time management, meta-cognition and regulation of effort produce learners with improved academic achievements [91–93].

Unlike learning style frameworks, which describe learners’ preferences for different content modality, learning styles, cognitive style is a construct that defines an individual’s consistent approach to organizing and processing information during learning, which is different from cognitive ability [94]. Under this construct, learners can differ in terms of two dimensions—the *wholist-analytic dimension* and the *verbaliser-imager dimension* [95]. The *wholist-analytic dimension* describes a habitual approach to organize and structure information, where individual high on analytics may deconstruct information

to its components while wholists would retain a global view of information. Prior studies have shown that neither wholist or analytics styles are optimum for memorization of content and learners may benefit from the alternatives [96]. Preferences in learning materials may also be influenced by external factors. For example, cultural differences may result in learners with different preferences for content modality, help-seeking and off-task behaviors [97–99].

2.4 Existing studies identifying various learning behaviors

Initial efforts for identifying learning behaviors employed various data mining algorithms to map learning-activity data to the learning styles that were described in these frameworks. In general, to capture these precise definitions concisely, these studies relied mostly on rule-based algorithms to map the learning styles. For example, the components of FSLSM were determined through navigational and engagement patterns via a rule-based algorithm [101], multilayer perceptron (MLP) [102] and fuzzy inference system [103]. These learning styles were also identified through assessment scores with the use of decision trees and naive Bayes methods [100]. Since FSLSM differentiated learning styles into four major components, with two sub-categories each, a fuzzy c-means algorithm was first employed to determine learning styles that were demonstrated by the learners through the eight sub-categories before a neural network was applied to determine the four learning styles. This approach takes into account the various learning styles that were undertaken by learners throughout the semester [111]. Similar approaches were also undertaken to identify learning styles defined by the VARK model [115] and Kolb’s learning styles [116]. Consequently, they also do not uncover undefined learning styles that were induced due to the paradigm shift with the introduction of VLEs.

In addition, some studies undertook an exploratory approach to discover online learning behaviors [117, 118]. Learners were differentiated by their platform usage patterns (such as time spent on platform and number of clicks to access various segments) via an enhanced k -nearest neighbor (KNN) algorithm [104] and a neural network [105]. Learn-

Table 2.1: Summary of established learning style framework and related works identifying learning styles within VLEs.

Theme	Sub-category	Literature
Established learning style frameworks	-	Keefes learning styles [65] VARL model [70] Kolbs model [71] FSLSM [72] Honey and Mumford model [65]
Identify learning styles from existing frameworks	Assessment scores	Decision trees [100] Nave Bayes [100]
	Navigation/engagement patterns	Rule-based systems [101] Neural networks [102] Fuzzy inference [103]
Discovery of learning styles	Platform usage	k -nearest neighbour [104] Neural networks [105] k-Means [106–108] Hierarchical agglomerative clustering [107–109]
	Attendance	Fuzzy c-means [110]
	Navigation/engagement patterns	Fuzzy c-means [111] Neural networks [111] Hidden Markov models [77] Other statistical methods [76, 109]
	In-video interactions	k-means [39] Markov models [40, 41] Hidden Markov models [42] Other statistical methods [112]
	Intelligent tutoring systems	k-means [113] Markov models [112] Other statistical methods [114]

ing preferences were also identified through the amount and type of content learners consume during their learning using k -means [106–108], hierarchical agglomerative clustering (HAC) [107, 109]. Learning styles were also inferred from class attendance data via fuzzy C-means [110].

The reason for the popularity of learning style concepts being widely adopted by the research community and general public in the past stems from its successes in fostering learning and instruction and that people generally would believe that they are different and preferred personalized experiences [77]. Regardless how attractive this idea may be,

a meta-analysis conducted by Pashler et. al [77] on existing learning-style studies reveal that only limited studies among many studies can demonstrate a positive effect [76]. Nevertheless, there are merits for identifying learning styles, particularly for learning styles hypotheses, since they can have action-relevant conclusions.

Rather than analyzing instantaneous snapshots of learning, various studies have considered the consumed learning resources in totality and the evolution of preferences over an extended duration to identify learners' dominant learning styles. Different learning resources were first categorized based on the FSLSM's learning styles and their accesses were differentiated via fuzzy C-means. A neural network then utilizes the learning style information from the clustering algorithm and the sequences pertaining to document accesses throughout the semester to determine their preferred learning styles [111]. Latent class analysis was applied on identified weekly learning styles to quantify their learning strategies throughout the semester [109]. Hidden Markov models (HMMs) were also used to identify learners with similar strategies in Betty's Brain, an open-ended learning environment [112]. Instances of changes to their preferences and strategies were also captured using statistical methods [114] and HMMs [112].

Beyond analyzing general learning styles and strategies, micro-analyses focusing on interactions with specific content types were also conducted. As video content remains the primary source of content within these VLEs [119], in-video interactions were studied to understand the relationship to in-video drop-out and course drop-out. Two general approaches were observed with analyses that focus on video-watching behavior. The video-watching behaviors were either characterized using static or dynamic interaction patterns. For static interaction patterns, click frequencies were aggregated to identify peak interaction segments and this was examined in relation to in-video drop-out, i.e., learners navigating away from video content before completion [113]. The study revealed that segments with high interactions indicated either points of interests or confusion for learners. This information is critical for instructors to address any concerns that their learners have. Furthermore, by considering the type of video content, the authors were able to identify five different video-watching behaviors from these peak interactions. Time-based features such as the frequency and duration of pauses, seeking and replays were also clustered to identify different video-watching behaviors [39]. Since learners do

not often perform independent clicks but rather a series of clicks to achieve their desired outcome, video-watching clickstream data were also analyzed as clickstream sequences to capture learner behavior. The dynamics between clicks were captured using n -gram modeling to extract frequently occurring interaction sub-sequences [119]. Video-watching sequences were also represented using Markov chains [40,41] and HMMs [42]. Due to the ability to better differentiate nuances between video-watching sequences, these studies were able to identify more variety of video-watching behaviors compared to methods that employed static interaction patterns.

Analyses of problem-solving approaches within intelligent tutoring systems were also of interest. These analyses presented a perfect opportunity to precisely diagnose learning in order to design and administer interventions to overcome these challenges. This was not possible in conventional classrooms since coding for these behavior requires very precise and timely observations that are too costly and not scalable to bigger learner populations. Various problem-solving approaches were identified by applying visualization-based methodology [120], k -means clustering [121] and Markov decision processes [122] on clickstream interaction data.

While most students had undertaken productive learning experiences as expected, occasionally learners may also partake in off-task behaviors. These off-task behaviors include on-task conversation, off-task conversation, off-task solitary behavior, inactivity, playing, skipping, and/or gaming the system (systematic exploit of platform regularities for progression) [5, 123–126]. Gaming behaviors were further differentiated to help abuse and systematic trial-and-error [125, 127]. These behaviors had been studied in conventional classrooms [128–131] and was often associated with poorer learning due to the reduced learning opportunities [63, 132]. However, these studies considered all types of off-task instances under the same label. Arguably, not all off-task behaviors affect learning in the same way. For example, while help-seeking behaviors were positively correlated to learning [127, 133, 134], when differentiated into legitimate help-seeking versus help abuse, help abuse is negatively correlated to learning [5, 135, 136]. These studies also concluded that poorer learning due to off-task behaviors was likely due to fewer learning opportunities [125]. Furthermore, the type of off-task behavior demonstrated by an individual was more significant than the frequencies at which these behaviors were

demonstrated. The fact that harmful gaming behavior was negatively correlated to their preferences towards computers implies that these behaviors were induced by personal preferences rather than the learning environment [124, 136, 137]. Off-task detectors generally rely on human-coded classroom observations via an established protocol [138] to obtain the labels. The labels were determined using time-based and frequency-based platform usage features via a latent response model [139–141], decision trees, KNNs, Bayesian methods, neural network, propositional-logic rule based system and logistic regression [124]. It is worth noting that these studies achieved similar results regardless of the prediction models [124].

Without thinking fastidiously (WTF) behavior is another type of off-task behavior that is defined as a haphazard inquiry strategy [142, 143]. Learners can appear to be approaching the goal, but ended up further from them due to a lack of understanding on how to conduct inquiry [142, 143]. While prior studies have considered this as a type of off-task behavior [144], it has also been argued that WTF behavior is driven by different intent and, therefore, should be considered as a distinct off-task behavior [142]. Unlike off-task behaviors, wheel-spinning behaviors are learning efforts that do not help learner master a skill as they lack a lack of understanding of pre-requisite knowledge [145]. Synonymous to the perseverance of effort sub-scale of the grit inventory, persistence when faced with questions that a learner has yet to obtain relevant skills and knowledge to solve is considered counterproductive [146]. Greater utility may be gained by attempting easier questions or reinforcing pre-requisites that may aid the learner to overcome this challenge. Detection of wheel-spinning in VLEs has been demonstrated using correctness of question responses, response times, and use of hints via logistic regression and random forests [147, 148].

On the other hand, procrastination is considered a failure of time management. While other factors may determine whether a learner would begin the learning activity, procrastination, in general, has been associated with poorer learning. This may be modeled via the naive method of computing the average delay since learners have been given an assignment or the number of days until the deadline when learners started working on an assignment. However, this approach does not account for seasonality of one’s learning. To address this limitation, a data-driven approach by analyzing temporal repetition in

behaviors via Poisson mixture models have instead been proposed [149].

Even when learners are expected to have mastered the knowledge, they can provide wrong responses to questions due to carelessness. This can happen due to several reasons. Learners may be in a hurry, being over-confident about completing a task, doing routine tasks, or performing tasks that are deemed as lower importance. This is similar to the slip variable in Bayesian Knowledge Tracing, albeit individualized to the learner rather than a global estimate. Since most studies investigate various factors and learning behavior in relation with learning outcomes, these studies would benefit from accurate measurements of mastery that account for carelessness. Although carelessness may be investigated by analyzing future responses given current responses, logistic regression models have been employed to model carelessness using time-based and frequency-based aggregated features from interaction logs [150].

Finally, stop-out is defined as stopping of any further learning activity which can result from either prolonged struggle to progress further or disengagement from the learning activity. Successful models of stop-out have been trained using aggregated features distilled from learning activities via gradient-boosted decision trees [151].

2.5 Chapter summary

In this chapter, the objectives within learning analytics and educational data mining have been discussed. Stakeholders and their objectives as well as the data available to support learning analytics is then presented. Subsequently, existing studies that examined the identification of learning styles, video-watching behavior, problem-solving approaches, and off-task behaviors are reviewed. Most studies that have focused on the validation of existing theories did not explore the possibility of new learning behaviors that arose from the paradigm shift; the reverse is also true. The lack of either component will hinder the pursuit of knowledge about learning. These challenges will serve as a motivation for the proposed frameworks in the following chapters.

Chapter 3

Identifying Lesson Preparation Preferences via Clustering

In Chapter 2, the identification of learner groups with similar learning behaviors was reviewed. These behaviors were analyzed in relation to the existing learning style frameworks. In this chapter, general lesson preparation preferences within a blended course is investigated. To identify these preferences exhibited by different group of learners, a clustering framework that differentiates learners' lesson preparation, is proposed. The clustering algorithm employs a Gaussian mixture model (GMM) to differentiate learners based on their use of different digital media. A set of heuristics then labels these lesson preparation in terms of their commonality and effort. Finally, the chapter concludes by performing a likelihood analysis to study the impact of lesson preparation preferences on learning outcomes.

3.1 Introduction

Conventional education systems often boast curriculum structures designed by expert teachers that aim to provide pedagogical support to allow learners to succeed. As they

Part of this chapter has been published as Kelvin H. R. Ng, K. Hartman, N. Goodkin, Andy W. H. Khong, "Mining for preparatory processes of transfer learning in a blended course," in *Proc. Am. Geophys. Union Fall Meet., AGU Fall Meeting*, New Orleans, LA, USA, Dec 11 - 15 2017.

are refined over multiple course runs, these curricula are slowly adapted to support most learners in their learning. Since the curriculum structure in most courses are relatively rigid, learners who are ahead of their peers may often find themselves having to wait for new learning materials to be released while the latter are put at risk of dropping out if no additional support is available for them. With many institution adopting blended learning approaches and moving their courses online, the onus is now on the learners to strategize and regulate their lesson preparations. Despite the rigid curricula, learners are driven by their cognitive and learning preferences, and regulated by their non-cognitive traits such as motivation and self-regulation. Consequently, they may still undertake different learning experiences to achieve their goals.

Identification of general learning preferences in online courses is also useful for instructors and course designers. While most end-users are expected to undertake the recommended learning pathway, some users may deviate away from this recommendation, albeit not all users who deviate achieve desirable outcomes. In the case of intelligent tutoring systems (ITSs), these type of behaviors are sometimes categorized as off-task behaviors. Instructors are, however, oblivious to these off-task behaviors in the context of online learning and would not be able to administer interventions to curb the off-task behaviors early. As most prior studies that analyses learning behaviors require multiple course runs to study the relationship of learning behaviors to learning outcomes, they are usually not suitable to be adapted onto new courses. In this study, a course-agnostic clustering framework is proposed to identify general learning preferences via the usage of learning materials.

3.2 Dataset

The framework is developed and validated on an interdisciplinary science course conducted in Nanyang Technological University, Singapore. This course enrolled 585 third- and fourth-year learners as a requirement by their undergraduate program. The fourteen-week course was developed as two self-contained halves with minimal content overlap between them. The first half of the course focused on the inter-relationships between

Figure 3.1: Average online activities for different categories of learning resources across the teaching weeks.

Earth’s physical systems and the impacts humans have had on these systems whereas the latter focused on solutions that have been implemented to address sustainability issues. Between these two halves, learners received a week-long recess period where instructors did not introduce any new course material or assignments. These contents were taught using activities such as weekly in-class lecture, online assessments, projects and mid-term exams. The overall course structure is shown in Figure 3.1.

Prior to attending the weekly lectures, learners were asked to prepare for their class using the provided online reading materials. Accesses to these reading materials were logged in the database. For eight of these weeks, online assessments were also administered to gauge the comprehension of each learner and to gather feedback regarding any doubts on the topics. Learners were only expected to complete at least 75% of the online

assessments, i.e., at least three assessments from each half of the course. While these online assessments were not graded, their responses were reviewed by the instructors and incorporated into the content of the weekly lectures. During face-to-face lectures, instructors would post multiple-choice questions regularly and polled the learners for their responses. These responses were logged for the evaluation of their lecture participation. Lectures were recorded and made available through an LMS within a few hours after its completion. The recorded lectures remained available for learners to review throughout the semester.

Beside weekly assessments and lectures, learners had to complete three projects throughout the semester. These projects include an individual essay, a group essay, and an online group discussion. For two of the group projects, time that was normally dedicated to face-to-face lectures were re-allocated for learners to work on the assignments with their groups. An examination was administered during the scheduled lecture in the seventh and fourteenth weeks. Each examination only covered materials that were taught in their respective independent halves. The examination questions were also differentiated in terms of three levels of cognitive reasoning—recall, application, and transfer [152]. Recall-type questions referred to materials that were explicitly mentioned in the course lectures or readings, whereas application-type questions asked learners to solve problems by following closely the steps that were demonstrated by the lecturer in class. On the other hand, transfer-type questions required learners to use the information they have obtained within the course to solve novel problems that they had not been explicitly prepared for. For most educational contexts, while performing well for recall- and application-type questions is important, the true level of understanding is mainly evaluated by their ability to transfer knowledge between different domains.

3.2.1 Data Pre-Processing

Each log entry within the database describes the type of actions performed and user information. To determine the weekly lesson preparation preferences, the material access logs for each learner are first segregated into four usage categories—*relevant*, *review*, *advance* and *irrelevant*—based on their relevance to the current teaching week. For

instance, during the third week, learners are expected to read the provided materials, complete an online assessment, and attend the face-to-face lecture. Materials that are released for the third week were considered as *relevant*, whereas those that were released in the previous weeks are regarded as *review*. Naturally, at the onset of the fourth week, these resources that were released in the third week would be labeled as *review*. Occasionally, materials were released ahead of schedule and accessing these materials before they become *relevant* would be regarded as an *advance* access. Materials that were used for the project works were one such instance of the change in relevance. Once an examination has completed, any resources that were evaluated in the examination would thereafter be labeled as *irrelevant* since subsequent exams do not cover these materials. The frequency of material accesses under each category is then computed and used to determine the amount of effort put in by the learners in each teaching week.

3.3 Proposed Clustering Framework

To identify weekly learning preparation preferences, a clustering framework based on the GMM clustering algorithm is applied onto the computed access frequencies. An overview of the proposed clustering framework is depicted in Fig. 3.2. GMM is chosen for several of its attractive properties. As a soft clustering approach, GMM is able to assign cluster membership to multiple components and identify overlapping cluster components. GMM also employs a small set of parameters to represent the cluster that can account for cluster covariances. This is advantageous since the total number of lesson preparation preferences over all teaching weeks can be high. Furthermore, these parameters can be estimated efficiently via the expectation-maximization (EM) algorithm. GMM, however, does assume that the sub-populations are also normally distributed. While it works relatively well for real-world applications, if the assumption is violated, this may result in poor performance. In cases where sub-populations are known to not be normally distributed but follow other known distribution, a generic mixture model that employs other fundamental distributions can be used in place of GMM. Each sub-population is parameterized by mean vector $\boldsymbol{\mu}_k$ and covariance matrix $\boldsymbol{\Sigma}_k$. The probability of data point $\mathbf{x}_i$ belonging to the mixture of Gaussian distribution can, therefore, be expressed

Figure 3.2: Overview of the proposed clustering framework for identifying learning preparation preferences.

as

$$P(\mathbf{x}_i) = \sum_{k=1}^N \phi_k \mathcal{N}(\mathbf{x}_i | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k), \quad (3.1)$$

where ϕ_k denotes the mixture weight for the k th component, $\mathcal{N}(\mathbf{x}_i | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$ the conditional probability of $\mathbf{x}_i$ given a Gaussian distribution that is parameterized by $\boldsymbol{\mu}_k$, and $\boldsymbol{\Sigma}_k$ and N is the total number of instances in the dataset. These parameters are optimized to maximize the likelihood of the dataset via the expectation-maximization (EM) algorithm. This is performed by alternating between two computation steps. The algorithm starts with the expectation step that computes the assignment of data point $\mathbf{x}_i$ to the clusters. The probability of assigning the i th data point to the k th cluster is mathematically expressed as

$$\gamma(i, k) = \frac{\phi_k \mathcal{N}(\mathbf{x}_i | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)}{\sum_j \phi_j \mathcal{N}(\mathbf{x}_i | \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)}, \quad (3.2)$$

The maximization step then updates the parameters such that the likelihood of the data given these parameters is maximized, that is given by

$$\mu_k = \frac{\sum_{i=1}^N \gamma(i, k) \mathbf{x}_i}{\sum_{i=1}^N \gamma(i, k)}, \quad (3.3)$$

$$\Sigma_k = \frac{\sum_{i=1}^N \gamma(i, k) (\mathbf{x}_i - \boldsymbol{\mu}_k)^\top (\mathbf{x}_i - \boldsymbol{\mu}_k)}{\sum_{i=1}^N \gamma(i, k)}, \quad (3.4)$$

$$\phi_k = \frac{1}{N} \sum_{i=1}^N \gamma(i, k). \quad (3.5)$$

To strike a balance between specificity and sensitivity, four to six clusters are identified for the weekly lesson preparation. The number of clusters is determined by the Bayesian information criterion (BIC) scores. The BIC scores for different number of clusters identified during a regular teaching week is depicted in Fig. 3.3. The optimal number of clusters is determined by a decrease in the rate of change in BIC score when the number of clusters is increased, e.g., 4 clusters in Fig. 3.3. In addition, an additional group is added for learners who did not show any in-class or online activity related to the course during the week. This also ensures that there are sufficient members in each clusters for

Figure 3.3: BIC scores for clusters identified during a regular teaching week.

the subsequent analysis.

A set of heuristic enforced via a rule-based system is then employed to describe each cluster in terms of its behavioral normalcy and the amount of effort that learners belonging to the clusters had demonstrated. Behavioral normalcy was described as a function of the cluster membership size. Clusters with at least 30% of the learners were labeled as *common* behaviors. However, when no cluster comprised at least 30% of the learners, clusters with membership sizes that were within 5% from the largest cluster were labeled as *common* behaviors. On the other hand, clusters that contained fewer than 5% of the learners were labeled as *rare* behaviors. The remaining clusters were labeled as *uncommon* behaviors.

Preparation effort was defined as a relative measure of the clusters that demonstrated common behaviors. To determine if a non-common cluster demonstrates similar, more or less effort as compared to the common behavior, the mean and standard deviation of the total effort for the common behavior clusters are computed. Thereafter, non-common clusters with a mean effort that is more than one standard deviation from that

Table 3.1: Composition of learning preparation preferences obtained using the GMM clustering algorithm for each activity category.

Period Category	Offline	Normal	Uncommon (Alternative)	Uncom- mon (Effort)	Rare (Alternative)
Teaching weeks 1	✓	✓	✓	✓	✓
No class	✓	✓	✓		
Examination 1	✓	✓	✓		
Recess	✓		✓	✓	✓
Teaching weeks 2	✓	✓	✓	✓	✓
E-learning	✓	✓	✓	✓	✓
Examination 2	✓	✓	✓		✓

of common clusters in the positive direction were labeled as being *effortful* while clusters whose mean effort was more than one standard deviation in the negative direction were labeled as showing *reduced effort*. Non-common clusters with a mean effort similar that were within one standard deviation from the common clusters were instead labeled as *alternatives*.

3.4 Results and Discussion

3.4.1 Lesson Preparation Preferences

The average accesses across the teaching weeks for different categories of learning resources are depicted in Fig. 3.1. As can be seen from the figure, the amount of lesson preparation that happened within each week varied throughout the course, with increased online activities prior to the two exams as learners ramp up their revision efforts. Online activities prior to the second examination were significantly higher since this was synchronized with the examinations of other courses that learners had subscribed to. As learners were accessing the materials for different purposes, it is, therefore, expected that lesson preparation preferences would also vary across the teaching weeks.

To analyze variations in lesson preparation preferences across the teaching weeks, the

teaching weeks were first categorized into broader activity categories of regular teaching weeks, examination revision weeks, the recess week and the e-learning week, as denoted in Fig. 3.1. Lesson preparation preferences for each activity category were identified by the GMM clustering algorithm using the proposed clustering framework and tabulated in Table 3.1. Overall, five different lesson preparation preferences were identified throughout the semester. Besides common behavior(s) demonstrated by the majority of the learner population, learners were also observed to engage in lesson preparations that are considered either uncommon alternatives, uncommon approaches with increased effort, rare alternatives and offline preparations. Lesser varieties were discovered during the examination revision weeks, recess week and the week when no class was held. For instance, during the first examination preparation week, learners only engaged in three of the five different preferred lesson preparations. This is expected since learners have yet undergone any summative assessment in this course. Consequently, they do not have any prior knowledge of the conduct of the examination and are, therefore, unable to strategize their learning. Most learners reverted to revising offline, which may indicate that they were relying on hard-copy materials that had been amassed throughout the semester thus far. Unlike the first examination, since learners had a better understanding of their expectations for the examinations, an additional rare alternative cluster surfaced in the second examination preparation week, even though the two examinations were structured similarly. Since the average online activities had increased during the second examination preparation week, clusters with higher average effort became the norm and no cluster stood out from the rest in terms of effort exerted. Even though no classes were conducted during the fifth week, lesson preparation preferences were similar to that of the first examination preparation week. It is believed learners had already started their examination revision during the fifth week due to their proximity. While most learners did not participate in any online learning activity during the recess week, surprisingly, some learning activities were observed despite existing materials being irrelevant for learners going forward and no new materials had yet been released.

3.4.2 Likelihood Analysis

To evaluate the relationship between lesson preparation preferences and learning outcomes, the likelihood ratio of learners obtaining an outcome after engaging in the different preferred lesson preparations compared to learners who has an uniform chance of obtaining any of the outcomes is analyzed. Given a learner who had engaged in his preferred lesson preparation $\mathcal{R}_i$ and obtained $\mathcal{Q}_j$, the likelihood ratio is mathematically expressed as

$$\mathcal{L}(\mathcal{R}_i, \mathcal{Q}_j) = \frac{P(\mathcal{R}_i, \mathcal{Q}_j)}{P(\mathcal{R}_i)}, \quad (3.6)$$

where $P(\mathcal{R}_i)$ and $P(\mathcal{R}_i, \mathcal{Q}_j)$ denote the probability of learners performing $\mathcal{R}_j$ and the joint probability of learners who performed $\mathcal{R}_i$ and obtained an outcome that belongs to grade-quintile $\mathcal{Q}_j$, respectively, and is given by

$$P(\mathcal{R}_i) = \frac{\sum_j C(\mathcal{R}_i, \mathcal{Q}_j)}{\sum_{i,j} C(\mathcal{R}_i, \mathcal{Q}_j)}, \quad (3.7)$$

and

$$P(\mathcal{R}_i, \mathcal{Q}_j) = \frac{C(\mathcal{R}_i, \mathcal{Q}_j)}{\sum_i C(\mathcal{R}_i, \mathcal{Q}_j)}, \quad (3.8)$$

where $C(\mathcal{R}_i, \mathcal{Q}_j)$ denotes the number of learners who performed $\mathcal{R}_i$ and obtained $\mathcal{Q}_j$. A likelihood ratio above 1 would, therefore, indicate that a learner who had engaged in a preferred lesson preparation is more likely to obtain a grade outcome that belongs to a grade-quintile compared to learners who had engaged in any type of lesson preparation. The converse applies to likelihood ratios below 1. For instance, if 75% of learners were found to be engaged in the *common* lesson preparation for a given week but only 50% of learners in first grade-quintile came from this group of learners, then the likelihood of a learner who engaged in *common* lesson preparation and achieving the first grade-quintile is 0.67.

The likelihood ratio of learners obtaining different grade-quintiles in the first examination based on their lesson preparation preferences throughout the course except the recess

Figure 3.4: Likelihood ratio of learners obtaining grade-quintiles after undertaking various lesson preparation preferences for the first examination.

week is plotted in Fig. 3.4. The five grade-quintiles, labeled from low to high in ascending order, are represented using warm to cool shades. Each bar plot, therefore, represents the likelihood of learners who had performed the preferred lesson preparation to obtain learning outcomes that belong to a grade-quintile. As can be seen, there is no significant impact for the first half of the course (Teaching Week 1, No Class, Exam 1). However, learners who had engaged in rare alternative lesson preparation approaches in the second half of the course (Teaching week 2, E-learning, and Examination 2) were more likely to get a lower grade-quintile. For instance, learners who engaged in rare alternative learning preparation approaches in the second teaching week achieved a likelihood ratio of 1.85 for the lowest grade-quintile while achieving only 0.49 for the highest quintile. On the other hand, Fig. 3.5 that depicts the likelihood ratio of obtaining different grade-quintiles based on learners' lesson preparation during the examination weeks shows that there is a significant impact of lesson preparation preferences on grade outcomes for the two examinations. Learners who had engaged in any online learning activity during the recess week are better off than their peers who had not done so. In fact, these learners were more than twice as likely to find themselves in the first quintile for the first examination than would be expected based on the representation of lesson preparation preferences within the whole cohort. Although the recess week only happened after the first examination, this is still unexpected since there are no materials that were relevant to the upcoming teaching weeks. Furthermore, this association holds for the second examination, albeit weaker.

These results suggest that learners who show a higher ability to transfer their knowledge tend to employ different lesson preparation preferences. In this case, learners who spent their recess period accessing course materials performed more accurately on novel problems than learners who used the common method of retreating from their studies during breaks in the course. Citing Louis Pasteur's "Fortune favors the prepared mind" [153], it is believed that these learners are more adept at incorporating the material into their existing cognitive maps and can better retrieve it in flexible ways when necessary. This fits with the idea that persistence, returning to a line of thinking until all options are exhausted, can be observed through the use of educational data mining algorithms. If persistence is associated with regular intervals of review and reflection, then

Figure 3.5: Likelihood ratio of lesson preparation preferences during the recess week on grades obtained for the first examination (top) and second examination (bottom).

it is not surprising that learners who review materials when the majority do not, would also perform better on measures of creativity. It is possible that they simulate possible transfer questions when they are reviewing resources such that when those questions appear in an examination in the future, they are prepared to answer them. Interestingly, the preparatory process used during the recess week was indicative of performance on both exams, but preparation for the weekly lectures and the exams was not.

There are two possible explanations that will need further investigation. First, it is possible that learners who prepare during lulls in course activity possess higher motivation for being self-directed learners. Therefore, they engage the material more deeply and prepare themselves in a more flexible manner. It is also possible that learners who prepare during these lulls tend to exhaust all of their options when given extra time to prepare. In other words, they might be preparing with an eye for transferring or they might be going through every imaginable combination of the material so that they can fluidly access the knowledge when the examination requires it. While providing supportive evidence is beyond the scope of this thesis, it is believed that the second explanation is much less likely than the first. The nature of the transfer questions was not a direct extension of the lecture topics and course readings. They required learners to take those concepts,

deconstruct them, and reassemble them in new ways to solve novel problems and learners could not have been able to prepare for these questions directly.

3.5 Chapter summary

In this chapter, general lesson preparation preferences was investigated and discussed. To ensure that frameworks are transferable to other courses or course runs of the same course, a data-driven clustering framework was proposed for the identification of different lesson preparation preferences is proposed. The proposed clustering framework employs a GMM to identify learner sub-populations who had undertaken similar lesson preparation behaviors. These learning preparation preferences are then described in terms of their effort and creativity via a set of heuristics. Five different types of learning preparation preferences were identified across the teaching weeks. To understand the impact of creativity on learning, a likelihood analysis is performed to quantify the probability that learners who had undertaken their preferred learning preparation are able to achieve better learning outcomes. For teaching weeks where new content were made available, there is mixed effect on the use of different lesson preparation on learning outcomes. Surprisingly, the analysis revealed that learners who demonstrated any hint of lesson preparation activities when the course does not demand it were different from their peers and this difference was correlated to better learning outcomes.

Chapter 4

Problem-Solving Action Sequence Archetypes and the Relationship to Learner Drop-out

In Chapter 3, a clustering framework was proposed to identify lesson preparation preferences of learners within virtual learning environments (VLEs). Similar to most existing studies, the framework relies on aggregated frequency-based features to analyze the data. As the understanding of nuances in the learning process is critical to discern good studying habits, this aggregation process would obscure the otherwise critical information. With recent development in data storage and modeling capability, large-scale collection of fine-grain clickstream data is made possible. A common theme among existing studies that analyzed clickstream problem-solving approaches is the employment of clustering algorithms to identify different problem-solving approaches. While these approaches can uncover interesting learning behaviors, nuances within these behaviors are filtered through the process, rendering it challenging to analyze these behaviors in relation to the underlying pedagogies employed within these VLEs. On the other hand, sequential pattern mining (SPM) approaches that were employed to identify frequently occurring

Part of this chapter has been published as Kelvin H. R. Ng, K. Hartman, K. Liu, and Andy W. H. Khong, “Modelling the way: Using action sequence archetypes to differentiate learning pathways from learning outcomes,” in *Proc. 9th Int. Conf. Educ. Data Min., EDM*, 2016, Raleigh, NC, USA, Jun 29 - Jul 2 2016, pp. 167174.

sub-sequences retain the subtleties but often generate large sets of sequential patterns that are tedious to analyze manually. In this chapter, a sequential clustering framework that balances between identifying short bursts of repetitive sequential patterns and aggregating these motifs into bigger labels is proposed. To differentiate clickstream sequences, four similarity measures that capture various aspects of the sequences are employed to differentiate pairs of clickstream sequences. Independently, these similarity measures are not sufficient to differentiate the sequences. The similarity matrix derived from aggregating these measures is then employed to identify different problem-solving approaches of learners in an elementary grade mathematics online module. The identified problem-solving approaches are then analyzed in conjunction with the scripted pedagogy. Finally, the chapter concludes by discussing the impact of these problem-solving approaches on learners' persistence in their learning.

4.1 Introduction

As discussed in Chapter 2, intelligent tutoring systems (ITSs) were designed with the sole purpose of tracking and diagnosis of problem-solving approaches in order to administer interventions at the earliest opportunity to ensure their success. Staying true to its mission, topics that were taught via ITSs are often methodological and the general approach for solving its problems can often be deconstructed into finite steps. As such, ITSs have been largely deployed for science, technology, engineering, and mathematics (STEM) education. Similar to these topics, Singapore's elementary mathematics education for solving arithmetic word problems is taught via the model method [17]. This approach is based on Polya's problem-solving technique [154], where the complete process is deconstructed into five fundamental steps that are collectively known as the *RIGHT* sequence. During its application, learners are expected to

1. **R**ead the word problem,
2. **I**dentify the nouns, numerical values and unknown variables,
3. **G**raph these values using a box diagram,

4. (**H**ave it done) by performing necessary reasoning and calculations, and
5. (**T**riple check) to review their work.

This was proposed as a learning mnemonic and served as a script for learners to execute the model method.

As a cognitive construct, scripts are collections of discrete actions that enable its users to achieve a goal when followed [16]. A classic example of a cognitive script is one that is used to order food at a restaurant [16]. Following the same set of steps in most dining establishments, albeit some allowance for minor deviations, will lead the patron to receive a meal. These sets of steps have been found to reduce cognitive load for novice learners because it does not require them to plan prior to the execution while experts may experience greater expressions of automaticity [155]. Scripts, however, can often be considered contextually bound and rather inflexible [156] and one that works in a specific context, such as purchasing food in dining establishments, may not lend itself to achieve benefits in other contexts.

On the other hand, by consolidating these steps into generic and abstract knowledge [157], generalizable schemas serve several cognitive functions such as the integration of information, regulation of attention and aid inferences [158]. While they sacrifice in terms of automaticity, generalizable schemas make up with flexibility [158]. Furthermore, without a schema, learning can be slow and uncertain [158]. Although some schemas are innate, most schemas are acquired through learning. Returning to the above-mentioned example, rather than obtaining meals in a restaurant, the generalizable schema for acquiring food would include all known methods of gaining nourishment. Similarly, while following the *RIGHT* sequence will lead learners to arrive at the correct answer for the word problem, the goal of imparting the *RIGHT* sequence is to facilitate learners to obtain their critical thinking and problem-solving schemas to tackle other novel problems [17]. In practice, learners often instantiate the schema in the form of a word problem-solving script [159]. However, both strategies would work in the short term and it is impossible for instructors to differentiate if learners had internalized the model method as a schema or simply utilizing the script from their problem-solving accuracies.

Figure 4.1: The structured activity of the online elementary mathematics module.

4.2 Experiment Design

To understand how learners employ the *RIGHT* method to solve these arithmetic word problems, an online elementary mathematics course is designed to emulate the pen-and-paper approach that was taught to the second-grade learners in a local primary school. The course is divided into two halves; the former (Phase 1) covers addition and subtraction while the latter (Phase 2) multiplication and division. Learners had received formal instructions on addition and subtraction in a physical class, but had yet to encounter multiplication and division type questions. Four types of activities were present for each of the two phases. They include video content, structured activities, unstructured activities and multiple-choice questions (MCQ). The video content reiterated the *RIGHT* sequence and the use of the model method from a pen-and-paper standpoint. At the start of the structured and unstructured activity, additional videos were also provided to guide learners through various features that were designed to scaffold their use of the model method to solve the arithmetic word problem.

The structured activity focused on the selection of appropriate bar diagrams for the word problem, and hence the first three steps of the *RIGHT* method. An example of the structured activity is shown in Fig. 4.1. The structured activity took the form of multiple-choice questions with four different bar diagrams as its selections. To scaffold the *RIGHT* method, a checklist consisting the first three steps of the *RIGHT* method was provided on the right column of the platform. Although the activity did not enforce

Figure 4.2: The unstructured activity of the online elementary mathematics module.

that learners follow these three steps, learners were advised to highlight the nouns within the word problem and tick off the respective check boxes as they complete each step in the *RIGHT* sequence. Once they had identified appropriate bar diagrams that fit the word problem, learners would be given feedback on their choices as well as options to review, ask for hints if they had chosen the wrong diagram or move on to the next question. Reviewing a question will revert the environment to the last instance prior to their submission. Hints, on the other hand, are partially completed solutions that can be given in succession up to a maximum of two hints. At this point, the learners would require minimal effort to complete the question. If a learner chose to proceed without reviewing any mistake, the system would note that the learner had ignored an error. Having this feature allows the algorithm to differentiate learners who are trying to master the approach as opposed to those who are gaming the system.

For the unstructured activity, learners were given full control over all the available features to solve the problems using the *RIGHT* sequence. An example of the unstructured activity is shown in Fig. 4.2. Instead of having four bar diagrams to choose from, learners must now decide from the four arithmetic operations before selecting a bar diagram from a variety of diagrams that can be used for the selected arithmetic operations. On top of that, nouns within the word problem are now available as interactive blocks for learners

to drag onto the bar diagram. Text boxes were also provided for learners to complete the mathematical expressions. The checklist for the unstructured activity comprises the five steps of the *RIGHT* method. Although the activity expects learners to perform these steps, once learners have internalized the *RIGHT* method, these sub-processes may be performed implicitly. Therefore, the platform allows learners to forego performing any or all of these actions. However, they must submit a final answer before they can receive feedback and proceed to the next question.

For the MCQ activity, learners were posed with ten multiple-choice questions, which presented four possible numerical answers to the word problem. Beyond highlighting of nouns and a floating checklist, no other scaffolding (i.e., model and equation templates) is provided within this activity. While the checklist is available to the learners, whenever the learner interacts with the options for a question, the checklist would reset in lieu of the next question. Learners must complete the MCQ activity for addition and subtraction before they can proceed to Phase 2.

4.3 Dataset

Learners' interactions with the scaffolding and their navigation around the online module were logged. Besides interactions within the module, indications of off-task behavior such as learners activating another browser tab and returning to the online module were also collected. Pre-processing was performed on the logs to reduce the complexity of these clickstream sequences and to make them more schematic. For example, instead of considering highlighting of individual keyword, only the first instances of highlighting and canceling of highlights for each question were retained to signal that the highlighting resource was used. In addition, when tracking learners as they navigate the model template selection interface, only their final template selection was analyzed. To differentiate answer revisions, multiple-choice selections that do not constitute the first selections were labeled as edits. After pre-processing, a total of 9918 logs were collected from 36 second-grade learners who had completed the first phase of activities. The final list of actions for each type of activity is shown in Table 4.1.

Table 4.1: List of logged actions for the elementary Mathematics online course.

Action	Video	Structured	Unstructured	MCQ
Leave page				
Return to page	✓	✓	✓	✓
Start activity				
Stop activity				
Start video				
Stop video				
Pause video				
Scrub video (forward)				
Scrub video (backward)	✓			
End video				
End (and complete) video				
Replay video				
Select same video				
Select different video				
Attempt question				
Highlight word				
Undo highlight word				
Check checklist		✓	✓	✓
Submit answer ^a				
Review error				
Ignore error				
Request for hint				
Select MCQ option ^a		✓		✓
Amend MCQ option ^a				
Confirm model template ^a				
Drag model component ^a			✓	
Label model component ^a				
Label equation component ^a				

^a The correctness of these action is recorded but it has been removed for the purpose of this analysis.

4.4 Proposed Action Sequence Archetype Mining Algorithm

To identify different problem-solving approaches in relation to the *RIGHT* method, the proposed action sequence archetype mining algorithm employs a clustering algorithm to first identify similar sequences of actions performed by learners before summarizing each cluster via an SPM algorithm, as depicted in Fig. 4.3. As the relationship between clickstream sequences and learning outcomes are not well-understood, it is important

Figure 4.3: Overview of the proposed action sequence archetype mining algorithm.

to incorporate human judgment as part of the mining process. The objective of the proposed method is, therefore, to incorporate human judgment in the identification of problem-solving approaches while reducing the cognitive load on these human annotators. The clustering module employs four pairwise similarity measures to differentiate input sequences. These measures capture different aspects of sequences and include the Jaccard distance, difference in lengths and action rarity and an adapted variant of the common word order (CWO) measure [160].

The Jaccard distance, as the complement of the Jaccard similarity, captures the degree of dissimilarity between the term compositions of the two sequences. Given two arbitrarily defined input sequences $\mathbf{x}_i$ and $\mathbf{x}_j$, the Jaccard distance is mathematically expressed as

$$\mathbb{D}_{\text{Jaccard}}(\mathbf{x}_i, \mathbf{x}_j) = 1 - \frac{|\mathbf{x}_i \cap \mathbf{x}_j|}{|\mathbf{x}_i \cup \mathbf{x}_j|}, \quad (4.1)$$

where $|\mathbf{x}_i \cap \mathbf{x}_j|$ and $|\mathbf{x}_i \cup \mathbf{x}_j|$ denote the number of unique terms between both sequences and the total number of unique terms across both sequences, respectively. The Jaccard distance is bounded between zero and one, where two input sequences without any common terms between them would achieve $\mathbb{D}_{\text{Jaccard}}(\mathbf{x}_i, \mathbf{x}_j) = 1$. Prior studies have found that Jaccard distance can discover interesting insights when used as a metric to evaluate the interestingness of association rules between disengagement and affect [161]. Although the analysis did not highlight reasons for the finding, the fact that the antecedents comprise of items that are different from its consequent (as measured by Jaccard distance) may be the reason behind why these rules are more interesting to the annotators. In a similar manner, the use of Jaccard distance to differentiate pairs of input sequences can achieve similar effects. That is, clusters identified when Jaccard distance is used as a measure of similarity can potentially identify clusters that are more interesting.

The CWO measure [160] was originally proposed to quantify the alignment of the common actions between two input sequences. Conceptually, CWO identifies the set of common terms between two input sequences. Defining L_x as the length of the longer sequence after the actions are filtered down to the common actions, CWO measures the difference in index positions between the sequences for the same term that is mathemat-

Figure 4.4: Pre-processing process for the modified common word order measure.

ically given by

$$\mathbb{D}_{MCWO}(\mathbf{x}_i, \mathbf{x}_j) = \begin{cases} 1 - \frac{2 \sum_{l=1}^{L_x} |x_i^k - x_j^k|}{L_x^2} & \text{if } L_x \text{ is even;} \\ 1 - \frac{2 \sum_{l=1}^{L_x} |x_i^k - x_j^k|}{L_x^2 - 1} & \text{if } L_x \text{ is odd;} \\ 1 & \text{if } L_x = 1. \end{cases} \quad (4.2)$$

As it was originally proposed for text applications, it was assumed that the number of occurrences for each term is similar and low between the two sequences. This is, however, not true for sequences that describe online learning activities since they tend to comprise a small collection of actions that are repeated regularly within the sequence. As a consequence, the common word order measure between two online learning activity sequences is not bounded when applied as is.

To account for this difference, the common word order is adapted with an additional pre-processing phase, as illustrated in Fig. 4.4. Similar to the CWO measure, common terms between two input sequences $\mathbf{x}_i$ and $\mathbf{x}_j$ are first extracted from each sequence. Due to differences in the number of occurrences for each term between the two sequences, this results in the truncated sequences having different lengths. To ensure that these two sequences are most closely aligned, a local sequence alignment algorithm, e.g., the Smith-Waterman [162], is employed to identify a pair of reference indices that best align the two sequences. This reduces the differences in positional indices of actions that do not occur at the same indices. Once the two sequences are aligned, null tokens are appended

Algorithm 1: The modified common word order algorithm

Input: Input sequences by $\mathbf{x}_i = [x_i^1, x_i^2, \dots, x_i^{L_i}]$ and $\mathbf{x}_j = [x_j^1, x_j^2, \dots, x_j^{L_j}]$

Output: Modified common word order score $\mathbb{D}_{\text{MCWO}}$

$\bar{\mathbf{x}}_i, \bar{\mathbf{x}}_j \leftarrow \text{ExtractCommonTerm}(\mathbf{x}_i, \mathbf{x}_j)$

$\text{offset}_i, \text{offset}_j \leftarrow \text{Align}(\bar{\mathbf{x}}_i, \bar{\mathbf{x}}_j)$

$\bar{\mathbf{x}}_i \leftarrow \text{Pad}(\bar{\mathbf{x}}_i, \text{offset}_i)$

$\bar{\mathbf{x}}_j \leftarrow \text{Pad}(\bar{\mathbf{x}}_j, \text{offset}_j)$ // $\bar{\mathbf{x}}_i, \bar{\mathbf{x}}_j \in \mathbb{R}^{L_k}$

$$\mathbb{D}_{\text{MCWO}}(\bar{\mathbf{x}}_i, \bar{\mathbf{x}}_j) \leftarrow \begin{cases} 1 - \frac{2 \sum_{i=1}^{L_k} |x_i - x_j|}{L_k^2} & \text{if } L_k \text{ is even;} \\ 1 - \frac{2 \sum_{i=1}^{L_k} |x_i - x_j|}{L_k^2 - 1} & \text{if } L_k \text{ is odd;} \\ 1 & \text{if } L_k = 1. \end{cases}$$

to the shorter sequence such that the two sequences are equivalent in length. Finally, the CWO measure can be computed using (4.2). The modified common word order (MCWO) measure is summarized in Algorithm 1.

The action rarity difference measure differentiates trivial navigational sequential patterns from more complex activity patterns by quantifying differences between frequencies of actions that make up a pair of sequences. The rarity of an action is quantified with reference to all other actions that were performed by the learners. This is computed using the inverse document frequency (idf) score from the term-frequency inverse-document frequency (TF-IDF) [163] that is given by

$$\mathbb{D}_{\text{idf}}(x_i, \mathcal{D}) = \log \frac{|\mathcal{D}|}{|d \in \mathcal{D} : x_i^j \in d|}, \quad (4.3)$$

where $|d \in \mathcal{D} : x_i^j \in d|$ denotes the total number of sequences that the action x_i^j appeared in in the dataset $\mathcal{D}$ and $|\mathcal{D}|$ is the total number of sequences in the dataset. The action rarity difference between sequences $\mathbf{x}_i$ and $\mathbf{x}_j$, defined as the difference in rarity of the most uncommon term in $\mathbf{x}_i$ and $\mathbf{x}_j$, is, therefore, given by

$$\mathbb{D}_{\text{AR}}(\mathbf{x}_i, \mathbf{x}_j) = |\max\{\mathbb{D}_{\text{idf}}(x_i^k, \mathcal{D}) : x_i^k \in \mathbf{x}_i\} - \max\{\mathbb{D}_{\text{idf}}(x_j^k, \mathcal{D}) : x_j^k \in \mathbf{x}_j\}|. \quad (4.4)$$

Since the action rarities are non-negative, similarly, the action rarity difference measure is zero when two sequences are composed of actions with identical rarity, and non-zero

otherwise.

4.4.1 Proposed Similarity Measure

The proposed similarity measure aims to capture the similarity between a pair of sequences by considering different aspects as quantified by the four similarity measures. The four similarity measures are combined using a weighted sum mechanism and is given by

$$\begin{aligned} \mathbb{D}_{\text{Aggr}}(\mathbf{x}_i, \mathbf{x}_j) = & w_{\text{Jaccard}} \times \mathbb{D}_{\text{Jaccard}}(\mathbf{x}_i, \mathbf{x}_j) + w_{\text{L}} \times \mathbb{D}_{\text{L}}(\mathbf{x}_i, \mathbf{x}_j) \\ & + w_{\text{MCWO}} \times \mathbb{D}_{\text{MCWO}}(\mathbf{x}_i, \mathbf{x}_j) + w_{\text{AR}} \times \mathbb{D}_{\text{AR}}(\mathbf{x}_i, \mathbf{x}_j), \end{aligned} \quad (4.5)$$

where $\mathbb{D}_{\text{L}}(\mathbf{x}_i, \mathbf{x}_j) = |L_i - L_j|$ denotes the difference in sequence lengths between $\mathbf{x}_i$ and $\mathbf{x}_j$, and $\mathbf{w} = \{w_{\text{Jaccard}}, w_{\text{MCWO}}, w_{\text{AR}}, w_{\text{L}}\}$ denote the mixture weights for the four similarity measures. The weighting ratio is chosen to ensure that all four sequence measures contribute uniformly to the aggregated measure. That is, w_{Jaccard} and w_{MCWO} are set to 1 (since the corresponding measures are bounded between zero and one) while w_{AR} and w_{L} are set to the inverse of maximum of the measures, i.e., $w_{\text{AR}} = \frac{1}{\max \mathbb{D}_{\text{AR}}(\mathbf{x}_i, \mathbf{x}_j)}$, $w_{\text{L}} = \frac{1}{\max \mathbb{D}_{\text{L}}(\mathbf{x}_i, \mathbf{x}_j)}$. This is because while all four measures have a lower bound of zero, action rarity and length differences have higher upper bounds $|x| > 1$ compared to their differences between the common word order and Jaccard distance, both having an upper bound of $|x| = 1$. As will be shown in Section 4.5.1, pairs of sequences that are different may be identified by some of these similarity measures but not the others. As such, when considered independently, sequences can appear to be similar when they are not. By incorporating various similarity measures, differences between the sequences are effectively captured. This enables the clustering algorithm to differentiate them.

4.4.2 Action Sequence Archetype Clustering

Different types of learning activity sequences are subsequently identified via the Affinity Propagation (AP) algorithm [164] applied on the proposed similarity measure. AP

is a non-parametric clustering algorithm that identifies clusters via a pairwise similarity matrix. Unlike centroid-based clustering such as the k-means algorithm, AP is able to determine an appropriate number of clusters automatically via its set of shared preferences. This is advantageous for applications where the number of clusters is unknown but large. Furthermore, AP is known to be less sensitive to parameter initialization since it relies on the fixed pairwise similarity matrix, therefore often outperforming other clustering algorithms [164]. In addition, the pairwise similarity matrix is not limited to symmetrical pairwise similarity matrices. This allows the use of non-symmetrical similarity measures, whenever applicable.

The pairwise similarity matrix Ξ comprises two components—the diagonal shared preferences $\{\xi_{ij}\}_{i=j}$ and the off-diagonal pairwise similarities $\{\xi_{ij}\}_{i \neq j}$. The contribution of the shared preferences to the clustering algorithm is twofold. First, the relative amplitude of the shared preference for a given input sequence determines its likelihood to serve as an exemplar for other inputs. An input sequence with higher shared preference is more likely to represent other input sequences as their exemplars. Secondly, all shared preferences are, collectively, used to determine the number of generated clusters. The number of clusters as well as the suitability of an input sequence to represent a set of inputs as its exemplar are determined via a message passing mechanism. Two types of messages—responsibility and availability—are passed between data points. The responsibility of input sequence $\mathbf{x}_i$ to input sequence $\mathbf{x}_j$, denoted by ρ_{ij} , represents the amount of evidence such that $\mathbf{x}_j$ is suitable to serve as the exemplar for $\mathbf{x}_i$. On the other hand, the availability of $\mathbf{x}_i$ for $\mathbf{x}_j$, α_{ij} , signifies its suitability to be chosen by $\mathbf{x}_j$ as its exemplars after considering all other candidates. When pairwise similarity information is incomplete, these messages are passed only between data points with known similarities. The messages are initialized to zeros and updated iteratively as follows:

$$\rho_{ij} \leftarrow \xi_{ij} - \max_{k \neq j} \{\alpha_{ik} + \xi_{ik}\}, \quad (4.6)$$

$$\alpha_{ij} \leftarrow \min\{0, \rho_{jj} + \sum_{k \neq i, j} \max\{0, \rho_{kj}\}\}, \quad (4.7)$$

$$\alpha_{jj} \leftarrow \sum_{k \neq j} \max\{0, \rho_{kj}\}. \quad (4.8)$$

To determine the cluster exemplars, the candidate exemplar for each input is determined using

$$\gamma(i, k) = \operatorname{argmax}_k \{\alpha_{ik} + \rho_{ik}\}. \quad (4.9)$$

Equation (4.9) implies that either $\mathbf{x}_i$ will serve as the exemplar for $\mathbf{x}_k$ or vice versa. The updates terminate when the number of clusters remains constant across the iterations. While the identified exemplars are representatives for their respective clusters, some variations between the exemplars and their cluster members are expected. To summarize these variations, a sequential pattern mining algorithm is employed to identify frequent sequential patterns that are common to members of a cluster.

4.4.3 Descriptive cluster labels via sequential pattern mining

To generate a descriptive label for each cluster, input sequences that belong to a cluster are summarized using the position-coded pre-order linked web access pattern (PLWAP) algorithm with multiple minimum support (PLMS) algorithm [165]. The PLMS algorithm is chosen for two main reasons. The tree-based sequential pattern mining algorithm is efficient in terms of its space and time complexity. Furthermore, the multiple minimum support constraint ensures that rare actions are not filtered out by the minimum support threshold. The functioning of the PLMS algorithm is detailed as follows.

The PLMS algorithm employs the PLWAP tree to encode the complete dataset, i.e., sequences that belong to the same cluster. Since sequences containing actions that do not satisfy the global minimum support threshold would not qualify as a frequent sequence, the frequency of occurrences for each action is first computed. Actions that satisfy the global minimum support threshold are referred to as frequent 1-sequences whereas those that do not satisfy the minimum support are discarded. The process of constructing the PLWAP tree begins by initializing the tree with a root node. Considering only actions that belong to the set of frequent 1-sequences, starting from the root node, each action within a sequence is appended to the node as its child node if the current node does not have one that corresponded to this action. These new nodes have a frequency

Table 4.2: Header table example for the illustrative example depicted in Fig. 4.5

Label	Support	Position Code
Start video	10	{01}
End (and complete) video	6	{011}, {0110001}
End video	4	{0110}, {011001}, {011000101}
Pause video	2	{011000}
Scrub video (backward)	1	{01100010}
Scrub video (forward)	1	{01100}

count of one. However, if the child node with the corresponding action label has already been created, its frequency count will be increased by one. The process is repeated by appending the next action to this child node until the complete sequence is traversed. The final PLWAP tree is, therefore, a concise representation of sequences that belong to a cluster. An illustrative example of the PLWAP tree is depicted in Fig. 4.5. To further ease the identification of prefixes and suffixes, each node in the tree is assigned a binary position code that emulates the Huffman coding scheme. This coding scheme uniquely identifies each node and its sub-tree. Labeling the root node as a digit ‘0’, the remaining tree is labeled via the following process. Given a parent node, its leftmost child is assigned a position code that has a digit ‘1’ appended to the back of that for the parent node. The position codes of other children are derived by appending a ‘0’ to the position codes of its nearest sibling to its left. From the position codes, the level of a node can be determined through the sum of individual indices whereas the sub-tree of a node comprises nodes that have the position code of its parent node as its prefixes. As the tree is being constructed, a header table containing a collection of nodes with unique labels, a list of their binary codes and its support is also maintained. The corresponding header table for the PLWAP tree in Fig. 4.5 is shown in Table 4.2.

Moreover, the PLMS algorithm differs from the PLWAP algorithm by having multiple minimum support (MMS) thresholds in addition to the single global minimum support (GMS) threshold. This is because GMS performs well when all events are equiprobable within the dataset. However, when some actions occur less frequently, these actions would be filtered by GMS during the tree construction phase. Therefore, these rare actions would not be represented in the set of frequent patterns. For MMS, a different minimum action support (MAS) is set for each action as determined by its native frequency within

Figure 4.5: An illustrative example of a PLWAP tree.

the dataset. This ensures that even though these actions occur less frequently, sequences that contain them can still qualify as frequent patterns. Consequently, the GMS is set to the minimum of the MASs. Each node in the PLMS tree has an additional minimum possible support (MPS) variable that corresponds to the lowest minimum support of actions in its suffixes.

The process of mining a PLMS tree begins by identifying frequent actions before longer sequential patterns are mined iteratively from these actions. To identify the frequent actions, entries in the header table (after it has been sorted in the order of decreasing frequencies) are followed to identify nodes that correspond to the same action. If the total frequency of an action satisfies its MAS, then it qualifies as a 1-sequence and is added to the list of frequent patterns. However, if the action satisfies the GMS but not its MAS, then the 1-sequence is kept as a pattern candidate. Otherwise, actions that do not meet the requirement of the GMS are discarded. Actions that precede these 1-sequences are identified by scanning the header table and matching its binary position code. Similarly,

frequencies of newly identified nodes are aggregated and used to determine if the longer sequences are frequent patterns. On top of that, if the new candidate has a lower MPS, the MPS for the new node is updated with the lower MPS. The process is repeated until the aggregated frequency of newly identified nodes is below the GMS.

Finally, the set of frequent sequential patterns is analyzed to determine the descriptive summary for each cluster. Furthermore, clusters with similar descriptive summary are aggregated into a greater label that is termed the action sequence archetypes (ASAs).

4.5 Results and discussion

4.5.1 Impact of various similarity measures to identify clusters

To analyze the quality of clusters generated via the proposed aggregated measure described in (4.5), the pairwise similarity matrices for individual similarity measures and the aggregated similarity measure are illustrated in Fig. 4.6. Each data point represents the similarity between a pair of inputs, where a darker shade indicates that this pair of inputs are similar. Since the objective of clustering algorithms is to minimize inter-cluster similarity and intra-cluster differences, cluster quality can be inferred visually via these heatmaps. An illustrative example of the clusters identified by the aggregated feature is illustrated in Fig. 4.7. Since well-defined clusters place inputs that are similar together, this naturally creates square neighborhoods along the diagonal that are dark in color. Data points that form these darker neighborhood are similar. Similarly, cluster quality can be inferred via the amount of noise within the square neighborhoods along the diagonal of the matrices. On the other hand, intra-cluster similarity are determined by the dissimilarity of data points between the clusters that are represented by the off-diagonal points. When fewer clusters are desired, two adjacent clusters can be combined as a single bigger cluster, vice versa. As can be seen from Fig. 4.6, heatmaps for the four similarity measures are different. The similarity between the Jaccard distance and length difference measure is the highest. In contrast to Jaccard distance and length differences, MCWO identifies one main cluster whereas there is no clear clusters from action rarity. However, for the clusters identified using Jaccard distance or length difference, action

Figure 4.6: Pairwise similarity of action sequences measured using individual similarity measure and the proposed aggregated similarity measure for the video activity.

Figure 4.7: Illustrative example of clusters identified via the proposed aggregated similarity measure for the video activity.

Table 4.3: Action sequence profile.

Activity	No. of Attempts	No. of Profiles
Video	89	10
Structured	286	11
Unstructured	303	11
MCQ	33	12

rarity and MCWO is able to further differentiate data points, leading to better-defined clusters. This can be seen from the clusters learnt by the aggregated feature. While differences between cluster are reduced, inter-cluster similarity is reduced. The ability to capture these nuances results in better cluster quality.

4.5.2 Identification of action sequence archetypes

The total number of ASAs identified from the video, structured, unstructured and MCQ activities when the shared preference is set to similarity quantile of 0.75 are tabulated in Table 4.3. This value of similarity quantile is selected to ensure that the generated clusters have very similar sequences such that sequences that may have different interpre-

Table 4.4: Action sequence archetypes for the video activity.

ASA	Description
V1	Offtask
V2	Pre-mature termination
V3	Complete video without other actions
V4	Complete video with pauses
V5	Complete video with off-task
V6	Complete video with pauses and off-task
V7	Complete video with pauses and scrub (backward)
V8	Incomplete video
V9	Incomplete video with pauses
V10	Incomplete video with scrub (forward)

Table 4.5: Action sequence archetypes for the structured activity.

ASA	Description
S1	Pre-mature termination
S2	Answer directly with off-task
S3	Answer directly
S4	Answer directly with some choice edits
S5	Answer with highlights
S6	Answer with highlights and choice edits
S7	Answer with use of the checklist
S8	Answer with use of the checklist and highlights
S9	Answer with highlights, choice edits and use of the checklist
S10	Submission after use of the checklist and highlights without selecting the answer
S11	Submission without selecting the answer

tation when it comes to the scripted pedagogy are not mixed. As expected, the number of clusters identified increases with the flexibility of each activity. Since the MCQ activity contained multiple questions within the same page, this makes it difficult to differentiate learners' interactions with different assessment questions. This freedom of choice yielded ASAs with indefinite boundaries. The nebulousness of the ASAs associated with each question provides little utility to infer the probability of learner's persistence and therefore will not be addressed in the following section. Descriptions for individual ASAs for the video, structured and unstructured activities are listed in Tables 4.4, 4.5 and 4.6, respectively. As can be seen, these ASAs can be categorized into major themes with

Table 4.6: Action sequence archetypes for the unstructured activity.

ASA	Description
U1	Attempt without submission
U2	Attempts with off-task but no submission
U3	Submission without attempts
U4	Submission without answer
U5	Submission with answer and highlights
U6	Submission without highlights and drags
U7	Submission without highlights
U8	Submission without drags
U9	Submission without drags with one change of model template
U10	Submission without drags with multiple changes of model template
U11	Submission without drags with off-task
U12	Scripted steps
U13	Scripted steps with one change of model template
U14	Scripted steps with multiple change of model template
U15	Scripted steps with off-task

minor variations within each theme. For example, the video ASAs described learners who either did not demonstrate any video watching behaviors, watched the videos to completion or terminated the activity before the video is completed. The ability to label each problem-solving approach to these ASAs allow the problem-solving approaches to be analyzed in relation to scripts and schemas.

4.5.3 Drop-out prediction performance of action sequence archetypes

To establish a baseline for evaluating performance of ASAs to predict learner drop-out, the relationship between mean activity scores and learners' persistence is analyzed. The mean activity scores of learners who drop out after Phase 1 and those who persisted into Phase 2 are tabulated in Table 4.7. As seen from 4.7, when activities are considered independently, only performance from the MCQ activity is associated with learners persisting into Phase 2. At the same time, the MCQ is a pre-requisite to access Phase 2. Therefore, rather than the scores that learners had achieved being indicative of their persistence, the fact that learners completed the MCQ activity signaled that they are

Table 4.7: Mean activity scores for learners who stop during Phase 1 and persist to Phase 2.

Activity	Accuracy	
	Drop-out	Persist
Structured	53.08%	55.23%
Unstructured	54.56%	69.81%
MCQ	0%	91.84%

Table 4.8: Drop-out prediction performance via 2x3 fold cross-validated logistic regression.

Variable Set	Variables	Accuracy	F1	Kappa	AUC
Score-based	Structured	0.461±0.046	0.508±0.100	-0.056±0.089	0.415±0.124
	Unstructured	0.651± 0.059	0.787±0.044	0.00	0.443±0.204
	Structured + Unstructured	0.667	0.800	0.00	0.250±0.236
Sequence-based	Videos	0.540±0.051	0.666±0.037	0.112±0.104	0.516±0.071
	Structured	0.613±0.055	0.518±0.105	0.055±0.096	0.546±0.103
	Unstructured	0.726±0.072	0.761±0.041	0.218±0.178	0.797±0.134
	MCQ	0.824±0.016	0.902±0.001	0.000	1.000

more likely to proceed to Phase 2. Learners who did not complete the MCQ activity would, therefore, obtain a score of zero and will be unable to proceed further. When considered as a series of activities, performance for learners who persisted increase as they progress through the prescribed curriculum, whereas that of those who dropped out remain consistent across all activities.

To highlight the predictive performance of ASAs to detect drop-out, logistic regression models are trained independently on activity grades and the proportion of ASAs demonstrated by each learner to predict whether they persisted into Phase 2. Performance obtained via a 2x3-fold cross validation is tabulated in Table 4.8 in terms of the accuracy, F1, Cohen’s kappa and area under operating characteristics curve (AUC) scores. Similar to the above grade analysis, performance of these logistic regression models increases as activities approach Phase 2. Sequence-based models achieve better classification performance than score-based models for the same activity or preceding activity in the case of video activity. For example, sequence-based models for the structured activity is able to achieve 15.2% better accuracy and 0.131 higher AUC scores than the corresponding

Figure 4.8: Decision tree for the video activity.

score-based models. More importantly, it can be inferred from the Kappa score that score-based models only perform as good as random chances whereas sequence-based models performed moderately better than random chances for the activity and unstructured activity. The ability to perform reliable early detection of drop-out is paramount to administer intervention strategies to alter the learning experiences of these at-risk learners.

To analyze possible reasons of drop-out, drop-out prediction is performed using the proportion of ASAs that learners have engaged in via a classification and regression tree (CART). The resulting decision trees for the video, structured and unstructured activities are illustrated in Figures 4.8, 4.9 and 4.10. Each node in the decision tree is labeled with two variables, the proportion of learners that is represented at that node and the probability of learner drop-out if the prediction is performed at this node. Nodes where

Figure 4.9: Decision tree for the structured activity.

learners were more likely to drop out are colored orange while nodes for learners who are more likely to persist in their learning is represented in green. The rule for splitting learner population that belongs to the current node is labeled below each node. From Fig. 4.8, pre-mature termination (ASA V2) is identified as the top behavior to determine if learners would persist into Phase 2. Learners who terminated the video activities before it is completed in more than 25% of their attempts has 83% chance of dropping out in Phase 1. Furthermore, higher compliance to finish watching the videos (lower ASA V10) and completing video content are indicative of persistence. It is expected that good video

Figure 4.10: Decision tree for the unstructured activity.

watching habits are associated with their persistence.

For the structured activity, learners who demonstrated higher frequencies of schematic variations such as attempts without submission (ASA S10) and attempts with the use of noun highlighting (ASA S5) and the scripted steps were more likely to proceed to Phase 2. Similarly, in the unstructured activity, where learners were given the freedom to carve their own paths in the unstructured activity, their choice to attempt the questions but deviate from the script (ASA U12) was predictive of their persistence into Phase 2. While it remains counter-intuitive that deviations from the taught steps are associated with learners dropping out, learners who did not perform all the steps explicitly are indications that they had already internalized the steps. As compared to studies that employ frequent sub-sequences to predict learning outcomes, the ability to analyze clickstream sequences under semantic labels can provide insights to their problem-solving approaches, such as the different approaches that had been undertaken for the same outcomes.

4.6 Chapter Summary

In this chapter, a sequential clustering framework for mining clickstream problem-solving approach is presented. This framework balances the automaticity of clustering algorithms to identify similar problem-solving approaches and the interpretability of using SPM algorithm to summarize clickstream data. The ability to achieve these two objectives simultaneously is also important to facilitate discussion and inter-disciplinary collaboration between the computer sciences and the learning sciences. The proposed clustering framework is validated on an online elementary mathematics module that used a scripted pedagogy. Staying in line with the scripted pedagogy, the identified problem-solving approaches are characterized for their deviations (or lack of) from the scripted steps and grouped under a bigger label of action sequence archetypes. The study revealed that learners who deviated from the scripted steps were more likely to persist further in their learning. Although it is counter-intuitive to suggest that learners who follow the steps more closely would be less likely to persist in an activity, novices who consistently exhibit script-like behaviors, meant to minimize cognitive load, could indicate a routine that involves lesser assimilation of new concepts.

Part II

Deep Neural Modeling of Educational Click-stream Data

Chapter 5

Review of Grade Prediction Models and Deep Neural Models

Part I of the thesis involves the use of data mining algorithms to identify different types of learning behaviors. The ability to identify these learning behaviors is useful to facilitate decision making for education policies as well as to aid instructors devise group-based intervention strategies for physical classrooms. As the use of virtual learning environments (VLEs) becomes prevalent, personalization for individual learning also becomes achievable with reasonable effort and cost. However, the pre-requisite of being able to perform just-in-time evaluation of one's learning is paramount in order to make real-time adaptation to individual learning preferences so as to maximize their learning. This can be achieved through the use of machine learning algorithms. In Part II of this thesis, the focus is on the prediction of learning outcomes. This chapter begins with an introduction of the grade prediction task. Mathematical operations of various neural network components are then introduced. Subsequently, the relationship between deep neural grade prediction model and linear regression models and their underlying assumptions are established. These assumptions will later be employed for the development of the proposed algorithm in Chapter 7.

5.1 Types of grade prediction

As described in Chapter 2, various stakeholders evaluate learning differently. Institutions and policymakers are interested in learner success and drop-outs while instructors would want to know how their students are coping in their course. These learning outcomes can be represented on a spectrum and include, with increasing granularity, course completion and dropouts, completion certification types, passing or failing a course, banded letter grades, and numerical grade outcomes. Usually, the coarser labels of course completion and certification types can be derived from finer labels of letter and numerical grade outcomes. As the granularity increases, it, however, becomes increasingly harder for prediction models to discern the learning outcomes since the finer labels have tighter tolerances when it comes to modeling error. Grade prediction tasks also differ in terms of context. Machine learning algorithms can be employed to predict learning outcomes based on pre-admission data, historical academic performances within a program or analyzing their learning habits within a single course. While studies for each context rely on different data and algorithms, as can be seen in Section 5.3, most algorithms were employed as black-boxes and can be adapted onto the other contexts.

5.2 Grade prediction task

Consider a learner who had engaged with learning resources hosted within a VLE. The grade prediction task is defined as the forecasting of learners' future grade outcomes using historical learning interactions that were logged within the VLE. Given the historical learning interactions for the i th learner as $\mathbf{x}_i$, and his/her numerical grade outcome as $\mathbf{y}_i$, the grade prediction task is formalized as

$$\hat{\mathbf{y}}_i = f_{\theta}(\mathbf{x}_i), \quad (5.1)$$

where f_{θ} denotes a model that is parameterized by θ to capture the underlying relationship between historical inputs and target grades $P(\mathbf{y}_i|\mathbf{x}_i)$. The type of learning interactions that are logged within VLEs has been reviewed in Chapter 2.

Conventionally, $\mathbf{x}_i$ is analyzed via two approaches—aggregated frequency/time-based features or raw sequential inputs. To analyze learning interactions via aggregated features, a window of analysis has to be determined first. Depending on the curriculum and prediction targets, the designated analysis window may vary from individual teaching weeks to thematic segments, half-semester (often when there are mid-term assessments) or over the complete semester. The total duration spent performing different actions or their frequency of occurrences are then aggregated over the window duration and represented numerically as a single dimension x_i . The interaction vector containing all input features is, therefore, represented by $\mathbf{x}_i = [x_i^1, x_i^2, x_i^3, \dots, x_i^{L_x}]$, where L_x denotes the total number of input features that are considered.

Models that analyze raw sequential inputs, on the other hand, admit collections of ordered interactions as their inputs, i.e., $\mathbf{x}_i = \{x_i^1, x_i^2, x_i^3, \dots, x_i^{L_x}\}$, where L_x denotes the sequence length. Although temporal dependencies between interaction actions are preserved within these sequential inputs at the expense of higher computational and complexity loads, earlier studies have highlighted that these temporal dependencies are crucial in allowing models to discern learners who achieved different grade outcomes [166, 167]. One study demonstrated that models trained using sequential inputs can achieve greater stability compared to those trained using aggregated features when the analysis window is varied [166]. This is because learners can appear to behave similarly when viewed through their aggregated features over an extended duration [166]. However, since discrete symbolic actions of interaction sequences differ from numerical feature spaces of prediction models, the onus to map these discrete interactions onto numerical feature vectors is placed on the models before they can learn the underlying relationship. This also implies that the prediction performance is dependent on both the discrete-to-continuous mapping as well as the model’s capability to represent the underlying relationship between the numerical features and target grades.

Pre-processing techniques may be applied to the discrete interaction sequences to quantify various aspects such as the overall composition and sets of contiguous actions. These numerical features include the bag-of-words representation [168] (which counts the occurrence of individual actions within the sequence) to n -grams [169] (that model the composition of contiguous sub-sequences). As the feature engineering process is

labor-intensive and ad-hoc in nature, these features are often non-exhaustive and may be biased. Furthermore, since the utility of these features cannot be tested prior to the modeling process, an additional feature selection and pruning are required to achieve good predictive performance. An alternative to feature engineering is the use of embedding layers within deep neural networks (DNNs). The embedding layer maps each x_i^k into L_{emb} -dimensional continuous vectors via a look-up table

$$T = \{x_i^k : \mathbf{v}_i^k \in \mathbb{R}^{L_{\text{emb}}}\}, \quad (5.2)$$

where $\mathbf{v}_i^k$ is randomly initialized according to any probabilistic distribution. The embeddings are updated iteratively throughout the training process. As the embeddings are learnt simultaneously with parameters for other computational units within the DNN through an end-to-end backpropagation (BP) gradient descent optimization, these embeddings, therefore, offer high utility without introducing excessive biases that are prominent with the feature engineering process. Instead of initializing the embedding random, an auto-encoder may be employed to refine the embedding vector. This is achieved by training an encoder (ENC) and a decoder (DEC), both of which are multilayer perceptrons (MLPs), as illustrated in Fig. 5.1. Actions in a clickstream sequence are first represented using a one-hot encoding, i.e., a binary vector that has a length equivalent to the total number of unique actions in the dataset and has a value of one at positional index of a given action and zero otherwise. The one-hot vector x_i^k is then encoded onto a compressed vector representation that is given by

$$\mathbf{v}_i^k = \text{ENC}(x_i^k), \quad (5.3)$$

before DEC reconstructs the one-hot vector using $\mathbf{v}_i^k$, that is given by

$$\hat{x}_i^k = \text{DEC}(\mathbf{v}_i^k). \quad (5.4)$$

To learn an appropriated compressed vector representation for each action, the auto-

Figure 5.1: Overview of an autoencoder.

encoder minimizes the reconstruction loss that is given by

$$\mathcal{L}_{AE} = ||\hat{x}_i^k - x_i^k||. \quad (5.5)$$

When AE is trained, only ENC is retained and new inputs are parsed into ENC to be encoded onto the compressed vector representation, as described in (5.3). Like the embedding layer, ENC can continue to be updated throughout the training process to identify better representations that are suited for the task.

5.2.1 Components of deep neural networks

Once the inputs are converted into numerical vectors, DNNs employ a combination of computational units such as fully-connected (FC) layers, convolutional layers, and recurrent units to refine the features and predict the target grade outcomes. Using the output of the previous layer as its inputs (the inputs for the first hidden layer), each layer employs different mathematical operations to transform its inputs.

The FC layer, convolutional layer and recurrent units are designed to capture different aspects of the inputs. As the fundamental component, the FC layer transforms its inputs

via a weighted sum mechanism that is given by

$$\begin{aligned}\mathbf{h}_i &= \text{FC}(\mathbf{x}_i) \\ &= \psi(\mathbf{W}_{\text{FC}}^T \cdot \mathbf{x}_i + b_{\text{FC}}),\end{aligned}\tag{5.6}$$

where ψ denotes a non-linear activation function, $\mathbf{W}_{\text{FC}}$ and b_{FC} represent the hidden weights and biases, and $\mathbf{h}_i$ is the transformed output of the FC layer. The convolutional unit extracts spatial/temporal information from two-dimensional inputs by applying its rectangular kernels across the input space [170]. Given a spatial/temporal location (m, n) on a two-dimensional input $\mathbf{X}_i$, and a rectangular kernel κ , the convolution operation at (m, n) can be expressed as

$$\begin{aligned}\mathbf{H}[m, n] &= (\mathbf{X}_i * \kappa)[m, n] \\ &= \sum_k \sum_j \kappa[j, k] \mathbf{X}_i[m - j, n - k].\end{aligned}\tag{5.7}$$

Hereafter, the application of the convolutional operator across the input matrix is referred to as

$$\mathbf{h}_i = \text{CONV}(\mathbf{x}_i).\tag{5.8}$$

For the recurrent units, the inputs are processed sequentially to incorporate new information within the hidden cells of the recurrent unit. The recurrent unit employs a cell state to maintain the representation of historical inputs and new information is fused into this representation iteratively. This is given by

$$\mathbf{c}^\tau = \psi(\mathbf{W}_c \cdot [\mathbf{x}_i^\tau, \mathbf{c}_i^{\tau-1}] + b_c),\tag{5.9}$$

$$\mathbf{h}^\tau = \psi(\mathbf{W}_h \cdot \mathbf{x}_i^\tau + b_h)\tag{5.10}$$

where $\mathbf{c}^\tau$ and $\mathbf{h}^\tau$ denote the cell and feature representation of the input sequence after the τ th interaction within $\mathbf{x}_i$, respectively, and $\{\mathbf{W}_c, b_c, \mathbf{W}_h, b_h\}$ denotes the set of parameters for the recurrent cell. However, it is well-known that the recurrent unit suffers from the vanishing/exploding gradient problem where the error signals are amplified/attenuated

as it propagates through time along the input sequences. This results in instability in the recurrent unit. The gated recurrent unit (GRU) and long short-term memory (LSTM) recurrent neural network was proposed to solve this challenge by employing gates to control the flow of information through time [171]. The GRU unit is given by

$$\mathbf{f}^\tau = \psi(\mathbf{W}_f \cdot [\mathbf{h}^{\tau-1}, \mathbf{x}_i^\tau] + b_f), \quad (5.11)$$

$$\mathbf{i}^\tau = \psi(\mathbf{W}_i \cdot [\mathbf{h}^{\tau-1}, \mathbf{x}_i^\tau] + b_i), \quad (5.12)$$

$$\bar{\mathbf{h}}^\tau = \psi(\mathbf{W}_h \cdot [\mathbf{i}^\tau \otimes \mathbf{h}^{\tau-1}, \mathbf{x}_i^\tau] + b_h), \quad (5.13)$$

$$\mathbf{h}^\tau = (1 - \mathbf{f}^\tau) \otimes \mathbf{h}^{\tau-1} + \mathbf{f}^\tau \cdot \bar{\mathbf{h}}^\tau, \quad (5.14)$$

while the LSTM unit is mathematically expressed as

$$\mathbf{f}^\tau = \psi(\mathbf{W}_f \cdot [\mathbf{h}^{\tau-1}, \mathbf{x}_i^\tau] + b_f), \quad (5.15)$$

$$\mathbf{i}^\tau = \psi(\mathbf{W}_i \cdot [\mathbf{h}^{\tau-1}, \mathbf{x}_i^\tau] + b_i), \quad (5.16)$$

$$\mathbf{o}^\tau = \psi(\mathbf{W}_o \cdot [\mathbf{h}^{\tau-1}, \mathbf{x}_i^\tau] + b_o), \quad (5.17)$$

$$\bar{\mathbf{c}}^\tau = \psi(\mathbf{W}_c \cdot [\mathbf{h}^{\tau-1}, \mathbf{x}_i^\tau] + b_c), \quad (5.18)$$

$$\mathbf{c}^\tau = \mathbf{f}^\tau \otimes \mathbf{c}^{\tau-1} + \mathbf{i}^\tau \otimes \bar{\mathbf{c}}^\tau, \quad (5.19)$$

$$\mathbf{h}^\tau = \mathbf{o}^\tau \otimes \psi(\mathbf{c}^\tau), \quad (5.20)$$

where $\mathbf{f}^\tau$, $\mathbf{i}^\tau$, $\mathbf{o}^\tau$ denote the forget, input and output gates for the τ th interaction within $\mathbf{x}_i$, respectively, and $\mathbf{c}^\tau$ and $\mathbf{h}^\tau$ are the respective cell and hidden states for the τ th interaction. $[\cdot, \cdot]$ and $\otimes$ denote the vector concatenation operator and Hadamard product, respectively. Equations (5.15)-(5.20) are summarized as

$$\mathbf{h}_i = \text{LSTM}(\mathbf{x}_i). \quad (5.21)$$

The input, output and forget gates for the t th input, denoted by i^t , o^t and f^t , respectively, are responsible to control the amount of information flow within the LSTM cell.

In principle, any non-linear function may be employed as activation functions. However, the sigmoid function, hyperbolic tangent (tanh) and rectified linear units (ReLU) are among the top favorites due to their ability to achieve good performance from exist-

ing studies. Defining x as an arbitrary variable, the sigmoid function that compresses its inputs to values that are bounded between 0 and 1 is given as

$$\psi(x) = \frac{1}{1 + \exp(-x)}, \quad (5.22)$$

whereas the tanh function maps its inputs onto a range of -1 to 1 and it is mathematically expressed as

$$\psi(x) = \frac{\exp(x) - \exp(-x)}{\exp(x) + \exp(-x)}. \quad (5.23)$$

The ReLU function gains its popularity in DNNs since, similar to recurrent units, the gradients can diminish quickly with an increasing number of layers being added to the network. Since ReLU filters its inputs instead of compressing it, the gradient of positive inputs are kept at unity and this allows the gradient to pass through the network layers easily. The ReLU activation function is given by

$$\psi(x) = \max(0, x). \quad (5.24)$$

From the formulation of various neural network components, it can be implied that these computational units typically perform linear operations while the activation functions apply non-linear compression onto the outputs. Both the computational units as well as the non-linear activation functions are crucial such that DNNs can automatically extract and refine features as it passes through the numerous layers. However, activation functions are typically omitted from output layers of DNNs. As such, the relationship between the refined features of a DNN and its target output follows a linear relationship.

5.2.2 Linear regression paradigm

When considering all the hidden layers as a deterministic operator, a DNN operates similar to a linear regression (LR) model. A general formulation for this type of models

is similar to (5.1) and can be expressed as

$$\hat{\mathbf{y}}_i = f_{\boldsymbol{\theta}}(\mathbf{x}_i) + \epsilon_i, \quad (5.25)$$

where ϵ_i denotes an error term that represents the discrepancy between the predicted grade $\hat{\mathbf{y}}_i$ and actual grade $\mathbf{y}_i$, $f_{\boldsymbol{\theta}}$ denotes the aggregated operation of all the hidden layers, and $\boldsymbol{\theta}$ is the set of parameters from these hidden layers. While both DNN (with continuous target outputs) and LR often minimize the residual sum of squares that is given by

$$\begin{aligned} \text{RSS} &= \sum_i (\mathbf{y}_i - \hat{\mathbf{y}}_i)^2 \\ &= \sum_i (\mathbf{y}_i - f_{\boldsymbol{\theta}}(\mathbf{x}_i))^2, \end{aligned} \quad (5.26)$$

batch-wise gradient descent is preferred for the former while the latter employs ordinary least square methods. Consequently, DNNs inherit some of the assumptions that are normally associated with LR, which include an underlying linear relationship between its input and target variables, Gaussian distributions for all variables, little to no multicollinearity and autocorrelation between its input variables, and homoscedasticity (the error terms have similar variances across all values of the input variables). As will be demonstrated in the following chapters, interaction clickstream data may not satisfy these criteria due to its inherent MV characteristics. This will serve as the foundation for the development of the proposed method described in Chapter 7.

5.3 Existing algorithms and motivation of proposed algorithms

Existing studies for the prediction of at-risk learners and grade outcomes are now presented. Initial efforts to predict at-risk learners posit that learners' demographics may affect their learning. These studies were typically carried out at institutional level, where demographic information is readily available, so that appropriate intervention strategies

can be designed and administered to prevent drop-out. To understand the impact of demography on learning outcomes, existing studies involved the use of non-cognitive traits such as their scholastic aptitude [213] and psychosocial aspect of learners [213], personality traits [220], family social economic status [174,207,214], gender and age [174, 177,183,198,207,210,214–219], and pre-admission academic achievements [197,207]. Such analysis was performed using correlational analyses and various classification techniques. Pre-admission academic achievements, gender and age were among variables that had a significant effect on learning outcomes [207, 213, 215]. However, these studies reported mixed effects for the three variables. It is, therefore, believed that while demographic may be correlated to learners' persistence, the context of how VLEs are deployed within the course also plays an important role in affecting one's learning.

Since learners are expected to perform similarly for courses that they have not taken and those that they had already taken, some studies have also explored the use of historical academic achievements obtained within a program for forecasting [174, 177, 178, 183, 188, 196–212]. Furthermore, some studies hypothesized that course structures, concept and topic dependencies between courses and even instructional differences between educators can affect the learning outcomes. Therefore, these factors were also explicitly represented within grade prediction models [200, 202, 206, 228, 229]. However, unlike studies that utilize interactions that are logged within VLEs throughout the course, these predictions were performed at program- or course-level and did not analyze the impact of nuances in learning efforts on learning outcomes.

Learning efforts are often synonymous with learning opportunities, which are, in turn, correlated with learning outcomes. To quantify the amount of learning effort, interactions within VLEs were considered in terms of the amount of time spent within the platform [172–175] and frequency of accessing various learning resources such as video content, assessment problems, discussion forums and other collaborative exercises such as micro-blogs. Video watching behaviors were also analyzed through the frequency of various video playback actions (play, pause, rewind, fast-forward) [176–179], proportions and the total duration of video played [175–178, 182–184], the total number of views [176–179], and video watching frequency and intervals [180, 181]. While video content forms the primary source of information [119], due to the lack of face-to-face opportunities

with instructors, forum discussions are, instead, often considered the main driving force within MOOC to retain learners. This is because MOOC generally have a low entry cost, leading to learners having different motivations subscribing to these MOOC. It is expected that learners who are motivated in their learning are more likely to participate in forum discussions as opposed to those who merely subscribed to MOOC due to its novelty and wanted to explore a new mode of learning. Therefore, analyses of forum participation activities should allow algorithms to discern learners who are at-risk of dropping out within these MOOC. To understand the type of engagement learners have in forum discussions, learners' activities were described in terms of their posting and reply frequencies [175, 179, 184, 186, 190, 191], the topics that they had posted or replied within discussion boards [192–195], their contribution quantified using social network analysis features such as their centrality, eccentricity, authority and hub scores [173, 187–189] and the sentiments of their posts [185, 186] were employed to determine if learners would complete the course.

Predictions in the above studies were performed using a multitude of machine learning algorithms. These algorithms can be broadly categorized into instance-based algorithm such as KNN [181, 204], matrix factorization and collaborative filtering methods [181, 200–203, 211, 212], and discriminative models such as naïve Bayes methods [174, 176, 214], random forests [174, 176, 180, 197, 199], rule-based classifiers [197, 207, 209, 221], support vector machines [179–181, 193, 195, 198, 214], decision trees [174, 176, 180, 183, 194, 196–198, 210, 214], regression models [173, 174, 176–178, 184, 188, 191, 192, 198, 205, 206, 213, 216, 217], Bayesian networks [197, 210, 214], correlational analysis [185, 189, 220], factor graphs [190], Cox proportional hazard model [182, 187, 215], and neural networks [177, 186, 200, 208–210, 219, 228, 229]. Given the infancy of learning analytics at the time of these studies, and the limited understanding about learning, these algorithms have generally been employed as black-boxes to discern learning outcomes.

Due to improvement in modeling techniques, recent studies have viewed clickstream data as interaction sequences to model the temporal dependencies within the inputs. These temporal dependencies represent direct manifestations of the infinitesimal decision-making process that is spontaneously happening during learning. Therefore, it is expected that the ability to capture these temporal dependencies would improve prediction per-

formance. Prior to the popularity of temporal models, features that capture temporal aspects of inputs were proposed. For example, weekly accesses of learning resources were preserved as individual vectors but vectors across several weeks were concatenated into a single vector [167]. On the other hand, sequential features such as n-grams and behavioral motifs were also proposed to identify contiguous interaction actions [119, 230–232]. Multiple models trained on clickstream data from individual weeks [233, 234] were aggregated and smoothen via transfer learning algorithm [234]. Models that are able to capture temporal dependencies were also applied to model learners’ grades. A temporal variant of the Cox proportional hazard model and HMMs were employed to capture variations between learners’ grades across the semester [235–239]. However, the Markov assumption of HMMs implies that the models are only capable of capturing limited temporal information. While hand-crafted sequential features can capture some semblance of temporal dependencies within interaction sequences, methods involving feature engineering process achieve limited predictive performances. More recently, studies have also explored the use of recurrent neural networks to exploit their ability to identify temporal dependencies via a data-driven approach. Recurrent neural networks [166, 237, 240–244], convolutional neural network [245] or a hybrid neural network comprising both convolutional and recurrent layers [246] have been applied to model raw sequential clickstream data.

As discussed in Section 5.2.2, most prediction models that predict numerical grade outcomes operate under the linear regression paradigm. Therefore, two other aspects of grade prediction that were less often discussed but equally important are the uncertainty surrounding the predictions and the generalizability of models for future course runs. Uncertainties within grade prediction translate into undesirable consequences where poor-performing learners may be demoralized to the point of dropping out while learners who have the ability to excel are not presented with sufficient opportunity to shine. To address this issue, studies have estimated the uncertainties surrounding their predictions through an analysis of predicted grades for similar behaving learners [204] or modeled these uncertainties within the parameter spaces of deep Bayesian models via Monte Carlo (MC) dropout [247]. Furthermore, prediction uncertainties are typically higher when the predictions are performed at earlier opportunities. This is believed to be affected by the

lack of data and other aspects of learners. To identify the best opportunity and study the trade-off between accuracy and data sufficiency, difference in classification performance at various gaps between observations and evaluations are analyzed [184] and modeled via a genetic algorithm to simultaneously maximize classification accuracy at the earliest opportunity [199].

Generalizability of prediction models, on the other hand, remains one of the greatest challenges within learning analytics [14, 248, 249]. It is well-known that prediction models are usually not generalizable due to, and not limited to, differences in cohort distribution, pedagogical differences and teaching paces. To overcome this limitation, transfer learning algorithms were employed to transfer the knowledge instilled from one dataset onto new datasets that have yet to be trained on [249].

5.4 Other applications of deep neural networks in educational data mining and learning analytics.

Another popular research direction that employs deep neural networks in educational data mining (EDM) and learning analytics (LA) is the modeling of knowledge components and skills as a surrogate of knowledge mastery. While Bayesian knowledge tracing model originally employ a hidden Markov model (HMM), each HMM model only encodes a single skill. The training of numerous models is, therefore, required to cover all the concepts covered within a course. Deep knowledge tracing models are direct extensions of Bayesian knowledge tracing [31] that employs recurrent neural networks (RNNs) that have been trained using historical question responses. As opposed to Bayesian knowledge tracing (BKT) models, which splits the dataset by knowledge components (KCs) and train a hidden Markov model for each KC, DKT considers responses from all KCs holistically and encodes this information within a single model. To train a DKT, question responses are first differentiated by KC and correctness. These responses are, then, encoded using one-hot encoding and projected onto an arbitrary feature space. The recurrent neural network encodes the responses sequentially and produce the probability of the learner correctly answers the next question for each KC. In addition to being able to model

mastery levels, DKT models have also been employed to estimate concept relationships via the probability of correctly answering questions related to a concept after answering a series of questions [31]. However, interactions between knowledge and skills that are captured by the feature space cannot be inferred.

Although DKT unifies the many BKT models into a single model, it does not address the issue of sparsity in the dataset. Given that a single course can cover multiple knowledge components but would not be able to administer sufficient assessment items for each knowledge component to its learners, the number of responses per knowledge component remain sparse and often affect modeling performance adversely. To over this limitation, qDKT which incorporates a graph Laplacian regularization to smoothen prediction for each skill is proposed [222]. Furthermore, the incorporation of prerequisites to link assessment items with similar concepts was also proposed [223].

To address this issue of interpretability for DKT, dynamic key-value memory networks (DKVMNs) that employ memory-augmented neural network (MANN) were proposed [224]. Similar to MANN, information in a MANN is stored and retrieve via memory matrices using a read [250–252] and write process [250]. However, unlike the single matrix in MANN, DKVMN uses two matrices a key matrix for latent concepts and value matrix for student mastery level. More recently, transformers have also been applied to train DKTs [225–227]. Since they employ a self-attention mechanism, information encoded by transformers may also be interpreted.

5.5 Chapter summary

In this chapter, sequential clickstream grade prediction is discussed. Various types of grade prediction tasks is first presented, followed by the formalization of the grade prediction task. Two approaches that process clickstream data into aggregated features and using clickstream data as ordered sequences of actions are reviewed. Deep neural architectures are then introduced for the modeling of clickstream data as sequences of interaction actions. The relationship between DNN-based grade prediction model and linear regression models is established. Finally, existing studies that analyzed different

types of learning behaviors to establish their relationship with learning outcomes are reviewed. Due to the inherent characteristic of sequential clickstream data, performance of grade predictions is, however, limited. Furthermore, the use of transient learning habits collected within the course often do not allow models to be transferable to newer course runs. These challenges will serve as a motivation for the proposed frameworks in the following chapters.

Table 5.1: Summary of prediction models for detecting at-risk learners and their corresponding feature sets.

Topics	Features	Methods
Video Watching Behaviour	Time spent [172–175], Action frequency [176–179], Video views [176–179], Video-watching frequency and intervals [180, 181], Proportion of video watched [175–178, 182–184]	Random forests [174, 176, 180], k -nearest neighbours [181], Cox proportional hazard model [182], Neural networks [177], Regression models [173, 174, 176–178, 184], Nave Bayes [176], Decision trees [174, 176, 180, 183], Support vector machines [179–181], Matrix factorization & collaborative filtering [181]
Forum Discussion	Sentiment [185, 186], Social network analysis features [173, 187–189], Post and reply frequency [175, 179, 184, 186, 190, 191], Discussion topics [192–195]	Cox proportional hazard model [187], Neural networks [186], Regression models [173, 191, 192], Decision trees [194], Correlational analysis [185, 189], Support vector machines [179, 193, 195], Factor graphs [190]
Demographics	Historical academic achievements [174, 177, 178, 183, 188, 196–212], Pre-admission academic achievements [197, 207] Scholastic aptitude [213], Socioeconomic factors [174, 207, 214], Gender & age [174, 177, 183, 198, 207, 210, 214–219], Psychosocial aspect [213], Personality traits [220],	Random forests [174, 197, 199], k -nearest neighbours [204], Neural networks [177, 200, 208–210, 219], Bayesian networks [197, 210, 214], Regression models [174, 177, 178, 188, 198, 205, 206, 213, 216, 217], Nave Bayes [174, 214], Rule-based systems [197, 207, 209, 221], Decision trees [174, 183, 196–198, 210, 214], Correlational analysis [220], Support vector machines [198, 214], Matrix factorization & collaborative filtering [200–203, 211, 212], Cox proportional hazard model [215]
Deep knowledge tracing	-	Deep knowledge tracing [31, 222, 223], Memory-augmented neural networks [224], Transformers [225–227]

Chapter 6

Regularizing Sequential Clickstream Grade Prediction Model using Non-Cognitive traits

In Chapter 5, the problem of grade prediction has been reviewed. Existing grade prediction studies that focused on the use of feature engineering to quantify learning behaviors have seen successes for forecasting learners outcomes. Since learning is a dynamic process, these aggregated features are unable to capture temporal changes in learning behaviors. As such, the focus gradually shifts from analyzing efforts via aggregated time-based measurements to modeling learning behaviors through raw interaction sequences. In this chapter, the impact of online clickstream sequences to forecast learning outcomes is examined. To further enhance grade prediction performance, a multi-modal deep learning schema that captures the transient behaviors and persistent non-cognitive traits of learners is developed.

Part of this chapter has been published as Kelvin H. R. Ng, S. Tatinati, and Andy W. H. Khong, “Online education evaluation for signal processing course through student learning pathways,” in *Proc. IEEE Int. Conf. Acoust., Speech and Signal Process., ICASSP*, 2018, Calgary, AB, Canada, Apr 15-20 2018, pp. 6458-6462.

Figure 6.1: The 2x2 Hierarchical Motivation Framework.

6.1 Introduction

As discussed in Chapter 5, existing studies rely mainly on the use of aggregated time- and frequency-based features to forecast learning outcomes. Given limited knowledge of learning behaviors within virtual learning environments (VLEs) [177, 253] on top of the laborious, costly, and often biased feature engineering process, it is therefore difficult to design a comprehensive set of features that can predict learning outcomes well while remaining platform-agnostic [242]. While exploiting deep neural networks (DNNs) for their ability to identify high-utility features automatically from the data is attractive, it is also well-known that learning behaviors are heterogeneous such that even learners who engaged in same learning behavior can obtain very different outcomes [190]. This is because the cognitive process is simultaneously regulated by non-cognitive factors such as their personality, the learning environment, study habits, and pedagogy [23, 24, 254].

Some of the most notable non-cognitive traits include learners' motivation [22], theory of intelligence [255], academic buoyancy [256] and grit [257]. The hierarchical motivation framework proposed by Elliot et al. describes three achievement orientations—performance-approach goal, performance-avoidance goal and mastery goal—and is constructed via a hierarchical combination of the achievement goals and the approach-avoidance construct [22], as depicted in Fig. 6.1. Achievement goal is defined as the purpose for engaging in a task to achieve a potential outcome and this is further deconstructed into performance and mastery achievement goals. Performance goal can

be viewed as a preference to demonstrate achieving success that are stereotypical for the tasks and are, therefore, often associated with a preference of easier task, decreased task enjoyment and the withdrawal of effort in the face of failure [22]. On the other hand, mastery goal is driven by the need to develop competency and comprehension. As such, mastery-driven behaviors are often described by preferences for challenging tasks, persistence when faced with failure and greater task enjoyment. Fundamentally, these goals are directional functions in motivation and serves as a cognitive end-point that regulates behaviors. While they are strategic manifestation of motivation, it is insufficient to consider only the goal orientation. The framework, therefore, considers the underlying processes via the approach-avoidance construct. Under this construct, behaviors are described as responses towards positive and negative stimuli. This is fundamental to motivation since humans have been shown to evaluate stimuli based on positive or negative valences, and this evaluation would, in turn, evoke approach or avoidance behaviors. Coming back to the education context, the ability to differentiate learners based on these three dimensions would allow for the understanding of their learning behaviors.

The theory of intelligence, or self-theory, describes the belief that one can either view intelligence as a fixed entity, and therefore cannot be improved, or malleable variable that is developed through effort and guidance. Learners who consider their intelligence as fixed may often find themselves concerned with appearing incapable when faced with challenges and the lack of early success can quickly impact their self-esteem. Consequently, these learners may not persist in their learning. On the other hand, learners who view intelligence as malleable put effort into avoiding being unprepared and therefore favor opportunities to learn, which is beneficial in the long run. These views are important in an education context since they drive learners' behaviors towards goal choices. As a caveat, these goals are not always defined in the positive sense and have been associated with self-handicapping behaviors [258].

Grit is defined as perseverance and passion for long-term goals [257]. While it is not correlated to the intelligence quotient (IQ), grit is often cited as a talent that distinguish successful persons in the real world. Similarly, grit has a favorable impact on learning since, unlike achievement goals, it does not consider the need for achievement. Therefore, learners high in grit can set extremely long-term goal and not sway in the absence of

positive feedback. Consequently, this results in learners receiving more opportunities for learning and this would ultimately benefit individuals. While grit is portrayed as a desirable trait in existing literature [257], a meta-analysis is conducted to verify the validity of the grit inventory and whether grit should be a single higher-order variable instead of two independent sub-scales of perseverance of effort and consistency of interest [146]. While some of these hypotheses remain true, it was concluded that the aggregation of the two sub-scales into a single higher-order variable is not supported because it results in a significant loss in the ability to predict performance.

Like grit, academic buoyancy describes responses of individuals when faced with challenging circumstances [256]. However, the construct considers an additional dimension of the capacity to adapt under difficult circumstances. Therefore, learners with higher academic buoyancy are more likely to excel in future when they are faced with poor performance now. As learners with different non-cognitive traits have been shown to display significantly different learning strategies [60] and learning outcomes [55–59], it is, therefore, important for prediction models to consider these effects.

To support the incorporation of transient clickstream behavior and persistent non-cognitive traits, one of the approach is to augment the feature space via joint embeddings. Information extracted from non-cognitive traits can be incorporated into the feature space learnt on clickstream behavior. This approach has been widely studied among text classification applications, where textual information such as knowledge bases, sentence- and word-level features as well as visual factors from images are combined to create better text embeddings for knowledge inference [259–262].

6.2 Proposed joint-space sequential grade prediction model

The proposed joint-space sequential grade prediction model comprises two DNNs for the encoding and extracting sequential patterns from input clickstream sequences, and the encoding of the non-cognitive covariates, respectively, and a joint embedding layer that fuses latent features extracted from these two inputs, as depicted in Fig. 6.2.

Figure 6.2: Overview of the proposed joint-space sequential grade prediction model.

Given a sequence of clickstream traces $\mathbf{x}_i = [x_i^1, x_i^2, \dots, x_i^{L_x}]$, with x_i^k representing the k th action of the i th input sequence, the DNN begins by mapping each x_i^k into an L_{emb} -dimensional continuous vector v_i^k via the embedding layer, as described in (5.2). The embedded sequence is, therefore, denoted by $\mathbf{v}_i = [v_i^1, v_i^2, \dots, v_i^{L_x}]$. Multiple DNN components are then applied to extract sequential patterns from the embedded features $\mathbf{v}_i$. These sequential patterns are represented via a feature vector representation

$$\mathbf{h}_i = \text{SE}(\mathbf{v}_i), \quad (6.1)$$

where $\text{SE}(\cdot)$ denotes a deep neural sequence encoder such as the RNN, gated recurrent unit (GRU) and long short-term memory (LSTM) network. Finally, fully-connected (FC) layers are applied to reduce $\mathbf{h}_i$ into hidden feature vectors $\mathbf{z}_i^{\text{seq}}$ of lower dimension L_{seq} such that

$$\mathbf{z}_i^{\text{seq}} = \text{FC}(\mathbf{h}_i), \quad (6.2)$$

where $\text{FC}(\cdot)$ has been defined in Section 5.2.1.

Non-cognitive covariates are obtained from self-reported questionnaires, where participants provide their responses to each question on a Likert scale. Since the questionnaire is constructed and validated on a different participant population, these responses have to be validated with the original study population to ensure that the discovered latent variables and their factor loadings are similar. This is achieved via the confirmatory factor analysis [263]. Assuming that the set of learners is identical to the original population that was used to construct the questionnaire, the non-cognitive covariates are recovered from question responses by summing and standardizing responses that corresponding to the same latent factor. Defining a set of question responses $\{q_i^k\}$ obtained from the i th learner for the k th latent factor and the sum of these responses as $p_i^k = \sum_i q_i^k$, the k th latent variable can be recovered via

$$\bar{p}_i^k = \frac{p_i^k - \mu_k}{\sigma_k}, \quad (6.3)$$

where μ_k and σ_k denote the corresponding mean and standard deviation of p_i^k . FC layers

are then employed to transform the vector of non-cognitive variables $\bar{\mathbf{p}}_i$ into compressed vector representations $\mathbf{z}_i^{\text{psych}}$ with dimensions L_{psych} such that

$$\mathbf{z}_i^{\text{psych}} = \text{FC}(\bar{\mathbf{p}}_i). \quad (6.4)$$

To create a joint feature space between $\mathbf{z}_i^{\text{seq}}$ and $\mathbf{z}_i^{\text{psych}}$, two mechanisms are employed to fuse the two feature vectors. The first mechanism creates the joint feature space by concatenating $\mathbf{z}_i^{\text{seq}}$ and $\mathbf{z}_i^{\text{psych}}$ that is mathematically given by

$$\mathbf{z}_i^{\text{joint}} = [\mathbf{z}_i^{\text{seq}}, \mathbf{z}_i^{\text{psych}}], \quad (6.5)$$

where $[\cdot, \cdot]$ denotes the concatenation operator and $\mathbf{z}_i^{\text{joint}} \in \mathbb{R}^{(L_{\text{seq}} + L_{\text{psych}})}$ is a joint feature vector representation of length $(L_{\text{seq}} + L_{\text{psych}})$. Using this joint embedding scheme, feature spaces corresponding to the two independent components are stitched together. As such, subsequent layers in the DNN has the ability to learn and assign different weights for each feature dimensions, allowing the model to select and refine relevant features. Hereafter, this approach of fusing two feature vector via concatenation is referred to as the parallel space (PS) joint embedding.

The second mechanism emulates a bilinear mapping [260] and employs a multiplication operator to create the joint feature space that is given by

$$\mathbf{z}_i^{\text{joint}} = (\mathbf{z}_i^{\text{seq}} \otimes \mathbf{z}_i^{\text{psych}}), \quad (6.6)$$

where $\otimes$ denotes the Hadamard product. Compared to PS, the feature space learnt from clickstream sequences is augmented with that of the non-cognitive variable, giving rise to a new feature space. Following the inference of neural model outputs as conditional probabilities, this joint feature space is synonymous to a joint probability distribution. As a caveat, since the joint feature space is created via the Hadamard product, the dimensions of compressed vector representations for both the input sequences and non-cognitive covariates must be equal, i.e., $L_{\text{seq}} = L_{\text{psych}}$. However, since any point in the new feature space is simultaneously characterized by the two inputs, unlike parallel space embedding, computational units that are applied subsequently will suffer from biased

emphasis to either of the feature vectors. This approach of fusing two feature vector is denoted by the multiplicative space (MS) embedding.

Once the joint space is created, FC layers are applied to refine the joint feature vectors and to perform the grade prediction

$$\hat{\mathbf{y}}_i = \text{FC}(\mathbf{z}_i^{\text{joint}}). \quad (6.7)$$

6.3 Evaluation framework

6.3.1 Dataset

Performance of the proposed joint-space sequential grade prediction model was evaluated on the DSP dataset. The DSP dataset was collected from an undergraduate level DSP course that was taught over 13 teaching weeks using flipped classroom pedagogy at the Nanyang Technological University, Singapore. Each week, learners are expected to complete a set of online learning activities—video lectures, interactive media contents, multiple-choice practice questions, and open-ended questions—that were hosted on an LMS, where instructors can designate linear sequences of activities for each teaching week. At the start of each face-to-face seminar, a quiz was administered to assess learners’ readiness levels. Thereafter, based on their assessment performances, instructors would then facilitate discussions to clarify any doubts and reinforce the taught concepts.

For the online multiple-choice practice questions, learners had been conveyed that these questions would serve only as self-monitoring mechanisms. Therefore, their participation is voluntary and would not affect the course outcomes. In the seventh teaching week, a summative assessment was conducted to assess learners’ efficacy on all the content that had been covered. This quiz would contribute to their course outcomes. Throughout the first six weeks, learners’ interactions with video lecture resources and practice questions were collected as transactional logs in a database. Furthermore, at the beginning of the course, learners were invited to participate in a 44-items questionnaire that evaluates four non-cognitive traits [22,255–257]. Responses from the survey were validated against

Table 6.1: List of logged actions for the DSP dataset.

Learning Content Types	Actions
Video	Start Pause Resume End video Seek forward Seek back End activity Enter full screen Exit full screen Speed up Speed down
Multiple-choiced Questions	Select choice Edit choice Submit answers
Personality Traits	Grit [257], Theory of intelligence [255], Academic Buoyancy [256], Motivation [22]

the original inventories. The list of actions that were logged and non-cognitive covariates are listed in Table 6.1 and approximately 91,000 logs obtained from 148 learners were analyzed. As discussed in Section 6.1, the combination of sub-scales into higher-order variable may impact prediction performance. Therefore, the four non-cognitive traits employed in this study are included in the proposed method as independent sub-scales.

6.3.2 Performance comparison

Grade prediction performance was evaluated across one frequency-based model, three sequential-based DNNs and the six proposed joint-space models. These three sequential-based DNNs were each implemented based on (6.5) and (6.6) giving six proposed joint-space models.

The frequency-based model sets the baseline for existing methods that analyze time- or frequency-based features. In this case, the frequency of each action performed is considered. To perform the grade prediction via frequency-based features, a three-layer

MLP is employed. This model is denoted by FREQ.

For the sequential-based models, two convolutional neural networks (CNNs) and a LSTM network are employed to encode the clickstream sequences. All three models share a common feature of employing an embedding layer to perform the mapping from discrete actions to continuous mapping. Both CNNs and LSTMs have seen successes in applications such as text and document classification, which uses discrete ordered sequences as their inputs. CNNs, however, learn differently from RNNs. For CNNs and RNNs, individual symbols within the sequential inputs are first vectorized. RNNs then process these inputs sequentially, encoding the inputs onto an arbitrary feature space that represents the relationship between these inputs to the prediction targets. CNNs, on the other hand, stack these vectors into a rectangular matrix $\mathbf{V}_i \in \mathbb{R}^{L_x \times L_{emb}}$, then apply its kernels across the input feature space to identify regularities. Convolutional layers are then applied to the matrix. The two CNNs differ in their convolutional filter dimensions. The standard CNN employs n -dimensional square filters $\kappa \in \mathbb{R}^{n \times n}$ to identify spatial/temporal features within $\mathbf{V}_i$ whereas the other CNN that employs rectangular filters $\kappa \in \mathbb{R}^{n \times L_{emb}}$. These rectangular kernels are synonymous to n -gram modeling since each filter is applied onto the complete vector length of the embedded vectors. These CNNs are denoted by CONV and nCONV, respectively. The LSTM model follows a standard two-layer configuration with one hundred hidden units each. In addition to the four baseline models, an additional model that analyzes only the non-cognitive covariates is included. This model also employs a three-layer MLP. However, due to its smaller input dimensions, the number of hidden units is reduced compared to FREQ.

For the proposed joint-space grade prediction model, DNNs and MLPs are employed to encode the clickstream sequences and non-cognitive covariates, respectively. Six proposed methods considered are derived from a combination of three DNNs (CONV, nCONV and LSTM) and two joint embedding (PS and MS). The corresponding models are denoted by CONV-PS, CONV-MS, nCONV-PS, nCONV-MS, LSTM-PS and LSTM-MS, respectively. Architectures of all evaluated models and their parameters are summarized in Fig. 6.2 and 6.3.

Figure 6.3: Overview of the baseline frequency-based models (left) and baseline deep neural models (right).

Figure 6.4: Average model loss at each learning rate for the proposed Bayesian-regularized sequence encoder at input sequence length $L_x = 20$. The optimal learning rate is chosen from the region with the highest rate of decrease of model loss.

6.3.3 Model implementation

All models are implemented on Tensorflow 1.4 and trained on a GPU-accelerated machine. Hyper-parameters for each model was initialized using the Xavier initialization method [264]. For each model, an optimal learning rate was determined, similar to the systematic approach proposed in [265]. In this approach, the learning rate is first initialized with a small value, e.g., 1×10^{-7} . The model loss is computed, and the hyper-parameters are updated subsequently. The learning rate is then increased exponentially, and this process is repeated until the learning rate reaches to the upper limit considered, e.g., 10. This systematic approach is repeated for ten random initializations to account for perturbations due to the random parameter initialization, and the average model loss is depicted in Fig. 6.4. An optimal learning rate for the model is then determined from the region where the model experienced the highest rate of decrease in model losses. In this illustrative example, the optimal learning rate can be identified as 0.7×10^{-4} . Using the identified optimal learning rate, each grade prediction model is then trained using a student-stratified 70/30% training/testing split for 200 epochs and the Adam optimizer [266] .

6.4 Results and discussion

6.4.1 Frequency-based versus sequence-based grade prediction model

Performance of grade prediction models was evaluated via the mean absolute error (MAE) that is given by

$$\begin{aligned} \text{MAE} &= \frac{1}{N} \sum_{i=1}^N |\hat{\mathbf{y}}_i - \mathbf{y}_i| \\ &= \frac{1}{N} \sum_{i=1}^N \sum_{k=1}^{L_y} |\hat{y}_i^k - y_i^k|, \end{aligned} \quad (6.8)$$

and root mean-squared error (RMSE), that is given by

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^N \|\hat{\mathbf{y}}_i - \mathbf{y}_i\|^2}, \quad (6.9)$$

where $\mathbf{y}_i = [y_i^1, y_i^2, \dots, y_i^{L_y}]$ is defined as a vector containing the actual grades associated with the i th input sequence, and $\hat{\mathbf{y}}_i$ is the predicted value of $\mathbf{y}_i$. When being trained with non-cognitive covariates alone, PSYCH achieves a RMSE score of 0.299. This is similar with most of the frequency- and sequence-based models that were trained on weekly clickstream sequences. This corroborates existing studies that suggest that non-cognitive variables, like learning behaviours, are correlated with learning outcomes. Among models trained using weekly sequences, CONV achieved the lowest RMSE of 0.247 while the highest RMSE of 0.290 is achieved by nCONV. This is surprising since these two models only differ by their feature kernels. The use of rectangular kernels may be restricted the performance of nCONV since more kernels are required (compared to CONV) to learn the same amount of information from the input feature space. Since LSTM encode the input sequences onto its feature space, it is expected to achieve better performance than CNN-based models. However, LSTM only achieved moderate performance that is slightly better than CONV and comparable to FREQ.

Table 6.2: Grade prediction performance using weekly clickstream sequences and complete historical clickstream data

Model Type	Model	Weekly sequences		Complete history	
		MAE	RMSE	MAE	RMSE
Non-cognitive covariates	PSYCH	0.267±0.135	0.299	-	-
Frequency-based	FREQ	0.221±0.155	0.270	0.162±0.120	0.201
Sequence-based	CONV	0.207±0.135	0.247	0.150±0.143	0.206
	nCONV	0.232±0.149	0.276	0.161±0.131	0.207
	LSTM	0.224±0.155	0.273	0.185±0.124	0.222

Table 6.3: Grade prediction performance using weekly clickstream sequences and non-cognitive covariates

Model Type	Weekly sequences		Complete history	
	MAE	RMSE	MAE	RMSE
CONV-PS	0.217±0.129	0.253	0.179±0.132	0.221
CONV-MS	0.196±0.133	0.237	0.731±0.402	0.832
nCONV-PS	0.216±0.173	0.277	0.160±0.129	0.205
nCONV-MS	0.256±0.178	0.312	0.759±0.315	0.821
LSTM-PS	0.197±0.142	0.243	0.161±0.121	0.201
LSTM-MS	0.205±0.108	0.231	0.250±0.205	0.322

When the analysis window is increased to include all the teaching weeks, FREQ achieved the lowest RMSE of 0.201 while sequence-based models achieved an average RMSE of 0.267. However, a further analysis on FREQ reveals that the model collapse into singularity, mapping all the outcomes onto the same prediction. This implies that frequency-based models are sensitive to the analysis window. When the window length is increased, learner activities start to appear similar and, as a consequence, the models are unable to discern learning outcomes (c.f. Chapter 7). On the contrary, both CONV and LSTM achieved better performance than FREQ when trained with complete historical inputs and better than when being trained on weekly clickstream sequences.

6.4.2 Use of non-cognitive variables as prediction regulators

To evaluate the ability of non-cognitive variables to regulate clickstream grade prediction models, performance of models trained on both clickstream data and non-cognitive variables are evaluated via MAE and RMSE, as tabulated in Table 6.3. While grade prediction improvements are not consistent across all evaluated models, LSTM-PS and

LSTM-MS are able to achieve a lower RMSE of 0.243 and 0.231, respectively. This is equivalent to a 11.0% and 15.4% reduction in RMSE compared to its counterpart trained without the non-cognitive covariates. The CONV-MS model also achieves a similar RMSE of 0.237, or an improvement of 4% improvement in prediction performance. This improvement in prediction performance indicates that, to some extent, non-cognitive variables can aid grade prediction models in differentiating similar clickstream sequences. On the other hand, models with MS achieved higher RMSE for complete historical inputs while the models using PS improved in prediction performance. This implies that input clickstream sequences and non-cognitive covariates contributes independently to the grade prediction model rather than working in conjunction with the other. Nevertheless, the RMSE scores obtained from these models are high and may not be suitable for real-world applications. Chapter 7 explores one of the limitations that result in this high RMSE score and a solution is proposed to overcome this limitation.

6.5 Conclusion

In this chapter, performance of grade prediction models trained on frequency-based features is compared to that of models that are trained using sequential approaches. Furthermore, to incorporate learner non-cognitive traits to improve grade prediction model, a joint-space grade prediction algorithm is proposed. Two joint-space grade prediction schemes are proposed to fuse transient clickstream behavior and persistent non-cognitive traits. The proposed model is evaluated on a dataset that is collected from a DSP course. Results show that frequency- and sequence-based grade prediction models achieved similar performance when trained on weekly clickstream data. However, when the analysis window is increased, performance of frequency-based models deteriorates rapidly while sequence-based models can achieve similar performance. Furthermore, the results also show that the use of non-cognitive covariates is able to regulate clickstream grade prediction performance. This corroborates the fact that learning outcomes are affected by both non-cognitive traits and learning behaviors.

Chapter 7

Characterization of Fuzziness in Education Data using Deep Neural Networks

In Chapter 5, the sequential grade prediction task is reviewed. However, it is also well-known that performance of sequential clickstream grade prediction models (often averaging at three letter grade difference) is limited despite the use of deep neural networks (DNNs) and an increase in available training data. It is posited that performance of grade prediction models are limited by the multi-valued (MV) nature of education dataset. One of the cause for the MV characteristic of education data is due to the insufficiency of information. This is because learning is often regulated by cognitive variables such as working memory, inductive reasoning ability, information processing speeds, and non-cognitive variables such as grit and learning efforts. These variables determine the rate at which learners synthesize knowledge, yet are not quantifiable in the case of distance learning and e-learning (relative measures can be obtained via self-reported instruments). In this chapter, the detrimental impact of MV inputs on grade

Part of this chapter has been published as Kelvin H. R. Ng, S. Tatinati, and Andy W. H. Khong, “Characterization of fuzziness for grade prediction using deep neural networks,” in *Companion Proc. 9th Int. Learn. Anal. Knowl. Conf., LAK 2019*, Tempe, AZ, USA, May 4-8, pp. 6458-6462 and Kelvin H. R. Ng, S. Tatinati and Andy W. H. Khong, “Grade Prediction From Multi-Valued Click-Stream Traces via Bayesian-Regularized Deep Neural Networks,” in *IEEE Trans. Signal Process.*, vol. 69, pp. 1477-1491, 2021.

prediction is first highlighted. The proposed MV identification module that employs a DNN to quantify the MV characteristic of the dataset is then presented. As opposed to existing DNN-based grade prediction models, the proposed algorithm is able to capture the uncertainty associated with each click-stream sequence. This allows the model to organize its feature space such that the MV inputs can be differentiated from non-MV inputs. With the newly formulated surrogate measures, and by removing these identified MV inputs, experimental results show that the proposed models achieve lower prediction error compared to existing algorithms when evaluated on two online-learning datasets.

7.1 Introduction

Conventional enrollment-based small private online courses (SPOCs) offered by institutions and individual modules within thematic massive open online courses (MOOCs) often offer little flexibility in their curricula. While learners are expected to complete the learning activities in a prescribed order, albeit options being made available for learners to skip some learning activities, it is, however, important to note that external factors such as an individual’s efficacy to synthesize knowledge and revision efforts to enhance recall skills can result in different learning outcomes across learners undergoing the same online course. As such, while most learners perform activities that lead to similar click-stream sequences, they achieve contrasting grades. This phenomenon of similar online learning behavior resulting in different learning outcomes is termed as the MV problem.

Existing grade prediction methods such as deep learning techniques are optimized to achieve the best linear unbiased estimators of learners’ grades, i.e., the conditional expectation given a clickstream sequence. Since MV clickstream sequences are similar but associated with a diverse set of grades, the conditional expectation of these sequences may not be close to the desired estimates. Consequently, these predictive models are prone to erroneous grade estimates. This is undesirable since falsely identifying top-performing learners as at-risk could hurt morale while allowing at-risk learners to go undetected would incur additional cost within the education system [267]. While various algorithms have been proposed to estimate prediction confidences explicitly [204, 243, 268–270], these

Figure 7.1: Overview of the standard grade prediction model.

uncertainties are computed after the relationship between input sequences and target grades has been established. As such, features identified by these models are agnostic to underlying properties that generate these uncertainties, and hence unable to differentiate input sequences that resulted in these uncertainties. In order to highlight the impact of MV inputs, an illustrative example is first presented.

7.2 Multi-valued nature of sequential clickstream grade prediction

A typical grade prediction model that is based on clickstream sequences includes two main phases—feature extraction and output modeling—as depicted in Fig. 7.1. The feature extraction process first maps the discrete clickstream sequences into continuous numerical vectors via an embedding layer. Arbitrary features representing learning behaviors that can differentiate grades obtained by students who had performed these actions are then extracted and refined using a series of neural feature extractors such as the long short-term memory (LSTM) and fully-connected (FC) layers. These feature vectors are finally mapped onto the target grades via the output modeling process. Using least square optimization methods, these grade prediction models can be described in terms of its prediction error—defined as the difference between the actual and target grades—during the training phase. In the testing phase, these expected errors are interpreted as uncertainties associated with each prediction.

Grade prediction models based on DNNs are naturally associated with modeling un-

Figure 7.2: Identifying identical input sequences.

certainties due to numerical precisions of machines [271]. Since each computational layer operates as a deterministic operator, the DNN maps *identical* input sequences into the same feature vector, and therefore predicts the same grade. For input sequences that are *similar*, the DNN maps these input sequences into similar feature vectors if it does not recognise their differences as features that would differentiate their grades. This implies that target grades (corresponding to similar feature vectors) are expected to be similar for existing DNNs to perform well. When the above condition is not satisfied (such as for datasets that contain MV input sequences), predicted grades would deviate away from their target grades resulting in erroneous predictions.

7.2.1 Data perspective of the multi-valued problem

The existence of MV input sequences and its effect on grade prediction is illustrated via the dataset as described in Section 6.3.1.

To visualize the MV characteristic, grades corresponding to any two identical input sequences were extracted from the dataset to form a pair of grades. A pair of input sequences were considered to be identical if the composition and order of their leading sub-sequences were the same. For instance, given three exemplar input sequences (corresponding to that of a video content) in Fig. 7.2, all three sequences are considered as identical when the first three actions are considered, whereas only Sequences A and B are identical for the first four actions. This approach of identifying identical sequences is referred to as the truncation pre-processing method.

Six-hundred pairs of grades are extracted from the dataset when the length of trun-

Figure 7.3: Grade pair distribution of identical input sequences in an undergraduate digital signal processing course. Each location on this plot represents the grades achieved from two identical action sequences (each of length 50). The shade denotes the number of instances for each grade pair.

cated input sequences is taken as 50. These six-hundred grade pairs with identical input sequences are plotted on a two-dimensional histogram, using each grade as the ordinate and abscissa axes, as shown in Fig. 7.3. In this figure, the shades represent the total number of input-pairs that are associated with each grade pair. For example, the point highlighted in Fig. 7.3 represents seventeen pairs of input sequences which are identical but resulted in two different quiz grades—65% and 95% score. Therefore, for input sequences that are identical but do not exhibit any MV characteristic, these data points will be distributed along the diagonal. It can be noted that the data points are distributed over a wide range of grades between 20% and 100%, and the majority of these pairs can be found in the higher grade range. More importantly, out of the six-hundred pairs of identical input sequences, 87% of these identical action sequences result in at least 5% grade differences (corresponding to a grade band in most assessment criteria). Although these input sequences correspond to 13.8% of the input sequences within the complete dataset, this proportion of identical sequences leading to vastly different outcomes is not surprising. This is because while materials are generally delivered in a rigid linear cur-

Figure 7.4: Proportion of clickstream sequences with $>5\%$ grade differences within the truncated (solid) and augmented (dotted) dataset at various L_x .

Figure 7.5: Sliding window pre-processing data augmentation where the original action sequence is segmented into sequence each of length L_x .

riculum in an SPOC, learning efficacy is very much individualized, giving significantly different assessment outcomes.

The extent of MV characteristic at various input sequence length is also evaluated. The proportion of identical sequence-pairs having $>5\%$ grade difference for various sequence length L_x determined by the truncation method is shown in Fig. 7.4. As expected, this proportion decreases with increasing L_x given that with increasing L_x , learning behaviors vary across learners leading to a reduction in the number of identical sequences. While the above trend is observed, it is important to note that at least 13.8% of the identical sequence-pairs suffer from the MV characteristic, i.e., they have $>5\%$ grade differences. Many of these sequences correspond to learners watching multiple videos without interacting with the video player and to those who skipped through multiple videos towards the later parts of the course.

As opposed to data truncation, data augmentation is often used to deconstruct a learning activity sequence into smaller overlapping blocks to capture the short-term transient clickstream behaviors. To demonstrate the existence of MV characteristics in augmented data, a sliding window (each of length L_x) as illustrated in Fig. 7.5 is used to extract overlapping sequences. All the extracted sequences are associated with the grade that was originally associated with the entire learning activity. The proportion of identical sequence-pairs (generated by data augmentation) with $>5\%$ grade differences at various L_x is plotted as a dotted line shown in Fig. 7.4. As can be seen, the proportion of identical sequence-pairs exhibiting the MV characteristic reduces initially before increasing with L_x . This is because although there is a decrease in the number of identical sequence-pairs with $>5\%$ grade differences for larger L_x , the total number of input sequences (that comprise both MV and non-MV sequences) in the augmented dataset also decreases. Since the reduction in the total number of input sequences outweighs the reduction in the identical sequence-pairs with $>5\%$ grade differences, a modest increase in proportion of identical sequence-pairs with $>5\%$ grade differences within the augmented dataset is exhibited. Note that the MV characteristic is present for any L_x . Furthermore, although the proportion is low for some L_x , the impact of MV characteristic on the predictive performance of existing machine learning techniques can be significant, as will be illustrated below.

7.2.2 Model perspective of the multi-valued problem

Performance of grade prediction models was evaluated on the dataset in two scenarios—on the whole dataset and after removing input sequences that are known to exhibit the MV characteristic (MV inputs). For the latter, two naive procedures were employed to determine if input sequences exhibit the MV characteristic. The first procedure identified input sequences that belong to identical sequence-pairs with $>5\%$ grade differences as MV inputs, and after removing these MV inputs, the resultant dataset is referred to as the dataset *without identical* (WOID) sequences. For the second procedure, a more comprehensive set of MV inputs were identified. Since two input sequences are considered *similar* if a model does not recognize their differences as features that would differen-

Figure 7.6: Training and testing performances of existing grade prediction models trained using the complete dataset, WOID, and WOSIM at various sequence lengths L_x .

tiate its prediction, these similar sequences would, therefore, exhibit MV characteristic if the grade difference between them is greater than 5%. A formal definition based on this description will be presented in Section 7.3. To identify such sequences, a two-layer LSTM neural network was employed to encode the input sequences into feature vectors. The k -nearest neighbours (KNN) clustering algorithm was then applied on the feature vectors. For this analysis, the grade of an input sequence is compared to its most similar neighbor. The resultant dataset after removing these similar input sequences with a $>5\%$ grade difference is referred to as the dataset *without similar* (WOSIM) sequences.

Three sequential grade prediction models [166] were trained on the complete dataset, WOID, and WOSIM at various L_x . Each model was trained on input sequences obtained from 70% of the students for 200 epochs using an appropriate learning rate selected using the modified Smith's learning rate finder (c.f. Section 6.3.3), and thereafter evaluated on the remaining 30% students. Defining N as the total number of input sequences within the dataset, and L_y as the total number of summative assessments, the models were evaluated using the root mean-squared errors (RMSE), as described in (6.9). Grade prediction performance of these models for various L_x is depicted in Fig. 7.6. It can be

Proposed MV-sensitive Grade Prediction Paradigm

Figure 7.7: Overview of the proposed MV-sensitive grade prediction model.

observed that while the algorithms perform well for the training dataset with increasing L_x , RMSE for the testing dataset remains high particularly for WOSIM. This is due to the higher proportion of similar MV sequences present in the dataset. Therefore, compared with models trained with the complete dataset and WOID, WOSIM is able to achieve lower training RMSE due to the removal of substantial number of MV instances. However, the removal of these MV sequences results in worse testing RMSE scores since there are now fewer number of samples for the model to generalize to any unseen samples. As with existing machine learning algorithms, such reduction in the number of training samples increases the RMSE. This implies that the lack of training samples undermines the potential improvement in performance brought about by the removal of MV sequences. This illustrative experiment shows that the detection and removal of MV sequences using direct methods such as WOID and WOSIM do not address the problem and this serves to motivate the development of a new algorithm described in this work. Hereinafter, subsequent experiments are performed on datasets that have been treated with data augmentation.

7.3 The proposed MV-sensitive grade prediction model

7.3.1 Problem formulation

Consider a learner who had engaged in a number of learning sessions. Upon application of data augmentation, as described in Section 7.2.1, learning sequences from individual learning sessions are deconstructed into one or more sequences of length L_x . Consequently, each of these clickstream (input) sequences is denoted by $\mathbf{x}_i = [x_i^1, x_i^2, \dots, x_i^{L_x}]$ with x_i^k representing the k th action of the i th input sequence. Throughout the online course, the learners attempted L_y number of summative assessments (quizzes and exams). The assessment grades obtained by a learner who had performed $\mathbf{x}_i$ is denoted by $\mathbf{y}_i = [y_i^1, y_i^2, \dots, y_i^{L_y} | y_i^k \in [0, 1]]$. In general, $L_x \gg L_y$, and the complete dataset of sequence-grade pairs is, therefore, denoted by $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^N$, where N denotes the total number of sequence-grade pairs across all learners within the dataset.

A grade prediction model is employed to learn the underlying relationship between $\{\mathbf{x}_i\}$ and $\{\mathbf{y}_i\}$. This relationship is described as a cascaded function $f = f_2 \circ f_1$, where f_1 transforms each input sequence $\mathbf{x}_i \in \mathbb{R}^{L_x}$ into a hidden feature vector $\mathbf{h}_i \in \mathbb{R}^{L_{seq}}$, and these hidden feature vectors are subsequently mapped into outputs $\hat{\mathbf{y}}_i \in \mathbb{R}^{L_y}$ by f_2 . The MV characteristic of an input sequence can, therefore, be identified through the relationship between the hidden features vectors and its target grades that is modeled by f_2 . Formally, an input sequence $\mathbf{x}_\alpha$ is said to exhibit the MV characteristic due to another input sequence $\mathbf{x}_\beta$ if they satisfy the MV criterion defined in Definition 1.

Definition 1 (Multi-valued criterion) *Given two input sequences $\mathbf{x}_\alpha$ and $\mathbf{x}_\beta$, and a model that maps input sequence $\mathbf{x}_\alpha$ into hidden feature vector $\mathbf{h}_\alpha$, $\mathbf{x}_\alpha$ is said to exhibit MV characteristic with respect to $\mathbf{x}_\beta$ if the distance between their respective hidden feature vectors $\mathbf{h}_\alpha$ and $\mathbf{h}_\beta$ is less than the representation threshold ϵ_h , and that the difference between their target grades $\mathbf{y}_\alpha$ and $\mathbf{y}_\beta$ is greater than the output threshold ϵ_y . This is*

represented mathematically using a binary indicator function

$$\mathbb{1}_{MV}(\mathbf{x}_\alpha, \mathbf{x}_\beta) = \langle \mathbf{h}_\alpha, \mathbf{h}_\beta \rangle < \epsilon_h \wedge \langle \mathbf{y}_\alpha, \mathbf{y}_\beta \rangle > \epsilon_y, \quad (7.1)$$

where $\langle \cdot, \cdot \rangle$ denotes a distance measure between two vectors, and $\wedge$ denotes the logical AND operator.

The set of inputs that has been identified by model f to exhibit the MV characteristic at thresholds ϵ_h and ϵ_y is therefore given by

$$\mathbf{X}^{MV} = \left\{ \mathbb{1}_{MV}(\mathbf{x}_j, \mathbf{x}_k) \otimes \mathbf{x}_j \right\}, \quad (7.2)$$

where $\mathbf{x}_j$ and $\mathbf{x}_k$ are input sequences in $\mathcal{D}$, and $\otimes$ denotes the Hadamard product. While Definition 1 has been used to identify similar input sequences exhibiting MV characteristic in Section 7.2.2, ϵ_h was determined implicitly by the KNN algorithm.

Since the target grades are not available when predicting for unseen input sequences, MV and non-MV inputs have to be mapped onto different feature sub-spaces by the model (during the training phase) so as to discriminate between themselves. However, existing deep learning techniques optimized via least square methods generate a model that maps input sequences with similar target outputs onto the same feature sub-space to minimize the average prediction error. Consequently, input sequences that resulted in higher prediction errors (such as the MV inputs) are uniformly distributed throughout the feature space. To achieve good MV identification performance, a model that captures the prediction uncertainty of the input sequences is proposed so that an additional degree of freedom is provided within its feature space. This, in turn, allows the MV and non-MV inputs to be mapped onto different feature sub-spaces, making them differentiable. Furthermore, the effectiveness of a model to identify MV inputs is dependent on the classifier employed to learn a decision boundary that separates these two sub-spaces. Unlike conventional prediction problems where the objective is to organize a feature space such that grades can be differentiated, the proposed model aims to achieve a feature space that differentiates and identifies the MV inputs. Such identification and subsequent removal of these MV inputs is important as a pre-processing technique to

achieve good prediction performance.

7.3.2 The MV-sensitive grade prediction model

The proposed MV-sensitive grade prediction model comprises an MV identification module and a grade prediction model, as depicted in Fig. 7.7. The MV identification module is constructed via a Bayesian-regularized sequence encoder and an MV classifier, where the former is used to determine the MV characteristic. It consists of an embedding layer, multiple LSTM and FC layers, and a Bayesian-regression layer. Given a sequence of clickstream traces $\mathbf{x}_i$, the embedding layer first maps each x_i^k into L_{emb} -dimensional continuous vectors via a look-up table $T = \{x_i^k : \mathbf{v}_i^k \in \mathbb{R}^{L_{\text{emb}}}\}$, where $\mathbf{v}_i^k$ is randomly initialized according to any probabilistic distribution. To extract temporal information from the embedded input $\mathbf{V}_i = \{\mathbf{v}_i^k\}_{k=1}^{L_x}$, multiple LSTMs are employed to encode these inputs onto a feature space described by hidden feature vectors

$$\mathbf{h}_i = \text{LSTM}(\mathbf{V}_i), \quad (7.3)$$

where $\text{LSTM}(\cdot)$ has been defined in Section 5.2.1. Since high-dimensional hidden feature vectors result in excessive computational load to predict outputs of lower dimensionality, FC layers are subsequently employed to reduce $\mathbf{h}_i$ into hidden feature vectors of lower dimension

$$\mathbf{z}_i = \text{FC}(\mathbf{h}_i). \quad (7.4)$$

To capture the underlying relationship between $\mathbf{z}_i$ and the target grades $\mathbf{y}_i$ as well as the uncertainties surrounding each prediction $\hat{\mathbf{y}}_i$, the proposed Bayesian-regression layer estimates a mean and standard deviation (SD) component for each $\mathbf{z}_i$. Similar to mixture density networks [268], the mean and SD components are estimated independently using FC layers with different activation functions, and are, respectively, given by

$$\boldsymbol{\mu}_i = \mathbf{W}_\mu^\top \cdot \mathbf{z}_i + b_\mu, \quad (7.5)$$

and

$$\sigma_i = \exp(\mathbf{W}_\sigma^\top \cdot \mathbf{z}_i + b_\sigma), \quad (7.6)$$

The terms $\{\mathbf{W}_\mu, b_\mu\}$ and $\{\mathbf{W}_\sigma, b_\sigma\}$ denote the set of weights and biases for the mean and SD components, respectively. An exponential activation function is applied to the SD estimate in (7.6) to ensure that the estimate is non-negative. The predicted grade can then be expressed as a Gaussian distribution

$$\hat{\mathbf{y}}_i = \mathcal{N}(\boldsymbol{\mu}_i, \sigma_i^2 \mathbb{I}), \quad (7.7)$$

where $\mathbb{I}$ denotes an identity matrix of size L_y . The sequence encoder is optimized by minimizing the empirical negative log-likelihood

$$\mathcal{L}_{\text{NLL}} = -\frac{1}{N} \sum_{i=1}^N \log \mathcal{N}(\mathbf{y}_i | \boldsymbol{\mu}_i, \sigma_i), \quad (7.8)$$

where $\mathcal{N}(\mathbf{y}_i | \boldsymbol{\mu}_i, \sigma_i)$ denotes the conditional probability of $\mathbf{y}_i$ given $\boldsymbol{\mu}_i$ and σ_i .

As opposed to [204, 243] where the prediction uncertainties are inferred via the model predictions, and that these uncertainties do not affect the model, the proposed Bayesian-regularized sequence encoder employs σ_i to organize a feature space suitable for the identification and discrimination of MV inputs. In addition, while σ_i is jointly optimized with $\boldsymbol{\mu}_i$ in the proposed algorithm, both components capture different aspects of the underlying relationship between the input sequences and the target grades. Since $\boldsymbol{\mu}_i$ denotes the conditional expectation given an input sequence, this enables the proposed algorithm to differentiate similar input sequences with different target values. Consequently, $\boldsymbol{\mu}_i$ is, to a large extent, correlated with the target grades. On the other hand, σ_i quantifies the deviation between its mean component and the target grades. Unlike $\boldsymbol{\mu}_i$, σ_i is dependent on the set of input sequences that generates the prediction, which, in turn, determines their prediction errors. With σ_i and $\boldsymbol{\mu}_i$ learning underlying relationships that are different, feature dimensions that are engaged to model $\boldsymbol{\mu}_i$ are expected to be uncorrelated with, and therefore orthogonal to, those for σ_i . This allows the pro-

posed model to map input sequences with varying degree of predictive uncertainties onto different feature sub-spaces to facilitate the identification of MV inputs.

The MV classifier sub-module identifies MV inputs through $\mathbf{z}_i$ that are extracted from the Bayesian-regularized sequence encoder. This is achieved by employing a binary classifier to learn the non-linear decision boundary within the organized feature space. Support vector machines (SVMs) with radial basis function kernels [272] are chosen for their good performance and robustness towards noisy data. Since the MV characteristic of the dataset is unknown, the true labels of the data points are not readily available to train the classifier. Motivated by the fact that the presence of MV inputs leads to large prediction errors and uncertainty in modeling (SD component), two surrogate measures are proposed to label the data points during training. The first surrogate measure employs the absolute prediction error (AE) from the Bayesian-regularized sequence encoder to determine if an input sequence exhibits MV characteristic. An input sequence is deemed to exhibit the MV characteristic if its AE is greater than a pre-defined threshold ϵ_y , i.e.,

$$\lambda_i^{MV} = \begin{cases} 1 & \text{MV} & \langle \mathbf{y}_i, \hat{\mathbf{y}}_i \rangle > \epsilon_y; \\ 0 & \text{non-MV} & \text{otherwise.} \end{cases} \quad (7.9)$$

Similarly, the second surrogate measure identifies an input sequence as an MV input if the amplitude of its SD component is greater than a pre-defined threshold ϵ_y , i.e.,

$$\lambda_i^{MV} = \begin{cases} 1 & \text{MV} & |\boldsymbol{\sigma}_i| > \epsilon_y; \\ 0 & \text{non-MV} & \text{otherwise,} \end{cases} \quad (7.10)$$

where $|\boldsymbol{\sigma}_i|$ denotes the vector norm of $\boldsymbol{\sigma}_i$. These two surrogate measures are highly correlated with $0.82 \leq r \leq 0.88$.

The SVM is trained to learn a decision boundary that separates the MV inputs from the non-MV inputs, determined either from (7.9) or (7.10), in the feature space. Thereafter, the SVM is employed to predict the MV label for each input based on the learned

Algorithm 2: MV Identification Module

Input: Dataset $D = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^N$, initial model parameters θ
Output: Bayesian-regularized sequence encoder f , MV classifier MVClass
Train f :
for *epoch* **from** 1 **to** 200 **do**
 forall $\mathbf{x}_i$ **do**
 $\mathbf{h}_i \leftarrow f_1(\mathbf{x}_i)$
 $\mathbf{z}_i \leftarrow f_2(\mathbf{h}_i)$
 $\boldsymbol{\mu}_i, \boldsymbol{\sigma}_i = \text{BayesianRegressionLayer}(\mathbf{z}_i)$
 $\text{NLL} = -\log N(\mathbf{y}_i | \boldsymbol{\mu}_i, \boldsymbol{\sigma}_i)$
 $\theta \leftarrow \theta - \frac{\delta \text{NLL}}{\delta \theta}$
 end
end
Determine MV labels:
forall $\mathbf{x}_i$ **do**
 $\hat{\mathbf{y}}_i, \mathbf{z}_i \leftarrow f(\mathbf{x}_i)$
 switch *Surrogate measure type* **do**
 case *AE* **do**
 if $\langle \mathbf{y}_i, \hat{\mathbf{y}}_i \rangle > \epsilon_y$ **then**
 $\lambda_i^{MV} \leftarrow 1$
 else
 $\lambda_i^{MV} \leftarrow 0$
 end
 case *SD* **do**
 if $|\boldsymbol{\sigma}_i| > \epsilon_y$ **then**
 $\lambda_i^{MV} \leftarrow 1$
 else
 $\lambda_i^{MV} \leftarrow 0$
 end
 end
end
MVClass $\leftarrow \text{SVM}(\{(\mathbf{z}_i, \lambda_i^{MV})\}_{i=1}^N)$

decision boundary. This is mathematically expressed as

$$\hat{\lambda}_i^{MV} = \text{SVM}(\mathbf{z}_i). \quad (7.11)$$

Denoting the model parameters $\{\mathbf{W}_{\text{LSTM}}, \{\mathbf{v}_i^k\}, \mathbf{W}_\mu, b_\mu, \mathbf{W}_\sigma, b_\sigma, \mathbf{W}_{\text{FC}}, b_{\text{FC}}\}$ collectively by θ , Algorithm 2 provides a formal description of the proposed MV identification module.

Algorithm 3: MV-sensitive Grade Prediction model

Input: Input sequence $\mathbf{x}_i$, Bayesian-regularized sequence encoder ENC, MV classifier MVClass, non-MV grade prediction model GP
Output: Predicted Grade $\hat{\mathbf{y}}_i$
 $\mathbf{z}_i \leftarrow \text{ENC}(\mathbf{x}_i)$
 $\hat{\lambda}_i^{MV} \leftarrow \text{MVClass}(\mathbf{z}_i)$
if $\hat{\lambda}_i^{MV} = 1$ **then**
 | No prediction performed
else
 | $\hat{\mathbf{y}}_i \leftarrow \text{GP}(\mathbf{x}_i)$
end

7.3.3 MV-sensitive Grade prediction model

The proposed MV-sensitive grade prediction model generates the predictions for the input sequences according to the outcome of the MV identification module. Depending on the MV classification outcomes, one of two actions is performed:

- For the set of input sequences that are labeled as MV, i.e., $\hat{\lambda}_i^{MV} = 1$, the grades are not predicted since their predictions are expected to be erroneous.
- For the remaining non-MV input sequences, an additional grade prediction model GP is trained to perform the grade prediction for the non-MV inputs, i.e.,

$$\hat{\mathbf{y}}_i = \text{GP}(\mathbf{x}_i). \quad (7.12)$$

In this context, any existing grade prediction model or the proposed Bayesian-regularized sequence encoder can be used for GP.

The MV-sensitive grade prediction model is provided in Algorithm 3.

7.4 Evaluation framework

7.4.1 Datasets

Performance of the proposed MV-sensitive grade prediction model was evaluated on two datasets. The DSP dataset has already been presented in Section 6.3.1.

In addition to the DSP dataset, the proposed MV-sensitive grade prediction model is evaluated on the 2019 Learning Analytics and Knowledge Data Challenge (LAKDC) dataset. The LAKDC dataset was collected from an e-book platform accessed by 1545 learners across fifteen classes conducted within three tertiary institutions [273, 274]. An e-portfolio learning management system (LMS) was introduced for the purpose of improving and recommending learning materials, understanding how learners utilize these digital media and their knowledge, and predicting learning outcomes within the courses. Within this e-book platform, clickstream traces are logged as learners interact with the digital reading media. As tabulated in Table 7.1, a total of nineteen types of clickstream traces that include navigating and annotating of reading materials, inserting and removing bookmarks, and navigating through the use of keyword searches are recorded. The final course grades of each learner are also available within this dataset. The objective of this dataset is, therefore, to predict a learner’s final grades based on their interactions with these digital reading materials. A total of two million logs amounting to an average of 1304 traces per learner were recorded.

7.4.2 Performance comparison

Classification performance was evaluated across five MV model configurations constructed using a combination of two sequence encoders and three MV classifiers. The proposed Bayesian-regularized (BR) sequence encoder is compared with the baseline (BL) sequence encoder; they both share the same feature extraction architecture—a ten-dimensional embedding layer, two LSTM layers with fifty hidden units each, an FC layer with five hidden units, and an output layer. The output layer of the baseline BL sequence encoder consists of an FC layer and the encoder was optimized using the mean-squared

Table 7.1: List of logged actions for the LAKDC dataset.

Operation Types	Actions
Navigation	Open reading materials Close reading materials Navigate to the next page Navigate to the previous page Jump to a page Navigate via a link
Bookmark	Add bookmark Delete bookmark Jump to bookmark
Memo	Add memo Delete memo Change memo Jump to memo
Search	Search keyword Jump to searched keyword
Marker	Add marker (important) Delete marker (important) Add marker (difficult) Delete marker (difficult)

error (MSE), given by

$$MSE = \frac{1}{N} \sum_{i=1}^N \|\hat{\mathbf{y}}_i - \mathbf{y}_i\|, \quad (7.13)$$

where $\|\cdot\|$ denotes the L2-norm operator. On the other hand, the proposed BR sequence encoder models the underlying relationship between the feature vectors and the target outputs as defined by (7.5)-(7.7).

MV classifiers were evaluated for their ability to identify *identical* inputs and *similar* inputs that exhibit MV characteristics. The former is termed as inputs of known labels (KLs) since these inputs were identified using an exhaustive greedy search technique. On the other hand, input sequences that are similar and exhibiting MV characteristics were identified using the surrogate measures defined by (7.9) and (7.10). For these surrogate measures, the output threshold ϵ_y was set to the average of the training datasets, i.e., the mean absolute (training) error (AE) and the mean SD estimate. SVM classifiers

Table 7.2: Model configurations based on which performance was evaluated.

Model Config	MV Identification Module			Grade Prediction Model	
	Complete Dataset	BR-AE	BR-SD	BL ^a	BR ^b
GP1	✓			✓	
GP2		✓		✓	
GP3		✓			✓
GP4			✓	✓	
GP5			✓		✓

^a Vanilla LSTM sequence encoder

^b Proposed Bayesian-regularized sequence encoder

trained on KL and the two surrogate measures are denoted by SVM-KL, SVM-AE, and SVM-SD, respectively.

For compactness in notation, the baseline and proposed identification modules that employ BL and BR (as its sequence encoder) followed by SVM-KL (for the identification of known MV inputs) are denoted by BL-KL and BR-KL, respectively. For identification modules that employ the surrogate measures, the module that employed BL (to learn the mapping) and SVM with the AE surrogate measure (to identify the MV inputs) is denoted by BL-AE. Similarly, modules that employ the proposed BR sequence encoder with SVM and the AE/SD surrogate measures are denoted by BR-AE and BR-SD, respectively.

Prediction performance was evaluated across a baseline grade prediction model and four variants of the proposed MV-sensitive grade prediction model. The baseline grade prediction (GP) model that employs BL to encode the complete dataset is denoted by GP1 model configuration, whereas four variants of the proposed MV-sensitive grade prediction models are denoted by GP2-GP5 model configurations. Details for the various grade prediction models are summarized in Table 7.2. All models were trained on the training dataset that consists clickstream traces from 70% of the student population, and evaluated on the testing dataset, which contains the remaining clickstream traces using the methodology described in Section 6.3.3.

7.4.3 Performance measures

Classification performance of MV identification module was evaluated via the recall (R) and the area under operating characteristics curve (AUC) scores. The recall score determines the ability of MV identification modules to identify the known labels and can be expressed as

$$R = \frac{TP}{TP + FP}, \quad (7.14)$$

where TP and FP denote the true and false positive rates of the classifier, respectively. Therefore, R is bounded between 0 and 1, where $R = 1$ implies perfect identification of all MV inputs. Since R only accounts for TP (Type-I errors), it does not account for the FP that contributes to poor classification performances. The AUC score, on the other hand, is defined as the area under the curve when the true positive rate is plotted against the false positive rate for various probability thresholds. An AUC score of 0.5 implies that the proposed MV identification module performed no better than random chance in identifying the MV inputs, and any improvement in AUC score up to a maximum score of 1 implies more accurate identification of MV inputs.

Performance of grade prediction methods was evaluated via the MAE (defined in (6.8)) and RMSE (defined in (6.9)).

7.5 Results and discussion

7.5.1 Organization of feature space

To highlight the organization of feature space due to the proposed sequence encoder, feature spaces obtained from BL and BR for the DSP dataset with $L_x = 20$ are shown in Fig. 7.8. The figure is generated by first extracting $\{\mathbf{z}_i\}_{i=1}^N$ from the input sequences. For visualization, these hidden feature vectors were then projected into their principal components PC1 and PC2 identified via principal component analysis. Each input sequence is then represented as a point on the scatter plots constructed with these two principal

Figure 7.8: Visualisation of feature space representing (a) the predicted grades, (b) target grades, (c) identical input sequences with $>5\%$ grade differences, (d) absolute errors, and (e) SD measures for BL (left) and BR (right). Decision boundaries of the MV classifiers derived from SVM are illustrated in Figs. (c)-(e).

components as its axes.

Fig. 7.8(a) illustrates the actual (target) grades for each input sequence with a darker shade for 0% (represented as 0.0 in the figure legend) and a lighter shade for 100% score (represented as 1.0 in the figure legend). The left column plot shows results for the BL model, while the right column shows that for the proposed BR model. The predicted grades for the same input sequences in Fig. 7.8(a) is depicted in Fig. 7.8(b). From these figures, it can be inferred that the predicted grades are dependent on the feature sub-spaces that the input sequences are mapped onto since Fig. 7.8(b) appears as a smoothened plot of Fig. 7.8(a). For instance, predicted grades identified by the arrow in Fig. 7.8(b) can approximately be determined by the target grades of inputs that are mapped within its vicinity, as illustrated via the circle in Fig. 7.8(a). This implies that a model with a mapping function that is able to place similar input sequences with similar target grades together will achieve higher grade prediction accuracy. In the presence of MV inputs, however, a mixture of MV and non-MV inputs within the same proximity raises the average absolute error for these data points due to differences in their target grades. This phenomenon is apparent for the BL model, where identical inputs exhibiting MV characteristics (i.e., the known labels) are distributed across the feature space, as shown in Fig. 7.8(c)(left). It can be seen from this figure that the decision boundary (depicted as contour lines) learned by BL-KL covers a large segment of the feature space, with similar proportions of MV and non-MV inputs for both the positive and negative MV classes. Due to the widespread distribution of known labels (identical input sequences that exhibit MV characteristics), BL-KL is unable to learn a concise decision boundary that segregates MV inputs from non-MV inputs. While a boarder coverage by the decision boundary increases the recall performance, a high false positive rate and hence lower AUC scores can be expected. The absolute differences between the actual grades in Fig. 7.8(a) and predicted grades in Fig. 7.8(b) is shown in Fig. 7.8(d). As expected, due to the distribution of the known labels for BL, most data points have similar absolute differences between their actual and predicted grades. Therefore, similar to BL-KL, the decision boundary of BL-AE covers a large region of the feature space. In this case, since non-MV inputs may also result in higher prediction errors, BL-AE (that employs the surrogate measure) identifies more data points as MV inputs compared to

that of BL-KL.

Representation of input sequences formed by BL and the proposed BR model is different due to the Bayesian-regression output layer, which introduces an additional degree of freedom within the feature space. As observed from Fig. 7.8(c)(right), a large proportion of MV inputs (shown by the data points with darker shade) are mapped onto the same feature sub-space. Moreover, the decision boundary of the proposed BR-KL model clearly demarcates the neighbourhood that these MV inputs are mapped onto. The two distinct regions correspond to a region of low and high prediction errors, as shown in Fig. 7.8(d). The organization of the feature space placing predictions with varying degrees of absolute errors into different local neighbourhoods implies that information derived from prediction uncertainties is captured within the feature space by the model. In addition, the neighbourhood associated with higher prediction errors corresponds to the same neighbourhood that the known labels and inputs associated with higher SD components are mapped onto, as depicted in Fig. 7.8(c) and 7.8(e), respectively. The consistency between the feature spaces implies that the MV classifiers can identify MV inputs effectively using either the known labels or the surrogate measures. It is important to note that unlike absolute error, which considers the tuple of actual and predicted grades, the SD components of BR considers the actual grades across multiple data points. This allows the classifiers to learn a tighter class boundary, as can be seen through the steep boundary contours in Fig. 7.8(e), to better differentiate inputs that are associated with higher predictive uncertainties compared to a vague bound obtained by BR-AE and BR-KL. These plots highlight that the proposed BR-SD is able to organize the feature space such that MV inputs can better be differentiated from the non-MV inputs.

7.5.2 Classification performance

Classification performance of the MV identification methods was computed via recall and AUC scores on the DSP dataset, as tabulated in Table 7.3. When trained with known labels, both BL-KL and BR-KL are able to achieve good recall and AUC scores for all L_x . However, as a consequence of the organized feature space, BR-KL outperforms BL-KL for all L_x when evaluated on the unseen inputs. For instance, for $L_x = 35$, BL-

Table 7.3: MV classification performances for the DSP dataset

	Input Sequence Length L_x	Identical inputs (Known labels)		Surrogate Measures		
		BL-KL	BR-KL (Proposed)	BL-AE	BR-AE (Proposed)	BR-SD (Proposed)
Recall	10	0.753	0.787	0.78	0.757	0.695
	20	0.52	0.719	0.672	0.806	0.733
	35	0.684	0.921	0.697	0.934	0.947
AUC	10	0.659	0.742	0.615	0.741	0.74
	20	0.631	0.746	0.569	0.702	0.70
	35	0.76	0.843	0.659	0.814	0.819

KL and the proposed BR-KL model are able to achieve AUC scores of 0.76 and 0.843, respectively. The differences in classification performance between BL-KL and BR-KL is smaller at shorter L_x due to the higher proportion of MV inputs.

Classification performance of MV identification modules that employ the two surrogate measures are comparable to those that employ the known labels. Since the known labels are not readily available until the final exam, these surrogate measures serve as appropriate replacement measurements for identifying MV inputs. Furthermore, the proposed BR-AE and BR-SD models outperform BL-AE in terms of recall scores for most L_x , and AUC scores across all L_x . For example, BR-AE and BR-SD achieved AUC scores of 0.814 and 0.819, respectively, compared to 0.659 for BL-AE. These results highlight the effectiveness of the proposed surrogate measures to identify the MV inputs in place of the known labels. Although BL-AE achieves modestly higher recall score of 0.78 compared to the proposed MV identification models that employ the two surrogate measures for $L_x = 10$, the lower AUC score of 0.615 obtained for BL-KL imply that this method suffers from high false positive rate compared to the proposed BR-KL model.

Improvements in classification performance for the proposed BR model over BL are listed in Table 7.4. Overall, BR-SD is able to achieve an improvement of up to 35.87% and 24.28% for the recall and AUC scores over BL-AE, respectively. This significant improvement in classification performance implies that BR with the proposed surrogate measures are effective in identifying the known labels and, more importantly, for other

Table 7.4: Percentage improvement of MV classification performances for BR over BL (%).

	Input Sequence Length L_x	Known Labels	AE Surrogate Measure	SD Surrogate Measure
Recall	10	4.52	-2.95	-10.9
	20	38.27	19.94	9.08
	35	34.65	34	35.87
AUC	10	12.59	20.49	20.33
	20	18.23	23.37	23.02
	35	10.92	23.52	24.28

Table 7.5: Proportion of MV inputs identified from the DSP dataset (%).

Input Sequence length L_x	Known Labels	BR-AE	BR-SD
10	51.4	56.7	50.4
20	13.3	45.6	35.5
35	2.18	37.9	33.5

MV inputs that may not be known in advance. The proportions of MV inputs identified by these methods are summarized in Table 7.5. As expected, the proportion of known labels is similar to the proportion of MV inputs that is identified by BR-AE and BR-SD at shorter L_x . This is because of the large number of input sequences and the lower diversity of each sequence at these input sequence lengths. At higher values of L_x , while the known labels account for a smaller proportion of MV inputs, BR-AE and BR-SD are able to identify up to 32.3% more inputs over the known labels. These larger number of input sequences arise due to the relaxed conditions defined by the surrogate measures in (7.9) and (7.10) which identified similar inputs (as opposed to identical inputs in the known labels) that exhibit MV characteristic.

7.5.3 Grade prediction performance

Performance of the grade prediction models on the DSP dataset before and after the removal of MV inputs is depicted in Fig. 7.9 and tabulated in Table 7.6. Results

Figure 7.9: Regression performance in terms of MAE and RMSE for DSP for input sequence length of (i) 10, (ii) 20 and (iii) 35.

highlighted that all the proposed GP model configurations (GP2 to GP5) achieve lower MAEs compared to GP1. Furthermore, grade prediction models that employ SVM with the AE/SD surrogate measures as the MV classifiers achieve similar grade prediction performance. When considered individually, the learners have, on average, 39.3% of their generated input sequences deemed as non-MV inputs, with only one out of the forty-seven learners having none of their input sequences being non-MV. Performance improvement for the proposed grade prediction model over the baseline GP1 configuration, where the LSTM model has been applied to the complete dataset, is tabulated in Table 7.7. The largest RMSE improvement of 0.081 (40.79%) is observed when the Bayesian-regularized sequence encoder employs the AE surrogate measure (BR-AE) is used to identify the

Figure 7.10: Regression performance in terms of MAE and RMSE for LAKDC dataset for input sequence length of (i) 10, (ii) 20 and (iii) 35.

MV sequences and thereafter employing the BR model to perform grade prediction (GP3 configuration) for the case of $L_x = 10$. This improvement translates to a letter grade improvement in terms of the grade prediction errors. It is also useful to note that the improvements are more significant at shorter L_x due to the higher proportion of MV inputs.

7.5.4 External validation for grade prediction performances

To highlight the prevalence of MV inputs and the generalizability of the proposed model, performance of the proposed grade prediction models was also evaluated on the

Table 7.6: Grade Prediction Performances of GP1-GP5.

		Input Sequence Length L_x	Model Configuration				
			GP1	GP2	GP3	GP4	GP5
MAE	DSP	10	0.190	0.122	0.116	0.121	0.129
		20	0.213	0.178	0.183	0.161	0.171
		35	0.212	0.194	0.215	0.179	0.203
	LAKDC	10	0.0644	0.0678	0.0712	0.0725	0.0676
		20	0.0724	0.0628	0.0578	0.0623	0.0581
		35	0.0868	0.0612	0.0563	0.0606	0.0560
RMSE	DSP	10	0.227	0.153	0.146	0.152	0.177
		20	0.248	0.231	0.239	0.202	0.215
		35	0.250	0.256	0.261	0.235	0.271
	LAKDC	10	0.0828	0.0817	0.0852	0.0871	0.0812
		20	0.0911	0.0770	0.0725	0.0780	0.0727
		35	0.1070	0.0763	0.0735	0.0744	0.0700

2019 LAK Data Challenge dataset (LAKDC) dataset [273, 274]. Performance of the GP1-GP5 model configurations on the LAKDC dataset is plotted in Fig. 7.10, and the relative improvements in predictive performance with GP2-GP5 over GP1 is summarized in Table 7.7.

With reference to Fig. 7.10, the RMSE of the proposed algorithms (GP2-GP5) perform at least comparatively with the baseline algorithm (GP1) when L_x is very small or large. For small $L_x = 10$, a high number of MV sequences is present in the datasets. This is due to many learners performing the same set of actions since the course has been designed in a linear fashion and learners were not given the option to skip any sub-topics but achieving different grade outcomes. The subsequent removal of these large number of MV sequences result in insufficient training samples for the model to learn the complex underlying relationship between the click-stream sequences and the target grades. This leads to comparable performance with the baseline (to within the third decimal place) as can be seen in GP2-GP5 for the LAKDC dataset (also shown in Table 7.7). On the other hand, it is important to note that for large L_x , fewer number of samples are

Table 7.7: Improvement in Grade Prediction Performances with GP2 to GP5 over GP1 in terms of MAE and RMSE.

		Input Sequence Length L_x	Model Configuration			
			GP2	GP3	GP4	GP5
ΔMAE	DSP	10	0.068	0.074	0.069	0.061
		20	0.035	0.03	0.052	0.042
		35	0.018	-0.003	0.033	0.009
	LAKDC	10	-0.003	-0.007	-0.008	-0.003
		20	0.01	0.015	0.01	0.014
		35	0.026	0.031	0.026	0.031
ΔRMSE	DSP	10	0.074	0.081	0.075	0.05
		20	0.017	0.009	0.046	0.033
		35	-0.006	-0.011	0.015	-0.021
	LAKDC	10	0.001	-0.002	-0.004	0.002
		20	0.014	0.019	0.013	0.018
		35	0.031	0.034	0.033	0.037

available to the model. This is consistent with batch-based processing methods where the chunking of long sequences result in lower number of effective samples for the model. Therefore, for large L_x , while the proportion of MV instances is low (since there are now greater diversity in the actions generated by the learners), fewer number of data samples available to the model leads to the models failure to generalize well. This results in modest reduction in RMSE scores as can be seen in GP3 and GP5 for the DSP dataset. Both the above conditions imply that there is a tradeoff between the proportion of MV sequences present in the dataset and the number of data samples available for the model to generalize well. Nevertheless, compared to Fig. 7.6, where the direct removal of the MV sequences results in a worse performance than the complete dataset, the efficacy of the proposed algorithms is highlighted in Figs. 7.9 and 7.10.

The two datasets also comprise different proportions of identical MV inputs (known labels) and, therefore, the proportion of MV inputs identified by the proposed MV identification module differs. In particular, as can be seen in Figs. 7.9 and 7.10, the proportion

of known labels at $L_x = 10$ for the LAKDC dataset is 81% compared to 51.4% for the DSP dataset. When comparing similar proportion of known labels (such as when $L_x = 10$ for DSP and $L_x = 35$ for LAKDC), it can be observed that the improvement brought about by the proposed algorithm is similar. Therefore, while analyzing click-stream behaviors at shorter sequence length is attractive for early prediction, the high proportion of MV instances presents a challenge for accurate prediction of learning outcomes.

It is worthwhile to note, from these results, that the choice of model configurations and input sequence length may affect grade prediction performance. More importantly, the feasibility and reliability of training a grade prediction model may be pre-determined. Since the proposed MV-sensitive grade prediction model can handle up to approximately 80% MV inputs and yet achieve good predictive performance, it is recommended that L_x is set such that the identical sequences form less than 80% of the dataset to ensure good predictive performance. With the improved prediction performance achieved via GP2-GP5 model configurations, instructors can identify suitable intervention opportunities (such as targeted remedial for at-risk learners or recommending challenging problems to learners who are excelling) more effectively to improve learning experiences.

7.6 Conclusions

In this chapter, the notion of MV inputs in sequential clickstream datasets and its adverse impact on grade prediction performance is highlighted. Evidence for the MV phenomenon across various L_x is presented. The removal of potential MV inputs demonstrated potential for grade prediction performance improvement. To provide a systematic approach for identifying the MV characteristic, a MV identification model that employs a DNN with the proposed Bayesian-regularized output layer is proposed. The proposed output layer allows the DNN to incorporate prediction uncertainty within its feature space, and doing so allow it to differentiate MV inputs from non-MV inputs. To identify the MV inputs from the model's feature space, two surrogate measures derived from the training model are employed to label the inputs. SVM classifiers with radial basis function kernels are then employed to identify these MV inputs. The proposed model

is validated on two education clickstream datasets. Experimental results show that the identification, and subsequent removal, of these MV inputs improves grade prediction performances.

Chapter 8

Conclusions and Future Research Directions

8.1 Conclusions

In this thesis, the task of identifying different learning behaviors via data mining algorithms and forecasting of learning outcomes via DNNs have been considered.

Part I of this thesis focused on the problem of automatic identification of learning behaviors. Firstly, learning behaviors via accesses of online learning resources is considered. To differentiate resource accesses in terms of learners' preferences and efforts, a clustering framework based on the Gaussian mixture model (GMM) clustering algorithm and a set of heuristics is proposed. Differences in learning resource consumption are then described in terms of behavioral normalcy and effort. Compared to existing studies, the domain-agnostic data-driven approach to defining learning behavior preferences lends itself to be applied onto new courses without prior knowledge on the course. In the study, it was found that learners who spent effort during the recess break to revisit learning resources that would otherwise be irrelevant in coming teaching weeks were more likely to perform better in upcoming examinations. These results support the hypothesis that learners with higher ability to transfer knowledge tend to behave differently from their peers.

Secondly, problem-solving approaches undertaken by learners in an online elementary Mathematics course is analyzed. In this course, the curriculum taught learners to solve arithmetic word problems via a series of steps that would allow learners to achieve the outcomes. Furthermore, since these steps comprise independent critical thinking sub-processes, these steps were designed with the goal of allowing learners to generalize their approaches to novel problems. To differentiate problem-solving approaches undertaken by learners, a sequential clustering framework that employs four string distance metrics to discern these approaches before a sequential pattern mining algorithm is employed to summarize learning behaviors of various learner groups is proposed. The framework allow various problem-solving approaches to be aggregated into greater labels that are in line with the scripted steps. Problem-solving approaches identified via the framework also achieved an F1 score of 0.712 and AUC score of 0.715 for predicting student drop-out. A further analysis between script-like and schematic approaches revealed that learners who demonstrated schematic variants tend to persist in their learning.

Part II of this thesis focused on the use of deep neural networks to model sequential clickstream data to perform grade prediction. Compared to conventional grade prediction approaches that distill features from the logged clickstream data, the ability of deep neural network to automatically extract features from the inputs is advantageous since the relationship between input clickstream sequences and grade targets are not well studied. Furthermore, it has been established that non-cognitive traits are highly correlated to learning outcomes. To improve performance of grade prediction model, a joint-space grade prediction model that incorporates transient clickstream learning behavior and persistent learner characteristics is proposed. The use of non-cognitive trait enable prediction models to better differentiate clickstream sequences and improve grade prediction performance by up to 12%.

Next, the thesis raise the notion that sequential clickstream data is multi-valued. Due to its inherent multi-valued characteristics, the assumption that the underlying relationship between its inputs and targets is bijective is violated, and this limits grade prediction performance. A deep neural architecture is proposed to identify and differentiate multi-valued inputs from non-multi-valued inputs. The proposed MV identification module is able to identify 32.3% more instances as multi-valued compared to greedy methods. The

identification, and subsequent removal, of these multi-valued inputs also improve grade prediction performance by 38.95%.

8.2 Future research directions

The following are some of the possible suggestions for future research:

1. **Determination of multi-valued mappings.** Part II of the thesis has highlighted the multi-valued nature of interaction clickstream data. This is expected given the dynamics of human behavior. While the proposed algorithm is able to identify multi-valued inputs, the study stop short of determining the possible outcomes that are associated with each prediction. The ability to determine possible outcomes would allow practitioners to better identify at-risk learners.
2. **Generalizability of prediction models.** While prediction of learning outcomes have been studied extensively in the past decade, one of the notable problems that have yet been solved is the ability of prediction models to generalize to unseen dataset. While several studies have employed transfer learning algorithms to achieve similar prediction performance in future course runs of the same course, the ability of prediction models to be able to be adapted for any similar course remains an open question. In order to overcome this limitation, algorithms that take into account course characteristic and data distribution differences between student populations is required.
3. **Dynamic weighting scheme for sequence clustering.** Chapter 4 proposes a clustering algorithm that employs a fixed weighting scheme to combine the four pairwise similarity metrics to differentiate clickstream sequences. This is due to the fact that the relationship between clickstream sequences and learning outcomes are not well studied. As more studies reveal insights that allow for better understanding of clickstream sequences, a dynamic weighting scheme that allocates different weightage to each metric may reveal other interesting patterns.

Publication list

The author's publications are summarized as follows.

- Kelvin H. R. Ng, K. Hartman, K. Liu, and Andy W. H. Khong, "Modelling the way: Using action sequence archetypes to differentiate learning pathways from learning outcomes," in *Proc. 9th Int. Conf. Educ. Data Min., EDM, 2016*, Raleigh, NC, USA, Jun 29 - Jul 2 2016, pp. 167174.
- Kelvin H. R. Ng, K. Hartman, N. Goodkin, Andy W. H. Khong, "Mining for preparatory processes of transfer learning in a blended course," in *Proc. Am. Geophys. Union Fall Meet., AGU Fall Meeting*, New Orleans, LA, USA, Dec 11 - 15 2017
- Kelvin H. R. Ng, S. Tatinati, and Andy W. H. Khong, "Online education evaluation for signal processing course through student learning pathways," in *Proc. IEEE Int. Conf. Acoust., Speech and Signal Process., ICASSP, 2018*, Calgary, AB, Canada, Apr 15-20 2018, pp. 6458-6462.
- Kelvin H. R. Ng, S. Tatinati, and Andy W. H. Khong, "Characterization of fuzziness for grade prediction using deep neural networks," in *Companion Proc. 9th Int. Learn. Anal. Knowl. Conf., LAK 2019*, Tempe, AZ, USA, May 4-8 2019, pp. 64586462.
- Kelvin H. R. Ng, S. Tatinati and Andy W. H. Khong, "Grade Prediction From Multi-Valued Click-Stream Traces via Bayesian-Regularized Deep Neural Networks," in *IEEE Trans. Signal Process.*, vol. 69, pp. 1477-1491, 2021.

References

- [1] G. Siemens and P. Long, “Penetrating the fog: Analytics in learning and education.” *EDUCAUSE Review*, vol. 46, no. 5, p. 30, 2011.
- [2] R. S. Baker and K. Yacef, “The state of educational data mining in 2009: A review and future visions,” *J. Educ. Data Mining*, vol. 1, no. 1, pp. 3–17, 2009.
- [3] C. Romero and S. Ventura, “Educational data mining: A review of the state of the art,” *IEEE Trans. Syst. Man Cybern. Part C*, vol. 40, no. 6, pp. 601–618, 2010.
- [4] G. Castillo, J. Gama, and A. M. Breda, “An adaptive predictive model for student modeling,” in *Adv. in Web-based education: Personalized learning environments*. IGI Global, 2006, pp. 70–92.
- [5] R. S. Baker, A. T. Corbett, K. R. Koedinger, and A. Z. Wagner, “Off-task behavior in the cognitive tutor classroom: When students “game the system”,” in *Proc. SIGCHI Conf. Hum. Factor. Comput. Syst., SIGCHI 2004*, Vienna, Austria, Apr 24 - 29 2004, pp. 383–390.
- [6] S. Craig, A. Graesser, J. Sullins, and B. Gholson, “Affect and learning: an exploratory look into the role of affect in learning with autotutor,” *J. Educ. Media*, vol. 29, no. 3, pp. 241–250, 2004.
- [7] P. Blikstein, “Multimodal learning analytics,” in *Proc. 3rd Int. Learn. Anal. Knowl. Conf., LAK 2013*, Leuven, Belgium, April 8-12 2013, pp. 102–106.
- [8] G. Siemens and R. S. Baker, “Learning analytics and educational data mining: towards communication and collaboration,” in *Proc. 2nd Int. Conf. Educ. Data Min., EDM 2012*, Chania, Greece, Jun 19-21 2012, pp. 252–254.
- [9] A. M. Johnson, M. E. Jacovina, D. G. Russell, and C. M. Soto, *Challenges and solutions when using technologies in the classroom*. ERIC Clearinghouse, 2016.
- [10] K. H. R. Ng, S. Tatinati, and A. W. H. Khong, “Characterization of fuzziness for grade prediction using deep neural networks,” in *Companion Proc. 9th Int. Learn. Anal. Knowl. Conf., LAK 2019*, Tempe, AZ, USA, May 4-8 2019, pp. 6458–6462.

REFERENCES

- [11] P. M. Moreno-Marcos, C. Alario-Hoyos, P. J. Muñoz-Merino, and C. D. Kloos, “Prediction in MOOCs: A review and future research directions,” *IEEE Trans. Learn. Technol.*, vol. 12, no. 3, pp. 384–401, 2019.
- [12] J. M. Braxton, A. S. Hirschy, and S. A. McClendon, *Understanding and reducing college student departure: ASHE-ERIC higher education report, volume 30, number 3*. John Wiley & Sons, 2011, vol. 16.
- [13] R. Stillwell and J. Sable, “Public school graduates and dropouts from the common core of data: School year 2009-10. first look (provisional data). NCES 2013-309rev.” *National Center for Education Statistics*, 2013.
- [14] N. Gitinabard, Y. Xu, S. Heckman, T. Barnes, and C. F. Lynch, “How widely can prediction models be generalized? performance prediction in blended courses,” *IEEE Trans. Learn. Technol.*, vol. 12, no. 2, pp. 184–197, 2019.
- [15] B. A. Nijstad, C. K. De Dreu, E. F. Rietzschel, and M. Baas, “The dual pathway to creativity model: Creative ideation as a function of flexibility and persistence,” *Eur. Rev. Soc. Psychol.*, vol. 21, no. 1, pp. 34–77, 2010.
- [16] R. P. Abelson, “Psychological status of the script concept,” *Am. Psychol.*, vol. 36, no. 7, p. 715, 1981.
- [17] T. H. Kho, S. M. Yeo, and L. Fan, “Model method in singapore primary mathematics textbooks,” in *Proc. Int. Conf. Math. Textb. Res. and Dev., ICMT 2014*, Glasgow, United Kingdom, Jul 7-9 2014, p. 275.
- [18] B. Dietz-Uhler and J. E. Hurn, “Using learning analytics to predict (and improve) student success: A faculty perspective,” *J. Interact. Online Learn.*, vol. 12, no. 1, pp. 17–26, 2013.
- [19] K. Verbert, N. Manouselis, H. Drachsler, and E. Duval, “Dataset-driven research to support learning and knowledge analytics,” *J. Educ. Techno. Soc.*, vol. 15, no. 3, pp. 133–148, 2012.
- [20] Z. Cemalcilar and F. Gökşen, “Inequality in social capital: Social capital, social risk and drop-out in the turkish education system,” *Br. J. Sociol. Educ.*, vol. 35, no. 1, pp. 94–114, 2014.
- [21] M. Letseka and S. Maile, *High university drop-out rates: A threat to South Africa’s future*. Human Sciences Research Council Pretoria, 2008.
- [22] A. Elliot and M. Church, “A hierarchical model of approach and avoidance achievement motivation,” *J. Pers. Soc. Psychol.*, vol. 72, pp. 218–232, 01 1997.
- [23] S. Mangal, *Advanced educational psychology*. PHI Learning Pvt. Ltd., 2002.
- [24] A. Benjamin, “Factors influencing learning,” 2014.

REFERENCES

- [25] K. V. Petrides, P. Vernon, J. Schermer, L. Ligthart, D. Boomsma, and L. Veselka, “Relationships between trait emotional intelligence and the big five in the netherlands,” *Pers. Individ. Differ.*, vol. 48, no. 8, pp. 906–910, 2010.
- [26] Z. Ren, X. Ning, and H. Rangwala, “ALE: Additive latent effect models for grade prediction,” in *Proc. SIAM Int. Conf. Data Min., SDM 2018*, San Diego, CA, USA, May 3-5 2018, pp. 477–485.
- [27] A. N. Kluger and A. DeNisi, “The effects of feedback interventions on performance: A historical review, a meta-analysis, and a preliminary feedback intervention theory,” *Psychol. Bull.*, vol. 119, no. 2, p. 254, 1996.
- [28] P. Black and D. Wiliam, “Assessment and classroom learning,” *Assess. Educ.*, vol. 5, no. 1, pp. 7–74, 1998.
- [29] L. Shepard, “The role of classroom assessment in teaching and learning,” 2001.
- [30] A. T. Corbett and J. R. Anderson, “Knowledge tracing: modeling the acquisition of procedural knowledge,” *User Model. User Adapt Interact.*, vol. 4, no. 4, pp. 253–278, 1994.
- [31] C. Piech, J. Spencer, J. Huang, S. Ganguli, M. Sahami, L. J. Guibas, and J. Sohl-Dickstein, “Deep knowledge tracing,” 2015, arXiv:1506.05908.
- [32] Blackboard Inc. Blackboard. [Online]. Available: <https://www.blackboard.com/>
- [33] Moodle. Moodle. [Online]. Available: <https://moodle.org/>
- [34] edX Inc. edX. [Online]. Available: <https://www.edx.org/>
- [35] Coursera Inc. Coursera. [Online]. Available: <https://www.coursera.org/>
- [36] Khan Academy, Inc. Khan academy. [Online]. Available: <https://www.khanacademy.org/>
- [37] Udemy, Inc. Udemy. [Online]. Available: <https://www.udemy.com/>
- [38] Udacity, Inc. Udacity. [Online]. Available: <https://www.udacity.com/>
- [39] N. Li, L. Kidzinski, P. Jermann, and P. Dillenbourg, “MOOC video interaction patterns: What do they tell us?” in *Proc. 10th Eur. Conf. Technol. Enhanc. Learn., EC-TEL 2015*, vol. 9307, Toledo, Spain, Sept 15-18 2015, pp. 197–210.
- [40] S. Mongy, C. Djeraba, and D. A. Simovici, “On clustering users’ behaviors in video sessions,” in *Proc. Int. Conf. Data Min., DMIN 2007*, Las Vegas, NV, USA, Jun 25-28 2007, pp. 99–103.

REFERENCES

- [41] P. Branch, G. Egan, and B. Tonkin, “Modeling interactive behaviour of a video based multimedia system,” in *Proc. IEEE Int. Conf. Commun., ICC 1999*, vol. 2, Vancouver, BC, Canada, Jun 6-10 1999, pp. 978–982 vol.2.
- [42] T. F. Syeda-Mahmood and D. B. Ponceleon, “Learning video browsing behavior and its application in the generation of video previews,” in *Proc. 9th ACM Int. Conf. Multimedia, ACM Multimedia 2001*, Ottawa, ON, Canada, Sept 30 - Oct 5 2001, pp. 119–128.
- [43] S. Ritter, J. R. Anderson, K. R. Koedinger, and A. Corbett, “Cognitive tutor: Applied research in mathematics education,” *Psychon. Bull. Rev.*, vol. 14, no. 2, pp. 249–255, 2007.
- [44] A. C. Graesser, P. Chipman, B. C. Haynes, and A. Olney, “Autotutor: an intelligent tutoring system with mixed-initiative dialogue,” *IEEE Trans. Educ.*, vol. 48, no. 4, pp. 612–618, 2005.
- [45] T. Crow, A. Luxton-Reilly, and B. Wuensche, “Intelligent tutoring systems for programming education: a systematic review,” in *Proc. 20th Australas. Comput. Educ. Conf., ACE 2018*, Brisbane, QLD, Australia, Jan 30 - Feb 2 2018, pp. 53–62.
- [46] D. Di Mitri, J. Schneider, M. Specht, and H. Drachsler, “From signals to knowledge: A conceptual model for multimodal learning analytics,” *J. Comput. Assist. Learn.*, vol. 34, no. 4, pp. 338–349, 2018.
- [47] K. Sharma, Z. K. Papamitsiou, J. K. Olsen, and M. N. Giannakos, “Predicting learners’ effortful behaviour in adaptive assessment using multimodal data,” in *Proc. 10th Int. Learn. Anal. Knowl. Conf., LAK 2020*, Frankfurt, Germany, Mar 23-27 2020, pp. 480–489.
- [48] O. AlZoubi, S. K. D’Mello, and R. A. Calvo, “Detecting naturalistic expressions of nonbasic affect using physiological signals,” *IEEE Trans. Affect. Comput.*, vol. 3, no. 3, pp. 298–310, 2012.
- [49] L. P. Prieto, K. Sharma, P. Dillenbourg, and M. Jesús, “Teaching analytics: towards automatic extraction of orchestration graphs using wearable sensors,” in *Proc. 6th Int. Learn. Anal. Knowl. Conf., LAK 2016*, Edinburgh, United Kingdom, Apr 25-29 2016, pp. 148–157.
- [50] A. A. Edwards, A. Massicci, S. Sridharan, J. Geigel, L. Wang, R. Bailey, and C. O. Alm, “Sensor-based methodological observations for studying online learning,” in *Proc. 2017 ACM Workshop Intell. Interfaces Ubiquitous Smart Learn., SmartLearn 2017*, Limassol, Cyprus, Mar 13-16 2017, pp. 25–30.
- [51] N. Bosch, “Multimodal affect detection in the wild: Accuracy, availability, and generalizability,” in *Proc. 2015 ACM Int. Conf. Multimodal Interact., ICMI 2015*, Seattle, WA, USA, Nov 9-13 2015, pp. 645–649.

REFERENCES

- [52] M. Raca and P. Dillenbourg, “Holistic analysis of the classroom,” in *Proc. 2014 ACM Int. Conf. Multimodal Interact., ICMI 2014*, Istanbul, Turkey, Nov 12-16 2014, pp. 13–20.
- [53] N. Alyuz, E. Okur, E. Oktay, U. Genc, S. Aslan, S. E. Mete, B. Arnrich, and A. A. Esme, “Semi-supervised model personalization for improved detection of learner’s emotional engagement,” in *Proc. 18th ACM Int. Conf. Multimodal Interact., ICMI 2016*, Tokyo, Japan, Nov 12-16 2016, pp. 100–107.
- [54] M. S. Hussain, H. Monkarezi, and R. A. Calvo, “Categorical vs. dimensional representations in multimodal affect detection during learning,” in *Proc. Int. Conf. Intell. Tutoring Syst., ITS 2012*, Chania, Crete, Greece, Jun 14-18 2012, pp. 78–83.
- [55] J. Lee and Y. Lee, “Personality types and learners’ interaction in web-based threaded discussion,” *Q. Rev. Distance Educ.*, vol. 7, no. 1, p. 83, 2006.
- [56] T. J. Butler and G. Pinto-Zipp, “Students’ learning styles and their preferences for online instructional methods,” *J. Educ. Technol. Syst.*, vol. 34, no. 2, pp. 199–221, 2005.
- [57] T. Irani, R. Telg, C. Scherler, and M. Harrington, “Personality type and its relationship to distance education students course perceptions and performance,” *Q. Rev. Distance Educ.*, vol. 4, no. 4, pp. 445–453, 2003.
- [58] H. Kanuka and N. Nocente, “Exploring the effects of personality type on perceived satisfaction with web-based learning in continuing professional development,” *Distance Educ.*, vol. 24, no. 2, pp. 227–244, 2003.
- [59] E. B. Kim and M. J. Schniederjans, “The role of personality in web-based distance education courses,” *Commun. ACM*, vol. 47, no. 3, pp. 95–98, 2004.
- [60] H.-G. Wolff and S. Kim, “The relationship between networking behaviors and the big five personality dimensions,” *Career Dev. Int.*, 2012.
- [61] V. Trowler, “Student engagement literature review,” *The Higher Education Academy*, vol. 11, no. 1, pp. 1–15, 2010.
- [62] N. Karweit and R. E. Slavin, “Measurement and modeling choices in studies of time and learning,” *Amer. Educ. Research J.*, vol. 18, no. 2, pp. 157–171, 1981.
- [63] J. B. Carroll, “A model of school learning,” *Teachers college record*, pp. 723–733, 1963.
- [64] A. F. Gregorc and H. B. Ward, “A new definition for individual,” *NASSP Bull.*, vol. 61, no. 406, pp. 20–26, 1977.
- [65] J. W. Keefe, *Learning Style Theory and Practice*. ERIC, 1987.

REFERENCES

- [66] J. C. Tseng, H.-C. Chu, G.-J. Hwang, and C.-C. Tsai, "Development of an adaptive learning system with two sources of personalization information," *Comput. Educ.*, vol. 51, no. 2, pp. 776–786, 2008.
- [67] P. Brusilovsky *et al.*, "Adaptive and intelligent technologies for web-based education," *Int. J. Artif. Intell. Educ.*, vol. 13, no. 4, pp. 19–25, 2003.
- [68] P. Brusilovsky, J. Eklund, and E. Schwarz, "Web-based education for all: a tool for development adaptive courseware," *Comput. Netw. ISDN Syst.*, vol. 30, no. 1, pp. 291–300, 1998.
- [69] P. Brusilovsky and M. T. Maybury, "From adaptive hypermedia to the adaptive web," *Commun. ACM*, vol. 45, no. 5, pp. 30–33, 2002.
- [70] N. D. Fleming, *Teaching and learning styles : VARK strategies*. Christchurch, N.Z. : N.D. Fleming, 2001.
- [71] D. A. Kolb, *Experiential learning: Experience as the source of learning and development*. FT press, 2014.
- [72] R. M. Felder, L. K. Silverman *et al.*, "Learning and teaching styles in engineering education," *Eng. Educ.*, vol. 78, no. 7, pp. 674–681, 1988.
- [73] P. Honey and A. Mumford, *Manual of Learning Styles*. London: P Honey, 1982.
- [74] J. Feldman, A. Monteserin, and A. Amandi, "Automatic detection of learning styles: state of the art," *Artif. Intell. Rev.*, vol. 44, no. 2, pp. 157–186, 2015.
- [75] P. Brusilovsky, "Methods and techniques of adaptive hypermedia," *User Model. User Adapt. Interact.*, vol. 6, no. 2-3, pp. 87–129, 1996.
- [76] D. Rohrer and H. Pashler, "Learning styles: Where's the evidence?," *Online Submission*, vol. 46, no. 7, pp. 634–635, 2012.
- [77] H. Pashler, M. McDaniel, D. Rohrer, and R. Bjork, "Learning styles: Concepts and evidence," *Psychological science in the public interest*, vol. 9, no. 3, pp. 105–119, 2008.
- [78] M.-H. Cho and M. L. Heron, "Self-regulated learning: the role of motivation, emotion, and use of learning strategies in students learning experiences in a self-paced online mathematics course," *Distance Educ.*, vol. 36, no. 1, pp. 80–99, 2015.
- [79] R. M. Ryan and E. L. Deci, "Intrinsic and extrinsic motivations: Classic definitions and new directions," *Contemporary Educ. Psychol.*, vol. 25, no. 1, pp. 54–67, 2000.
- [80] P. G. de Barba, G. E. Kennedy, and M. Ainley, "The role of students' motivation and participation in predicting performance in a MOOC," *J. Comput. Assist. Learn.*, vol. 32, no. 3, pp. 218–231, 2016.

REFERENCES

- [81] R. F. Kizilcec, M. Pérez-Sanagustín, and J. J. Maldonado, “Self-regulated learning strategies predict learner behavior and goal attainment in massive open online courses,” *Comput. Educ.*, vol. 104, pp. 18–33, 2017.
- [82] J. C. Sun and R. Rueda, “Situational interest, computer self-efficacy and self-regulation: Their impact on student engagement in distance education,” *Br. J. Educ. Technol.*, vol. 43, no. 2, pp. 191–204, 2012.
- [83] T. Lee, P. Shen, and C. Tsai, “Applying web-enabled problem-based learning and self-regulated learning to add value to computing education in taiwan’s vocational schools,” *J. Educ. Technol. Soc.*, vol. 11, no. 3, pp. 13–25, 2008.
- [84] B. Samruayruen, J. Enriquez, O. Natakatoong, and K. Samruayruen, “Self-regulated learning: A key of a successful learner in online learning environments in thailand,” *J. Educ. Comput. Research*, vol. 48, no. 1, pp. 45–69, 2013.
- [85] C.-C. Tsai, “Conceptions of learning versus conceptions of web-based learning: The differences revealed by college students,” *Comput. & Educ.*, vol. 53, no. 4, pp. 1092–1103, 2009.
- [86] N. Hood, A. Littlejohn, and C. D. Milligan, “Context counts: How learners’ contexts influence learning in a MOOC,” *Comput. Educ.*, vol. 91, pp. 83–91, 2015.
- [87] A. Littlejohn, N. Hood, C. Milligan, and P. Mustain, “Learning in moocs: Motivations and self-regulated learning in moocs,” *The Internet and Higher Education*, vol. 29, pp. 40–48, 2016.
- [88] R. Azevedo and J. G. Cromley, “Does training on self-regulated learning facilitate students’ learning with hypermedia?” *J. Educ. Psychol.*, vol. 96, no. 3, p. 523, 2004.
- [89] D. H. Schunk, “Self-regulated learning: The educational legacy of paul r. pintrich,” *Educ. Psychol.*, vol. 40, no. 2, pp. 85–94, 2005.
- [90] B. Zimmerman, “Self-regulated learning: Theories, measures, and outcomes,” *Int. Encyclopedia Social Behavioral Sciences*, 12 2015.
- [91] R. Azevedo and V. Aleven, “Metacognition and learning technologies: An overview of current interdisciplinary research,” *International Handbook of Metacognition and Learning Technologies*, pp. 1–16, 2013.
- [92] J. Broadbent and W. L. Poon, “Self-regulated learning strategies & academic achievement in online higher education learning environments: A systematic review,” *The Internet and Higher Education*, vol. 27, pp. 1–13, 2015.
- [93] H. Niemi, A. Nevgi, and P. i. Virtanen, “Towards self-regulation in web-based learning,” *J. Educ. Media*, vol. 28, no. 1, pp. 49–71, 2003.

REFERENCES

- [94] R. J. Riding and F. Pearson, "The relationship between cognitive style and intelligence," *Educ. Psych.*, vol. 14, no. 4, pp. 413–425, 1994.
- [95] R. Riding and I. Cheema, "Cognitive styles: an overview and integration," *Educ. Psych.*, vol. 11, no. 3-4, pp. 193–215, 1991.
- [96] A. Hunt and S. Tickner, "Cultural dimensions of learning in online teacher education courses," *J. Open Flexible Dist. Learn.*, vol. 19, no. 2, pp. 25–47, 2015.
- [97] K. Kim and C. J. Bonk, "Cross-cultural comparisons of online collaboration," *J. Comput. Mediat. Commun.*, vol. 8, no. 1, p. 0, 2002.
- [98] A. Ogan, E. Walker, R. S. Baker, M. M. T. Rodrigo, J. C. A. Soriano, and M. J. Castro, "Towards understanding how to assess help-seeking behavior across cultures," *Int. J. Artif. Intell. Educ.*, vol. 25, no. 2, pp. 229–248, 2015.
- [99] M. M. T. Rodrigo, R. Baker, L. Rossi *et al.*, "Student off-task behavior in computer-based learning in the philippines: comparison to prior research in the usa," *Teachers College Record*, vol. 115, no. 10, pp. 1–27, 2013.
- [100] M. Abdullah, W. H. Daffa, R. M. Bashmail, M. Alzahrani, and M. Sadik, "The impact of learning styles on learner's performance in e-learning environment," *Int. J. Adv. Comput. Sci. and Appl.*, vol. 6, no. 9, pp. 24–31, 2015.
- [101] M. A. Abdullah, "Learning style classification based on student's behavior in moodle learning management system," *Trans. Mach. Learn. Artif. Intell.*, vol. 3, no. 1, pp. 28–28, 2015.
- [102] A. Latham, K. A. Crockett, and D. McLean, "Profiling student learning styles with multilayer perceptron neural networks," in *IEEE Int. Conf. Syst., Man, and Cybernetics, SMC 2013*, Manchester, United Kingdom, Oct 13-16 2013, pp. 2510–2515.
- [103] L. J. Deborah, R. Sathiyaseelan, S. Audithan, and P. Vijayakumar, "Fuzzy-logic based learning style prediction in e-learning using web interface information," *Sadhana*, vol. 40, no. 2, pp. 379–394, 2015.
- [104] Y. Chang, W. Kao, C. Chu, and C. Chiu, "A learning style classification mechanism for e-learning," *Comput. Educ.*, vol. 53, no. 2, pp. 273–285, 2009.
- [105] J. Bernard, T. Chang, E. Popescu, and S. Graf, "Using artificial neural networks to identify learning styles," in *Proc. 17th Int. Conf. Artif. Intell. in Educ., AIED 2015*, vol. 9112, Madrid, Spain, Jun 22-26 2015, pp. 541–544.
- [106] T. An, C. Krauss, and A. Merceron, "Can typical behaviors identified in moocs be discovered in other courses?" in *Proc. 10th Int. Conf. Educ. Data Min., EDM 2017*, Wuhan, Hubei, China, Jun 25-28 2017.

REFERENCES

- [107] M. I. López, C. Romero, S. Ventura, and J. M. Luna, “Classification via clustering for predicting final marks starting from the student participation in forums,” in *Proc. 5th Int. Conf. Educ. Data Min., EDM 2012*, Chania, Greece, Jun 19-21 2012, pp. 148–151.
- [108] M. M. T. Rodrigo, E. A. Anglo, J. O. Sugay, and R. S. Baker, “Use of unsupervised clustering to characterize learner behaviors and affective states while using an intelligent tutoring system,” in *Proc. 16th Int. Conf. Comput. in Educ., ICCE 2008*, Hoi An, Vietnam, Jan 9-13 2008, pp. 57–64.
- [109] A. Pardo, D. Gasevic, J. Jovanovic, S. Dawson, and N. Mirriahi, “Exploring student interactions with preparation activities in a flipped classroom experience,” *IEEE Trans. Learn. Technol.*, vol. 12, no. 3, pp. 333–346, 2019.
- [110] M. A. Hogo, “Evaluation of e-learners behaviour using different fuzzy clustering models: A comparative study,” 2010, arXiv:1003.1499.
- [111] S. V. Kolekar, R. M. Pai, and M. M. M. Pai, “Prediction of learner’s profile based on learning styles in adaptive e-learning system,” *Int. J. Emerg. Technol. Learn.*, vol. 12, no. 6, pp. 31–51, 2017.
- [112] Y. Dong and G. Biswas, “An extended learner modeling method to assess students’ learning behaviors,” in *Proc. 10th Int. Conf. Educ. Data Min., EDM 2017*, Wuhan, Hubei, China, Jun 25-28 2017, pp. 302–305.
- [113] J. Kim, P. J. Guo, D. T. Seaton, P. Mitros, K. Z. Gajos, and R. C. Miller, “Understanding in-video dropouts and interaction peaks in online lecture videos,” in *Proc. 1st ACM Conf. Learn. Scale, L@S 2014*, Atlanta, GA, USA, Mar 4-5 2014, pp. 31–40.
- [114] J. Park, K. Denaro, F. Rodriguez, P. Smyth, and M. Warschauer, “Detecting changes in student behavior from clickstream data,” in *Proc. 7th Int. Learn. Anal. Knowl. Conf., LAK 2017*, Vancouver, BC, Canada, Mar 13-17 2017, pp. 21–30.
- [115] M. Alian and A. Shaout, “Predicting learners styles based on fuzzy model,” *Educ. Inf. Technol.*, vol. 22, no. 5, pp. 2217–2234, 2017.
- [116] T. Chellatamilan, M. Ravichandran, R. Suresh, and G. Kulanthavel, “Effect of mining educational data to improve adaptation of learning in e-learning system,” in *Proc. Int. Conf. Sust. Energ. Intell. Syst., SEISCON 2011*, Chennai, India, Jul 20-22 2011, pp. 94–103.
- [117] U. Markowska-Kaczmar, H. Kwasnicka, and M. Paradowski, “Intelligent techniques in personalization of learning in e-learning systems,” in *Computational Intelligence for Technology Enhanced Learning*. Springer, 2010, vol. 273, pp. 1–23.

REFERENCES

- [118] H. M. Truong, “Integrating learning styles and adaptive e-learning system: Current developments, problems and opportunities,” *Comput. Hum. Behav.*, vol. 55, pp. 1185–1193, 2016.
- [119] T. Sinha, P. Jermann, N. Li, and P. Dillenbourg, “Your click decides your fate: Inferring information processing and attrition behavior from mooc video clickstream interactions,” 2014, arXiv:1407.7131.
- [120] A. Bauer, J. Flatten, and Z. Popovic, “Analysis of problem-solving behavior in open-ended scientific-discovery game challenges,” in *Proc. 10th Int. Conf. Educ. Data Min., EDM 2017*, Wuhan, Hubei, China, Jun 25-28 2017.
- [121] D. A. Filvà, M. A. Forment, F. J. García-Peñalvo, D. F. Escudero, and M. J. C. Guerrero, “Clickstream for learning analytics to assess students’ behavior with scratch,” *Future Gener. Comput. Syst.*, vol. 93, pp. 673–686, 2019.
- [122] A. N. Rafferty and T. L. Griffiths, “Diagnosing algebra understanding via bayesian inverse planning,” in *Proc. 7th Int. Conf. Educ. Data Min., EDM 2014*, London, United Kingdom, Jul 4-7 2014, pp. 351–352.
- [123] J. Mostow, J. Beck, R. Chalasani, A. Cuneo, P. Jia, K. Kadaru *et al.*, “A la recherche du temps perdu, or as time goes by: Where does the time go in a reading tutor that listens?” in *Proc. 6th Int. Conf. Intel. Tutoring Syst., ITS 2002*. Seoul, Korea: Springer, Aug 18-22 2002, pp. 320–329.
- [124] J. Walonoski and N. Heffernan, “Detection and analysis of off-task gaming behavior in intelligent tutoring systems,” in *Proc. 8th Int. Conf. Intel. Tutoring Syst., ITS 2006*, Beijing, China, Jun 12-16 2006, pp. 382–391.
- [125] M. Cocea, A. HersHKovitz, and R. S. Baker, “The impact of off-task and gaming behaviors on learning: Immediate or aggregate?” in *Proc. 14th Int. Conf. Artif. Intell. in Educ., AIED 2009*, vol. 200, Brighton, United Kingdom, Jul 6-10 2009, pp. 507–514.
- [126] N. Heller and F. Bry, “Predicting learners’ behaviours to get it wrong,” in *Proc. 8th Int. Conf. Methodolog. Intell. Syst. Technol. Enhanc. Learn., MIS4TEL 2018*, vol. 804, Toledo, Spain, Jun 20-22 2018, pp. 12–19.
- [127] H. Wood and D. Wood, “Help seeking, learning and contingent tutoring,” *Comput. Educ.*, vol. 33, no. 2-3, pp. 153–169, 1999.
- [128] N. Karweit and R. E. Slavin, “Time-on-task: Issues of timing, sampling, and definition,” *J. Educ. Psychol.*, vol. 74, no. 6, p. 844, 1982.
- [129] H. M. Lahaderne, “Attitudinal and intellectual correlates of attention: A study of four sixth-grade classrooms,” *J. Educ. Psychol.*, vol. 59, no. 5, p. 320, 1968.

REFERENCES

- [130] J. W. Lloyd and A. B. Loper, “Measurement and evaluation of task-related learning behaviors: Attention to task and metacognition,” *Sch. Psychol. Rev.*, vol. 15, no. 3, pp. 336–345, 1986.
- [131] S. W. Lee, K. E. Kelly, and J. E. Nyre, “Preliminary report on the relation of students’ on-task behavior with completion of school work,” *Psychol. Rep.*, vol. 84, no. 1, pp. 267–272, 1999.
- [132] B. S. Bloom, *Human characteristics and school learning*. McGraw-Hill, 1976.
- [133] A. Arbretton, “Student goal orientation and help-seeking strategy use,” 1998.
- [134] A. Renkl, “Learning from worked-out examples: Instructional explanations supplement self explanations,” *Learn. Instr.*, vol. 12, pp. 149–176, 2002.
- [135] V. Aleven, B. McLaren, I. Roll, and K. R. Koedinger, “Toward tutoring help seeking,” in *Proc. 7th Int. Conf. Intel. Tutoring Syst., ITS 2004*, Maceió, Brazil, Aug 30 - Sep 3 2004, pp. 227–239.
- [136] R. S. Baker, I. Roll, A. T. Corbett, and K. R. Koedinger, “Do performance goals lead students to game the system?” in *Proc. 12th Int. Conf. Artif. Intell. in Educ., AIED 2005*, vol. 125, Amsterdam, The Netherlands, Jul 18-22 2005, pp. 57–64.
- [137] I. Arroyo and B. P. Woolf, “Inferring learning and attitudes from a bayesian network of log file data,” in *Proc. 12th Int. Conf. Artif. Intell. in Educ., AIED 2005*, Amsterdam, The Netherlands, Jul 18-22 2005, pp. 33–40.
- [138] R. S. Baker, J. Ocumpaugh, and J. Andres, “Bromp quantitative field observations: A review,” in *Learning Science: Theory, Research, and Practice.*, R. Feldman, Ed. New York, NY: McGraw-Hill, 2015, ch. 5.
- [139] R. S. Baker, A. T. Corbett, and K. R. Koedinger, “Detecting student misuse of intelligent tutoring systems,” in *Proc. 7th Int. Conf. Intel. Tutoring Syst., ITS 2004*, vol. 3220. Maceió, Brazil: Springer, Aug 30 - Sep 3 2004, pp. 531–540.
- [140] R. S. Baker, A. T. Corbett, K. R. Koedinger, and I. Roll, “Detecting when students game the system, across tutor subjects and classroom cohorts,” in *Proc. 10th Int. Conf. User Model., UM 2005*, vol. 3538. Edinburgh, United Kingdom: Springer, Jul 24-29 2005, pp. 220–224.
- [141] R. S. Baker, “Modeling and understanding students’ off-task behavior in intelligent tutoring systems,” in *Proc. SIGCHI Conf. Hum. Factor. Comput. Syst., SIGCHI 2007*, San Jose, CA, USA, 01 2007, pp. 1059–1068.
- [142] B. C. Buckley, J. D. Gobert, P. Horwitz, and L. M. O’Dwyer, “Looking inside the black box: assessing model-based learning and inquiry in biologicatm,” *Int. J. Learn. Technol.*, vol. 5, no. 2, pp. 166–190, 2010.

REFERENCES

- [143] J. E. Beck and Y. Gong, “Wheel-spinning: Students who fail to master a skill,” in *Proc. 16th Int. Conf. Artif. Intell. Educ., AIED 2013*, ser. Lecture Notes in Computer Science, vol. 7926. Memphis, TN, USA: Springer, Jul 9-13 2013, pp. 431–440.
- [144] J. P. Rowe, S. W. McQuiggan, J. L. Robison, and J. C. Lester, “Off-task behavior in narrative-centered learning environments,” in *Proc. 14th Int. Conf. Artif. Intell. Educ., AIED 2009*, vol. 200. Brighton, UK: IOS Press, Jul 6-10 2009, pp. 99–106.
- [145] S. Kai, M. V. Almeda, R. S. Baker, N. Shechtman, C. Heffernan, and N. T. Heffernan, “Modeling wheel-spinning and productive persistence in skill builders,” in *Proc. 10th Int. Conf. Educ. Data Min., EDM 2017*, Wuhan, Hubei, China, Jun 25-28 2017.
- [146] M. Credé, M. C. Tynan, and P. D. Harms, “Much ado about grit: A meta-analytic synthesis of the grit literature.” *J. Person. Social Psych.*, vol. 113, no. 3, p. 492, 2017.
- [147] H. Wan and J. Beck, “Considering the influence of prerequisite performance on wheel spinning,” in *Proc. 8th Int. Conf. Educ. Data Min., EDM 2015*, Madrid, Spain, Jun 26-29 2015, pp. 129–135.
- [148] C. Zhang, Y. Huang, J. Wang, D. Lu, W. Fang, J. C. Stamper, S. Fancsali, K. Holstein, and V. Aleven, “Early detection of wheel spinning: Comparison across tutors, models, features, and operationalizations,” in *Proc. 12th Int. Conf. Educ. Data Min., EDM 2019*, Montréal, Canada, Jul 2-5 2019.
- [149] J. Park, R. Yu, F. Rodriguez, R. Baker, P. Smyth, and M. Warschauer, “Understanding student procrastination via mixture models,” in *Proc. 16th Int. Conf. Educ. Data Min., EDM 2018*, Buffalo, NY, USA, Jul 15-18 2018.
- [150] M. O. C. Z. S. Pedro, R. S. J. de Baker, and M. M. T. Rodrigo, “Detecting carelessness through contextual estimation of slip probabilities among students using an intelligent tutor for mathematics,” in *Proc. 15th Int. Conf. Artif. Intell. Educ., AIED 2011*, vol. 6738. Auckland, New Zealand: Springer, Jun 28 - Jul 2011 2011, pp. 304–311.
- [151] S. Karumbaiah, R. S. Baker, and V. J. Shute, “Predicting quitting in students playing a learning game,” in *Proc. 11th Int. Conf. Educ. Data Min., EDM 2018*, Buffalo, NY, USA, July 15-18 2018.
- [152] B. S. Bloom *et al.*, “Cognitive domain,” in *Taxonomy of Educational Objectives*. New York: McKay, 1956, vol. 1, p. 24.
- [153] R. M. Pearce, “Chance and the prepared mind,” *Sci.*, vol. 35, no. 912, pp. 941–956, 1912.

REFERENCES

- [154] G. Polya, *How to solve it: A new aspect of mathematical method*. Princeton university press, 2004, no. 246.
- [155] A. H. Schoenfeld, “Models of the teaching process,” *J. Math. Behav.*, vol. 18, no. 3, pp. 243–261, 1999.
- [156] R. C. Schank and R. P. Abelson, *Scripts, plans, goals, and understanding: An inquiry into human knowledge structures*. Psychology Press, 2013.
- [157] J. Sweller and P. Chandler, “Why some material is difficult to learn,” *Cogn. Instr.*, vol. 12, no. 3, pp. 185–233, 1994.
- [158] D. E. Rumelhart and A. Ortony, “The representation of knowledge in memory,” in *Schooling and the acquisition of knowledge*. Routledge, 2017, pp. 99–135.
- [159] Y. K. Cheong and M. Consultancy, “The model method in singapore,” *Math. Educ.*, vol. 6, no. 2, pp. 47–64, 2002.
- [160] A. Islam and D. Z. Inkpen, “Semantic text similarity using corpus-based word similarity and string similarity,” *ACM Trans. Knowl. Discov. Data*, vol. 2, no. 2, pp. 1–25, 2008.
- [161] D. L. Bazaldua, R. S. Baker, and M. O. S. Pedro, “Comparing expert and metric-based assessments of association rule interestingness,” in *Proc. 7th Int. Conf. Educ. Data Min., EDM 2014*, London, UK, Jul 4-7 2014, pp. 44–51.
- [162] T. F. Smith, M. S. Waterman *et al.*, “Identification of common molecular subsequences,” *J. Mol. Biol.*, vol. 147, no. 1, pp. 195–197, 1981.
- [163] A. Rajaraman and J. D. Ullman, *Data Mining*. Cambridge University Press, 2011, p. 117.
- [164] B. J. Frey and D. Dueck, “Clustering by passing messages between data points,” *Sci.*, vol. 315, no. 5814, pp. 972–976, 2007.
- [165] Y. Hu, F. Wu, and Y. Liao, “An efficient tree-based algorithm for mining sequential patterns with multiple minimum supports,” *J. Syst. Softw.*, vol. 86, no. 5, pp. 1224–1238, 2013.
- [166] K. H. R. Ng, S. Tatinati, and A. W. H. Khong, “Online education evaluation for signal processing course through student learning pathways,” in *Proc. IEEE Int. Conf. Acoust., Speech and Signal Process., ICASSP 2018*, Calgary, AB, Canada, Apr 15-20 2018, pp. 6458–6462.
- [167] W. Xing, X. Chen, J. Stein, and M. Marcinkowski, “Temporal predication of dropouts in moocs: Reaching the low hanging fruit through stacking generalization,” *Comput. Hum. Behav.*, vol. 58, pp. 119–129, 2016.

REFERENCES

- [168] J. Sivic and A. Zisserman, “Efficient visual search of videos cast as text retrieval,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 4, pp. 591–606, 2009.
- [169] M. Wen and C. P. Rosé, “Identifying latent study habits by mining learner behavior patterns in massive open online courses,” in *Proc. 23rd ACM Int. Conf. Inf. and Knowl. Manage., CIKM*, 2014, pp. 1983–1986.
- [170] Y. LeCun, P. Haffner, L. Bottou, and Y. Bengio, “Object recognition with gradient-based learning,” in *Shape, Contour and Grouping in Computer Vision*, vol. 1681. Springer, 1999, p. 319.
- [171] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [172] J. Whitehill, J. Williams, G. Lopez, C. Coleman, and J. Reich, “Beyond prediction: First steps toward automatic intervention in mooc student stopout,” *SSRN Electron. J.*, 2015.
- [173] L. Macfadyen and S. Dawson, “Mining LMS data to develop an “early warning system” for educators: A proof of concept,” *Comput. Educ.*, vol. 54, no. 2, pp. 588–599, 2010.
- [174] E. Aguiar, N. V. Chawla, J. Brockman, G. A. Ambrose, and V. Goodrich, “Engagement vs performance: using electronic portfolios to predict first semester engineering student retention,” in *Proc. 4th Int. Learn. Anal. Knowl. Conf., LAK 2014*, Indianapolis, IN, USA, Mar 24-28 2014, pp. 103–112.
- [175] H. Wan, J. Ding, X. Gao, and D. Pritchard, “Dropout prediction in moocs using learners’ study habits features,” in *Proc. 10th Int. Conf. Educ. Data Min., EDM 2017*, Wuhan, Hubei, China, Jun 25-28 2017.
- [176] D. J. Lemay and T. Doleck, “Grade prediction of weekly assignments in MOOCS: mining video-viewing behavior,” *Educ. Inf. Techol.*, vol. 25, no. 2, pp. 1333–1342, 2020.
- [177] X. Li, L. Xie, and H. Wang, “Grade prediction in moocs,” in *Proc. 2016 IEEE Int. Conf. Comput. Sci. Eng., CSE 2016, and IEEE Int. Conf. Embed. Ubiquitous Comput., EUC 2016, and 15th Int. Symp. Distrib. Comput. Appl. Bus. Eng., DCABES 2016*, Paris, France, Aug 24-26 2016, pp. 386–392.
- [178] Z. Ren, H. Rangwala, and A. Johri, “Predicting performance on MOOC assessments using multi-regression models,” in *Proc. 9th Int. Conf. Educ. Data Min., EDM 2016*, Raleigh, NC, USA, Jun 24 - Jul 2 2016, pp. 484–489.
- [179] M. Kloft, F. Stiehler, Z. Zheng, and N. Pinkwart, “Predicting MOOC dropout over weeks using machine learning methods,” in *Proc. 2014 Conf. Empir. Methods Nat. Lang. Process., EMNLP 2014*, Doha, Qatar, Oct 25-29 2014, pp. 60–65.

REFERENCES

- [180] B. Hong, Z. Wei, and Y. Yang, “Online education performance prediction via time-related features,” in *Proc 16th IEEE/ACIS Int. Conf. Comput. Inf. Sci., ICIS 2017*, Wuhan, China, May 24-26 2017, pp. 95–100.
- [181] C. G. Brinton and M. Chiang, “MOOC performance prediction via clickstream data and social learning networks,” in *Proc. 2015 IEEE Conf. Comput. Commun., INFOCOM 2015*, 2015, pp. 2299–2307.
- [182] T. Xie, Q. Zheng, W. Zhang, and H. Qu, “Modeling and predicting the active video-viewing time in a large-scale e-learning system,” *IEEE Access*, vol. 5, pp. 11 490–11 504, 2017.
- [183] Y. Zhang and B. Wu, “Research and application of grade prediction model based on decision tree algorithm,” in *Proc. ACM Turing Celebration Conf., ACM-TURC 2019*, Chengdu, China, May 17-19 2019, pp. 80:1–80:6.
- [184] C. Taylor, K. Veeramachaneni, and U. O’Reilly, “Likely to stop? predicting stopout in massive open online courses,” 2014, arXiv:1408.3382.
- [185] M. Wen, D. Yang, and C. Rose, “Sentiment analysis in mooc discussion forums: What does it tell us?” in *Proc. 7th Int. Conf. Educ. Data Min., EDM 2014*, London, United Kingdom, Jul 4-7 2014, pp. 130–137.
- [186] D. S. Chaplot, E. Rhim, and J. Kim, “Predicting student attrition in moocs using sentiment analysis and neural networks,” in *Proc. 17th Int. Conf. Artif. Intell. in Educ., AIED 2015*, vol. 1432, Madrid, Spain, Jun 22-26 2015.
- [187] D. Yang, T. Sinha, D. Adamson, and C. P. Rosé, “Turn on, tune in, drop out: Anticipating student dropouts in massive open online courses,” in *Proc. NIPS Workshop on Data Driven Educ., NIPS-DDE 2013*, vol. 11, Lake Tahoe, NV, USA, Dec 9-10 2013, p. 14.
- [188] S. Jiang, A. E. Williams, K. Schenke, M. Warschauer, and D. K. O’Dowd, “Predicting MOOC performance with week 1 behavior,” in *Proc. 7th Int. Conf. Educ. Data Min., EDM 2014*, London, United Kingdom, Jul 4-7 2014, pp. 273–275.
- [189] D. García-Saiz, C. Palazuelos, and M. Zorrilla, “The predictive power of the sna metrics for education,” in *Proc. 7th Int. Conf. Educ. Data Min., EDM 2014*, London, United Kingdom, Jul 4-7 2014, pp. 419–420.
- [190] J. Qiu, J. Tang, T. X. Liu, J. Gong, C. Zhang, Q. Zhang, and Y. Xue, “Modeling and predicting learning behavior in moocs,” in *Proc. 9th ACM Int. Conf. Web Search and Data Min., WSDM 2016*, San Francisco, CA, USA, Feb 22-25 2016, pp. 93–102.
- [191] M. M. Ashenafi, G. Riccardi, and M. Ronchetti, “Predicting students’ final exam scores from their course activities,” in *2015 IEEE Front. Educ. Conf., FIE 2015*, El Paso, TX, USA, Oct 21-24 2015, pp. 1–9.

REFERENCES

- [192] N. C. Ming and V. L. Ming, “Predicting student outcomes from unstructured data,” in *Proc. 20th Int. Conf. User Model. Adapt. Personalization, UMAP 2012*, vol. 872, Montreal, QC, Canada, Jul 16-20 2012.
- [193] S. E. Sorour, K. Goda, and T. Mine, “Student performance estimation based on topic models considering a range of lessons,” in *Proc. 17th Int. Conf. Artif. Intell. in Educ., AIED 2015*, vol. 9112, Madrid, Spain, Jun 22-26 2015, pp. 790–793.
- [194] F. Mi and B. Faltings, “Adaptive sequential recommendation for discussion forums on moocs using context trees,” in *Proc. 10th Int. Conf. Educ. Data Min., EDM 2017*, Wuhan, Hubei, China, Jun 25-28 2017.
- [195] K. Goda, S. Hirokawa, and T. Mine, “Correlation of grade prediction performance and validity of self-evaluation comments,” in *Proc. 14th Annu. ACM Conf. Inf. Technol. Educ., SIGITE 2013*, Orlando, FL, USA, Oct 10-12 2013, pp. 35–42.
- [196] M. A. Al-Barrak and M. Al-Razgan, “Predicting students final gpa using decision trees: a case study,” *Int. J. Inf. Educ. Technol.*, vol. 6, no. 7, p. 528, 2016.
- [197] G. Dekker, M. Pechenizkiy, and J. Vleeshouwers, “Predicting students drop out: A case study,” in *Proc. 2nd Int. Conf. Educ. Data Min., EDM 2009*, Cordoba, Spain, Jul 1-3 2009, pp. 41–50.
- [198] E. J. M. Lauría, J. D. Baron, M. Devireddy, V. Sundararaju, and S. M. Jayaprakash, “Mining academic data to improve college student retention: an open source perspective,” in *Proc. 2nd Int. Learn. Anal. Knowl. Conf., LAK 2012*, Vancouver, BC, Canada, Apr 29 - May 2 2012, pp. 139–142.
- [199] F. Jiménez, A. Paoletti, and G. S. and Guido Sciavicco, “Predicting the risk of academic dropout with temporal multi-objective optimization,” *IEEE Trans. Learn. Technol.*, vol. 12, no. 2, pp. 225–236, 2019.
- [200] A. Elbadrawy and G. Karypis, “Domain-aware grade prediction and top-n course recommendation,” in *Proc. 10th ACM Conf. Recommender Syst., Boston, MA, USA, September 15-19, 2016*, 2016, pp. 183–190.
- [201] M. Sweeney, J. Lester, and H. Rangwala, “Next-term student grade prediction,” in *Proc. IEEE Int. Conf. Big Data, IEEE BigData 2015*, Santa Clara, CA, USA, Oct 29 - Nov - 1 2015, pp. 970–975.
- [202] Z. Ren, X. Ning, and H. Rangwala, “Grade prediction with temporal course-wise influence,” in *Proc. 10th Int. Conf. Educ. Data Min., EDM 2017*, Wuhan, Hubei, China, Jun 25-28 2017.
- [203] H. Bydzovská, “Student performance prediction using collaborative filtering methods,” in *Proc. 17th Int. Conf. Artif. Intell. in Educ., AIED 2015*, vol. 9112, Madrid, Spain, Jun 22-26 2015, pp. 550–553.

REFERENCES

- [204] Y. Meier, J. Xu, O. Atan, and M. van der Schaar, "Predicting grades," *IEEE Trans. Signal Process.*, vol. 64, no. 4, pp. 959–972, 2016.
- [205] S. Morsy and G. Karypis, "Cumulative knowledge-based regression models for next-term grade prediction," in *Proc. 2017 SIAM Int. Conf. Data Min.* SIAM, 2017, pp. 552–560.
- [206] A. Polyzou and G. Karypis, "Grade prediction with models specific to students and courses," *Int. J. Data Sci. Anal.*, vol. 2, no. 3-4, pp. 159–171, 2016.
- [207] L. V. Morris, S.-S. Wu, and C. L. Finnegan, "Predicting retention in online general education courses," *The Am. J. Distance Educ.*, vol. 19, no. 1, pp. 23–36, 2005.
- [208] I. E. Livieris, K. Drakopoulou, and P. Pintelas, "Predicting students performance using artificial neural networks," in *Proc. 8th Pan-Hell. Conf. Int. Participation Inf. Commun. Technol. Educ.*, Volos, Greece, Sep 28-30 2012, pp. 321–328.
- [209] S. Borkar and K. Rajeswari, "Attributes selection for predicting students' academic performance using education data mining and artificial neural network," *Int. J. Comput. Appl.*, vol. 86, no. 10, 2014.
- [210] M. Tan and P. Shao, "Prediction of student dropout in e-learning program through the use of machine learning method," *Int. J. Emerg. Technol. Learn.*, vol. 10, no. 1, pp. 11–17, 2015.
- [211] N. Thai-Nghe, L. Drumond, A. Krohn-Grimberghe, and L. Schmidt-Thieme, "Recommender system for predicting student performance," in *Proc. 1st Workshop Recommender Syst. Technol. Enhanc. Learn., RecSysTEL 2010*, vol. 1, no. 2, Barcelona, Spain, Sep 29-30 2010, pp. 2811–2819.
- [212] N. Thai-Nghe, T. Horváth, and L. Schmidt-Thieme, "Factorization models for forecasting student performance," in *Proc. 4th Int. Conf. Educ. Data Min., EDM 2011*, Eindhoven, The Netherlands, Jul 6-8 2011, pp. 11–20.
- [213] S.-M. R. Ting and R. Man, "Predicting academic success of first-year engineering students from standardized test scores and psychosocial variables," *Int. J. Eng. Educ.*, vol. 17, no. 1, pp. 75–80, 2001.
- [214] A. Daud, N. R. Aljohani, R. A. Abbasi, M. D. Lytras, F. Abbas, and J. S. Alowibdi, "Predicting student performance using advanced learning analytics," in *Proc. 26th Int. Conf. World Wide Web Companion, Perth, Australia, April 3-7, 2017*, 2017, pp. 415–421.
- [215] R. M. Stein and G. Allione, "Mass attrition: An analysis of drop out from a principles of microeconomics mooc," *J. Econ. Educ.*, vol. 47, pp. 174–186, 2016.

REFERENCES

- [216] J. DeBoer, G. S. Stump, D. T. Seaton, A. D. Ho, D. E. Pritchard, and L. Breslow, “Bringing student backgrounds online: MOOC user demographics, site usage, and online learning,” in *Proc. 6th Int. Conf. on Educ. Data Min., EDM 2013*, Memphis, TN, USA, Jul 6-9 2013, pp. 312–313.
- [217] P. A. Willging and S. D. Johnson, “Factors that influence students’ decision to dropout of online courses.” *J. Asynchronous Learn. Netw.*, vol. 13, no. 3, pp. 115–127, 2009.
- [218] S. L. Hoskins and J. C. Van Hooff, “Motivation and ability: which students use online learning and what influence does it have on their achievement?” *Br. J. Educ. Technol.*, vol. 36, no. 2, pp. 177–192, 2005.
- [219] V. R. C. Martinho, C. Nunes, and C. R. Minussi, “Prediction of school dropout risk group using neural network fuzzy ARTMAP,” in *Proc. 2013 Federated Conf. Comput. Sci. and Information Syst., Kraków, Poland, September 8-11, 2013*, 2013, pp. 111–114.
- [220] R. Kappe and H. Van Der Flier, “Predicting academic success in higher education: whats more important than being smart?” *Eur. J. Psychol. Educ.*, vol. 27, no. 4, pp. 605–619, 2012.
- [221] S. Bouzayane and I. Saad, “Weekly predicting the at-risk MOOC learners using dominance-based rough set approach,” in *Proc. 5th. Eur. MOOCs Stakeholders Summit, EMOOCs 2017*, vol. 10254, Madrid, Spain, May 22-26 2017, pp. 160–169.
- [222] S. Sonkar, A. S. Lan, A. E. Waters, P. Grimaldi, and R. G. Baraniuk, “qdk: Question-centric deep knowledge tracing,” in *Proc. 13th Int. Conf. Educ. Data Min., EDM 2020*, Jul 10-13 2020.
- [223] P. Chen, Y. Lu, V. W. Zheng, and Y. Pian, “Prerequisite-driven deep knowledge tracing,” in *IEEE Int. Conf. Data Min., ICDM 2018*. Singapore: IEEE Computer Society, Nov 17-20 2018, pp. 39–48.
- [224] J. Zhang, X. Shi, I. King, and D. Yeung, “Dynamic key-value memory networks for knowledge tracing,” in *Proc. 26th Int. Conf. World Wide Web, WWW 2017*. Perth, Australia: ACM, April 3-7 2017, pp. 765–774.
- [225] S. Pandey and G. Karypis, “A self attentive model for knowledge tracing,” in *Proc. 12th Int. Conf. Educ. Data Min., EDM 2019*, Montréal, Canada, Jul 2-5 2019.
- [226] Y. Choi, Y. Lee, J. Cho, J. Baek, B. Kim, Y. Cha, D. Shin, C. Bae, and J. Heo, “Towards an appropriate query, key, and value computation for knowledge tracing,” in *Proc. 7th ACM Conf. Learn. Scale, L@S 2020*. Vancouver, BC, Canada: ACM, Aug 12-14 2020, pp. 341–344.

REFERENCES

- [227] D. Liu, Y. Zhang, J. Zhang, Q. Li, C. Zhang, and Y. Yin, “Multiple features fusion attention mechanism enhanced deep knowledge tracing for student performance prediction,” *IEEE Access*, vol. 8, pp. 194 894–194 903, 2020.
- [228] Z. Ren, X. Ning, A. S. Lan, and H. Rangwala, “Grade prediction with neural collaborative filtering,” in *Proc. 2019 IEEE Int. Conf. Data Sci. Adv. Anal., DSAA 2019*, Washington, DC, USA, Oct 5-8 2019, pp. 1–10.
- [229] S. Morsy and G. Karypis, “Context-aware non-linear and neural attentive knowledge-based models for grade prediction,” 2020, arXiv:2003.05063.
- [230] C. G. Brinton, S. Buccapatnam, M. Chiang, and H. V. Poor, “Mining MOOC click-streams: Video-watching behavior vs. in-video quiz performance,” *IEEE Trans. Signal Process.*, vol. 64, no. 14, pp. 3677–3692, 2016.
- [231] X. Li, T. Wang, and H. Wang, “Exploring n-gram features in clickstream data for MOOC learning achievement prediction,” in *Proc. 24th Int. Conf. Database Syst. Adv. Appl., DASFAA 2017*, vol. 10179, Suzhou, China, Mar 27-30 2017, pp. 328–339.
- [232] C. A. Brooks, C. Thompson, and S. D. Teasley, “A time series interaction analysis method for building predictive models of learners using log data,” in *Proc. 5th Int. Learn. Anal. Knowl. Conf., LAK 2015*, Poughkeepsie, NY, USA, Mar 16-20 2015, pp. 126–135.
- [233] S. Nagrecha, J. Z. Dillon, and N. V. Chawla, “MOOC dropout prediction: Lessons learned from making pipelines interpretable,” in *Proc. 26th Int. Conf. World Wide Web Companion, Perth, Australia, April 3-7, 2017*, 2017, pp. 351–359.
- [234] J. He, J. Bailey, B. I. P. Rubinstein, and R. Zhang, “Identifying at-risk students in massive open online courses,” in *Proc. 29th AAAI Conf. Artif. Intell., January 25-30, 2015, Austin, Texas, USA*, 2015, pp. 1749–1755.
- [235] S. Ameri, M. J. Fard, R. B. Chinnam, and C. K. Reddy, “Survival analysis based framework for early prediction of student dropouts,” in *Proc. 25th ACM Int. Conf. Inf. Knowl. Manage., CIKM 2016*, Indianapolis, IN, USA, Oct 24-28 2016, pp. 903–912.
- [236] G. Balakrishnan and D. Coetzee, “Predicting student retention in massive open online courses using hidden markov models,” *Electrical Engineering and Computer Sciences University of California at Berkeley*, vol. 53, pp. 57–58, 2013.
- [237] M. Fei and D. Yeung, “Temporal models for predicting student dropout in massive open online courses,” in *Proc. 2015 IEEE Int. Conf. Data Min. Workshop, ICDMW 2015*, Washington, DC, United States, Nov 14-17 2015, pp. 256–263.

REFERENCES

- [238] F. Wang and L. Chen, “A nonlinear state space model for identifying at-risk students in open online courses,” in *Proc. 9th Int. Conf. Educ. Data Min., EDM 2016*, Raleigh, NC, USA, Jun 24 - Jul 2 2016, pp. 527–532.
- [239] S. J. Lee, Y. Liu, and Z. Popovic, “Learning individual behavior in an educational game: A data-driven approach,” in *Proc. 7th Int. Conf. Educ. Data Min., EDM 2014*, London, United Kingdom, Jul 4-7 2014, pp. 114–121.
- [240] F. Okubo, T. Yamashita, A. Shimada, and H. Ogata, “A neural network approach for students’ performance prediction,” in *Proc. 7th Int. Learn. Anal. Knowl. Conf., LAK 2017*, Vancouver, BC, Canada, Mar 13-17 2017, pp. 598–599.
- [241] T. Yang, C. G. Brinton, C. Joe-Wong, and M. Chiang, “Behavior-based grade prediction for MOOCs via time series neural networks,” *J. Sel. Topics Signal Process.*, vol. 11, no. 5, pp. 716–728, 2017.
- [242] B. Jeon and N. Park, “Dropout prediction over weeks in moocs by learning representations of clicks and videos,” 2020, arXiv:2002.01955.
- [243] Q. Hu and H. Rangwala, “Reliable deep grade prediction with uncertainty estimation,” in *Proc. 9th Int. Int. Learn. Anal. Knowl. Conf., LAK 2019*, Tempe, AZ, USA, May 4-8 2019, pp. 76–85.
- [244] B. Kim, E. Vizitei, and V. Ganapathi, “Gritnet: Student performance prediction with deep learning,” in *Proc. 11th Int. Conf. Educ. Data Min., EDM 2018*, Buffalo, NY, USA, Jul 15-18 2018.
- [245] L. Qiu, Y. Liu, Q. Hu, and Y. Liu, “Student dropout prediction in massive open online courses by convolutional neural networks,” *Soft Comput.*, vol. 23, no. 20, pp. 10 287–10 301, 2019.
- [246] W. Wang, H. Yu, and C. Miao, “Deep model for dropout prediction in MOOCs,” in *Proc. 2nd Int. Conf. Crowd Sci. Eng., ICCSE 2017*, Beijing, China, Jul 6-9 2017, pp. 26–32.
- [247] Y. Gal and Z. Ghahramani, “Dropout as a bayesian approximation: Representing model uncertainty in deep learning,” in *Proc. 33th Int. Conf. Mach. Learn., ICML 2016*, New York, NY, USA, Jun 19-24 2016, pp. 1050–1059.
- [248] J. Whitehill, K. Mohan, D. T. Seaton, Y. Rosen, and D. Tingley, “Delving deeper into MOOC student dropout prediction,” 2017, arXiv:1702.06404.
- [249] H. Wan, K. Liu, Q. Yu, and X. Gao, “Pedagogical intervention practices: Improving learning engagement based on early prediction,” *IEEE Trans. Learn. Technol.*, vol. 12, no. 2, pp. 278–289, 2019.
- [250] A. Graves, G. Wayne, and I. Danihelka, “Neural turing machines,” *CoRR*, vol. abs/1410.5401, 2014. [Online]. Available: <http://arxiv.org/abs/1410.5401>

REFERENCES

- [251] A. Santoro, S. Bartunov, M. Botvinick, D. Wierstra, and T. P. Lillicrap, “Meta-learning with memory-augmented neural networks,” in *Proc. 33rd Int. Conf. Machine Learn., ICML 2016*, ser. JMLR Workshop and Conference Proceedings, vol. 48. New York City, NY, USA: JMLR.org, Jun 19-24 2016, pp. 1842–1850.
- [252] S. Sukhbaatar, A. Szlam, J. Weston, and R. Fergus, “End-to-end memory networks,” in *Proc. 28th Int. Conf. Neural Inf. Process. Syst., NIPS 2015*, Montreal, Quebec, Canada, Dec 7-12 2015, pp. 2440–2448.
- [253] K. Ishikawa, M. Furukawa, Y. Saito, T. Naito, and M. Shichida, “Log data analysis of learning histories in an e-learning course,” *Int. J. Inf. Educ. Technol.*, vol. 3, no. 6, p. 587, 2013.
- [254] K. A. Feldman, J. C. Smart, and C. A. Ethington, “Major field and person-environment fit: Using holland’s theory to study change and stability of college students,” *J. Higher Educ.*, vol. 70, no. 6, pp. 642–669, 1999.
- [255] C. S. Dweck, *Self-Theories: Their Role in Motivation, Personality, and Development*. Psychology Press, 1999.
- [256] A. Martin and H. Marsh, “Academic resilience and its psychological and educational correlates: A construct validity approach,” *Psychol. Sch.*, vol. 43, pp. 267 – 281, 03 2006.
- [257] A. L. Duckworth, C. Peterson, M. D. Matthews, and D. R. Kelly, “Grit: perseverance and passion for long-term goals,” *J. Pers. Soc. Psychol.*, vol. 92, no. 6, p. 1087, 2007.
- [258] F. Rhodewalt, “Conceptions of ability, achievement goals, and individual differences in self-handicapping behavior: On the application of implicit theories,” *J. Pers.*, vol. 62, no. 1, pp. 67–85, 1994.
- [259] Z. Wang, J. Zhang, J. Feng, and Z. Chen, “Knowledge graph and text jointly embedding,” in *Proc. 2014 Conf. Empir. Methods Nat. Lang. Process., EMNLP 2014*, Doha, Qatar, Oct 25-29 2014, pp. 1591–1601.
- [260] Z. Yu, J. Yu, J. Fan, and D. Tao, “Multi-modal factorized bilinear pooling with co-attention learning for visual question answering,” in *Proc. IEEE Int. Conf. Comput. Vis., ICCV 2017*, Venice, Italy, Oct 22-29 2017, pp. 1821–1830.
- [261] X. Dong, E. Gabrilovich, G. Heitz, W. Horn, N. Lao, K. Murphy, T. Strohmann, S. Sun, and W. Zhang, “Knowledge vault: a web-scale approach to probabilistic knowledge fusion,” in *Proc. 20th ACM Int. Conf. Knowl. Discov. Data Min., SIGKDD 2014*, New York, NY, USA, Aug 24 - 27 2014, pp. 601–610.
- [262] A. Neelakantan, B. Roth, and A. McCallum, “Compositional vector space models for knowledge base completion,” in *Proc. 53rd Annu. Meet. Assoc. Comput. Linguist., ACL 2015*, Beijing, China, Jul 26-31 2015, pp. 156–166.

REFERENCES

- [263] R. B. Kline, *Principles and practice of structural equation modeling*. Guilford publications, 2015.
- [264] X. Glorot and Y. Bengio, “Understanding the difficulty of training deep feedforward neural networks,” in *Proc. 13th Int. Conf. Artif. Intell. and Statist., AISTATS 2010*, Sardinia, Italy, May 13-15 2010, pp. 249–256.
- [265] L. N. Smith, “Cyclical learning rates for training neural networks,” in *Proc. IEEE Winter Conf. Appl. Comput. Vision, WACV 2017*, Santa Rosa, CA, USA, Mar 27-29 2017, pp. 464–472.
- [266] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” in *Proc. 3rd Int. Conf. Learn. Represent., ICLR 2015*, San Diego, CA, USA, May 7-9 2015.
- [267] F. Marbouti, H. A. Diefes-Dux, and K. Madhavan, “Models for early prediction of at-risk students in a course using standards-based grading,” *Comput. Educ.*, vol. 103, pp. 1–15, 2016.
- [268] E. Variiani, E. McDermott, and G. Heigold, “A Gaussian mixture model layer jointly optimized with discriminative features within a deep neural network architecture,” in *Proc. IEEE Int. Conf. Acoust., Speech and Signal Process., ICASSP 2015*, South Brisbane, QLD, Australia, Apr 19-24 2015, pp. 4270–4274.
- [269] C. E. Rasmussen, “The infinite Gaussian mixture model,” in *Proc. 12th Int. Conf. Neural Inf. Process. Syst., NIPS 1999*, Cambridge, MA, United States, Nov 29 - Dec 4 1999, pp. 554–560.
- [270] A. Kristiadi, S. Dubener, and A. Fischer, “Predictive uncertainty quantification with compound density networks,” 2019, arXiv:1902.01080.
- [271] C. Sakr, Y. Kim, and N. R. Shanbhag, “Analytical guarantees on numerical precision of deep neural networks,” in *Proc. 34th Int. Conf. Mach. Learn., ICML 2017*, Sydney, Australia, Aug 6-11 2017, pp. 3007–3016.
- [272] C. Cortes and V. Vapnik, “Support-vector networks,” *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, 1995.
- [273] H. Ogata, C. Yin, M. Oi, F. Okubo, A. Shimada, K. Kojima, and M. Yamada, “e-book-based learning analytics in university education,” in *Proc. 23rd Int. Conf. Comput. in Educ., ICCE 2015*, Hangzhou, China, Nov 30 - Dec 4 2015, pp. 401–406.
- [274] B. Flanagan and H. Ogata, “Integration of learning analytics research and production systems while protecting privacy,” in *Proc. 25th Int. Conf. Comput. in Educ. ICCE 2017*, Christchurch, New Zealand, Dec 4-8 2017, pp. 333–338.