A holistic approach to trust and reputation management in big data

Leonit Zeynalvand

2020

Leonit Zeynalvand (2020). A holistic approach to trust and reputation management in big data. Doctoral thesis, Nanyang Technological University, Singapore.
<https://hdl.handle.net/10356/146956>

<https://hdl.handle.net/10356/146956>

<https://doi.org/10.32657/10356/146956>

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

Downloaded on 26 Apr 2023 10:23:06 SGT

**NANYANG
TECHNOLOGICAL
UNIVERSITY**

SINGAPORE

A Holistic Approach to Trust and Reputation Management in Big Data

Leonit Zeynalvand

School of Computer Science and Engineering

A thesis submitted to the Nanyang Technological University
in partial fulfilment 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.

04 / Feb / 2021

Date

Leonit Zeynalvand

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.

04 / Feb / 2021

Date

Assoc. Prof. Jie Zhang

Authorship Attribution Statement

This thesis contains material from 4 papers published in the following papers accepted at conferences in which I am listed as an author.

Chapter 3 is published as Leonit Zeynalvand, Jie Zhang, Tie Luo, Shuo Chen. *MASA: Multi-agent subjectivity alignment for trustworthy Internet of Things*. In the Proceedings of 21st International Conference on Information Fusion (FUSION), pp. 2013-2020. IEEE, 2018.

The contributions of the co-authors are as follows:

- Assoc. Prof. Jie Zhang provided the initial project direction and commented on the manuscript drafts.
- I did the literature review and identified the research gaps.
- I formulated the problem and designed the models.
- I collected real data for the experiments.
- I designed the experiments, implemented the models, conducted all the experiments, and analysed the results.
- I prepared the manuscript draft. Assoc. Prof. Tie Luo and Shuo Chen gave me constructive comments on how to revise the manuscript.

One of the benchmark methods which we used for comparison in Chapter 3, Section 3.4.2 is published as Tie Luo, Leonit Zeynalvand. *Reshaping mobile crowd sensing using cross validation to improve data credibility*. In the Proceedings of 2017 IEEE Global Communications Conference (GLOBECOM 2017), pp. 1-7. IEEE 2017.

The contributions of the co-authors are as follows:

- Assoc. Prof. Tie Luo provided the project direction, design, manuscript and formulation.
- I performed model checking and formal verification for the initial and revised formulations.
- I designed the experiments, implemented the models, and conducted all the experiments.

Chapter 4 is published as Leonit Zeynalvand, Tie Luo, Jie Zhang. *COBRA: Context-Aware Bernoulli Neural Networks for Reputation Assessment*. In the Proceedings of AAAI, pp. 7317-7324. AAAI, 2020.

The contributions of the co-authors are as follows:

- Assoc. Prof. Jie Zhang provided the initial project direction and commented on the manuscript drafts.
- I did the literature review and identified the research gaps.
- I formulated the problem and designed the models.
- I collected real data for the experiments.
- I implemented the models, conducted all the experiments, and analysed the results.
- I prepared the manuscript draft. The manuscript was further revised together with Assoc. Prof. Tie Luo.

Chapter 5 is in submission as Leonit Zeynalvand, Tie Luo, Ewa Andrejczuk, Dusit Niyato, Teo SinGee, Jie Zhang. *A Blockchain Enabled Quantitative Approach to Trust and Reputation Management with Sparse Evidence (Extended Abstract)*. In the Proceedings of AAMAS 2021.

The contributions of the co-authors are as follows:

- Assoc. Prof. Jie Zhang provided the initial project direction and commented on the manuscript drafts.
- I did the literature review and identified the research gaps.
- I formulated the problem and designed the models.
- I implemented the models, conducted all the experiments, and analysed the results.
- I prepared the manuscript draft. Assoc. Prof. Tie Luo, Dr. Ewa Andrejczuk, Prof. Dusit Niyato, and Dr. Teo SinGee gave me constructive comments on how to revise the manuscript.

04 / Feb / 2021

Date

Leonit Zeynalvand

Acknowledgement

This journey would not be possible without the support of my family; Caring parents who raised me well, my dear wife who gave me strength to carry on, and my lovely sister for whom I was a role model.

I sincerely want to thank my supervisors, Assoc. Prof. Jie Zhang and Assoc. Prof. Tony T. Luo whose advice and guidance throughout my PhD has brought me where I am. Likewise, many thanks to the other members of my thesis advisory committee, Prof. Dusit Niyato, Assoc. Prof. Bo An, and Dr. Teo SinGee whose valuable comments helped me overcome the challenges along the way.

Finally, I want to express my gratitude to all my friends who shared their valuable experience with me and never denied me their support.

Contents

Acknowledgement	v
Abstract	ix
Lists of Figures	xiv
Lists of Tables	xv
1 Introduction	1
1.1 Motivation	1
1.2 Problems and Proposed Approaches	5
1.2.1 Subjective evidence space	5
1.2.2 Dense evidence space	6
1.2.3 Sparse evidence space	8
1.3 Major Contributions of the Dissertation	9
1.4 Organization of the Dissertation	11
2 Preliminaries and Literature Review	13
2.1 Preliminaries	13
2.1.1 Big Data	13
2.1.2 Internet of Things	15
2.1.3 Data Science and Machine Learning	16
2.1.4 Blockchain	23
2.2 Definition of Trust and Reputation	23
2.2.1 The Interpretations of Trust	24
2.2.2 The Interpretations of Reputation	26
2.2.3 Conclusion	28
2.3 Trust and Reputation Management Systems	28

2.4	Trust in IoT, MANET, and Ubiquitous Computing	31
2.4.1	Summary and Discussion	37
2.5	Trust and Social Media	40
2.6	Trust and Blockchain	45
2.7	Summary	50
3	Subjective Evidence in Trust and Reputation Management	51
3.1	Problem Statement	51
3.2	MASA	55
3.2.1	Advice Alignment	57
3.2.2	Variance-modulated Entropy	58
3.2.3	Rationale of using Decision Trees	61
3.2.4	Computational complexity	61
3.3	A Running Example	62
3.4	Evaluation	68
3.4.1	Simulation Scenario 1: Internet data plan sharing in smart cities	68
3.4.2	Simulation Scenario 2: IoT-based crowd-sensing application	71
3.4.3	Experiment with Real Data	72
3.4.4	Advantages and Limitations	76
3.5	Summary	77
4	Trust and Dense Evidence Space	79
4.1	Problem Statement	79
4.2	Model Encapsulation	81
4.3	Bernoulli Neural Network	83
4.3.1	Topology	84
4.3.2	Depth and width	86
4.3.3	Activation and loss functions	87
4.3.4	Assembling training data	87
4.3.5	Computational complexity	91
4.4	Evaluation	93
4.4.1	Experiment setup	93
4.4.2	Experiment procedure and results	95
4.4.3	Simulation setup	96

4.4.4	Simulation results	97
4.5	Summary	98
5	Trust and Sparse Evidence Space	100
5.1	Problem Statement	100
5.2	Design of BEQA	102
5.2.1	Process Overview	103
5.2.2	Trustworthiness Analysis	105
5.2.3	Proofs	112
5.3	Evaluation	112
5.3.1	Experiment setup	112
5.3.2	Experiment Procedure and Results	114
5.3.3	Simulation procedure and results	115
5.4	Summary	122
6	Conclusion and Future Work	123
6.1	Summary	123
6.2	Real-world Datasets	124
6.3	Suggestions for Future Research	125
	Appendix A	132

Abstract

Big Data is becoming an increasingly important part of everyday life. In this dissertation we consider two types of Big Data, in particular *user-generated data* and *IoT-generated data*. We refer to the Big Data generated by users of online services as user-generated data. Ratings submitted by users for online purchases are examples of user-generated data. We refer to the Big Data generated by devices in the Internet of Things (IoT) as IoT-generated data. The key difference between these two types of Big Data is the density of data. Density here means with what frequency an entity perceives information about another entity. Hence, user-generated data is sparse while IoT-generated data is dense due to the higher frequency of interactions between IoT entities compared to that of social entities. On one hand a sparse dataset results in less accurate knowledge extraction compared to a dense dataset. On the other hand a dense dataset risks more privacy exposure compared to a sparse dataset. Regardless of the sparsity, a dataset may contain subjective information which affects the knowledge extraction. Common examples of knowledge extraction are trust and reputation management (TRM) systems.

This dissertation addresses issues within TRM systems in the emerging Big Data environments. Particularly, issues concerning the subjectivity of the evidence space in such environments, the density of IoT-generated data, and the sparsity of user-generated data are identified and addressed.

First we address subjectivity as an important issue in TRM for Big Data. Subjectivity means that the information provided by each user, represented by an agent, is influenced by the user's individual preference, which can be misleading in trust evaluation. In this dissertation, we seek to align the potentially subjective information with the information seeker's own subjectivity so that the acquired second-hand information is more useful and personalized. Accordingly, we propose a multi-agent subjectivity alignment (MASA) mechanism, which models the subjectivity using a regression technique and exchanges the models among agents as the input to an alignment process. This mechanism substantially

counteracts biases incurred by different agents and improves the accuracy of second-hand information fusion as demonstrated by our simulations. In addition, we also conduct experiments using a real-world dataset which further validates the efficacy of MASA.

Second, we explain how TRM plays an increasingly important role in large-scale online environments with dense evidence space such as multi-agent systems (MAS) and the IoT. One main objective of TRM is to achieve accurate trust assessment of entities such as agents or IoT service providers. However, this encounters an *accuracy-privacy dilemma* when the evidence space is dense as we identify in this dissertation. Thus, we propose a framework called *Context-aware Bernoulli Neural Network based Reputation Assessment* (COBRA) to address this challenge. COBRA encapsulates agent interactions or transactions, which are prone to privacy leak, in machine learning models, and aggregates multiple such models using a Bernoulli neural network to predict a trust score for an agent. COBRA preserves agent privacy and retains interaction contexts via the machine learning models, and achieves more accurate trust prediction than a fully-connected neural network alternative. COBRA is also robust to security attacks by agents who inject fake machine learning models; notably, it is resistant to the 51-percent attack. The performance of COBRA is validated by our experiments using a real dataset, and by our simulations, where we also show that COBRA outperforms other state-of-the-art TRM systems.

Third, we elaborate on how user-generated data with the prevalence of e-commerce applications poses new security challenges that render traditional TRM approaches ineffective, especially in mitigating and discouraging trust attacks such as whitewashing and Sybil attacks. Specifically, there are three challenges. First, user-generated data is increasingly becoming sparse, making the derived trust models not robust to attacks. Second, the cost of attacks has significantly decreased over time due to the widespread presence of bots in e-commerce applications, and thus the traditional assumption of majority users being honest often no longer holds. This also further exacerbates the previous challenge. Third, e-commerce applications involve publicity (user-generated data originating from influencers and paid users), which is not formulated in existing trust models. In this dissertation, we propose a new TRM framework called BEQA. It uses Blockchain to transform multiple disjoint and sparse sets of user-generated data into a single dense-enough dataset, and formulates the cost of Sybil attacks using Blockchain transaction fees. In particular, publicity is formulated as a *whitewashing deposit* such that a higher level of publicity will impose higher cost on the Sybil attack. To evaluate the performance of BEQA, we conduct experiments using three real datasets, and additional simulations for more extensive scenarios.

Our results show that BEQA outperforms the state-of-the-art TRM models. For instance, BEQA yields more accurate trustworthiness assessment and more rapid mitigation on Sybil attacks.

To summarize, addressing these three issues enables us to take a holistic approach towards trustworthiness in Big Data. Particularly, issues concerning the subjectivity, density, and sparsity of the evidence space do not necessarily co-exist in one Big Data environment. Hence, a *one-fits-all* TRM solution is often not viable in Big Data. Instead, in this dissertation we propose a holistic collection of effective solutions which address such issues.

Keywords: Trust, Reputation, TRM.

List of Figures

1.1	Analysis of the necessity of trust assessment in IoT interactions based on interaction beneficiaries.	3
2.1	Global data growth. (a) The amount of data that is generated globally per year; (b) The ratio of data generated in each year to the total amount in a ten year period. We obtained the figures using measurements reported by [IDC, 2018].	14
2.2	Decision Tree assessment of borrower trustworthiness.	19
2.3	Feed-forward neural network architecture.	21
2.4	A mind-map to illustrate our proposed categorization of the different definitions of trust in the existing literature.	25
2.5	A mind-map to illustrate our proposed categorization of the different definitions of reputation in the existing literature.	27
2.6	Trend analysis for IoT from 2004 to 2018 [Google, 2018b]	39
2.7	Trend analysis for e-commerce from 2004 to 2018 [Google, 2018a]	39
3.1	An overview of MASA approach.	56
3.2	Variance-modulated entropy for posterior advice random variable A with a range of $\{1, 2, \dots, 10\}$ ($\sigma_{U\{1,10\}}^2 = 20.25$). (a) 3D plot; (b) Contour plot. . .	60
3.3	Decision trees of different agents. (a) Decision tree regression for environmental sensors; (b) Decision tree regression for smart cars.	65
3.4	Ratio of MAE for different approaches to that of the baseline under different sparsities where (a) 30% (b) 50% (c) 70% of the advisor agents are smart cars.	69
3.5	Ratio of RMSE for different approaches to that of the baseline under different sparsities where (a) 30% (b) 50% (c) 70% of the advisor agents are smart cars.	70

3.6	Simulation results for IoT-based crowd-sensing application. (a) probability distribution of the original and the reshaped crowd-sensed data; (b) probability distribution of the reshaped crowd-sensed data with and without MASA.	72
3.7	Subjectivity of a random Movie-lense user modeled as decision trees in MASA.	74
3.8	Subjectivity of a random Movie-lense user modeled as decision trees in MASA.	75
4.1	The COBRA framework.	82
4.2	Experiment results: (a,b) ACC and RMSE of different approaches; (c,d) Moving average of prediction and training time for BNN compared to Dense; (e,f) Convergence of validation accuracy and loss for BNN compared to Dense.	94
4.3	Simulation results: (a) Accuracy of COBRA (in percentage) versus different distributions of Φ parameterized by α and β (in log scale). (b) Contour plot of (a); The color spectrum from <i>dark</i> to <i>light</i> represents accuracy from <i>low</i> to <i>high</i> ; (c) The Beta distribution function (adapted from [Horas, 2018])	97
5.1	Sequence diagram of the main interactions in BEQA. Interaction 1 illustrates the feedback submission process. Interaction 2 illustrates trustworthiness enquiry process.	105
5.2	Publicity filter (vertical dashed line) applied to various distributions of transaction fee expenditure: (a) Normal distribution with $\mu = 100$ and $\sigma = 10$; (b) Lognormal distribution with $\mu = 0$ and $\sigma = 1$; (c) Chi-Square distribution with 5 degrees of freedom; (d)Beta distribution with $\alpha = 4$ and $\beta = 2$; (e) Mixture of Gaussian distributions with $\mu_1 = 15, \mu_2 = 35, \mu_3 = 120$ and $\sigma_1 = \sigma_2 = \sigma_3 = 4$ and $w_1 = 0.6, w_2 = 0.2, w_3 = 0.2$; (f) Mixture of Gaussian distributions with $\mu_1 = 15, \mu_2 = 100, \mu_3 = 120$ and $\sigma_1 = \sigma_2 = \sigma_3 = 4$ and $w_1 = 0.6, w_2 = 0.2, w_3 = 0.2$; (g) Mixture of Gaussian distributions with $\mu_1 = 30, \mu_2 = 150$ and $\sigma_1 = \sigma_2 = 5$ and $w_1 = 0.66, w_2 = 0.33$; (h) Beta distribution with $\alpha = 0.1$ and $\beta = 0.1$	106
5.3	Performance of BEQA compared to benchmark methods on different datasets using: (a) RMSE evaluation metric; (b) MAE evaluation metric; (c) MSLE evaluation metric.	114

5.4	Simulation results: (a) Cost of a successful Sybil attack for entities with different publicity deposits at each time-step of the simulation; (b) r_e for entities with different publicity deposits at each time-step of the simulation; (c) RMSE for entities with different publicity deposits at each time-step of the simulation.	118
5.5	RMSE of BEQA compared to benchmark methods for different distributions of Φ : (a) Φ follows log-normal distribution with a mean of $\mu = 0$ and standard deviation of $\sigma = 1$; (b) Φ follows log-normal distribution with a mean of $\mu = 0$ and standard deviation of $\sigma = 1.5$; (c) Φ follows log-normal distribution with a mean of $\mu = 0$ and standard deviation of $\sigma = 2$	119
6.1	UX Survey Results: (a) The polar bar plot of users' major concerns w.r.t different Big Data environments; (b) The box plot of the statistical distribution of how much each user's time is worth and how much they would pay to record a feedback, if they indicated Sybil as one of their concerns; (c) The violin plot of the density of the Iot-generated evidence and that of the user-generated evidence (y-axis is in logarithmic scale).	130

List of Tables

2.1	Comparison of the discussed approaches.	37
2.2	Comparison of the discussed approaches.	44
2.3	Comparison of the discussed approaches.	49
3.1	MAE and RMSE of different approaches.	76
5.1	Definitions and Notations	104

Chapter 1

Introduction

1.1 Motivation

Although there are various interpretations of trust in the existing literature, it is commonly accepted that trust in an entity is the belief that the information shared by the entity is truthful or a promise made by the entity is valid [Hussain and Chang, 2007]. Reputation is a historical indicator of this belief [Resnick et al., 2000]. That is, whenever one trusts an entity, the difference between the observed outcome and the expected outcome (belief) constitutes an *evidence*. This evidence is then used to update the entity's reputation. In this dissertation we adopt the above definition for trust and reputation. Hence, evidence is the fundamental element of Trust and Reputation Management (TRM) and plays a decisive role in computational trust. When computational methods are used to process the available evidence and measure trustworthiness and reputation, the approach is called computational trust and reputation [Braga et al., 2018]. This allows for the creation of systems for the automation of TRM.

TRM systems are critical to a broad spectrum of Big Data environments such as e-commerce¹, the Internet of Things (IoT), and multi-agent systems (MAS), where agents² are more autonomous and tend to have more interactions with each other. Without a reliable TRM system, such interactions will be significantly hindered due to lack of trust in services or information provided by other agents. Formal contracts such as Service Level Agreements (SLA) are hard to enforce in such open environments because of high cost and lack of central authorities. There is a vast literature on the TRM systems in a broad spectrum

¹According to a recent IDC report [Rydning, 2018], e-commerce accounts for a substantial and growing share of the total big data.

²Throughout this dissertation, we use the term agents in a broader sense which is not limited to agents in MAS, but also includes IoT service providers and consumers as well as other similar cases.

of application domains [Yu et al., 2010, Yu et al., 2013, Patel and Jhaveri, 2015]. The key difference between different TRM approaches is the mechanism which they use to collect, share, process, and aggregate evidence. Hence, in TRM, evidence plays a decisive role.

However, the characteristics of the evidence space in a TRM model is application specific. That is, whether or not the evidence possesses certain quantitative or qualitative properties depends on the application and the environment for which a particular TRM model is deployed. This is of great importance because such properties dictate what TRM challenges exist in what environment and application. In this thesis we have identified three classes of such properties for trust evidence in Big Data: 1) Subjective evidence space, 2) dense evidence space, 3) sparse evidence space, respectively covered in Chapter 3-5. Hence, a holistic (comprehensive) TRM thesis in Big Data should study all of these properties which is what we do in this thesis.

Because Big Data has a broad spectrum of applications in disparate environments, the above properties are often not observed all together in the same Big Data application or environment. A clear example of this, is easy to spot in sparse evidence space and dense evidence space. In TRM systems, density of evidence means with what frequency an entity perceives information about another entity. That is, although there are Big Data applications in which the evidence space for TRM is dense, and also there are Big Data applications with sparse evidence space. However, there is no Big Data application in which the evidence space for TRM is simultaneously dense and sparse because these two terms are antonyms and hence by definition what is dense is not sparse. Consequently, a TRM model which is particularly designed for dense evidence space (Chapter 4) benefits most from that particular evidence space. However, a holistic and comprehensive thesis on TRM in Big Data should also consider sparse evidence space which is what Chapter 5 of this thesis is about.

Different Big Data environments have different trust requirements. For instance, in e-commerce TRM is needed in all major interactions [Salam et al., 2005], however in IoT, TRM is only necessary in certain interactions. The reason is, the interactions in e-commerce always have different beneficiaries due to the commercial nature of those interactions (i.e. There is at least a buyer and a seller in an existing interaction). However, in IoT an interaction can have a single beneficiary. Such interactions do not require TRM since there is only one beneficiary involved. For example, the interactions between a user *Alice* and her *smart home*³ or the interactions between the *wearables* of Alice and her smart home do

³See Section 2.1.2 for definition.

Figure 1.1: Analysis of the necessity of trust assessment in IoT interactions based on interaction beneficiaries.

not require TRM. The reason is that the sole beneficiary for such interactions is *Alice* and hence her devices are often connected as trusted pairs. However, if two IoT devices have different beneficiaries, then TRM is crucial. Fig. 1.1 illustrates the necessity of trust in IoT interactions based on interaction beneficiaries.

Moreover, different Big Data environments generate different types of Big Data. In this dissertation we consider two types of Big Data adopted from [Ghani et al., 2019, Marjani et al., 2017] as follows:

- **User-generated data.** We refer to the Big Data generated by users of online platforms as User-generated data. Likes, Up-votes, user ratings and reviews are examples of User-generated data in the context of TRM.
- **IoT-generated data.** We refer to the Big Data generated by devices in the Internet of Things as IoT-generated data. For example, whether or not a promised service level

agreement (SLA) between two web-service agents is violated constitutes an example of IoT-generated data relevant to TRM.

Earlier in this chapter, we explained that evidence is the basis of computational trust. Evidence in trust can be first-hand or second-hand. First-hand evidence is the observations of the trustor with regards to the trustworthiness of others while second-hand evidence is based on the observations of others.

All evidence is data, but not all data is evidence. With this respect the two classes of data examined in this thesis, namely user-generated data and IoT-generated data, are relevant to a Trust and Reputation Management (TRM) system only if they are evidence. A feedback submitted by a customer about a merchant is an example of user-generated data which is also considered trust evidence because it reflects the customer's observation of the trustworthiness of the merchant. Similarly, the data generated during Quality of Service (QoS) monitoring of IoT services is the example of IoT-generated data which is also trust evidence. This is because such data reflects whether the service provider IoT agent can be trusted to uphold certain Service Level Agreement (SLA) or not.

In other words, IoT-generated data relevant to TRM are in fact feedbacks submitted by IoT agents about other IoT agents. While user-generated data relevant to TRM are feedbacks submitted by users (e.g. buyers) about other users (e.g. sellers). This distinguish is of great importance mainly because of the difference in the density of evidence. In TRM systems, density of evidence means with what frequency an entity perceives information about another entity. In IoT generated data the evidence space is dense while in User-generated Data the evidence is sparse. This is because IoT is a system of interconnected devices which exchange data without human interactions [Gubbi et al., 2013]. Not requiring human intervention allows for a high level of interactivity in IoT among devices. This leads to the generation of large volumes of dense interaction data. The reason is, the percentage of the devices in the system that a single device can interact with over a period of time is orders of magnitude greater than the percentage of the humans in a social system that a single human can interact with over the same period of time [Edemacu and Bulega, 2014]. For instance, a *buyer-seller* interaction on an e-commerce website requires a willing buyer to spend the certain amount of time, effort and money to purchase a product which is subject to the seller's willingness to sell and availability of the stock. While the buyer's resources such as time, and money are often limited, his options (sellers to choose) are many.

On a separate vein, regardless of the sparsity, Big Data evidence may often be subjec-

tive. Subjectivity means that the information provided by each user (represented by an agent in IoT) is influenced by the user’s individual preference, which can be misleading in trust evaluation [Jøsang et al., 2007]. Therefore, having explained the importance of evidence as the basis of TRM, the density and subjectivity of the evidence pose various challenges to TRM systems in Big Data. In the following section we expand on each of these challenges and discuss our proposed approach to tackle them.

1.2 Problems and Proposed Approaches

1.2.1 Subjective evidence space

Computational trust systems generally calculate trust values based on two types of data: first-hand evidence (experiences of self) and second-hand evidence (advice from others).

However, second-hand evidence often suffers from high subjectivity. To further clarify what we mean by subjectivity, let us first consider an analogy where Alice has high tolerance for spicy food while Bob’s tolerance for spicy food is low. Hence, whenever Alice advises Bob that a certain dish is *slightly spicy*, it is probably of Bob’s best interest to avoid the dish. This explains the subjectivity in user-generated Big Data. However, The issue of subjectivity is not limited to people and man-made data. In fact, data which is not sourced from humans can also be subjective. Consider a crowd-sensing example in smart transportation domain where agents *A* and *B* respectively represent a smart car and a smart phone. Agent *A* has a speed sensor precision of *1mph*, while agent *B* has a precision of *5mph*. So agent *B* will consider any crowd-sensed measurement which deviates less than *5mph* from its belief of ground truth as *truthful* while agent *A* may consider the same measurement as *not truthful*.

Hence, we explain that subjectivity in IoT-generated Big Data comes from the fact that IoT sensors have different sensing capabilities with different precision. So the evidence that each IoT device can provide on an arbitrary matter is subject to its sensing capabilities, sensing precision, etc. Hence, subjectivity in IoT generated data is a consequence of heterogeneous sensing capabilities.

To shed more light on the matter we explain a common crowd-sensing scenario in traffic prediction. Google Maps and many other navigation apps provide crowd-sensed traffic predictions [Nagy and Simon, 2018]. Heterogeneous IoT devices such as smart watches, smart cars, and smart phones contribute to this traffic prediction task. Their contribution

is two folds; Some contribute by anonymously providing the sensor readings in a periodic fashion. While some other devices contribute by rating the average of the sensor readings provided by the other group. Now consider that the speed sensor of a smart car might have a different precision compared to the speed sensor of a smart phone. Hence in a scenario where these two IoT devices have to submit a feedback about a crowd-sensed congestion on the same road where both of these devices are located, the device with less sensitive speed sensor might not record any movement while the more sensitive sensor might record some slow movement. This can result in a situation where the device with less sensitive speed sensor rates the crowd-sensed traffic prediction as inaccurate simply because its sensors are not sensitive enough to record movement. Hence in this thesis we say that the feedbacks provided by IoT devices can be subjective to the sensing capabilities of such devices and this is what we mean in this thesis by subjectivity in IoT generated Data.

Despite the vast literature on trust and reputation management systems [Yu et al., 2013], there is no generally accepted solution for subjectivity in trust. Authors in [Fang et al., 2012] assumed strong naive independence between the attributes that influence the subjectivity which does not hold for real-world applications (e.g. knowing that a phone has a low resolution camera implies an old model and hence decreases the odds for it to have a near-field communication sensor). Clustering is used in [Fang et al., 2014] to group similar-thinking users together based on their provided ratings. However, it dictates a centralized approach which is not feasible for decentralized applications.

In this dissertation, we propose a multi-agent subjectivity alignment (MASA) mechanism, in which the process of how each agent evaluates the outcome of its interactions with other agents, is modeled by a regression technique. The models are then exchanged when agents offer advice (referred to as second-hand evidence) to each other, followed by an alignment process. This alignment mechanism substantially counteracts biases incurred by other agents. Each aligned advice then is propagated to the underlying trust and reputation management system to be fused with the rest of second-hand evidence using uncertainty-aware weights which are computed with our proposed method. This improves the accuracy of the second-hand evidence fusion.

1.2.2 Dense evidence space

Earlier we explained that IoT-generated Big Data is dense. This allows for more accurate trustworthiness assessment. However, what form the second-hand evidence should take

has been largely overlooked. This engenders an important issue which we refer to as the *accuracy-privacy dilemma*. To illustrate this, suppose an arbitrary agent *Alice* consults another agent *Judy* about how trustworthy *Bob* is. One way is to let *Judy* give a trust score or rating about *Bob* [Yu et al., 2013], which is the approach commonly adopted in the trust research community. This approach is simple but loses the *context information* of the interactions between agents. For example, the context could be the transaction time and location, and service provided by an agent during off-peak hours could have higher quality (more SLA-conformant) than during peak hours. Without such context information, trust assessment based on just ratings or scores would have lower accuracy. On the other hand, another method is to let *Judy* reveal her entire interaction history with *Bob* (e.g., in the form of a detailed review), which is the approach commonly used in recommender systems. Although the information disclosed as such is helpful for trust assessment of *Bob*, it is likely to expose substantial privacy of *Bob* and *Judy* to *Alice* and possibly the public.⁴

Hence, we need to provide a clear definition of what information is considered private and from whom it should be protected. In this thesis, we consider the context information of a feedback as private. A feedback on an interaction (e.g. feedback of an agent about a service provided by a service provider) can have many context features. Examples of context features are: time of the service, date of the service, location of the service, etc. This information (as we show in our evaluation of Chapter 4) is very beneficial for an accurate trust analysis. However, as we consider them private, they need to be protected from honest-but-curious parties who would otherwise seamlessly gain knowledge on the private aspects of the users of a Trust and Reputation Management (TRM) system. Regardless of whether this information provides enough incentive for an adversary to attack, its exposure to honest-but-curious parties makes the users of the TRM system reluctant to participate. The goal in Chapter 4 of this thesis is to mitigate this type of privacy exposure while still benefiting from the positive effect of context information of a feedback on the accuracy of trust assessment.

Therefore, a coarse-grain quantification of the notion of privacy suits best in our discussion of the privacy. With a coarse-grain quantification, the direct exposure of the context features in the evidence space retains *no* privacy, the non-exposure (i.e. context agnostic TRM) retains *full* privacy, and obfuscated/distorted exposure retains *some* privacy which is more desired than *no* privacy. The goal in dense evidence space is to retain *some* privacy

⁴Recommender systems can take this approach because they are generally considered trusted intermediaries, and they focus on preference modeling rather than trust and reputation modelling.

and *high* accuracy. The former is achieved by model encapsulation as explained in Section 4.2, while the latter is achieved by our proposed BNN in Section 4.3 and evaluated in Section 4.4.

Specifically, to address this accuracy-privacy dilemma, and in the meantime avoid relying on a trusted third-party which is often not available in practice, we propose a framework called *Context-aware Bernoulli Neural Network based Reputation Assessment* (COBRA). It encapsulates the detailed second-hand evidence using machine learning models, and then aggregate these model using a Bernoulli neural network (BNN) to predict the trustworthiness of an agent of interest (e.g., an IoT service provider). The encapsulation protects agent privacy and retains the context information to enable more accurate trust assessment, and the BNN accepts the outputs of those ML models and the information-seeking agent's (*Alice*' as in the above example) first-hand evidence as input, to make more accurate trustworthiness prediction (of *Bob* as in the above example).

1.2.3 Sparse evidence space

Trust is crucial in Big Data systems which concerns protecting users from malicious activities. Trust Management is used in a broad range of applications such as e-commerce, e-learning, e-health, and social-media [Trcek and Brodnik, 2013]. In such applications, users share evaluation about products they bought, services they used, and information shared by others [Hassani et al., 2018]. Henceforth, we refer to such products, services and information as *entities*, and the evaluation shared by users about these entities as *feedback*, where the accumulation of feedbacks creates evidence, on which trust models can be built to assess the trustworthiness of various entities.

The proliferation of web-based and mobile applications has led to the formation of a sparse evidence space. In other words, users have more options on where to share information, making the evidence space become sparser. As a result, trust models calculated from the sparse data become less accurate [Kastidou et al., 2009]. For instance, a merchant can list its products on both Amazon and Alibaba, and hence, may have some user feedbacks on Amazon and some on Alibaba; in fact, users are free to leave their feedbacks on many social media as well, such as Facebook, Twitter, YouTube, LinkedIn, and many blogging sites. In such circumstances, each individual set of opinions is often insufficient to derive a reliable assessment of the trustworthiness of a merchant.

The sparsity of evidence space in User-generated Big Data has two consequences. First,

the traditional assumption that the majority of users in TRM are honest does not always hold [Sigala and Gretzel, 2017]. The reason is that attackers now have more incentives to launch malicious activities as web-based and mobile applications are becoming more and more popular, and submitting and disseminating untrustworthy feedback is free. This makes the TRM within such applications vulnerable to *whitewashing* and *Sybil attacks* [Levine et al., 2006]. In the whitewashing attack, a malicious user with bad reputation deletes its identity and rejoins the system with a new identity. In the Sybil attack, a malicious user uses multiple identities to take unwarranted advantage (e.g., to spread fake information).

Second, since publicity is a crucial catalyst for emerging e-commerce applications, it is common to make use of paid reviews or social media influencers to populate the sparse set of feedbacks. However, publicity may have adverse effect on the credibility of evidence aggregation in TRMs, because most TRMs do not distinguish between the data provenance being influencers or normal users.

In this dissertation, we propose a new trust framework called Blockchain Enabled Quantitative Approach (BEQA) to address the above problems. Blockchain is an immutable record of transactions maintained by a peer-to-peer network using proof-of-work [Zhang et al., 2019]. The immutability of records prevents stealthy data modification, and thus enables participating TRMs to safely access the feedbacks about common entities without having to trust one another. As a consequence, the TRMs have access to more evidence, and therefore maintain an immutable and much more reliable trust model. This decreases sparsity and allows for a more accurate trustworthiness assessment [Kastidou et al., 2009]. To address the two consequences of sparse evidence, we design a weighting mechanism for the feedbacks using Blockchain transaction fees. We use the transaction fees to distinguish between feedbacks posted by influencers and normal users, where the publicity of an entity is formulated as a whitewashing deposit. The cost of Sybil attacks is then jointly determined by the whitewashing deposit and the proposed weighting mechanism. This allows us to control the Sybil attack cost using the whitewashing deposit whereby increasing whitewashing deposit will increase the cost of the Sybil attack.

To the best of our knowledge, BEQA is the first approach to address trust attacks such as Sybil and whitewashing by quantifying the cost of feedbacks using Blockchain.

1.3 Major Contributions of the Dissertation

We make the following contributions in this dissertation.

- We address the subjectivity issue (problem [1.2.1](#)) and propose MASA which models the subjectivity using a regression technique and exchanges the models among agents as the input to an alignment process. MASA can be seamlessly incorporated into existing trust and reputation management systems as a plugin.
- We demonstrate the effectiveness of MASA by conducting extensive simulations and experiments with a real dataset.
- We identify the accuracy-privacy dilemma in dense evidence space (problem [1.2.2](#)) and propose COBRA to solve this problem using a model encapsulation technique and a Bernoulli neural network. COBRA preserves privacy by encapsulating second-hand evidence using ML models, and makes accurate trust predictions using BNN which fuses both first-hand and second-hand evidence, where the valuable context information was preserved by the ML models.
- The proposed BNN yields more accurate predictions than the standard fully-connected feed-forward neural networks, and trains significantly faster. In addition, it is also general enough to be applied to similar tasks when the input is a set of probabilities associated with Bernoulli random variables.
- The design of COBRA takes security into consideration and it is robust to fake ML models; in particular, it is resistant to the 51-percent attack, where the majority of the models are compromised.
- We evaluate the performance of COBRA using both experiments based on a real dataset, and simulations. The results validate the above performance claims and also show that COBRA outperforms other state-of-the-art TRM systems.
- We address evidence sparsity in user-generated Big Data (problem [1.2.3](#)) by proposing a novel trust framework called BEQA that uses Blockchain to transform multiple disjoint sets of TRM evidence into a single dense set of evidence. This enables different TRMs to safely access the feedbacks about common entities.
- We make a novel design for BEQA to be resistant to trust attacks (whitewashing and Sybil in particular). Specifically, we formulate publicity as a whitewashing deposit such that a higher level of publicity will impose higher cost on the Sybil attacks.

- We evaluate the performance of BEQA using experiments with three public datasets. The results validate the performance advantages of BEQA over state-of-the-art trust models.
- We also conduct extensive simulations to verify various computational aspects of BEQA and demonstrate its dynamics throughout the time steps in multiple scenarios.

1.4 Organization of the Dissertation

The remaining of this dissertation is organized as follows. In Chapter 2 we first discuss the preliminaries and provide background information about the methods and technologies used in this dissertation. Next, we conduct a thorough literature review and survey the related works.

In Chapter 3, we address subjectivity as an important issue in trust management for Big Data. First, we provide the problem statement in Section 3.1. Second, we propose a subjectivity alignment mechanism (MASA) in Section 3.2.1. Next we enhance MASA with our proposed entropy based variance-modulated weighting mechanism in Section 3.2.2. Then, we provide a running example of MASA in Section 3.3. We evaluate MASA using two simulation scenarios in Section 3.4.1 and Section 3.4.2 in addition to extensive experiments with real data in Section 3.4.3. Finally, we conclude Chapter 3 in Section 3.5.

In Chapter 4, we propose COBRA, a context-aware trust assessment framework for high-density Big Data environments such as IoT. First, we provide the problem statement in Section 4.1. Next, We demonstrate the technical aspects of COBRA in Section 4.2 and 4.3. Then we compare COBRA with state-of-the-art TRM systems using experiments with real data in Section 4.4.1-2. Moreover, to investigate the effectiveness of COBRA especially with respect to extreme scenarios which may not be often observed in real world we also conduct extensive simulations in Section 4.4.3-4. Finally, we conclude Chapter 4 in Section 4.5.

In Chapter 5, we address the sparsity of user-generated data and the TRM threats which it poses. First, we provide the problem statement in Section 5.1. Second, we discuss our proposed trust framework, BEQA, in Section 5.2. Next, we present the evaluation results of BEQA in Section 5.3. Our evaluation is two folds as we evaluate BEQA using experiments with real datasets in Section 5.3.1 as well as using extensive simulations with multiple scenarios in Section 5.3.2. Finally, we conclude Chapter 5 in Section 5.4.

In Chapter 6, we conclude this dissertation and provide several future research directions.

Chapter 2

Preliminaries and Literature Review

2.1 Preliminaries

2.1.1 Big Data

Data is becoming an increasingly important part of everyday life. Take the analogy of oil which led to industrial revolution. It is commonly accepted that data is the new oil [Spiekermann et al., 2015]. The commercial value of data can be attributed to its usage in various industries. [Labrinidis and Jagadish, 2012] categorise data usage as follows:

- **Business Intelligence** . Data is used to derive insights on various business processes. These insights enable the enterprise to optimise corresponding processes and minimize the cost-benefit ratio.
- **Social Media**. Content creation and dissemination are the core of all social media platforms. Henceforth, two key elements of such platforms are data and data communication.
- **Visualization**. Enterprises often use data dashboards to visualise the status of projects, tasks, manufacturing, etc,. This facilitates project management and allows for easier auditing.
- **Fraud Detection**. Data is evidence and hence storing and analysing data records enables detection of fraudulent activities.
- **Soft Security**. In a broad spectrum of large scale online systems such as e-commerce, e-learning, etc,. where bad actors exist, user reviews are commonly used as an invisible and unobtrusive solution.

- **Recommender Systems.** User data can reveal user preferences which is crucial for developing effective marketing strategies. Therefore, recommender systems often explore such preferences by analysing user data.

On one hand, as illustrated in Fig. 2.1(a), the amount of data that is being generated globally has increased 25 folds in the last ten years according to [IDC, 2018]. In fact, the data generated in the past three years account for more than half of all data generated in the past ten years (see Fig. 2.1(b)).

Figure 2.1: Global data growth. (a) The amount of data that is generated globally per year; (b) The ratio of data generated in each year to the total amount in a ten year period. We obtained the figures using measurements reported by [IDC, 2018].

The term Big Data is used to describe large amounts of data. In this dissertation we consider two types of Big Data as follows:

- **User-generated Data.** We refer to the Big Data generated by users of online platforms as user-generated Data. Likes, Up-votes, user ratings and reviews are examples of User-generated Data.
- **IoT-generated Data.** We refer to the Big Data generated by devices in the Internet of Things as IoT-generated Data.

The key difference between these two types of Big Data is the density of the data. Density here means with what frequency an entity perceives information about another entity. User-generated Data is sparse while IoT Generated Data is dense. What causes this difference is the distinctive characteristics of the Internet of Things.

2.1.2 Internet of Things

The Internet of Things (IoT) is a systems of interconnected devices which exchange data without human interactions [Gubbi et al., 2013]. Not requiring human intervention allows for a high level of interactivity in IoT among devices. This leads to the generation of large volumes of dense interaction data. The reason is, the percentage of the devices in the system that a single device can interact with over a period of time is orders of magnitude greater than the percentage of the humans in a social system that a single human can interact with over the same period of time. This explains why Social Big Data is sparse.

IoT has a broad spectrum of applications. In this dissertation we adopt the following categorization from [Liu and Zhou, 2012]:

- **Smart Home:** Referred to a living environment where home appliances are equipped with connectivity and artificial intelligence to deliver more convenient living experience.
- **Smart City:** Referred to the generalization of smart homes to the scale of a city. However, there is a major difference which makes the realization of smart cities fundamentally different and substantially more challenging than smart homes, that is, the IoT devices within a house belong to the individuals living inside the house (usually members of the family) resulting in the presence of intrinsic trust, while in a smart city the trust needs to be established and evaluated.
- **Smart Transport:** Ride-sharing applications are familiar examples of this domain. Moreover, with the prevalence of Artificial Intelligence (AI), driver less cars are approaching to realization.
- **Smart Retail:** Vending machines which are connected to the internet and can monitor sales and stocks are good examples of this domain. A more recent example is cashier-less shopping [AmazonGo, 2016], where IoT technologies are used to tag goods and products, which will enable tracking throughout the shopping experience to eliminate the need for scanning the purchased items at the cashier. Instead, the shoppers will tap their phone at the exit while leaving the shop and the purchased items will be automatically deducted from their balance.
- **Smart Manufacturing:** Industrial IoT is another promising domain and creates unique business opportunities, particularly for networking and data analytic compa-

nies who can offer manufacturers the ability to remotely monitor, analyse, and control the manufacturing processes in different factories.

- **Wearables:** From health monitoring to fitness tracking and smart accessories, wearables have retracted a significant market share.

In IoT, devices interact with one another for various reasons. We categorise IoT systems based on interactions as follows:

1. **Monitoring.** Some IoT systems such as [Hopkins and Hawking, 2018, Fiedler and Meissner, 2013], are designed to facilitate the tracking and monitoring of a certain process or situation. For instance, in logistics industry geo-positioning sensors are used to periodically record the exact location of the trucks. These location data is then shared with other trucks to optimise time management, and allow for better route planning.
2. **Control.** It is not uncommon to have IoT systems where interactions are only for the purpose of carrying out a task in a collective fashion. Examples of such systems are the Botnets [Kolias et al., 2017a] which are used to carry out a Distributed Denial of Service (DDoS) attack where the objective of the attacker is to make a certain resource temporarily unavailable.
3. **Monitoring and Control.** Most IoT systems such as [Patti and Acquaviva, 2016, Hossai et al., 2017] involve both types of interactions. For instance, in areas such as smart city, autonomous vehicles, automation and smart manufacturing, home and building security, etc., a combination of control and monitoring is used.

IoT applications which involve monitoring generate huge amount of data. This data is then processed using complex analytical and statistical methods to derive meaningful models and identify patterns. Such methods are referred to as *Data Science*.

2.1.3 Data Science and Machine Learning

Data Science is a field that uses methods from various disciplines such as statistics, computer science and domain knowledge for principled extraction of hidden patterns from raw data using certain amount of computational power [Provost and Fawcett, 2013]. The set of techniques used by data scientists is commonly referred to as Machine Learning.

[[Qiu et al., 2016](#)] break down Machine Learning techniques with regards to data pre-requisites into three main categories as follows:

- **Supervised Learning.** This type of techniques require the data to be labeled. They then use the labeled data to make predictions about a new data point.
- **Un-supervised Learning.** This type of techniques do not require the data to be labeled and are used to discover the underlying patterns in the unlabeled data.
- **Reinforcement Learning.** In this type of techniques predefined data schemes are not required, instead data is generated while the learner (e.g. a robot) is interacting with its surrounding environment.

On a separate vein, [[Arun and Jabasheela, 2014](#)] provides a task-oriented categorization of Machine Learning techniques as follows:

- **Classification.** This is a supervised learning approach in which a discrete (i.e. categorical) value is predicted given a data point. For instance, methods such as Decision Tree Classifier, Naive Bayes, Bayesian network, Gaussian Process Classifier, and Artificial Neural Network can all be used for classification.
- **Regression.** This is also a supervised learning approach. However, in a regression problem a continuous (i.e. numerical) value is predicted given a data point. Linear and polynomial regression, Logistic regression, Support Vector Regression, and Decision Tree Regression are all examples of such methods.
- **Clustering.** This is an unsupervised learning approach in which the goal is to divide the data points into groups such that the data points in each group are more similar to each other than to the data points in other groups. K-Means, DBSCAN, and Hierarchical Clustering are examples of such methods.

In the remaining chapters of this dissertation, from the above Machine Learning methods we use the followings:

Logistic Regression

This is a statistical model which uses a logistic function to explain the relationship between independent variables and a binary dependent variable [[Kleinbaum et al., 2002](#)]. For

instance the logistic regression of gender based on height can be expressed by:

$$\ln \left(\frac{P}{1-P} \right) = a + bX$$

where X is the independent variable representing the height of a person and P is the probability of the dependent variable representing the gender of the person. Therefore, P can be obtained from:

$$P = \frac{e^{a+bX}}{1 + e^{a+bX}}$$

In real world applications, Logistic Regression is often used to model the probability of a binary event such as trade (buy or sell), outcome (good or bad), trustworthiness (trustworthy or not), etc., [Akinci et al., 2007].

Decision Tree

This is a decision support mechanism which models the decisions and their consequences [Song and Ying, 2015]. In its simplest form, a Decision Tree is a tree-like data structure in which the root node represents a condition over an independent variable. Each child of the root node: 1) indicates a distinct outcome of the parent's condition. For instance, in a binary Decision Tree it indicates whether the condition holds or not. 2) represents a condition over an independent variable or is a leaf node that indicates a value for the dependent variable. Hence, each leaf node contains all the data points which meet the conditions of all its parents.

For instance, consider the scenario where a borrower applies for a loan from a lender. The borrower promises that he can pay the loan back. The lender then needs to assess whether he can trust the borrower to keep his promise or not. A Decision Tree is one of the methods that can be used for this assessment. Fig. 2.2 illustrates an example of this method.

The key factor in a decision tree construction is how to split a node into child branches. [Perner, 2015] categorises commonly used splitting criteria as follows:

- **Gini Impurity.** This measures how pure a node is. Hence, the less number of classes a node contain the more pure it becomes. We can formulate this concept by:

$$I_G(node) = 1 - \sum_{i=1}^N (p_i)^2$$

Figure 2.2: Decision Tree assessment of borrower trustworthiness.

where N is the total number of classes in the *node* and p_i is the distribution of class i in the *node*. With this measure the criteria for splitting a node is to minimize the impurity of that node.

- **Information Gain.** This measures how much information a node contains. According to Shannon Theory of Information, the higher uncertainty is involved in an event the less information the event contains. Henceforth, information gain is defined by:

$$I(\text{node}) = 1 - \sum_{i=1}^N -(p_i) \log_2(p_i)$$

where N is the total number of classes in the *node* and p_i is the distribution of class i in the *node*. With this measure the criteria for splitting a node is to maximize the information gain of that node.

- **Variance.** This measure is suitable for regression problems while the previous two measures consider the number of classes and hence are suitable for classification problems. With this measure the criteria for splitting a node is to minimize the av-

erage variance of the children of that node (see [Sprinthal and Fisk, 1990] for more information on variance).

Naive Bayes

This is a probabilistic classifier which uses the Bayes theorem with strong independence assumptions among the data features. According to Bayes theorem the posterior probability of a data label $L_k : k \in 1, 2, \dots, K$ where K is the total number of classes given a data feature x is expressed by:

$$p(L_k|x) = \frac{p(L_k)p(x|L_k)}{p(x)}$$

where $p(L_k)$ is the prior probability of a data label L_k . Respectively, $p(x|L_k)$ and $p(x)$ are commonly referred to as *likelihood* and *evidence*.

Strong independence assumption among data features $x_1, x_2, \dots, x_n$ allows for the following derivation:

$$p(L_k|x_1, x_2, \dots, x_n) = p(L_k) \prod_{i=1}^n p(x_i|L_k)$$

A naive Bayes classifier then applies the following rule to the the above derivation to predict the class label $\hat{l}$ given a data point with data features $x_1, x_2, \dots, x_n$:

$$\hat{l} = \underset{k \in \{1, 2, \dots, K\}}{\operatorname{argmax}} \left(p(L_k) \prod_{i=1}^n p(x_i|L_k) \right)$$

Artificial Neural Network

This is a Machine Learning technique inspired by biological neural networks which form the brain structure of intelligent organisms. An Artificial Neural Network (ANN) in its simplest form is a graph of units called *perceptron* [Gurney, 1997]. In fact, perceptron is simply a threshold function as follows:

$$y = \begin{cases} 1, & b + \sum_{i=1}^n w_i \cdot x_i > 0, \\ 0, & \text{otherwise} \end{cases}$$

where x_i is an input of the perceptron, b is a bias parameter, and w_i is a weight which today is often obtained using non-convex optimization algorithms such as gradient descent [Ruder, 2016], a process commonly referred to as training. Henceforth, if trained, the perceptron model can map a series of inputs to a binary output. That is, in a binary classification

problem given a data point with data features $x_1, x_2, \dots, x_n$, the trained perceptron model predicts a label for the data point. Literature such as [Haykin, 1994, Ripley, 2007, Nielsen, 2015] state that perceptron is only applicable to linearly separable problems. Hence for more complex problems, often a deeper network of perceptrons is used which fosters the notion of *hidden* layers.

Figure 2.3: Feed-forward neural network architecture.

Topology. In a $(N + 1)$ -layer feed-forward neural network architecture as illustrated in Fig. 2.3, the input layer is denoted by L_0 , the output layer by L_N , and the hidden layers by L_k where $k = 1, \dots, N - 1$. The weight of an edge (i, j) is denoted as w_{ij} where $i \in L_{k-1}$ and $j \in L_k$, $k = 1, 2, \dots, N$. The bias at layer L_{k-1} is b_{k-1} . Thus, the output of any node $j \in L_k$ is

given by y_j as follows:

$$y_j = \begin{cases} f_k \left(b_{k-1} + \sum_{i \in L_{k-1}} w_{ij} y_i \right), & j \in L_k : k = 1, 2, \dots, N \\ x_j, & j \in L_0 \end{cases}$$

where $f_k(\cdot)$ is the activation function of layer k and x_j is the input value to an input node $j \in L_0$.

Activation Functions. The sign function in the perceptron model can be extended to a broader spectrum of functions based on the problem specifications. For instance, in a binary classification problem, often the sigmoid logistic function $f_N(z) = 1/(1 + e^{-z})$ is used for the output layer because the aim is to output the probability of a class. For the hidden layers however, the rectified linear unit (ReLU) function is often used [LeCun et al., 2015] as $f.(z) = \max(0, z)$, because the focus of hidden layers is to exploit the compositional hierarchy of the inputs to compose higher level (combinatoric) features so that data become more separable in the next layer, and hence the speed of convergence is a main consideration.

Depth and Width. The depth of a feed-forward network is N since the input layer is not counted by convention. While N is application specific, a feed-forward network with two hidden layers can be trained to represent any arbitrary function, given sufficient width and density (number of edges) [Heaton, 2008].

The width of a layer is the number of units (nodes) in that layer, and accordingly we denote the width of a layer k by $|L_k|$. Determining the width is largely an empirical task, and there are many rule-of-thumb methods used by the practitioners. For instance, [Heaton, 2008] suggests that the width of a hidden layer be $2/3$ the width of the previous layer plus the width of the next layer.

Training. The weights in the feed-forward network are computed using *gradient descent back propagation* algorithm [Nielsen, 2015] during the training process. That is a non-convex optimization problem given a loss function. In binary classification, the sigmoid activation functions, as often chosen, have a saturation effect which will result in small gradients, while gradients need to be large enough for weight updates be propagated back into all the layers. Hence, *cross-entropy* is commonly used as the loss function to mitigate the saturation effect of the sigmoid function. Cross-entropy is defined as follows:

$$H(p, q) = - \sum_x p(x) \log(q(x))$$

where the $\log(\cdot)$ component in the loss function counteracts the $\exp(\cdot)$ component in the sigmoid activation function, thereby enabling gradient-based learning process.

2.1.4 Blockchain

Blockchain is a distributed record of transactions maintained by a peer-to-peer network using consensus mechanisms such as proof-of-work (PoW) [Zhang et al., 2019]. Blockchain transactions consist of data and cryptographic values. These cryptographic values serve as digital signature for the transaction data and also provide source of funds for transaction fees. The nodes of the Blockchain peer-to-peer network compete with each other to collect transaction fees, typically by providing PoW.

PoW makes the record of transactions extremely costly for an attacker to tamper with. This is because any malicious attempt by an attacker who did not perform the majority of the work in the network needs to be approved by non-attackers. In other words, the only way an attacker can compromise the integrity of a transaction is to perform at least half of the work in the network. This costs, for instance, 664,000 USD in Bitcoin and 121,000 USD in Ethereum, per hour (as of 2020-05-01 08:31:03 EST) [crypto51.app, 2020]. Hence, the security of Blockchain with respect to data integrity is in fact a function of economical incentives and the level of participation in the competition.

When participation in the network is open to everyone, the Blockchain is called *public Blockchain*; when participation is restricted to the parties within a private group, the Blockchain is called *private Blockchain*. The key difference is the consensus mechanism: public Blockchain uses incentive-based consensus mechanisms such as PoW, but private Blockchain typically uses non-incentive based consensus such as *selective endorsement*. A detailed discussion can be found in [Mingxiao et al., 2017, Hao et al., 2018].

2.2 Definition of Trust and Reputation

There are multiple interpretations of trust and reputation in the existing literature [Hussain and Chang, 2007]. In this section we first review the existing definitions of trust in the literature and explain their key differences. Second, we review the existing definitions of reputation and explain how they are different from one another. Finally, we explain the definition which we adopt in this dissertation with respect to trust and reputation.

2.2.1 The Interpretations of Trust

Over the past decades, scientists from a broad variety of disciplines have proposed definitions for trust. Although many of such definitions are similar, there are some key differences. Based on the existing literature, we categorize the definitions of trust as follows:

1. **Trust as a belief of benevolence.** Initially [Jøsang, 1996] proposed this definition of trust and later on others such as [Wang and Vassileva, 2003b, Wang and Vassileva, 2003b, Mui et al., 2002c, Mui et al., 2002a] partially adopted it. This definition, presents the notion of trust as the belief that the action of a certain entity is without malevolent intentions. It then proposes two types of entities, *passionate* and *rational* respectively associated with human and non-human entities. A user of a website is an example of a passionate entity while an agent in a multi-agent system is an example of a rational entity.
2. **Trust as the probability of a promised or expected action.** Some authors such as [Gambetta et al., 2000, Golbeck, 2005, Esfandiari and Chandrasekharan, 2001] define trust as a probability. In the literature who adopt this definition, this probability is commonly assigned to a promised or expected action. However, some such as [Gambetta et al., 2000, Esfandiari and Chandrasekharan, 2001] consider a *subjective* probability, while others such as [Golbeck, 2005] interpret trust as an *objective* probability.
3. **Trust as speculative contingency.** Some authors such as [Deutsch, 1985, Deutsch, 1973, Sztompka, 1999, Dasgupta, 2000] take a sociological approach to trust and define it as a bet on the contingent actions of others.
4. **Trust as risk assessment.** A separate thread of research in [Luhmann, 2000, Buist, 2006, Mezzetti, 2003, Mezzetti, 2004] consider trust as exposure of one to the risks associated with the actions of others.

We illustrate our categorization of the different definitions of trust in the existing literature with a mind-map in Fig. 2.4. The boundaries between different interpretations of trust are often fuzzy and there is some overlap between the definitions. For instance, [Nitti et al., 2014] fits both the objective and the subjective definition of trust as the probability of an expected action. Similarly, [Luhmann, 2000, Buist, 2006] which fit the objective definition of trust as the probability of an expected action also fit the definition of trust as risk assessment. A similar argument can be made about [Deutsch, 1985, Deutsch, 1973, Dasgupta,

Figure 2.4: A mind-map to illustrate our proposed categorization of the different definitions of trust in the existing literature.

2000] which consider trust not only as a speculative contingency but also as a belief about the benevolence of a passionate entity.

2.2.2 The Interpretations of Reputation

Similar to definitions of trust, there are also several interpretations of reputation in the existing literature. Although these definitions are similar in some aspects, there are key differences. Based on the existing literature, we categorise the definitions of reputation as follows:

1. **Reputation as a history of behaviour.** Some authors such as [Abdul-Rahman and Hailes, 2000, Abdul-Rahman and Hailes, 1998, Mui et al., 2002c, Mui et al., 2002a] consider reputation as a history of the past behaviour. This interpretation of reputation considers two types of history:

- *Observed history of behaviour*, in which reputation is a record of observations with respect to the past behaviour of a user/agent.
- *Informed history of behaviour*, in which reputation is a record of information with respect to the past behaviour of a user/agent.

The main difference between these two types of behavioural history is that in the informed history, unlike the observed history, there is no need for direct interaction with the user/agent whose reputation is of interest.

2. **Reputation as a history of advice.** A separate thread of research such as [Wang and Vassileva, 2003b, Wang and Vassileva, 2003a, Chang et al., 2006] interpret reputation as a history of advice, suggestions, and recommendations received from other users/agents about a user/agent of interest.
3. **Reputation as a social opinion.** Some authors such as [Sabater and Sierra, 2001a, Sabater and Sierra, 2002a, Sabater, 2004, Kreps and Wilson, 1982] define reputation as a social opinion and consider stereotypes or the reputation of the community which the user/agent of interest is in as a decisive factor in determining its reputation.
4. **Reputation as a history of events.** A less common definition of reputation is proposed by [Grishchenko, 1999, Grishchenko, 2004] which extends the definition to the

Figure 2.5: A mind-map to illustrate our proposed categorization of the different definitions of reputation in the existing literature.

history of events. In this approach reputation is defined as the average level of compliance achieved during past events. Unlike other definitions, [Grishchenko, 1999, Grishchenko, 2004] associates reputation with an event rather than an action. The main difference is that an action requires an actor while an event does not necessarily involve an actor. Hence, this interpretation broadens the definition of reputation.

We illustrate our categorization of the different definitions of reputation in the existing literature with a mind-map in Fig. 2.5. Similar to the definitions of trust, the boundaries between different interpretations of reputation are often imprecise and there is certain amount of overlap between the definitions. For instance, [Abdul-Rahman and Hailes, 2000, Abdul-Rahman and Hailes, 1998] fit the definition of reputation as both an observed and an informed history of behaviour. Similarly [Chang et al., 2006] not only defines reputation as a history of advice but also as an informed history of behaviour. While [Wang and Vassileva, 2003b, Wang and Vassileva, 2003a] only fit the definition of reputation as a history of advice but not as an informed history of behaviour as the authors in [Wang and Vassileva, 2003b, Wang and Vassileva, 2003a] do not associate the definition with past behaviour. A different definition of reputation is proposed in [Lin and Dong, 2017] which includes the goal that the trustor wants to fulfil and the competence and willingness of the trustee, consequently the proposed reputation model is different from other works in the field.

2.2.3 Conclusion

Although there are various interpretations of trust in the existing literature, it is commonly accepted that trust in an entity is the belief that the information shared by the entity is truthful or a promise made by the entity is valid [Hussain and Chang, 2007]. Reputation is a historical indicator of this belief [Resnick et al., 2000]. That is, whenever one trusts an entity, the difference between the observed outcome and the expected outcome (belief) constitutes an *evidence*. This evidence is then used to update the entity's reputation. In this dissertation we adopt the above definition for trust and reputation.

2.3 Trust and Reputation Management Systems

Trust has two key steps as suggested by [Yu et al., 2013]: 1) trust evaluation, when one assesses the trustworthiness of an entity, and trust-aware decision making, when one decides which entity to interact with based on their reputation. Authors in [Yu et al., 2013]

categorize all the proposed models as follows:

- Direct trust models
- Indirect trust models
- Socio-cognitive trust models
- Organizational trust models

Direct trust evaluation models compute reputation based on past interactions with the entity. The intuition is that the history of the past interactions between trustor and trustee reflects an estimate of the outcome of the future interactions. Under this principle, Beta Reputation System [Jøsang and Ismail, 2002] and Dirichlet Reputation System [Josang and Haller, 2007] are well known examples of such systems. In Beta Reputation System, a binary value indicates the outcome of an interaction (i.e. 0 or 1 where 0 represents *Distrust* while 1 represents *Trust*). However, in Dirichlet Reputation System instead of taking a binary approach, the outcome is modeled with a discrete value of i where $i \in \{1, \dots, k\}$. Binary trust models can not distinguish between different levels of trustworthiness since they are restricted to a binary output to represent the belief of an agent with regards to the trustworthiness of another agent. However, binary models are much easier to implement and to ensure the robustness.

Indirect/Reputation-based trust evaluation models use reputation together with the direct trust and most follow a general structure such as:

$$\lambda(DirectTrust) + (1 - \lambda)(RecommendedTrust) \quad (2.1)$$

where *DirectTrust* is calculated based on previous interactions with the trustee (first-hand evidence), and *RecommendedTrust* is calculated based on the advice that an agent receives from other agents about the trustee (second-hand evidence). The importance of utilising second-hand evidence becomes evident when the first-hand evidence space is sparse which is inevitable with the growing number of agents particularly in new application domains such as IoT.

To aggregate first-hand and second-hand evidence a weighting strategy is needed as demonstrated in Eq. (2.1). Many authors use constants weights. For instance, authors in [Liu et al., 2011, Yu et al., 2011, Weng et al., 2006] consider $\lambda = 0.5$ to weight first-hand and second-hand evidence equally, assuming that first-hand evidence and second-hand

evidence are equally important. However, the authors in [Mui et al., 2002b] take a more dynamic approach by initializing λ to zero and relying mainly on second-hand evidence in the beginning when the first-hand evidence is not enough. As more first-hand evidence becomes available, λ increases until it reaches 1. However, [Mui et al., 2002b] assumes that entity's behaviour remains unchanged over time. This may not always be true and limits the applicability of the approach under more dynamic scenarios in which agents behaviour is less static and keeps on changing with environmental factors (i.e. an agent may behave truthful throughout the week and become less trustworthy during the weekend when the demand for the service that it provides is high).

on a separate vein, the ReGreT model [Sabater and Sierra, 2002b] make use of the social relationships among the members of a community to calculate λ , [Chen et al., 2016a, Chen et al., 2016b] also took a similar approach. However, they rely on the availability of social network information among the agents which may not be available in many systems. The availability of social links is limited to a narrow range of application domains where agents form social relationships besides the interactions they engage in. Hence, they can leverage this social information in trustworthiness evaluation, but in practice the majority of applications lack this social aspect because either it may raise privacy concerns or it is not desired in the context. [Weng et al., 2006] on the other hand uses direct trust as the basis for entropy filtering recommendations before aggregating them to form a reputation evaluation. However, the main reason that justifies the necessity of relying on second-hand evidence is the sparsity of first-hand evidence (i.e. direct trust). Hence, performing entropy filtering based on unreliable direct trust can be misleading.

Socio-cognitive trust evaluation models such as [Liu et al., 2013, Zhou et al., 2015] which compute trust with techniques other than reputation. Yet, such models can be considered a complement to reputation trust, particularly when the first-hand evidence does not suffice for a reliable trust evaluation. Moreover, they are useful to address the cold start problem in trust and reputation management systems. In [Zhou et al., 2015], a deep belief network is used to train classifiers for seven different stereotypes. The minimum trust prediction is passed on to a Beta Reputation System to build complementary direct trust when the first-hand evidence is not dense enough to suffice reliable trust calculation. However, the additional information which [Liu et al., 2013, Zhou et al., 2015] require may not be always available. Besides, [Zhou et al., 2015] considers second-hand evidence as such information and treats it similarly to other socio-cognitive information sources. One can debate that second-hand evidence is fundamentally different from a stereotypical informa-

tion source and should not be considered as such. Moreover, [Zhou et al., 2015] considers trust as a binary value, however, binary trust models are not able to capture the dynamic nature of an heterogeneous application domain such as IoT.

Organizational trust evaluation models can be realized only if there exists at least one trusted third-party in a multi-agent system who can act as a supervising body for the transactions among other agents. Given the decentralized nature of IoT and scalability being one of the main challenges in IoT solutions, organizational trust models are not the focus of this research project.

2.4 Trust in IoT, MANET, and Ubiquitous Computing

Although trust is a crucial recruitment for IoT applications, there is little work on trust management for IoT [Yan et al., 2014]. However, there is a significant body of work on trust in other similar contexts which exhibit many of the properties that today we associate with IoT. Authors in [Chen et al., 2016a] and [Chen et al., 2016b] provided a trust-based service management for Social Internet of Things (SIoT) where IoT nodes can establish social relationships in favour of their human owners. Clearly, such scheme heavily depends on trust, hence trust evaluation is of great importance to maintain the integrity of the automatically formed social relationships. The Idea of SIoT however was formerly introduced by [Atzori et al., 2011] where the necessity of trust is highlighted for social IoT nodes which supposedly establish relationships and request/provide services autonomously.

Similarly, authors in [Chen et al., 2016a] propose an approach where each node keeps a friendlist, a frequently visited location list and a device interaction list for calculating trust. The intuition is that the devices which come across each other more often are more likely to behave truthful towards each other. Similarly, the devices that have more mutual friends or interact with the same entities are more likely to be trustworthy of one another. However, in the presented model, nodes have to share this information with each other which may raise privacy concerns, since it involves the exchange of critical information such as the location list (i.e. discloses the frequently visited locations of the IoT device). A later work [Chen et al., 2016b] considered three trust properties: honesty, community-interest and cooperativeness as the key essence of the proposed trust calculation model, arguing that the final trust composition is application-specific since each IoT application can have different trust requirements with regards to these three trust properties. Among such properties, cooperativeness to a great extent resembles the friend list of [Chen et al., 2016a]

while community-interest considers groups of agents with similar goals more trustworthy towards each other.

In all of the aforementioned approaches trust is treated as an objective measurement. We say that a TRM model is deemed as objective when the calculation of trust scores in that model are not subject to the limitations, preferences, and/or predispositions of an individual trustee and/or trustor. That is, in objective models the trustor (i.e. the one who trusts the other) assesses the trustworthiness of the trustee (i.e. the one who is being trusted by the trustor) without considering any characteristic or attribute which is particular to itself or the trustee. Hence, such attributes do not contribute to the weighting and aggregation of the evidence (e.g. feedbacks) in objective models. In contrast, subjective models make use of such attributes and propose a subjective weighting and aggregation mechanism for each trustor.

Moreover, we explain the advantages and disadvantages of both objective and subjective TRM models. With this respect, on one hand the key advantage that objective models have over subjective ones is simplicity. Subjective assessments are prone to rejection and mistrust due to the high complexity involved [Earle, 2010, Guo et al., 2005]. That is when the process of an assessment or judgment is hard to understand for the trustees, they resist to accept the judgment and it remains open to debate. However, when the judgment process is simple, it is more likely that there will be consensus among trustees and the outcome of the judgment (even if not fair) will often be commonly accepted.

On the other hand, objective TRM models have several disadvantages over subjective one. Among the significant disadvantages is the issue of fairness. What is good for the majority may not be good for everyone [Folger and Cropanzano, 2001]. For example, consider a merchant who sells food online and claims that they will pack the food in either microwave-safe (plastic) or oven-safe (metallic) packaging according to customer's instruction. Assume that the vast majority of the customers have microwave oven and hence order microwave-safe packaging. While a few customers do not have microwave oven and hence order the food with metallic packaging which is more suitable for traditional ovens. Assume that the merchant often ignores the instruction of those few who requested metallic packs either intentionally (i.e. out of stock) or unintentionally (i.e. staff mistake in packaging). An objective TRM system would not be able to provide fair and accurate trust assessment for those few customers who do not have microwave oven simply because they have a particular need which the vast majority do not have and hence the positive feedbacks submitted by the vast majority would outweigh the few negative feedbacks submitted by this minor-

ity group. However, a subjective TRM system would propose a subjective weighting and aggregation mechanism for each trustor. That is, while in an objective TRM model everyone sees the same score for a merchant, in a subjective one the vast majority (those with microwave oven) see a positive trust score while the minority (those with traditional oven) see a negative score.

Hence, [Nitti et al., 2014] took the above disadvantages into consideration by providing two models: subjective and objective models. In the subjective model, each node calculates the trustworthiness of other nodes using direct observations (first-hand evidence) and the indirect observations (second-hand evidence) from mutual friends. However, it leverages the assumption of similar subjectivity among friends which is highly open to debate (e.g. two friends can have fundamentally different points of view. Hence, friendships do not necessarily form on the basis of perspective similarities).

On a different note, a cloud utility is proposed by [Al-Hamadi et al., 2019] to act as a reputation center, recommender system, identification and access controller. Applications installed on users' smartphones then use this cloud utility for above-mentioned purposes. The main trust model does not differ that much from that of classic trust models such as Beta Reputation System. The main difference is the use of a time decay factor and socio-cognitive parametric weights. Similarly, [Chen et al., 2018] proposes a trust protocol and cloud based architecture for service management while its contribution with regards to trust model is limited to a parametric weighted sum of direct trust and indirect trust of which the former is calculated using Beta Reputation System while the later is calculated based on socio-cognitive similarities. Authors in [Qiu et al., 2013] also take a similar approach and proposes a framework for trust based QoS prediction in which the reputation model is merely an iterative average function Hence, the main contribution of [Chen et al., 2018] and [Qiu et al., 2013] is architecture and protocol design rather than the trust model.

In the remaining of this section we expand our TRM literature review and broaden its focus to include a significant body of work which are not explicitly labeled as Internet of Things (IoT) but exhibit many of its properties. For example, a large body of work focus on TRM in the context of mobile ad-hoc networks (MANET). Although MANETs are not explicitly referred to as a form of IoT, but we agree with the examiner that in essence and spirit they have a lot in common with IoT and have existed for a long time even before the prevalence of IoT. A MANET [Corson and Macker, 1999] is a network of interconnected mobile nodes which are connected to on another in a wireless fashion without any central networking infrastructure.

To understand where and why TRM is used in MANETs it is crucial to explain the difference between MANETs and other types of wireless networks. In a normal wireless network, nodes are stationary and are connected to a central networking infrastructure using devices called Access Points. An example of such network is the WiFi network at home or office where the WiFi router is the access point.

Although the mobility strikes as a significant difference between MANET and WiFi, but it is not the key difference that necessitates TRM in MANETs. This is because there exist another type of wireless networking that is in fact very common and supports mobility: The cellular networks. A cellular network is a type of communication network where, like WiFi, nodes can connect to one another using a central networking infrastructure but, unlike WiFi, nodes can also move. In a cellular network, each mobile node is connected to at least one, but usually three, Base Transceiver Stations (BTS). When the mobile node moves, it may register with a new BTS and sign out from the previous one in a graceful fashion called handover. 4G, and more recently 5G, are examples of cellular networks.

Hence what really distinguishes MANETs from other wireless networks is not the mobility factor. It is the lack of a trusted central infrastructure that strikes as the main difference between MANETS and other wireless networks. In MANET if a mobile node A wants to send a packet to a mobile node B, it gives the packet to an arbitrary mobile node C who happens to be in its wireless transmission at the time and trusts node C to pass along the packet towards node B. Hence TRM is crucial for such system to work in practice simply because there is no centrally trusted infrastructure to provide reliable networking to the participants.

The existing literature extensively studies TRM and related issues in MANETs. Among the earliest approaches is [Marti et al., 2000]. The authors in [Marti et al., 2000] incorporate elements of TRM in the routing protocol of MANET by extending Dynamic Source Routing (DSR) to rate paths based on first-hand evidence. However, it does not consider second-hand evidence and hence the TRM used in [Marti et al., 2000] lies in the category of Direct Trust Evaluation models. We explained the architecture of such TRM models in Section 2.3 with details. Similarly, authors in [Pissinou et al., 2004, Wang et al., 2005] calculate direct trust using first-hand evidence. While [Pissinou et al., 2004] takes a Bayesian approach similar to those explained in Section 2.3, [Wang et al., 2005] uses Finite-state Automaton (FSA) to build a statistical model of the routing behaviour of each neighbor node and relies on statistical tests to detect abnormal (not trustworthy) nodes. The key difference between the work proposed by [Pissinou et al., 2004, Wang et al., 2005] and the work proposed by [Marti et al., 2000] is that the authors in [Pissinou et al., 2004, Wang et al., 2005] incorporate their

proposed direct trust evaluation model into Ad-hoc On-demand Distance Vector (AODV) routing protocol rather than extending the DSR protocol.

In contrast to [Marti et al., 2000, Pissinou et al., 2004, Wang et al., 2005] that only consider direct trust evaluation, a significant body of work also considers indirect trust evaluation. For example, the authors in [Michiardi and Molva, 2002] propose a collaborative reputation mechanism for MANETs to identify lazy/selfish nodes. That is, in [Michiardi and Molva, 2002] nodes share information indicating which nodes helped them forward packets and hence, selfish nodes who do not participate in routing other's packets remain with little or no positive reputation. When a node with low reputation asks others to forward its packets, others can refuse to do so. Hence, this incentivizes the lazy/selfish nodes to be more active and less lazy otherwise they will be abandoned by others. Similarly, the authors in [Zouridaki et al., 2006] use second-hand evidence which is weighted using first-hand evidence to provide an indirect trust evaluation model which is robust towards common reputation attacks. Hence, the key difference between the work proposed by [Zouridaki et al., 2006] and the work proposed by [Michiardi and Molva, 2002] is the robustness towards TRM attacks in which malicious nodes are present. Unlike [Michiardi and Molva, 2002] which only propagates positive second-hand evidence, the work proposed by [Zouridaki et al., 2006] also considers the potential existence of attacker nodes.

Ubiquitous/pervasive computing and ambient intelligence: MANET is not the only context that exhibits many of the IoT properties and for which a significant body of work on TRM exist. There is also a large body of work on TRM in the context of ubiquitous computing, pervasive computing, and ambient intelligence. Luckily, the TRM models proposed for such abstract domains were generic enough to remain relevant to what today is derived from those abstract domains and is known as IoT. In this thesis we discussed many of such generic TRM models in Chapter 2 Section 2.3 before narrowing our focus. We amend Section 2.3 to also include more recent works in such domains. Particularly, the authors in [Mhetre et al., 2016] present a distributed TRM model based on fuzzy rules. In specific, a linguistic fuzzy model referred to as Mamdani Fuzzy Inference System is used to calculate trust based on three factors: First-hand evidence, Second-hand evidence, and uncertainty. The authors in [Mhetre et al., 2016] define uncertainty as the conditional probability of the first-hand evidence given the second-hand evidence. A large body of work on TRM in ubiquitous and pervasive computing use MANETs as examples of such computing systems. For example, the authors in [Boukerche and Ren, 2008] propose a trust-based security system for ubiquitous and pervasive computing environments. They propose wireless

and mobile ad hoc networks as examples of such environments. Particularly, the authors in [Boukerche and Ren, 2008] calculate the trust score for a node in a MANET using a hybrid trust evaluation model. That is they use a combination of first-hand and second-hand evidence in addition to a socio-cognitive approach which increases the trust score of old nodes in a wireless neighbourhood assuming that older nodes are more likely to be trustworthy otherwise they would have been detected and removed from the network.

In the existing literature on trust management of IoT systems and similar systems such as MANETs, VANETs, etc., the three issues of discrimination, context-awareness, and subjectivity are not well studied. However, there are a few works outside the IoT domain which attempted to address these issues while they have restrictive assumptions which do not fit IoT well. For instance, [Fang et al., 2012] assumed strong naive independence between the attributes that influence the subjectivity, which does not hold for IoT applications (e.g. knowing that a phone has a low resolution camera implies an old model and hence decreases the odds for it to have a near-field communication sensor). Clustering is used in [Fang et al., 2014] to group similar-thinking users together based on their provided ratings but it dictates a centralized approach. Moreover it does not perform well when the first-hand evidence is sparse which makes it unfit for IoT applications. BLADE [Regan et al., 2006] took a probabilistic approach which utilises a Bayesian Network to address subjectivity. However, it provides no mechanism for Bayesian structure learning. Without such mechanism the applicability of BLADE is limited only to scenarios where the features are independent. TRAVOS [Teacy et al., 2006], on the other hand, attempted to address discrimination by downgrading exaggerative second-hand evidence. However, it implies a centralized approach by assuming that the reputation of all users is available upon demand. Given that scalability is a crucial recruitment for any IoT solution, a centralized architecture is a serious barrier to this achievement. REGRET [Sabater and Sierra, 2001b], tackled context-awareness by exploiting social relationships which often either do not exist in the IoT context or are not defined well.

Table 2.1: Comparison of the discussed approaches.

Approach	Bin.	Mul.	Edge.	Subj.	Cont.	Disc.
[Chen et al., 2016a]	Y	Y	N	N	N	N
[Chen et al., 2016b]	Y	Y	N	S	N	N
[Wang et al., 2005]	Y	Y	Y	N	N	N
[Fang et al., 2012]	Y	Y	N	Y	N	N
[Fang et al., 2014]	Y	Y	N	Y	N	N
[Jøsang and Ismail, 2002]	Y	N	Y	N	N	N
[Josang and Haller, 2007]	Y	Y	Y	N	N	N
[Regan et al., 2006]	Y	Y	U	N	Y	S
[Pissinou et al., 2004]	Y	Y	Y	N	N	N
[Teacy et al., 2006]	Y	Y	N	N	Y	N
[Sabater and Sierra, 2001b]	Y	Y	N	N	Y	S
[Zhou et al., 2015]	Y	N	N	N	Y	N
[Marti et al., 2000]	Y	Y	Y	N	N	N
[Mhetre et al., 2016]	Y	Y	U	Y	N	N
[Boukerche and Ren, 2008]	Y	Y	Y	N	N	N

2.4.1 Summary and Discussion

There are certain characteristics that affect the reliability and applicability of a trust model in different scenarios and environments. For instance, whether a trust model is binary or not affects its application in a highly dynamic domain such as IoT. Binary models does not embed sufficient amount of information to reflect the dynamic nature of IoT devices. When it comes to trust evaluation in IoT, a multi-level trust and reputation model is preferred. Another important consideration in IoT is the possibility of edge computing, that is, the computations of IoT application are preferred to be conducted on the edge devices rather than central servers. Under this principle, it is of great importance that the computational trust model takes a centralized approach or a decentralized one in trustworthiness computation. A decentralized approach can facilitate edge computing. This is mainly important due to the scalability requirement of IoT applications, since with edge computing, the IoT application can be infinitely scalable. Moreover, subjectivity, discrimination, and context-awareness are other major challenges associated with the existing trust and reputation management systems in the IoT context. For a trust system in the IoT context, subjectivity means that the information (advice) provided by each autonomous IoT device

is likely to have been influenced by the device's individual preference, which can be misleading in trust evaluation. Discrimination, on the other hand, is referred to the lack of integrity in providing services and information to particular types of IoT devices. Finally, the highly dynamic nature of IoT stress context-awareness as a necessity of any trust and reputation management system in this domain. Hence, in Table 2.1, we compare the most cited approaches among the reviewed existing literature with regards to the aforementioned characteristics.

As illustrated in Table 2.1, none of the existing approaches address all of the specified issues. This can be attributed to the fact that the trust research in IoT is still naive and the mature approaches are mainly designed for e-commerce applications. Clearly, IoT does not share its characteristics with e-commerce, which makes it a newly emerged application domain (see Fig. 2.6) with crucial trust needs that are not addressed in any of the existing approaches that mainly target e-commerce and are being concluded (see Fig. 2.7).

Issues such as subjectivity, context, and discrimination are less beneficial and often hard to address in e-commerce, and with the research interest for e-commerce decreasing over time, it is explainable why such problems are remained open in the e-commerce research field. However, it is impossible to maintain a sound and effective trust system in IoT without addressing such issues. Due to the high variance in IoT devices' sensor limitations, power and bandwidth constraints, privacy concerns, etc., IoT is featured with unprecedented heterogeneity which significantly aggravates challenges typically associated with the existing trust and reputation management systems. For example, What we refer to subjectivity of IoT generated data in this thesis comes from the fact that IoT sensors have different sensing capabilities with different precision. So the judgment/feedback that each IoT device can provide on an arbitrary matter is subject to its sensing capabilities, sensing precision, etc. Moreover, as the scale of the data grows in IoT generated Big Data, the evidence space becomes more dense and context-rich. This allows for more accurate trustworthiness assessment. However, what form the second-hand evidence should take has been largely overlooked. This engenders an important issue which we refer to as the *accuracy-privacy dilemma*. To illustrate this, suppose an arbitrary agent *Alice* consults another agent *Judy* about how trustworthy *Bob* is. One way is to let *Judy* give a trust score or rating about *Bob* [Yu et al., 2013], which is the approach commonly adopted in the trust research community. This approach is simple but loses the *context information* of the interactions between agents. For example, the context could be the transaction time and location, and service provided by an agent during off-peak hours could have higher quality

Figure 2.6: Trend analysis for IoT from 2004 to 2018 [Google, 2018b]

Figure 2.7: Trend analysis for e-commerce from 2004 to 2018 [Google, 2018a]

(more SLA-conformant) than during peak hours. Without such context information, trust assessment based on just ratings or scores would have lower accuracy. On the other hand, another method is to let *Judy* reveal her entire interaction history with *Bob* (e.g., in the form of a detailed review), which is the approach commonly used in recommender systems. Although the information disclosed as such is helpful for trust assessment of *Bob*, it is likely to expose substantial privacy of *Bob* and *Judy* to *Alice* and possibly to the rest of honest-but-curious public.¹

¹Recommender systems can take this approach because they are generally considered trusted intermediaries, and they focus on preference modeling rather than trust and reputation modelling.

2.5 Trust and Social Media

Studies such as [Highfield and Leaver, 2016, Duggan and Brenner, 2013, Nielsen and Schröder, 2014] show that social media has become an important part of people's everyday life. This is because the influence of social media extends to various aspects of one's life. Politics, education, business and economy are examples of such aspects where social media has a decisive influence [Stieglitz and Dang-Xuan, 2013, Greenhow and Lewin, 2016, Qualman, 2012]. Authors in [Madrigal, 2017] reveal that Facebook users who had exposure to a political advertisement were 17% more likely to vote. Similarly, authors in [Gibs and Bruich, 2010] show that users are 68% more likely to remember an advertisement if they see it in a social context.

Social media platforms have also become a source of information for many [Hopkins and Hawking, 2018, Mansour, 2018, Bu et al., 2010]. For instance, [Bartosik-Purgat et al., 2016] show that social media platforms are a crucial part of information seeking about products and services in many cultures around the world. Likewise, authors in [Lim et al., 2017] propose that social media influencers play an increasingly important role in shaping the purchase intention of the online shoppers.

However, social media can be untrustworthy [Wessel et al., 2016]. Hence, when used as a source of information, untrustworthy social media can have adverse influence on various aspects of one's life. For instance, it can mislead the voters in an election [Allcott and Gentzkow, 2017] or it can mislead the online shoppers and lure them into a scam [Malbon, 2013]. In a study, [Streitfeld, 2012] reveals that 33% of online reviews are fake. Similarly, authors in [Audrezet et al., 2018] propose that social media influencers are posing a threat to information authenticity. Thus, it is crucial to be able to assess the trustworthiness data and users in social media.

In the existing literature, various approaches have addressed trust assessment for social media over the past two decades [Urena et al., 2019]. In fact some approaches in this field such as [Schneider et al., 2000, Sabater and Sierra, 2001c] are as old as social media itself, while some others such as [Wu et al., 2018a, Wu et al., 2018b] are more recent. In the remaining of this section we categorize the existing approaches base on their contribution, methodology, architecture, and prerequisites. We then discuss how approaches have evolved over time, and identify existing open problems.

We categorize existing approaches based on their prerequisites as follows:

- **First-hand Evidence (FHE).** Some approaches such as [Guha et al., 2004, Hoffman

et al., 2009, Tian and Yang, 2011] rely on first-hand evidence to assess the trustworthiness of social entities. Examples of first-hand evidence in a social setting include the purchase history of a user in an e-commerce website, interaction history of a user in a social network, previous downloads from a peer/seeder in a p2p file sharing scheme, etc.

- **Second-hand Evidence (SHE).** Others such as [Resnick and Zeckhauser, 2002a, Resnick and Zeckhauser, 2002b] rely on second-hand evidence to assess the trustworthiness of social entities. Examples of second-hand evidence in a social setting include the user reviews in an e-commerce website, the rating of a peer/seeder in a p2p file sharing scheme, etc.
- **Socio-cognitive Features (SCF).** A different strand of research such as [Wu et al., 2015, Yan et al., 2015] supplement the evidence space with socio-cognitive features. For instance, the age of an account in social media platforms is considered a socio-cognitive evidence of trustworthiness as most of the fake accounts happen to be newly registered accounts [Gurajala et al., 2015].
- **Mixed Evidence.** Some approaches such as [Malik and Bouguettaya, 2009, Sabater and Sierra, 2001c] rely on multiple sources of evidence to assess trustworthiness of social entities.

In practice, what greatly affects the effectiveness and scalability of a trust system is its architecture. We categorize existing approaches based on their proposed architecture as follows:

- **Centralized.** Some approaches such as [Resnick and Zeckhauser, 2002b, Sabater and Sierra, 2001c, Schneider et al., 2000] propose a centralized architecture to assess the trustworthiness of social entities. Such approaches assume a reputation center which oversees the interactions between the social entities. Hence, the trust management in such systems is dependant on the trust center and its business plan. A rational reputation center will only host evidence if there exists a sustainable business plan which benefits from hosting such evidence. In other words, such trust center will only host data that coincide with its business plan. We refer to such trust systems as non-autonomous trust systems.
- **Decentralized.** Others such as [Resnick and Zeckhauser, 2002a, Tian and Yang, 2011, Yan et al., 2015] take a decentralized approach to assess trustworthiness of

social entities. In decentralized trust systems, users share advice in a peer-to-peer fashion which makes scalability a major issue for such systems.

- **Agnostic.** Some approaches such as [Wu et al., 2016, Victor et al., 2009] are architecture agnostic and propose an abstract model of trust which may be applicable to both centralized and decentralized systems.

Another important aspect of a trust management solution is how descriptive and prescriptive they are. Hence, the contribution of a proposed approach may constitute a model, a system, or a framework. We categorize existing approaches based on their contribution as follows:

- **Model.** Some approaches such as [Wu et al., 2015, Wu et al., 2018b, Yan et al., 2015] propose a model to assess the trustworthiness of social entities. Such approaches are often formal and very descriptive. They present the problem of trust management with a mathematical definition and offer high level and abstract solutions for such problems.
- **System.** Others such as [Resnick and Zeckhauser, 2002a, Resnick and Zeckhauser, 2002b] take a more prescriptive approach by proposing a TRM system often tailored for a specific set of applications. A TRM system can also contain the one or more implemented models.
- **Framework.** Some approaches such as [Hendrikx et al., 2015, Tian and Yang, 2011] propose a framework which is a collection of components and structured thoughts on how to solve a TRM problem. A model or a system can be one of the components in a framework.

Different approaches tackle TRM with different methodologies. Although the fundamental problem which they offer a solution for is common among them, the methods they use to derive the solution vary widely from one approach to another. Analysing the differences between such methodologies is crucial as it helps to identify the computational limitations of each approach. We categorize existing approaches based on their methodologies as follows:

- **Weighted Average (CAT1).** Approaches in this category such as [Guha et al., 2004, Resnick and Zeckhauser, 2002a, Resnick and Zeckhauser, 2002b] formulate trust

and reputation of social entities as an aggregation function such as weighted average of the evidence. Such approaches are often fast and require little computational power [Papadimitriou, 2003]. What all such approaches have in common is that they aggregate existing evidence with a weight. One important short coming of such approaches is that the credibility of their assessment is tightly coupled with the weight which they assign to the evidence. Hence, assigning equal weights, not specifying how those weights are obtained, or making it a design parameter limits the effectiveness of the approach.

- **Statistical Inference (CAT2).** Approaches in this category such as [Hoffman et al., 2009, Teacy et al., 2012] take a more mathematical approach by proposing a trust and reputation inference mechanism based on statistics and probability theory. One advantage of such approaches is that the derived trustworthiness model is mathematically sound. However, they often do not provide implementations which makes it difficult for researchers to incorporate such approaches in their systems.
- **Fuzzy Logic (CAT3).** Some approaches such as [Sabater and Sierra, 2001c, Wu et al., 2016] propose a fuzzy method to assess the trust and reputation of social entities. Fuzzy methods are a good candidate when there is uncertainty and vagueness involved in the solution of a problem [Ross, 2005].
- **Hybrid (CAT4).** Some approaches such as [Yan et al., 2015, Liu et al., 2017] use a mixture of other methods to assess the trust and reputation of social entities.

We present a summary of the discussed approaches and their key differences in Table 2.2. Our comparison reveals that there is a shortage of CAT2 methodology in the existing literature. This can be due to the fact that statistical methods are often more complex compared to weighted average and hence are less commonly adopted by the researchers in the field. Another interesting observation that we make is the proliferation of CAT4 methodology. We attribute this to the fact that hybrid approaches facilitate seamless expansion of topic space within research fields [Small, 2006].

On a separate vein, Table 2.2 shows that first-hand and second-hand evidence remain the most effective prerequisite of TRM systems over past two decades. This highlights the importance of evidence in computational trust. Although Socio-cognitive features are good

Table 2.2: Comparison of the discussed approaches.

Approach	Prerequisites	Architecture	Type	Method
[Guha et al., 2004]	mixed	decentralized	model	CAT1
[Hendriks et al., 2015]	mixed	decentralized	framework	CAT4
[Hoffman et al., 2009]	FHE	decentralized	model	CAT2
[Liu et al., 2017]	mixed	agnostic	system	CAT4
[Resnick and Zeckhauser, 2002a]	mixed	decentralized	system	CAT1
[Malik and Bouguettaya, 2009]	mixed	decentralized	framework	CAT4
[Sabater and Sierra, 2001c]	mixed	centralized	model	CAT3
[Resnick and Zeckhauser, 2002b]	SHE	centralized	system	CAT1
[Tian and Yang, 2011]	FHE	decentralized	framework	CAT2
[Schneider et al., 2000]	SHE	centralized	system	CAT1
[Victor et al., 2009]	SHE	agnostic	system	CAT4
[Wu et al., 2015]	SCF	agnostic	model	CAT4
[Wu et al., 2018a]	mixed	agnostic	model	CAT3
[Wu et al., 2016]	mixed	agnostic	system	CAT3
[Wu et al., 2018b]	mixed	agnostic	model	CAT3
[Yan et al., 2015]	SCF	decentralized	model	CAT4
[Teacy et al., 2012]	mixed	agnostic	model	CAT2

supplement to evidence based TRM systems, they do not replace evidence. Another observation from Table 2.2 concerns the architecture of proposed approaches. The increasing number of decentralized approaches signals the popularity of the concept, meanwhile requires more attention from the research community as scalability is one major drawback of decentralized approaches [Bondi, 2000].

2.6 Trust and Blockchain

Various *hard security* mechanisms and protocols using Blockchain have been proposed to solve security problems in IoT and big data. For instance, some methods use a public Blockchain to inherit its pre-existing security properties for preventing Denial of Service² (DoS) and Distributed Denial of Service³ (DDoS) attacks [Tariq et al., 2019]. However, just ensuring availability is not enough because attackers who want to gain unauthorised access to the system also benefit from the availability. Therefore, access control is enforced by some other mechanisms using smart contracts. For example, [Hammi et al., 2018] uses smart contracts on a public Blockchain to create virtual zones for IoT devices by associating a group ID with an object ID and a public key. As such, a device can only communicate with other devices within the same zone. Also, [Yu et al., 2018] designs a smart contract on both private and public Blockchains to regulate data access for proper data ownership and management.

Another important issue in hard security is the integrity and confidentiality of data communication. Various security methods use the Blockchain technology to benefit from its immutability and built-in asymmetric cryptography. For instance, [Biswas and Muth, 2016] uses Blockchain as an immutable database layer in their proposed security framework for smart cities, in which every transaction is broadcast to all the entities in the network and is confirmed and added to the Blockchain only if the majority of the entities verify the transaction.

Hard security mechanisms do not typically consider threat models about social behaviors or trust. For example, firewalls cannot detect untrustworthy buyers and sellers on Amazon, DoS and DDoS prevention do not discourage sharing and promoting disinformation on Facebook. To address such problems, *soft security* mechanisms provide an invisible and unobtrusive solution. For instance, [Yang et al., 2018] proposes a private Blockchain for a vehicular network in which road side units store trust assessment data submitted by the vehicles. A combination of PoW and proof-of-stake is then used to reach consensus among the road side units. Similarly, [Lu et al., 2018a, Lu et al., 2018b] use three private Blockchains to provide transparency for actions taken by authorities and to facilitate anonymous authentication.

²Denial of Service (DoS) is a type of attack in which the objective of the attacker is to make a certain resource temporarily unavailable.

³A type of DoS attack in which multiple compromised systems referred to as zombies are used to carry out a denial of service attack on the target victim.

On a different vein, some authors use Blockchain as a distributed database to accommodate existing trust models. For instance, [Moinet et al., 2017] adopts a trust model called Human-like Knowledge based Trust (HKT) in its proposed Blockchain for decentralized authentication of network nodes. In [Di Pietro et al., 2018], the authors propose an obligation Blockchain to record promises which are yet to be fulfilled, and then use a public Blockchain to reward and mark fulfilled promises.

Moreover, some authors redesign the architecture of existing multi-party processes to foster trust and transparency without a formal trustworthiness assessment mechanism. For example, [Lin et al., 2017] proposes a Blockchain architecture for long range wide area networks to verify the existence of data transactions for certain time intervals, where Blockchain is used as a trustworthy source of timestamp information.

The authors in [Kouicem et al., 2020] propose a blockchain-based TRM for IoT. Their proposed TRM calculates the trust scores for service providers in IoT simply using a weighted average of a direct trust score and an indirect trust score while the weight is a design parameter. Among these two trust scores, the calculation mechanism of direct trust is simply the average of direct observations. However, the calculation of indirect trust is itself based on two indirect sub-scores: 1) The indirect trust score about a service provider submitted by other service providers. 2) The indirect trust score about a service provider submitted by IoT devices. The former is simply the average of the second-hand evidence provided by the service providers. Similarly, the latter is the average of the second-hand evidence provided by IoT devices. Hence, we argue that the TRM model proposed in [Kouicem et al., 2020] is trivial and lacks novelty. The use of blockchain in their proposed method is minimal and is only limited to storing the evidence. However, instead of using existing public blockchain they proposed a new blockchain. Based on how they describe the underlying consensus mechanism of their proposed blockchain, we guess that they use proof-of-stake (PoS), however they consistently keep referring to it as “proof-of-stack” throughout the paper. Stack and stake are two different concepts in computer science terminology and the term “proof-of-stack” has been consistently used throughout the paper and hence cannot be a typo. This also points to an underlying problem with the preliminaries and assumptions adopted by [Kouicem et al., 2020] which we explain later in the following paragraphs.

Similarly, the authors in [Kumar et al., 2020] propose a secure TRM for smart cities by leveraging blockchain and machine learning. The proposed TRM model itself is summation based and hence more trivial than that of [Kouicem et al., 2020]. However, the authors

in [Kumar et al., 2020] also propose a machine learning (ML) approach for anomaly detection using an existing tree-based ML model called XGBoost. However, the proposed TRM and the ML-based anomaly detection are two stand-alone contributions of the paper with regards to smart cities. That is, the proposed anomaly detection mechanism is not used in the proposed TRM model.

The authors in [Kumar et al., 2020], similar to the work proposed by the authors in [Kouicem et al., 2020], propose a new blockchain rather than using the existing one although the use of blockchain in their proposed approach is minimal and is limited to data storing. Unlike [Kouicem et al., 2020], the authors in [Kumar et al., 2020] claim that they design an enhanced proof-of-work consensus mechanism rather than proof-of-stake which assumingly is used by [Kouicem et al., 2020]. However, similar to [Kouicem et al., 2020], the authors in [Kumar et al., 2020] do not present the details of their proposed consensus mechanism nor they prove its economical soundness. This is of great importance since the consensus mechanism and its underlying economical incentives of participation is the key guarantor of the blockchain's security. Without a sound consensus mechanism backed by strong economical incentives, the the blockchain is not secure [Gervais et al., 2016]. Hence, we argue that the body of work that propose a new blockchain rather than using an existing secure blockchain, should explain in great details if/how a matching level of security can be achieved using their proposed blockchain. We expand on this and include several other recent work in the in the remaining of this section.

Therefore, we clarify the particular properties of blockchain systems that the existing body of work used to support trust and if/how this is achieved. For this purpose, we first explain different properties of the blockchain. Then we describe how such properties are utilized by the existing body of work on TRM in blockchain and to what extent their utilization is justified and supports trust.

Tamper resistance: One key property, perhaps what blockchain has long been known for, is its resistance to tampering and alteration of the records once they are added to the blockchain. Hence, one appealing use case of the blockchain is as a tamper resistant ledger. This is, for instance, what the blockchain has been used for since 2009 in Bitcoin. Tamper resistance provides a potential solution to the problem of double spending in the context of electronic cash by making it extremely hard and economically irrational for an attacker to tamper with the records. This solution has been battle-tested in the past twelve years. However, it is important to understand that this high level of security in the battle-tested blockchain is achieved thanks to the underlying strong economical incentives in place [Ger-

[vais et al., 2016](#)]. In other words, what makes the blockchain tamper resistant is not simply the chain of blocks of cryptographic hashes. Instead, it is the underlying economically incentivized consensus mechanism, proof-of-work in case of Bitcoin, which makes use of that cryptographic techniques as a tool to achieve tamper resistance. Hence, when one proposes a new blockchain, even if they keep using proof-of-work as the consensus mechanism, the tamper resistance property of their proposed blockchain can be open to debate and it is regardless of what cryptographic techniques they use.

Now that we explained the tamper resistance property of blockchain, we can clarify how this property is used in a large body of work on TRM in blockchain. Particularly, authors in [[Yang et al., 2018](#), [Kouicem et al., 2020](#), [Kumar et al., 2020](#), [Moinet et al., 2017](#), [Di Pietro et al., 2018](#)] use this specific property in their TRM approach. That is they store trust evidence in blockchain to maintain integrity and transparency. Among them [[Di Pietro et al., 2018](#)] uses an existing public blockchain while [[Yang et al., 2018](#), [Kouicem et al., 2020](#), [Kumar et al., 2020](#), [Moinet et al., 2017](#)] each propose their own new blockchain however they do not evaluate to what extent their proposed blockchain is tamper resistant.

Smart contracts: In the context of present contract law, a smart contract is a legally-binding agreement that has some of its elements expressed in a machine readable language [[Savelyev, 2017](#)]. However in the existing literature on smart contracts in blockchain, the term is mainly used to refer to any general purpose computation that is stored as code in the blockchain and is performed using the computational resources of the blockchain participants. In the literature, this participants are referred to as *block producers* or *miners* and compete with one another to profit from the incentive system that, as explained in the previous paragraphs, also secures the tamper resistance property of the blockchain.

The authors in [[Malik et al., 2019](#)], use this property of blockchain to propose a TRM model for Vehicular Ad-hoc Networks (VANETs). Particularly, the role that blockchain plays in [[Malik et al., 2019](#)] is to store the implementation (code) of the TRM model and provide the computational resources required for the execution of the code. The proposed TRM model itself is a summation based approach similar to that of [[Kumar et al., 2020](#)] which we discussed earlier. That is, each positive feedback increases the trust score by one and each negative feedback decreases the trust score by one.

Electronic cash: Although not a property of the blockchain, electronic cash is among the first applications of the blockchain. Electronic cash enables micro payments which appeals to a large group of blockchain enthusiasts. The authors in [[Di Pietro et al., 2018](#), [Singh et al., 2020](#)] make use of this aspect of the blockchain to reward trustworthy entities.

However, their contribution to the state-of-the-art in TRM is marginal in the sense that they do not propose a novel TRM model and the trust assessment mechanism that they use is similar to those in [Kumar et al., 2020, Malik et al., 2019].

Transaction fee: Another unique property of the blockchain which can be used to support trust is the transaction fee. Blockchain provides a tamper resistant record of time stamped non-gratuitous participation history. To the best of our knowledge, we are the first to incorporate this property of the blockchain into a new TRM model which contributes meaningfully to the state-of-the-art in TRM. Unlike the existing body of work on TRM in Blockchain, in the fifth Chapter of this thesis we propose a novel use case that: 1) Proposes a novel TRM model rather than simply incorporating basic TRM models into blockchain. 2) Offers direct and meaningful use of a crucial blockchain element, the transaction fee, into the proposed TRM model rather than using it as a data store or computational resource. 3) Benefits from the security (i.e. tamper resistance) of existing public blockchain rather than re-inventing the wheel, given that this is a thesis on TRM and hence our focus in this thesis is to contribute to the state-of-the-art in TRM.

We present a summary of the discussed approaches and their key differences in Table 2.3.

Table 2.3: Comparison of the discussed approaches.

Approach	Security	Smart Contract	Blockchain Architecture
[Yang et al., 2018]	Soft	No	Private
[Di Pietro et al., 2018]	Soft	No	Public / Private
[Lu et al., 2018a]	Soft	No	Private
[Moinet et al., 2017]	Soft	No	Private
[Hammi et al., 2018]	Hard	Yes	Public
[Lin et al., 2017]	Soft	No	Private
[Biswas and Muth, 2016]	Hard	No	Private
[Yu et al., 2018]	Hard	Yes	Public / Private
[Lu et al., 2018b]	Soft	No	Private
[Malik et al., 2019]	Soft	Yes	Public
[Kumar et al., 2020]	Soft	No	Private
[Kouicem et al., 2020]	Soft	No	Private

2.7 Summary

In this chapter we first discuss the preliminaries and provide background information about the methods and technologies used in this dissertation. This includes Big Data and various disciplines related to Big Data such as IoT, Machine Learning and Blockchain.

Next, we conduct a thorough literature review and survey the state-of-the-art in trust. Specifically, we demonstrate that there are multiple interpretations of trust and reputation in the existing literature. Hence, we review the existing definitions of trust in the literature and explain their key differences. Next, we review the existing definitions of reputation and explain how they are different from one another. Then, we explain the definition which we adopt in this dissertation with respect to trust and reputation. Moreover, we explain the fundamentals of trust and reputation management and review state-of-the-art trust models in emerging domains. Specifically, the state-of-the-art in trust and reputation management in IoT is extensively reviewed followed by a comprehensive survey of the related works around trust in social media and Blockchain.

Chapter 3

Subjective Evidence in Trust and Reputation Management

3.1 Problem Statement

This chapter addresses subjectivity as an important issue in TRM for Big Data. In this dissertation we consider two types of Big Data, in particular *user-generated Big Data* and *IoT-generated Big Data* (see Chapter 1 for definitions). Subjectivity in user-generated Big Data means that the information provided by each user, represented by an agent, is likely to have been influenced by the user's individual preference and sensitivity, which can be misleading in TRM. For example, if Alice has high tolerance for spicy food while Bob's tolerance for spicy food is low. Hence, whenever Alice advises Bob that a certain dish is *slightly spicy*, it is probably of Bob's best interest to avoid the dish. Hence, Bob should first adjust the subjective advice to consider the fact that what tastes *slightly spicy* for Alice may taste differently for him due to the difference in the sensitivity of his tongue compared to that of Alice.

This explains the subjectivity in user-generated Big Data in the context of TRM. However, The issue of subjectivity is not limited to people and man-made data. In fact, data which is not sourced from humans can also be subjective. Consider a crowd-sensing example in smart transportation domain where agents *A* and *B* respectively represent a smart car and a smart phone. Agent *A* has a speed sensor precision of *1mph*, while agent *B* has a precision of *5mph*. So agent *B* will consider any crowd-sensed measurement which deviates less than *5mph* from its belief of ground truth as *truthful* while agent *A* may consider the same measurement as *not truthful*.

Hence, we explain that subjectivity in IoT-generated Big Data with respect to TRM, comes from the fact that IoT sensors have different sensing capabilities with different precision. So the evidence that each IoT device can provide on an arbitrary matter is subject to its sensing capabilities, sensing precision, etc. Hence, subjectivity in IoT generated data is a consequence of heterogeneous sensing capabilities.

To shed more light on the matter we explain a common crowd-sensing scenario in traffic prediction. Google Maps and many other navigation apps provide crowd-sensed traffic predictions [Nagy and Simon, 2018]. Heterogeneous IoT devices such as smart watches, smart cars, and smart phones contribute to this traffic prediction task. Their contribution is two folds; Some contribute by anonymously providing the sensor readings in a periodic fashion. While some other devices contribute by rating the average of the sensor readings provided by the other group. Now consider that the speed sensor of a smart car might have a different precision compared to the speed sensor of a smart phone. Hence in a scenario where these two IoT devices have to submit a feedback about a crowd-sensed congestion on the same road where both of these devices are located, the device with less sensitive speed sensor might not record any movement while the more sensitive sensor might record some slow movement. This can result in a situation where the device with less sensitive speed sensor rates the crowd-sensed traffic prediction as inaccurate simply because its sensors are not sensitive enough to record movement. Hence in this thesis we say that the feedbacks provided by IoT devices can be subjective to the sensing capabilities of such devices and this is what we mean in this thesis by subjectivity in IoT generated Data.

The focus of this chapter is subjective evidence space. As explained above, this manifests in both User-generated Big Data and IoT-generated Big Data. In the remaining of this section, first, we motivate the research problems presented above from the perspective of IoT-generated Big Data using several discussions. Then, we motivate the research problems presented above from the perspective of user-generated Big Data using several discussions. Next, we also motivate the research problems from a generic perspective with respect to Big Data. Finally, we introduce our proposed solution.

First, we motivate the research problems presented in this chapter from the perspective of IoT-generated Big Data using several discussions as follows:

- The application proposed by [Simplify.Inc., 2018] provides a mechanism for the smart-phones to sell/rent their cellular internet connection (e.g. 3G,4G,LTE, etc.,) with nearby smart-phones. That is, using the smart-phone application proposed by [Sim-

[plify.Inc., 2018\]](#) each user can turn their smart-phone into a non-gratuitous hot-spot and charge those who wish to connect to internet using this mobile hot-spot.

The real-world incentives for such an application are many. For example, some users might have spare data left in their prepaid data plan which is about to expire and they would like to share it with others for a reasonable price. The challenge however, is how to know if a nearby hot-spot is trustworthy or not. This can be achieved by using the second-hand evidence provided by other smart-phones which interacted with the hot-spot before. This process is automatic and user does not get involved. However, the research problem that arise is how to deal with the subjectivity of this evidence. As explained earlier, Subjectivity in IoT-generated Big Data comes from the fact that devices have different capabilities with different precision. So the evidence that each device can provide on an arbitrary matter is subject to its capabilities and sensing precision. For example, consider a mobile hot-spot that promises a speed not less than *2GB* per second in its advertised SLA. If a 4G smart phone report the hot-spot as untrustworthy, this can be due to its own limitation. Because the peak cellular speed for a 4G smart phone is *1GB* per second and even less when it comes to their peak WiFi speed. However, the peak 5G speed for a 5G smart phone is *20GB* per second and most of them come with a 802.11ac WiFi adapter which is also referred to as WiFi 5 and has a theoretical peak speed of about *7GB* per second [[Van Nee, 2011](#), [Parkvall et al., 2017](#)]. Hence a mechanism is needed to account for the subjectivity of the evidence space in IoT. Our proposed subjectivity alignment mechanism in Chapter 3 provides a solution to this problem.

- In Section 3.3 We adopt the scenario which we explained above and expand it to a smart city application where smart-phones can share their cellular data connection with heterogeneous IoT devices in the smart city such as the smart cars and environmental sensors. These devices do not have a sim card but can benefit from opportunistic internet connections. The smart-phones advertise their SLA, but they are self-interested agents so they can lie, which justifies the necessity of trust systems being used in such scenarios. Meanwhile, since the devices are heterogeneous, they may evaluate the trustworthiness of the smart phones differently. For example, smart cars have high degree of mobility, so they have a significantly narrow encounter window (the time they will be in the wireless range of a smart-phone). As a consequence, they are more sensitive to response time and latency, which leads to a different preci-

sion and priority for those specific metrics compared to that of environmental sensors which are static. However, the environmental sensors in our example are not connected to a power grid and operate on small solar panels. As a result, re-transmission due to packet loss is very costly for them while smart cars do not have this limitation, and consequently they are not as sensitive to packet loss. The difference between this scenario and that of [Simplify.Inc., 2018] is that in this scenario the SLA consists of multiple criteria. This makes the research problem of the subjectivity of evidence space in IoT even more challenging. Our proposed subjectivity alignment mechanism in Chapter 3 provides a solution to this problem.

- In Section 3.4.2, we adopt a different scenario: The IoT-based crowd-sensing application in the transportation domain proposed in [Luo and Zeynalvand, 2017]. It monitors the average traffic speed of a major road (ground truth is $45mph$), by collecting GPS readings from crowd agents. This is a common scenario in many existing real-world applications such as Google-Maps, Uber, etc., which provide real-time traffic information to drivers to help them avoid routes with heavy traffic. We have two sets of crowd agents: the drivers' car-borne GPS sensors dominantly as the contributing crowd and the passengers' smart phone built-in GPS sensors dominantly as the advising crowd. One important difference between the contributing crowd and the advising crowd is that the latter are reluctant to provide the raw GPS data of their smart-phones due to privacy concerns. However they are willing to authorise their smart-phone to rate the accuracy of sensed data provided by the contributing crowd. These ratings are utilised to reshape the probability distribution of the crowd-sensed data using the mechanism proposed in [Luo and Zeynalvand, 2017]. The research problem that arise is that each agent in the advising crowd may vote *subjectively* on the accuracy of the data sensed by the contributing crowd, subjected to its sensor precision. Our proposed subjectivity alignment mechanism in Chapter 3 provides a solution to this problem.

Second, we also motivate the research problems presented in this chapter from the perspective of User-generated Big Data as follows:

User-rating databases of Video-on-Demand (VoD) giants such as Netflix and Hulu, are good examples of subjective User-generated Big Data [Chen et al., 2014, Zhu et al., 2017]. Hence, the evidence provided by each user in such platforms is often influenced by the user's individual preference. This interferes with the normal function of a TRM system

and misleads the trust evaluation [Jøsang et al., 2007]. Our proposed subjectivity modeling mechanism in Chapter 3 provides a solution to this problem.

Third, we also motivate the research problems presented in this chapter from a generic perspective with respect to Big Data as follows:

Given the continuously growing scale of Big Data, multi-agent approaches are increasingly being used for dynamic processing of the Big Data [Twardowski and Ryzko, 2014, Belghache et al., 2016, Ryzko, 2020]. This chapter provides a multi-agent subjectivity alignment (MASA) mechanism which allows for the parallel processing of the evidence space. Parallelization is the key technique in Big Data analytics and hence is of great importance.

Therefore, subjective evidence space is an issue for both user-generated Big Data and IoT-generated Big Data as we show by conducting simulations and experiments in both disciplines. In this chapter, we seek to align the potentially subjective information with the information seeker’s own subjectivity so that the acquired second-hand information is more useful and personalized. Accordingly, we propose a multi-agent subjectivity alignment (MASA) mechanism¹, which models the subjectivity using a regression technique and exchanges the models among agents as the input to an alignment process. This mechanism substantially counteracts biases incurred by different agents and improves the accuracy of second-hand information fusion as demonstrated by our simulations². Specifically with respect to IoT-generated data. In addition, we also conduct experiments using a real-world dataset of user-generated data.

3.2 MASA

Computational trust systems generally calculate trust scores based on two types of data: first-hand evidence (self experiences) and second-hand evidence (advice). We refer to an agent who offers second-hand evidence as *advisor* and an agent who requests second-hand evidence as *advisee*. An agent can be an advisor, an advisee, or both. Each request by an agent for a specific information/service together with the response from the other agent, is called an interaction and can have several features. For instance, interactions between

¹The work in this chapter is published as Leonit Zeynalvand, Jie Zhang, Tony T Luo, Shuo Chen. *MASA: Multi-agent subjectivity alignment for trustworthy Internet of Things*. In the Proceedings of 21st International Conference on Information Fusion (FUSION), pp. 2013-2020. IEEE, 2018.

²One of the benchmark methods which we used for comparison in Chapter 3, Section 3.4.2 is published as Tie Luo, Leonit Zeynalvand. *Reshaping mobile crowd sensing using cross validation to improve data credibility*. In the Proceedings of (GLOBECOM 2017), pp. 1-7. IEEE 2017.

Figure 3.1: An overview of MASA approach.

IoT agents have features such as response time, latency, etc. Similarly, interactions which involve users³ such as watching a movie or purchasing a product have features such as genre of the movie and shipment delay of the purchase. In the remaining of this chapter, the process of how each agent evaluates the outcome of its interaction with another agent is referred to as interaction outcome evaluation (IOE). The inputs of IOE are interaction features and the output is a first-hand evidence. In MASA, the IOE of each agent is modelled using decision tree regression, where the inputs of IOE are used as the feature set and the output is used as the dependent variable to train the model. We use the classification and regression tree (CART) algorithm [Strobl et al., 2009] to build the decision trees. The agents then exchange these models when offering advice to each other, followed by an alignment process (Section 3.2.1). Each aligned advice then is propagated to the underlying trust and reputation management system to be fused with the rest of the second-hand evidence using uncertainty-aware weights which are computed with our proposed entropy-base method (Section 3.2.2). This improves the accuracy of second-hand evidence fusion. Finally (Section 3.2.3), we explain the rationale why we use decision tree regression as our design choice. Fig. 3.1 provides an overview of MASA.

³In this chapter we represent users with user agents and henceforth we use the term agent to refer to both IoT agents and user agents unless specified.

Algorithm 1 Advice Alignment Algorithm

Inputs: The advice x , the regression trees T of advisor and T' of advisee.

Output: $cpmf$ which is the probability vector of the random variable representing the aligned advice.

```
1: procedure TUNE( $x, T, T'$ )
2:   for each decision path  $r_m^{Tx}$  in  $T$  do
3:      $Set_x \xleftarrow{\text{add}} \text{Conjunction\_of\_Elements}(C(r_m^{Tx}))$ 
4:   end for
5:    $B \leftarrow \text{Disjunction\_of\_Elements}(Set_x)$ 
6:    $Denominator \leftarrow \text{count}(B)$ 
7:   for  $j$  in ratingRange do  $\triangleright$  (e.g.  $j \in \{1, 2, 3, \dots, 10\}$ )
8:     for each decision path  $r_n^{T'j}$  in  $T'$  do
9:        $Set_j \xleftarrow{\text{add}} \text{Conjunction\_of\_Elements}(C(r_n^{T'j}))$ 
10:    end for
11:  end for
12:  for  $j$  in ratingRange do  $\triangleright$  (e.g.  $j \in \{1, 2, 3, \dots, 10\}$ )
13:     $Temp \leftarrow \text{Disjunction\_of\_Elements}(Set_j)$ 
14:     $A \leftarrow \text{Conjunction}(Temp, B)$ 
15:     $Numerator \leftarrow \text{count}(A)$ 
16:     $cpmf[j] \leftarrow Numerator / Denominator$ 
17:  end for
18:  return  $cpmf$   $\triangleright$  (e.g.  $[p_1, p_2, \dots, p_{10}]$ )
19: end procedure
```

3.2.1 Advice Alignment

We denote the decision tree which represents an agent's IOE as T . The global set of interaction features (IOE inputs) used by agents to train their decision trees is denoted by $F = \{f_1, f_2, \dots\}$ (e.g. response time, error rate, delay, etc.). The set of distinct leaves in the decision tree of agent i , or in other words the set of possible outputs of agent i 's IOE, (e.g. excellent, good, fair, bad, etc.) is denoted as $D_i = \{d_{i1}, d_{i2}, \dots\}$. We convert categorical values to numeric form as $\tilde{D}_i = \{1, 2, \dots, |D_i|\}$ with order maintained. The set of paths (rules) in the corresponding decision tree T for agent i from the root to the leaves with a value equal to $j \in \tilde{D}_i$ is denoted as $R^{Tj} = \{r_k^{Tj} | k = 1, 2, \dots\}$ and the set of decision nodes in rule k is denoted as $C(r_k^{Tj}) = \{c_t | t = 1, 2, \dots\}$ where c_t is a condition over a specific $f \in F$ (e.g. $delay < 5ms$). Finally, the condition set, i.e. the conjunction of all $c_t \in C(r_k^{Tj})$ is denoted as $\tilde{C}(r_k^{Tj}) = (c_1 \cap c_2 \cap \dots)$.

In MASA, whenever an advisor agent i offers an advice $x \in \tilde{D}_i$ to advisee agent i' , it also offers T . Having x and T , for each outcome $j \in \tilde{D}_{i'}$, advisee computes the posterior probability of j being observed by itself given that outcome x is observed by advisor i . That

is, in MASA the advisee uses Algorithm 1 to calculate⁴ the posterior probability of Eq. 3.1.

$$p(T'(j)|T(x)) = \frac{|\bigcup_{n=1}^{|R^{T'}|} \tilde{C}(r_n^{T'}j) \cap \bigcup_{m=1}^{|R^{Tx}|} \tilde{C}(r_m^{Tx})|}{|\bigcup_{m=1}^{|R^{Tx}|} \tilde{C}(r_m^{Tx})|} \quad (3.1)$$

Here, given advice x and the decision trees T and T' of the advisor and the advisee as input, Algorithm 1 obtains the denominator in Eq. 3.1 by traversing T from the root to leaf node(s) with a value equal to x (lines 2-6). In each traverse, Algorithm 1 performs logical conjunction on all $c_i \in C(r_m^{Tx})$ or in other words all the vertices in each path from the root to the corresponding leaf node(s). Then, Algorithm 1 applies logical disjunction on the results Set_x obtained from each path. The value of *Denominator* (line 6) can either be computed by counting the number of occurrences of B over a representative sample set of either the advisor's first-hand evidence, or a global first-hand evidence (e.g. maintained by a third party acting as an auditing centre). Intuitively both computation strategies yield equivalent results under the assumption of no discrimination and given enough observations. However, the former is more prone to cold start. Moreover, to ensure privacy both sample sets can be anonymized since B contains no identity information.

Similarly (lines 7-11), Algorithm 1 obtains the numerator of Eqn. 3.1 by also traversing T' (the decision tree of the advisee) in addition to T . Finally, Algorithm 1 obtains a probability vector in the form of $[p_1, p_2, \dots, p_{|D_{T'}|}]$ from the calculated values of the posterior probabilities for all $j \in \tilde{D}_{T'}$ (lines 12-18). This probability vector represents the probability mass function of the subjectively aligned advice as a random variable. MASA passes the expected value of this random variable to the underlying trust and reputation system as the subjectively aligned advice together with a entropy based weight calculated in the next section (3.2.2).

3.2.2 Variance-modulated Entropy

The probability distribution obtained by Algorithm 1 may contain little or no information representing the subjectively aligned advice (e.g. a discrete uniform distribution which means that all values in the advice range are equally likely to be observed). Hence, a measure is needed to capture how informative an advice is after being transformed into advisee's subjectivity domain. This measure can then be utilised to fuse the subjectively aligned advice with the rest of the second-hand evidence. In MASA, we propose such a measure by

⁴A step-by-step example of the procedure is provided in Section 3.3.

combining two metrics in probability theory and statistics: entropy and variance. While either metric partially indicates the amount of information embedded in the random variable representing the transformed advice, neither alone provides a robust measure. To clarify, consider four random variables with given probability vectors:

- $A_1 = [0.10, 0.10, 0.10, 0.10, 0.10, 0.10, 0.10, 0.10, 0.10, 0.10]$
- $A_2 = [0.00, 0.00, 0.00, 0.00, 0.01, 0.09, 0.10, 0.60, 0.10, 0.10]$
- $A_3 = [0.50, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.50]$
- $A_4 = [0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.50, 0.50]$

Each random variable represents a subjectively aligned advice. Entropy justifies picking A_2 over A_1 since the entropy of A_1 is equal to one which means the highest uncertainty, while A_2 has a much lower entropy. However, A_3 and A_4 have the same entropy while A_3 is not the better pick because of its high variance compared to A_4 (i.e. it is not clear the advice is 1 or 10 while in A_4 we are almost certain that the advice is somewhat around 9.5). This shows that entropy alone is not descriptive enough for our case. Hence, we define:

$$\lambda(A) = (1 - H(A))^{\frac{12\sigma^2(A)}{|\tilde{D}_I|^2 - 1}} \quad (3.2)$$

which we refer to as *variance-modulated entropy* as a measure of how informative an advice is after being transformed into an advisee's subjectivity domain. In Eq. 3.2, A is the random variable representing the transformed advice whose probability vector is calculated using Algorithm 1, $H(A)$ is the entropy of A , and $\tilde{D}_I$ is the range of A . To clarify the rationale behind Eq. 3.2, consider a random variable A' with discrete uniform distribution $U\{1, |\tilde{D}_I|\}$. Since A' taking any of the values in its range is equally probable, the uncertainty in the aligned advice is maximum. Hence, we consider A' as the benchmark and compare A with that benchmark to derive $\lambda(A)$. Given its uniform distribution, $H(A')$ is equal to one. As a result, smaller values for $H(A)$ indicate more certainty as captured by the base of Eq. 3.2. Similarly, $\sigma^2(A') = (|\tilde{D}_I|^2 - 1)/12$. Hence, greater values of $\sigma^2(A)$ implies less certainty as captured by the exponent of Eq. 3.2. As illustrated in Fig. 3.2, the value of $\lambda(A)$ is always in range $[0, 1]$ where 1 indicates absolute certainty while 0 means complete uncertainty. Although not all the points on the surface in Fig. 3.2 can be observed in practice since entropy and variance are not completely independent, it is not an issue for MASA as we are only interested in those that are observed.

(a)

(b)

Figure 3.2: Variance-modulated entropy for posterior advice random variable A with a range of $\{1, 2, \dots, 10\}$ ($\sigma_{U\{1,10\}}^2 = 20.25$). (a) 3D plot; (b) Contour plot.

Existing trust and reputation management systems [Chen et al., 2016a, Chen et al., 2016b] often employ a weighted sum of second hand evidence in their fusion mechanisms to aggregate the advice received from different advisors. Hence, MASA propagates $\lambda(A)$ to the underlying trust system where it can be utilised as the weight to fuse the subjectively aligned advice with the second-hand evidence obtained from other agents. This said, as illustrated in Fig. 3.1 MASA is a plugin to existing TRM systems, one that transforms a prior subjective advice to a posterior advice which is better aligned with the advisee’s own subjectivity model and also provides an entropy-based weight for the posterior advice.

3.2.3 Rationale of using Decision Trees

In our approach, the IOE of each agent is modelled by using decision tree regression. The inputs of IOE are used as the feature set and the output is used as the dependent variable to train the decision tree regression model. In this section, we briefly discuss the rationales behind this design choice and its advantages over alternative ML models in the context of subjectivity modeling (see Section 2.1.3 in Chapter 2 for the definitions of various ML methods).

One crucial consideration for MASA is the comparability of the regression model. To clarify, consider an advisor agent and an advisee agent with heterogeneous IOEs. The main goal of modelling their IOEs is to transform them into a common space to compare and derive possible correlations between them. Hence, black-box models such as neural networks are not much helpful for this purpose. Also, logistic regression and support vector machine (SVM) do not produce output models which can seamlessly be used to derive possible correlations. Since decision trees are flowchart-like hierarchical data structures, they can be easily used in an automatic mechanism (tree traversal) to derive the aforementioned correlations, as described in this chapter in Section 3.2.1. Besides, decision trees are highly robust on small datasets which is the case here since the IOE of each agent is individually modelled and the training set is the agent’s own first-hand evidence space.

3.2.4 Computational complexity

Given the continuously growing scale of Big Data, multi-agent approaches are increasingly being used for dynamic processing of the Big Data [Twardowski and Ryzko, 2014, Belghache et al., 2016, Ryzko, 2020]. This chapter provides a multi-agent subjectivity alignment (MASA) mechanism which allows for the parallel processing of the evidence space.

In MASA, a decision tree is built for each agent using the first-hand evidence of that agent. The agents then exchange these trees when offering advice to each other, followed by an alignment process. Hence the discussion with regards to the computational complexity of MASA is two folds:

- Firstly, the process of building the decision tree for each agent is independent from that of other agents. Hence, the decision trees can be built in parallel. The computational complexity of building each decision tree depends on the exact variant of the decision tree algorithm. Given that, the complexity is between $O(m \cdot |F| \log |F|)$ and $O(m \cdot |F|^2)$ where m is the number of first-hand evidence and $|F|$ is the number of the subjective attributes of the promise. In TRM, what constitutes Big Data is the second-hand evidence space and not the first-hand evidence space. Therefore, m and $|F|$ are relatively small values.
- Second, each second-hand evidence then undergoes the alignment process of Algorithm 1 as explained in Section 3.2.1. The alignment process for each evidence is independent from that of other evidence. Hence, the alignments can be performed in parallel. Given an arbitrary second-hand evidence x , the decision tree of the evidence producer (i.e. the adviser agent) and that of the evidence consumer (i.e. the advisee agent) is traversed in Algorithm 1. Hence, the computational complexity of Algorithm 1 is $O(v + e)$ where v is the number of vertices in the decision tree, and e is the number of edges in the decision tree. Since the decision trees are binary trees, the complexity of Algorithm 1 can be expressed as $O(v + v - 1) = O(2v - 1) = O(v)$.

Moreover, we do not restrict MASA to offline assessment. Specifically, in Section 3.2 online versions of the decision tree algorithms shall be used to allow for online assessment. A classic example of such algorithms is proposed by the authors in [Crawford, 1989] where they extend the CART decision tree algorithm to incorporate the training data in an incremental and online fashion. A more recent example of an online decision tree algorithm is proposed by the authors in [Manapragada et al., 2018] where they modify the MOA implementation of Hoeffding Tree.

3.3 A Running Example

It is easy to find examples of subjective user-generated data. For instance movie ratings in Netflix, Hotel Ratings, and more generally item ratings in e-commerce are all examples

of subjective user-generated data. However, due to the lack of widespread presence of IoT enabling technologies such as 5G, existing real world applications of trust in IoT are not as abundant as that of user-generated data but still not hard to find. Hence, in this section, we introduce an existing real-world example of subjective IoT-generated data, and demonstrate how MASA improves the performance of the TRM in such application. Specifically, we adopt the scenario of Internet data plan sharing from [Simplify.Inc., 2018] and scale it to a smart city environment. To better clarify, we first provide a brief summary of the scenario proposed by [Simplify.Inc., 2018]. Then, we expand that scenario to a smart city and describe the agents in our proposed scenario. Next, we explain the details of Internet data plan sharing in the context of smart city and elaborate on where trust is needed and why subjectivity is an issue for the trust system in that scenario. Finally, we explain in full details how we apply MASA to address that issue.

To make the description of the scenario self-contained we explain that the application proposed by [Simplify.Inc., 2018] provides a mechanism for the users of smart-phones to sell/rent their cellular internet connection (e.g. 3G,4G,LTE, etc.,) with nearby smart-phone users. That is, using the smart-phone application proposed by [Simplify.Inc., 2018] each user can turn their smart-phone into a non-gratuitous hot-spot and charge those who wish to connect to internet using this mobile hot-spot.

The real-world incentives for such an application are many. For example, some users might have spare data left in their prepaid data plan which is about to expire and they would like to share it with others for a reasonable price. The challenge however, is how to know if a nearby hot-spot is trustworthy or not. We adopt the scenario which we explained above and expand it to a smart city application where smart-phones can share their cellular data connection with heterogeneous IoT devices in the smart city that do not have a sim card but can benefit from opportunistic internet connections. There are three types of agents in our smart city Internet sharing scenario :

- **Environmental Sensors:** These sensors monitor environment related variables (e.g. noise level, air pollution, etc.) and need to periodically report to the base station, which requires brief opportunistic Internet connectivity (i.e. few times a day, small batches of data transmission).
- **Smart Cars:** They need to periodically check for security updates which again requires brief opportunistic Internet connectivity. Furthermore, vehicular ad hoc networks (VANET) can significantly benefit from this type of connectivity since it would

decrease the dependency on road-side units.

- **Smart Phones:** With the prevalence of enabling cellular technologies (e.g. 4G, LTE, etc.), many phones are connected to the Internet and also most of them have the ability to act as a wifi hotspot.

Smart phones can then sell their spare data plan to other agents who require brief periodic Internet connectivity. In the rest of this example we refer to smart phones as service-provider agents and refer to environmental sensors and smart cars as service-requester agents. One important challenge in realization of such applications is the feasibility of maintaining a sound and scalable trust and reputation management system for this highly heterogeneous multi-agent space. So the smart cars and environmental sensors need to maintain a trust system based on their previous Internet data purchases (first-hand evidence) from the service-provider agents (smart phones) and request advice from each other to enrich their second-hand evidence space. The challenge that any trust system would face in this example scenario is that the potentially different IOE of heterogeneous adviser agents may result in subjective advice which is not the same as dishonest advice. To better clarify on this, consider three quality of service (QoS) metrics as defined below:

- **Response Time:** The time it takes for the service-provider agent to accept or reject a data purchase request.
- **Latency:** The delay between sending a data packet and receiving its delivery acknowledgement.
- **Packet Loss:** The total number of lost data packets in each 15KB of data transmission.

The service-provider agents advertise their QoS metric values. Since they are self-interested agents, they can lie, which justifies the necessity of trust systems being used in such scenarios. Meanwhile, since service-requester agents are heterogeneous, they may evaluate the trustworthiness of the service-provider agents differently. For example, smart cars have high degree of mobility, so they have a significantly narrow encounter window (the time they will be in the wireless range of a service-provider agent). As a consequence, they are more sensitive to response time and latency, which leads to a different precision and priority for those specific metrics in their IOE compared to that of environmental sensors which are static. However, the environmental sensors in our example are not connected

(a)

(b)

Figure 3.3: Decision trees of different agents. (a) Decision tree regression for environmental sensors; (b) Decision tree regression for smart cars.

to power grid and operate on small solar panels. As a result, re-transmission due to packet loss is very costly for them while smart cars do not have this limitation, and consequently they are not as sensitive to packet loss. In our example, these differences are accordingly reflected in the IOE mechanisms of environmental sensors and smart cars as given respectively by Eq. 3.3 and Eq. 3.4:

$$IOE_E = 3 - \left\lceil 0.1 \left\lceil \frac{\Delta L}{5} \right\rceil + 0.9 \left\lceil \frac{\Delta P}{5} \right\rceil \right\rceil \quad (3.3)$$

$$IOE_S = \begin{cases} 1 & \Delta L > 4 \\ 1 & \Delta R > 2 \\ 3 - \left\lceil 0.9 \left\lceil \frac{\Delta L}{2} \right\rceil + 0.1 \left\lceil \frac{\Delta P}{5} \right\rceil \right\rceil & \text{Otherwise} \end{cases} \quad (3.4)$$

where $\Delta L, \Delta R, \Delta P \in [0, 10)$ are the deviation of advertised values from perceived values for latency, response time and packet loss QoS metrics, respectively (e.g. $\Delta L = 4$ means that the perceived latency deviates 4 units from the advertised latency). Also, the output range for both IOE mechanisms is $\{1, 2, 3\}$ (e.g. dishonest, fairly honest, very honest).

As explained in Section 3.2, the IOE of each agent is modelled by using decision tree regression. The corresponding decision trees for environmental sensors and smart cars are depicted in Fig. 3.3. Therefore, whenever an environmental sensor i offers an advice $x \in \tilde{D}_i$ to smart car i' , where $\tilde{D}_i = \{1, 2, 3\}$, it also offers the decision tree T illustrated in Fig. 3.3(a). Having x and T , for each outcome $j \in \tilde{D}_{i'}$, where $\tilde{D}_{i'} = \{1, 2, 3\}$, the smart car computes the posterior probability of j being the outcome observed by itself given that the outcome x is observed by the environmental sensor i . Following Algorithm 1, we demonstrate a complete example for this process with $x = 3$ (very trustworthy according to advice) in Eq. 3.5-3.7.

$$\begin{aligned} & p(T'(3)|T(3)) : \\ &= p(\Delta L, \Delta P \leq 0.5 \wedge \Delta R \leq 1.5 \mid \Delta L, \Delta P \leq 0.5) \\ &= \frac{p(\Delta L, \Delta P \leq 0.5 \wedge \Delta R \leq 1.5)}{p(\Delta L, \Delta P \leq 0.5)} \\ &= \frac{|\Delta L, \Delta P \leq 0.5 \wedge \Delta R \leq 1.5|}{|\Delta L, \Delta P \leq 0.5|} \end{aligned} \quad (3.5)$$

$$\begin{aligned}
& p(T'(2)|T(3)) : \\
& = p((\Delta R, \Delta L \leq 1.5 \wedge \Delta P > 5) \vee \\
& (\Delta R \leq 1.5 \wedge \Delta L \leq 2.5 \wedge 0.5 < \Delta P \leq 5) \vee \\
& (\Delta R \leq 1.5 \wedge \Delta P \leq 0.5 \wedge 0.5 < \Delta L \leq 2.5) | \\
& \Delta L, \Delta P \leq 0.5) \\
& = 0
\end{aligned} \tag{3.6}$$

$$\begin{aligned}
& p(T'(1)|T(3)) : \\
& = p((\Delta R > 1.5) \vee (\Delta R \leq 1.5 \wedge \Delta L > 2.5) \vee \\
& (\Delta R \leq 1.5 \wedge \Delta P > 5 \wedge 1.5 < \Delta L \leq 2.5) | \\
& \Delta L, \Delta P \leq 0.5) \\
& = p(\Delta R > 1.5 | \Delta L, \Delta P \leq 0.5) \\
& = \frac{p(\Delta L, \Delta P \leq 0.5 \wedge \Delta R > 1.5)}{p(\Delta L, \Delta P \leq 0.5)} \\
& = \frac{|\Delta L, \Delta P \leq 0.5 \wedge \Delta R > 1.5|}{|\Delta L, \Delta P \leq 0.5|}
\end{aligned} \tag{3.7}$$

To further clarify, $p(T'(3)|T(3))$ in Eq. 3.5 gives the conditional probability of a leaf node with a value equal to 3 being observed on the decision tree T' of the smart car given the event that a leaf node with a value equal to 3 is observed on the decision tree T of the environmental sensor. The event $T(3)$ is equivalent to $(\Delta L, \Delta P) \leq 0.5$ which we obtain by traversing the decision tree of the environmental sensor from the root to the leaf node(s) with a value equal to 3. Similarly, the event $T'(3)$ is equivalent to $(\Delta L, \Delta P \leq 0.5 \wedge \Delta R \leq 1.5)$ which we obtain by traversing the decision tree of the smart car from the root to the leaf node(s) with a value equal to 3. Under the same principle, $p(T'(2)|T(3))$ and $p(T'(1)|T(3))$ are then obtained in Eq. 3.6 and Eq. 3.7, respectively.

We calculate the final values for the probabilities in Eq. 3.5-3.7 by counting over a representative sample of the first-hand evidence space. For this example, we assume the values to be 30%, 0, and 70% respectively for $p(T'(3)|T(3))$, $p(T'(2)|T(3))$, and $p(T'(1)|T(3))$. These probabilities form a vector which represents the probability mass function of the subjectively aligned advice. We then apply our proposed variance-modulated entropy (Eq. 3.2)

on the calculated probability vector to weight the transformed advice as illustrated in Eq. 3.8.

$$\begin{aligned}
A &= [0.7, 0, 0.3] \\
\lambda &= (1 - H(A))^{\frac{\sigma^2(A)}{\sigma^2(U_{\{1,2,3\}})}} \\
\lambda &= 0.679
\end{aligned} \tag{3.8}$$

The resulted $\lambda(A)$, then is passed to the underlying trust system as an evidence fusion weight and $\mu(A)$, the expected value of A, is passed as the subjectively aligned-second hand evidence.

3.4 Evaluation

Since subjectivity is an issue for both user-generated data and IoT-generated data, we conduct simulations and experiments in both disciplines. Specifically, we demonstrate how subjectivity can negatively impact the performance of the underlying trust system, and evaluate the effectiveness of MASA, by conducting two extensive simulations and also by an experiment with a public dataset. First, we simulate the scenario of Internet data plan sharing in a smart city. Second, we simulate an IoT-based crowd-sensing application in the transportation domain. Next, we apply our approach to real data obtained from MovieLens dataset to validate the efficacy of MASA.

3.4.1 Simulation Scenario 1: Internet data plan sharing in smart cities

Simulation Setup

Benchmark Methods. We compare our approach (MASA) with one previous approach [Fang et al., 2014] referred to as two layered clustering which addresses the subjectivity issue in trust management and a baseline which does not consider subjectivity at all and aggregates all the advice without subjectivity alignment.

Platform. We perform the simulation using a Linux workstation with 12 CPU cores and 32GB of RAM. We use `scikit-learn`⁵ for the implementation of decision trees.

⁵An open-source library for machine learning.

Figure 3.4: Ratio of MAE for different approaches to that of the baseline under different sparsities where (a) 30% (b) 50% (c) 70% of the advisor agents are smart cars.

Evaluation metrics. We employ two commonly used metrics. That is, root mean squared error (RMSE), mean absolute error (MAE) respectively defined as

$$RMSE(T, \hat{T}) = \sqrt{\frac{1}{m} \sum_{i=1}^m (T_i - \hat{T}_i)^2}$$

$$MAE(T, \hat{T}) = \frac{1}{m} \sum_{i=1}^m |T_i - \hat{T}_i|$$

where T is the ground-truth rating, $\hat{T}$ is the rating score predicted by MASA, and m is the total number of predictions.

Simulation Procedure and Results

In this simulation, we adopt the scenario of Internet data plan sharing in a smart city where smart phones sell their spare data plan to other agents (e.g. environmental sensors and smart cars) who require brief periodic Internet connectivity (see Section 3.3 for full details). The IOE functions and parameters of environmental sensors and smart cars are respectively presented by Eq. 3.3 and Eq. 3.4. We consider three different situations in which respectively 30%, 50% and 70% of the advisor agents are smart cars and the rest are environmental

Figure 3.5: Ratio of RMSE for different approaches to that of the baseline under different sparsities where (a) 30% (b) 50% (c) 70% of the advisor agents are smart cars.

sensors. We compare our approach (MASA) with the benchmark methods and calculate the *root mean square error* (RMSE) and *mean absolute error* (MAE) of the aforementioned approaches under different evidence sparsities. The ratio of total advisables (smartphones) to total number of advice in the simulation is considered as a measure of the sparsity of second-hand evidence space as shown in Eqn. 3.9.

$$sparsity = \frac{\#Advisibles}{\#Advice} \quad (3.9)$$

As illustrated in Fig. 3.4 and Fig. 3.5, with the sparsity of the second-hand evidence increasing, MASA outperforms the two layered clustering approach [Fang et al., 2014]. It is also notable that under high sparsity, two layered clustering loses its effectiveness and performs no better than the baseline while MASA converges to a fair effectiveness being approximately around 50 – 60% of the baseline in terms of MAE (60 – 70% in terms of RMSE). The reason is that the effectiveness of the clustering approach depends on the accuracy of clustering. Therefore, the increase in sparsity leads to a decrease in commonly rated service providers, and hence less effective clustering. Moreover, [Fang et al., 2014] is sensitive to the skewness of the evidence distribution and uses it as a smoothing factor. As a result, when the evidence is not skewed which is the case in Fig. 3.4(b) and Fig. 3.5(b), the

approach performs not significantly better than baseline while MASA does not suffer such restrictions, which makes it a suitable candidate for highly heterogeneous environments.

3.4.2 Simulation Scenario 2: IoT-based crowd-sensing application

Simulation Setup

Benchmark Methods. We compare the performance of [Luo and Zeynalvand, 2017] with and without subjectivity alignment using MASA. This is to show that MASA can be used as a plugin to improve the performance of existing TRM methods in crowd-sensing and crowd-sourcing applications.

Platform. We perform the simulation using a Linux workstation with 12 CPU cores and 32GB of RAM. We use `scikit-learn` for the implementation of decision trees.

Evaluation metrics. We employ a commonly used metric which measures the uncertainty of a probability distribution. That is, the entropy of a probability distribution defined as:

$$\varepsilon(s) = \frac{-1}{\log |S|} \sum_{s \in S} p(s) \log p(s)$$

where $p(s)$ is the probability of s , and S is the set of bins.

Simulation Procedure and Results

We adopt the scenario of the IoT-based crowd-sensing application in the transportation domain proposed in [Luo and Zeynalvand, 2017]. It monitors the average traffic speed of a major road (ground truth is $45mph$), by collecting GPS readings from crowd agents. This is a common scenario in many existing real-world applications such as Google-Maps, Uber, etc., which provide real-time traffic information to drivers to help them avoid routes with heavy traffic. We have two sets of crowd agents: drivers' car-borne GPS sensors dominantly as the contributing crowd and passengers' smart phone built-in GPS sensors dominantly as the advising crowd. One important difference between the contributing crowd and the advising crowd is that the latter are reluctant to provide the raw GPS data of their smart-phones due to privacy concerns. However they are willing to authorise their smart-phone to rate the accuracy of sensed data provided by the contributing crowd. These ratings are utilised to reshape the probability distribution of the crowd-sensed data using the mechanism proposed in [Luo and Zeynalvand, 2017]. In this simulation scenario each agent in the advising crowd votes *subjectively* on the accuracy of the data sensed by the contributing crowd, sub-

Figure 3.6: Simulation results for IoT-based crowd-sensing application. (a) probability distribution of the original and the reshaped crowd-sensed data; (b) probability distribution of the reshaped crowd-sensed data with and without MASA.

jected to its sensor precision. The sensor precision of the advisors is considered as a log normal random variable with the mean equals to the sensor precision of the advisee agent and the skewness equals to 75% of the sensor precision of the advisee agent.

As depicted in Fig. 3.6(a), in the presence of a subjective advising crowd without MASA, the advising crowd is not significantly helping the advisee to better reshape the probability distribution of the crowd-sensed data towards the ground truth 45mph. That is because advisors with different speed sensor precisions provide subjectively divergent feedbacks for the same crowd-sensed measurement. However, as depicted in Fig. 3.6(b), with incorporating MASA into the same system as a plugin to align the subjective feedbacks, the improvement is significant. This becomes evident when we measure the entropy of the probability distributions in Fig. 3.6 (b). The entropy of the probability distribution of the crowd-sensed traffic speed using the benchmark method without MASA is 0.90. In contrast, with MASA the measured entropy is 0.82. Hence, MASA increases the credibility of crowd-sensed data and decreases the amount of uncertainty by 9 percent.

3.4.3 Experiment with Real Data

We apply our approach to real data obtained from MovieLens dataset to further validate the efficacy of MASA.

Experiment Setup

Dataset. We use a subset of the public MovieLens dataset [Harper and Konstan, 2015] which contains 25,790 user ratings and we augment its feature space with additional data which we crawl from IMDB website. Our augmentation extends the feature space of the

dataset to contain:

- *Language*. That is the language of the movie.
- *Duration*. That is the duration of the movie expressed in hours.
- *MPAA content rate*. That is an alphanumeric symbol assigned to a movie by the the Motion Picture Association of America (MPAA) indication the fitness of the movie contents for different age groups. Symbols such as R, PG13, and G are examples of such content rate.
- *Primary genre*. That is the main genre of the movie.
- *Secondary genre*. That is the sub-genre of the movie (if any).
- *Third genre*. That is the auxiliary genre of the movie (if any).
- *Release year*. That indicates the year when the movie was released.

Benchmark Methods. We compare our approach (MASA) with one previous approach [Fang et al., 2014] referred to as two layered clustering which addresses the subjectivity issue in trust management and a baseline which does not consider subjectivity at all and aggregates all the advice.

Platform. We perform the experiments using a Linux workstation with 12 CPU cores and 32GB of RAM. We use `scikit-learn`

Evaluation metrics. We employ two commonly used metrics. That is, root mean squared error (RMSE), mean absolute error (MAE) respectively defined as

$$RMSE(T, \hat{T}) = \sqrt{\frac{1}{m} \sum_{i=1}^m (T_i - \hat{T}_i)^2}$$
$$MAE(T, \hat{T}) = \frac{1}{m} \sum_{i=1}^m |T_i - \hat{T}_i|$$

where T is the ground-truth rating, $\hat{T}$ is the rating score predicted by MASA, and m is the total number of predictions.

Experiment Procedure and Results

Here we apply MASA to the augmented dataset. We first model the subjectivity of every user in the dataset as a decision tree using his/her previous ratings. Fig. 3.7 and Fig. 3.8 are

Figure 3.7: Subjectivity of a random Movie-lense user modeled as decision trees in MASA.

Figure 3.8: Subjectivity of a random Movie-lense user modeled as decision trees in MASA.

examples of the obtained decision trees. We then randomly select a movie x , and among the users who rated the movie we randomly pick one. We refer to that randomly selected user as *advisee*. Next, we align the subjective ratings that previous users (*advisors*) have submitted for movie x with the subjectivity of the advisee taking into account the decision trees of the advisors and the advisee using Algorithm 1. The result is a set of objective ratings (relative to the subjectivity of the advisee). We then predict advisee’s rating on movie x based on the weighted average of the objective ratings of the advisors using our proposed weighting mechanism from Eq. (3.2). We repeat the procedure 1000 times. As illustrated in Table 3.1, MASA outperforms the benchmark methods. Specifically, MASA decreases the MAE and RMSE of the predictions respectively by 10% and 4% compared to the state-of-the-art in subjectivity alignment.

Table 3.1: MAE and RMSE of different approaches.

Methods	MAE	RMSE
Baseline	1.502	2.025
Two Layered Clustering	1.339	1.757
MASA	1.215	1.693

3.4.4 Advantages and Limitations

First, as illustrated in Fig. 3.4 and Fig. 3.5, MASA does not lose its efficacy when the first-hand evidence is sparse. This is of great importance since with the prevalence of online services and the growing number of service providers, the first-hand evidence has become too sparse.

Second, no assumptions are made in our simulations or the experiment about the underlying trust and reputation system. This highlights the fact that MASA can be incorporated into any of the existing trust or reputation systems, simply as a plugin. The significance of this independence becomes evident with the large heterogeneity in both user-agents and IoT-agents. To further clarify, consider two agents with heterogeneous trust systems (e.g. one binary trust and the other non-binary trust). MASA enables interoperability between these two agents with fundamentally different TRM systems.

Finally, each computation in MASA is performed either to model the IOE of an agent using decision tree or to align a recently received advice. The computation required to obtain the decision tree of each agent can be performed locally by the node which the agent

is representing, since it only requires the first-hand evidence of that agent. Moreover, the computation required to align any received advice can also be performed locally, since it mainly requires advisee’s decision tree, which is obtained from its first hand evidence and the decision tree of the adviser, which is received together with the advice. This enables all the computations of MASA to be performed in a decentralised manner which increases its scalability.

However, as indicated in Fig. 3.4 and Fig. 3.5, in the presence of extremely dense first-hand evidence, existing approaches such as [Fang et al., 2014] may slightly outperform MASA.

Another limitation of MASA is the reliance on interaction features, since these features together with the interaction outcome are used to train a regression model to represent the agent’s IOE. Although the interaction features are never shared with other agents, and hence pose no privacy risks, they may not be simply available in some systems. In practical TRM systems this issue is mitigated by exposing human advisers to additional questions after they submit their rating (e.g. in Uber this is framed as “What can be improved?”).

3.5 Summary

In this chapter we address subjectivity as an important issue in TRM for Big Data with respect to both user-generated and IoT-generated data. Subjectivity in user-generated Big Data means that the information provided by each user, represented by an agent, is likely to have been influenced by the user’s individual preference and sensitivity, which can be misleading in TRM. Subjectivity in IoT-generated Big Data comes from the fact that IoT sensors have different sensing capabilities with different precision. So the evidence that each IoT device can provide on an arbitrary matter is subject to its sensing capabilities, sensing precision, etc. Hence, subjectivity in IoT generated data is a consequence of heterogeneous sensing capabilities.

Specifically, in this chapter we seek to align the potentially subjective information with the information seeker’s own subjectivity so that the acquired second-hand information is more useful and personalized. Accordingly, we propose a multi-agent subjectivity alignment (MASA) mechanism, which models the subjectivity using a regression technique and exchanges the models among agents as the input to an alignment process. This mechanism substantially counteracts biases incurred by different agents and improves the accuracy of second-hand information fusion. Our extensive simulations as well as our conducted exper-

iments with real datasets demonstrate the effectiveness of MASA.

Chapter 4

Trust and Dense Evidence Space

4.1 Problem Statement

Trust and reputation management (TRM) plays an increasingly important role in large-scale online environments with dense evidence space such as multi-agent systems (MAS) and the Internet of Things (IoT). One main objective of TRM is to achieve accurate trust assessment of entities such as agents or IoT service providers. However, this encounters an *accuracy-privacy dilemma* when the evidence space is dense as we identify in this chapter.

We first provide a clear definition of the type of the privacy that we seek to protect and from whom. Specifically, we explain that in this thesis, we consider the context information of a feedback as private. A feedback on an interaction (e.g. feedback of an agent about a service provided by a service provider) can have many context features. Examples of context features are: time of the service, date of the service, location of the service, etc. This information (as we show in our evaluation of Chapter 4) is very beneficial for an accurate trust analysis. However, as we consider them private, they need to be protected from honest-but-curious parties who would otherwise seamlessly gain knowledge on the private aspects of the users of a Trust and Reputation Management (TRM) system. Regardless of whether this information provides enough incentive for an adversary to attack, its exposure to honest-but-curious parties makes the participants of the TRM system reluctant to share feedback. Hence, the goal in Chapter 4 of this thesis is to mitigate this type of privacy exposure while still benefiting from the positive effect of context information of a feedback on the accuracy of trust assessment.

Therefore, a coarse-grain quantification of the notion of privacy suits best in our discussion of the privacy. With a coarse-grain quantification, the direct exposure of the context

features in the evidence space retains *no* privacy, the non-exposure (i.e. context agnostic TRM) retains *full* privacy, and obfuscated/distorted exposure retains *some* privacy which is more desired than *no* privacy. The goal in Chapter 4 is to retain *some* privacy and *high* accuracy.

The focus of Chapter 4 is dense evidence space since the high density of context-rich evidence allows for building context-aware models. This manifests best in IoT-generated Big Data as explained in Section 1.2.2. We motivate the research problems presented in Chapter 4 from the perspective of IoT-generated Big Data using several discussions as follows:

- In Section 4.4.1, we use a public dataset of IoT-generated Big Data obtained from [Zheng et al., 2014] which contains the response-time values of 4,532 web services provider agents invoked by 142 service agents over 64 time slices. The dataset contains 30,287,611 records of data in total and retains a high degree of evidence density equal to 74.5%. That is, each agent has context-rich first-hand evidence about the majority of other agents.

The research problem that each arbitrary agent a faces is how to share evidence with another arbitrary agent b that does not have enough first-hand evidence about certain service provider. This needs to be done without letting b know that over what exact time-slots a interacted with the service provider agent since this would reveal the potential time-zone of device a to b which is not desired. Our proposed TRM in Chapter 4 provides a solution to this problem.

- In Section 4.4.3, we simulate a multi-agent system with 51 malicious agents and 49 legitimate agents, in consideration of the 51 percent attack. The condition where 51 percent (majority) of the advisers are malicious is a cliché example of where classical trust assessment methods such as BRS fail and in such methods one commonly accepted assumption is that the majority is not malicious. However, this assumption does not always hold in IoT-generated Big Data mainly because of the prevalence of bot-nets in IoT [Marzano et al., 2018, Kolas et al., 2017b, Alieyan et al., 2020]. IoT-based bot-nets are extremely big collection of IoT devices in which an attacker exploits a vulnerability to gain full control over the devices. The attacker therefore can utilise this control to carry out various attacks including trust attacks. In the context of TRM, the objective of such attacks is to poison the evidence space by submitting misleading evidence to change the outcome of the trust assessment in favor

of the malicious service providers.

Moreover, with the presence of IoT-based botnets in a TRM system, not only accurate trust assessment becomes a harder challenge, but also the privacy-accuracy dilemma becomes more important. That is, each arbitrary agent a still needs to share evidence with another arbitrary agent b that does not have enough first-hand evidence about certain service provider otherwise TRM is not possible. However, it is very likely that agent b is a compromised IoT device in the bot-net of the attacker. Hence, it is crucial for a to somehow mask or obfuscate the exact contextual attributes of its interactions with the service provider which otherwise would be revealed to the attacker, and meanwhile retain an accurate trust assessment. Our proposed TRM in Chapter 4 provides a solution to this problem.

In this chapter, we propose a framework¹ called *Context-aware Bernoulli Neural Network based Reputation Assessment* (COBRA) to address the challenges discussed above. COBRA encapsulates agent interactions or transactions, which are prone to privacy leak, in machine learning models, and aggregates multiple such models using a Bernoulli neural network to predict a trust score for an agent. COBRA preserves agent privacy and retains interaction contexts via the machine learning models, and achieves more accurate trust prediction than a fully-connected neural network alternative. COBRA is also robust to security attacks by agents who inject fake machine learning models; notably, it is resistant to the 51-percent attack. The performance of COBRA is validated by our experiments using a real dataset, and by our simulations, where we also show that COBRA outperforms other state-of-the-art TRM systems.

4.2 Model Encapsulation

COBRA encapsulates second-hand evidence in ML models, which achieves two purposes: (i) it preserves privacy of agents who are involved in the past interactions; (ii) it retains context information which will help more accurate trust prediction later (described in the next section).

In this technique, each agent trains a ML model using its past interaction records with other agents in different contexts. Specifically, an agent $u \in \mathcal{A}$ (the set of all the agents)

¹The work in this chapter is published as Leonit Zeynalvand, Tony T Luo, Jie Zhang. COBRA: Context-Aware Bernoulli Neural Networks for Reputation Assessment. In the Proceedings of AAAI ,pp. 7317-7324. AAAI, 2020.

Figure 4.1: The COBRA framework.

trains a model $\mathcal{M}_u^z(\zeta) = p$ based on its past direct interaction (i.e., first-hand evidence) with an agent z . The input to the model is a set ζ of context features (e.g., date, time, location), and the output is a predicted conditional probability p indicating how trustworthy z is for a given context ζ .

To build this model, the agent u maintains a dataset that records its past interactions with each other agent, where each record includes the context ζ and the outcome $t \in \mathbb{Z}_2 = \{0, 1\}$ with 0 and 1 meaning a negative and a positive outcome respectively (e.g., whether the SLA is met or not). For non-binary outcomes, they can be handled by using the common method of converting a multi-class classification problem into multiple binary classification problems (so there will be multiple models for each agent). Then, agent u trains a machine learning model for each agent, say z , using the corresponding dataset to obtain $\mathcal{M}_u^z(\zeta) = p$.

COBRA does not restrict the choice of ML models and this is up to the application and the agents. For example, agents hosted on mobile phones can choose simple models such as decision trees and Naive Bayes, while those on desktop computers or in the cloud can use more complex models such as random forests and AdaBoost. Furthermore, agents can choose different models in the same application, meaning that $\mathcal{M}_{u_1}^z$ may not be the same type as $\mathcal{M}_{u_2}^z$. However, the context feature set ζ needs to be fixed for the same application. That is, if an agent's model uses a subset of context features in ζ , then its models will simply be supplied with those context features. However, if some agents need more context parameters (features not in ζ), their shared models will be treated as missing models (we expand on this in Section 4.3.4).

Model Sharing. Whenever an agent, say a , seeks advice (i.e., second-hand evidence) from another agent, say u , about an agent of interest, say z , the agent u can share its model $\mathcal{M}_u^z(\xi)$ with a . This avoids exposing second-hand evidence directly and thereby preserves privacy of both u and z . It also retains context information as compared to u providing a with just a single trust score of z , and hence helps more informed decision making in the subsequent step (described in Section 4.3).

Note that the information we seek to keep private is the contextual details of the interactions between u and z , whereas concealing the identities of u and z is not the focus of this work.

Sharing a model is as straightforward as transferring the model parameters to the soliciting agent (i.e., a in the above example), or making it accessible to all the agents (in a read-only mode). This sharing process does not require a trusted intermediary because the model does not present a risk of privacy leaking about u and z . One can always apply the models to get trust assessment of the others in a context of interest but there is no guarantee that the adviser agent (i.e., u in the above example) has interacted in that context with the agent of interest (i.e., z in the above example). This helps to retain the interaction details private without losing accuracy. For example, exchanging any black-box ML model will never reveal the exact time of service consumption since we do not have the unrealistic assumption of *know it all* advisers. Hence a model may be more accurate in one context, say during night, while some other model may be more accurate during day and feeding BNN with context features in addition to model outputs keeps COBRA effective in such realistic situations while mitigating the privacy exposure. The required storage is also very low as compared to storing the original raw evidence.

Moreover, COBRA does not assume that all or most models are accurate. Unlike many existing work assuming *honest majority* and hence being vulnerable to 51-percent attack, COBRA use a novel neural network architecture (Section 4.3) that is more robust to model inaccuracy and even malice (e.g., models that give opposite outputs).

4.3 Bernoulli Neural Network

After model encapsulation which allows for a compressed transfer of context-aware second-hand evidence with privacy preservation, the next question is how to aggregate these models to achieve an accurate prediction of the trustworthiness of a target agent. Using common measures of central tendency such as mean, mode, etc. will yield misleading results because

an adviser agent's (u 's) model was trained on a dataset with likely different contexts than the advisee agent's (a 's) context. In a sense, this problem is akin to the problem found in *transfer learning*. Besides, COBRA aims to relax the assumption of *honest majority* and give accurate predictions even when the majority of the models are inaccurate or malicious.

In this section, we propose a solution based on artificial neural networks (ANN). The reasons for choosing ANN are two. First, the task of predicting trustworthiness in a specific context given other agents' models, is a linearly non-separable task with high dimensional input space (detailed in Section 4.3.1). Such tasks can specifically benefit from ANN's capability of discovering intrinsic relationships hidden in data samples [Zhang, 2010]. Second, the models are non-ideal due to the possibly noisy agent datasets, but ANN is highly noise tolerant and sometimes can even be positively affected by noise [Madić and Marinković, 2010, Luo and Yang, 2014].

Therefore, we propose a Bernoulli Neural Network (BNN) as our solution. BNN specializes in processing data that is a set of probabilities associated with random variables of Bernoulli distribution, which perfectly matches our input space which is a set of predicted trust scores between zero and one indicating the probability of an agent being trustworthy in a given context. In contrast to the widely used Convolutional Neural Network (CNN), BNN does not require data to have a grid-like or structured topology, and hence matches well with trust or reputation scores. Specifically, unlike CNN which uses the hierarchical pattern in data, BNN uses information entropy, to determine the connections in the network.

Fig. 4.1 provides an overview of COBRA, where the models on the left hand side are from the encapsulation technique described in Section 4.2, and the right hand side is the BNN described in this section. We make no centralized assumption. So BNNs in Fig. 4.1 can be trained and stored locally or on a central server or cloud. Each agent has its own BNN and there is no single global BNN. In the following, we explain the architectural design of BNN in Section 4.3.1-3 and assemble the data required for training the BNN in Section 4.3.4.

4.3.1 Topology

We propose a $(N + 1)$ -layer network architecture for the BNN, where the input layer is denoted by L_0 , the output layer by L_N , and the hidden layers by L_k where $k = 1, \dots, N - 1$. The weight of an edge (i, j) is denoted as w_{ij} where $i \in L_{k-1}$ and $j \in L_k$, $k = 1, 2, \dots, N$. The bias at layer L_{k-1} is b_{k-1} . Thus, the entire network can be compiled from Eq. 4.1 where

$$y_j = \begin{cases} f_k \left(b_{k-1} + \sum_{i \in L_{k-1}} \text{sgn} \left(\left| \frac{|L_k| - h(y_i)|L_k|}{\left| \left\{ y_t \mid \frac{\partial y_t}{\partial y_i} \neq 0, t \in L_k \right\} \right| + 1} \right| \right) w_{ij} y_i \right), & j \in L_k : k \in \mathbb{N} \\ x_j, & j \in L_0 \end{cases} \quad (4.1)$$

the output of any node $j \in L_k$ is given by y_j . The inputs x_j of the network in Eq. 4.1 are assembled from (1) the models explained in Section 4.2 and (2) the context features, where the assembling process is explained in Section 4.3.4. The former are values between zero and one which indicate the predicted trustworthiness probability of an entity in the given context sourced from different predictors.

The probabilistic nature of the inputs, enables us to calculate how informative an input is, by calculating the entropy of the predicted trustworthiness for which the input indicates the probability. This is used in Eq. 4.1 to ensure that the number of neural units an input connects to is inversely proportional to the average of information entropy calculated for input samples. Each sample of an input in the training data-set (exclusive of context features) is a probability (sourced prediction) associated with a random variable with Bernoulli distribution (trustworthiness). Hence, $h(\cdot)$ is defined recursively by Eq. 4.2 where $\bar{H}(x_j)$ is the average entropy of x_j in the training data-set. For context features $\bar{H}(x_j) = 0$ because the values of these features are not probabilities of Bernoulli random variables and hence the notion of entropy can only be applied to their entire feature space not individual values which is what is used in Eq. 4.1.

Moreover, $f_k(\cdot)$ in Eq. 4.1 is the activation function of layer k . The design choice of the activation functions is explained in Section 4.3.3.

$$h(y_j) = \begin{cases} \frac{\sum_{i \in I} h(y_i)}{|I|}, & j \in L_k : I = \{i \in L_{k-1} \mid \frac{\partial y_j}{\partial y_i} \neq 0\} \\ \bar{H}(x_j), & j \in L_0 \end{cases} \quad (4.2)$$

4.3.2 Depth and width

The depth of BNN is N since the input layer is not counted by convention. A feed-forward network with two hidden layers can be trained to represent any arbitrary function, given sufficient width and density (number of edges) [Heaton, 2008]. Our goal is to find the function which most accurately weights the predictions sourced from multiple predictors (i.e. high dimensional input space). Many of such sources can be unreliable or misleading either unintentionally (e.g. malfunction) or deliberately (e.g. malicious). There often does not exist a single source which is always reliable and some sources are more reliable in some contexts. Moreover the malicious sources sometimes collude with each other to make the attack harder to detect. Therefore, the function that we aim to estimate in this linearly non-separable task can have any arbitrary shape. Hence, we choose $N = 3$ in our design to benefit from two hidden layers which suffice to estimate the aforementioned function as we demonstrate by our experiment results in Section 4.4.

The width of a layer is the number of units (nodes) in that layer, and accordingly we denote the width of a layer k by $|L_k|$. Determining the width is largely an empirical task, and there are many rule-of-thumb methods used by the practitioners. For instance, [Heaton, 2008] suggests that the width of a hidden layer be $2/3$ the width of the previous layer plus the width of the next layer. Inspired by this method we propose a measure called *output gain* defined as the summation of the information gain of the inputs of a node and determine $|L_k|$ by Eq. 4.3. The width $|L_N|$ is set to 1 because the network has only a single output which is the trust score (probability of being trustworthy). And the width $|L_0|$ is set to the total number of input nodes denoted by n .

$$|L_k| = \begin{cases} n, & k = 0 \\ \left\lceil \sum_{j \in L_0} \frac{2}{3} (1 - \overline{H}(x_j)) \right\rceil + |L_2|, & k = 1 \\ \left\lceil \frac{2}{3} |L_1| \right\rceil + |L_3|, & k = 2 \\ 1, & k = 3 \end{cases} \quad (4.3)$$

4.3.3 Activation and loss functions

Let us recall the activation function $f_k(\cdot)$ from Eq. 4.1 in Section 4.3.1. Since we choose $N = 3$ as explained in Section 4.3.2, we need to specify three activation functions $f_1(\cdot)$, $f_2(\cdot)$, and $f_3(\cdot)$ for the first hidden layer, second hidden layer, and the output layer, respectively.

For the output layer, we choose the sigmoid logistic function $f_3(z) = 1/(1 + e^{-z})$ because we aim to output a trust score (the probability that the outcome of interacting with a certain agent is positive for a given context). For the hidden layers, we choose the rectified linear unit (ReLU) [LeCun et al., 2015] function as $f_{1,2}(z) = \max(0, z)$, because the focus of hidden layers is to exploit the compositional hierarchy of the inputs to compose higher level (combinatoric) features so that data become more separable in the next layer, and hence the speed of convergence is a main consideration.

The weights in the BNN are computed using *gradient descent back propagation* during the training process. However, sigmoid activation functions, as we choose, have a saturation effect which will result in small gradients, while gradients need to be large enough for weight updates to be propagated back into all the layers. Hence, we use *cross-entropy* $H(p, q) = -\sum_x p(x) \log(q(x))$ as the loss function to mitigate the saturation effect of the sigmoid function. Specifically, the $\log(\cdot)$ component in the loss function counteracts the $\exp(\cdot)$ component in the sigmoid activation function, thereby enabling gradient-based learning process.

4.3.4 Assembling training data

Having explained the architectural design aspects of our Bernoulli neural network, now we explain its computational aspects.

The output of the neural network is a predicted probability that a target agent z is trustworthy (e.g., meets SLA) in a certain context ζ , which (the probability) is what an agent a tries to find out. The input of the network consists of (1) all the context features $\varsigma \in \zeta$ and (2) all the probabilities predicted by models $\mathcal{M}_u^z(\zeta)$ shared by all the $u \in X_a$ where $X_a \subseteq \mathcal{A}$ is the agents a is seeking advice from. In the case that some agents $u \in X_a$ do not share their models with agent a , the corresponding input probability will be set to 0.5 to represent absolute no information. Feeding BNN with context features in addition to predictions from exchanged models, enables COBRA to learn which models are more credible in what contexts, as we make no assumption that a model should be equally accurate or inaccurate

in all contexts. Formally, the input from the models to the neural network is given by

$$G_a(x, z, \zeta) = \begin{cases} \mathcal{M}_x^z(\zeta) & \mathcal{M}_x^z \in R_a \\ 0.5 & \mathcal{M}_x^z \notin R_a \end{cases} \quad (4.4)$$

where R_a is the set of models available to a . Most precisely, each input variable (to layer L_0) is specified by

$$x_j = \begin{cases} \varsigma_j & j = 1, 2, \dots, |\zeta| \\ G_a(X_a^{j-|\zeta|}, z, \zeta) & j = |\zeta| + 1, \dots, |\zeta| + |X_a| \end{cases} \quad (4.5)$$

which also gives the number of input nodes (i.e. input dimension)

$$n = |X_a| + |\zeta|.$$

Algorithm 2 Training data initialization

Input: First-hand evidence of agent a :

$E_a = \{(z, \varsigma_1, \varsigma_2, \dots, \varsigma_k, t) \mid z \in \mathcal{S}_a \subseteq \mathcal{S}, k = |\zeta|, t \in \{0, 1\}\}$ where $\mathcal{S}$ is the set of target agents (whose reputation is to be predicted), $R_a = \{\mathcal{M}_u^z \mid x \in X_a \subseteq \mathcal{A}, z \in \mathcal{S}\}$

Output: Training dataset (*Features* and *Label*)

```

1: Features, Label, tmp  $\xleftarrow{\text{set}} \emptyset$ 
2:  $i, j \xleftarrow{\text{set}} 0$ 
3: for each  $(z, \varsigma_1, \varsigma_2, \dots, \varsigma_k, t)$  in  $E_a$  do
4:   for each  $u$  in  $X_a$  do
5:      $tmp[i] \xleftarrow{\text{set}} G_a(x, z, \varsigma_1, \varsigma_2, \dots, \varsigma_k)$ 
6:      $i \xleftarrow{\text{set}} i + 1$ 
7:   end for
8:    $Features[j] \xleftarrow{\text{set}} (\varsigma_1, \varsigma_2, \dots, \varsigma_k, tmp[0], \dots, tmp[|X_a|])$ 
9:    $Label[j] \xleftarrow{\text{set}} t$ 
10:   $j \xleftarrow{\text{set}} j + 1$ 
11: end for
12: return Features, Label
```

Algorithm 3 Update training data vertically

Input: A new first-hand evidence $(z, \varsigma_1, \varsigma_2, \dots, \varsigma_k, t)$ where $z \in \mathcal{S}_a \subseteq \mathcal{S}, k = |\zeta|, t \in \{0, 1\}$

Current training dataset: *Features* and *Label*

Model repository: $R_a = \{\mathcal{M}_u^z \mid x \in X_a \subseteq \mathcal{A}, z \in \mathcal{S}\}$.

Output: Updated training data-set (new *Features* and *Label*).

```
1:  $i \xleftarrow{\text{set}} 0$ 
2: for each  $u$  in  $X_a$  do
3:    $tmp[i] \xleftarrow{\text{set}} G_a(x, z, \varsigma_1, \varsigma_2, \dots, \varsigma_k)$ 
4:    $i \xleftarrow{\text{set}} i + 1$ 
5: end for
6:  $Features[j] \xleftarrow{\text{append}} (\varsigma_1, \varsigma_2, \dots, \varsigma_k, tmp[0], \dots, tmp[|X_a|])$ 
7:  $Label[j] \xleftarrow{\text{append}} t$ 
8: return Features, Label
```

Algorithm 4 Update training data horizontally

Input: A new model $\mathcal{M}_{u'}^{z'}$ where $u' \in \mathcal{A}$ and $z' \in \mathcal{S}$

Current training data-set: *Features* (no need for *Label*)

First-hand evidence of agent a :

$E_a = \{(z, \varsigma_1, \varsigma_2, \dots, \varsigma_k, t) \mid z \in \mathcal{S}_a \subseteq \mathcal{S}, k = |\zeta|, t \in \{0, 1\}\}$

Output: Updated training data-set (new *Features*).

```
1:  $i \xleftarrow{\text{set}} 0$ 
2: for each  $(z, \varsigma_1, \varsigma_2, \dots, \varsigma_k, t)$  in  $E_a$  do
3:   if  $z = z'$  then
4:      $Features[i].u' \xleftarrow{\text{set}} \mathcal{M}_{u'}^{z'}(\varsigma_1, \varsigma_2, \dots, \varsigma_k)$ 
5:   end if
6:    $i \xleftarrow{\text{set}} i + 1$ 
7: end for
8: return Features
```

Thus, we transform the recursive Eq. 4.3 into a system of linear equations:

$$\begin{cases} |L_1| = \frac{2}{3} \left(\left\lceil \sum_{j \in L_0} (1 - \overline{H}(x_j)) \right\rceil \right) + |L_2| \\ |L_2| = \frac{2}{3}(|L_1|) + 1 \end{cases} \quad (4.6)$$

Solving Eq. 4.6 yields the widths of all the layers of our neural network:

$$\begin{aligned} |L_1| &= 2 \left(\left\lceil \sum_{j \in L_0} (1 - \overline{H}(x_j)) \right\rceil \right) + 3, \\ |L_2| &= \left\lfloor \frac{4}{3} \left(\left\lceil \sum_{j \in L_0} (1 - \overline{H}(x_j)) \right\rceil \right) \right\rfloor + 3, \\ |L_3| &= 1. \end{aligned} \quad (4.7)$$

The weights are calculated using gradient descent back propagation based on training data. The training data is initialized once using Algorithm 2 and updated *vertically* upon acquiring new first-hand evidence using Algorithm 3 and updated *horizontally* upon acquiring a new model using Algorithm 4.

In Algorithm 2, the training data - which consists of *Features* as given by Eq. 4.4 and *Label* - is first initialized to $\emptyset$. Then, the first-hand evidence E_a is being iterated over (line 3) to find out historical information about the agent z , i.e., the outcome t and context $\varsigma_1, \dots, \varsigma_k$ of each interaction. This information is then supplied to $G_a(\cdot)$ (Eq. 4.4) to obtain the predicted conditional probability $P(t = 1 | \varsigma_1, \dots, \varsigma_k)$. The probabilities and the corresponding labels are then added to *Features* to form the training data (lines 8-12).

After initialization, all the subsequent updates are performed using Algorithm 3 and 4, where Algorithm 3 is executed when a new first-hand evidence is available at a and Algorithm 4 is executed when a receives a new model from a new advisor agent or an updated model from an existing advisor agent. This allows for COBRA to be an open MAS in which agents can join and leave any time.

Proposition 4.3.1 *The time complexity of Algorithm 2 is $O(|E_a| \times |X_a|)$.*

Proposition 4.3.2 *The time complexity of Algorithm 3 and 4 is $O(|X_a|)$ and $O(|E_a|)$, respectively.*

The training and retraining of the neural network using the above training dataset can be either performed by the agent itself or outsourced to fog computing [Yousefpour et al.,

2019]. Similarly is the storage of the neural network since the training only utilises the training-data obtained from Algorithm 2-4 in which there is neither a link between two different agents nor a link between a given context ζ and any specific target agent. Hence, the training/retraining process can be outsourced to intermediaries without requiring any *trusted* intermediary. Moreover, the agents do not even need to store the network or any of the shared models since the storage can also be outsourced to intermediaries without trusting them. Examples of such intermediaries are cloud computing, Blockchain, etc. The agents then simply need to retrieve the trained network from intermediaries to make a prediction using the network, which is a fast process without requiring any special computational capabilities.

Hence in summary, we train the network with the training-data obtained by Algorithm 2-4 to compute all the weights in Eq. 4.1. Then the network can predict the trustworthiness score for a given agent z and a given context ζ , i.e., probability that interacting with z under ζ will meet the SLA.

4.3.5 Computational complexity

Given the continuously growing scale of Big Data, multi-agent approaches are increasingly being used for dynamic processing of the Big Data [Twardowski and Ryzko, 2014, Belghache et al., 2016, Ryzko, 2020]. In this chapter we take a multi-agent approach where we propose COBRA, a context-aware trust assessment framework for high-density Big Data environments such as IoT. COBRA encapsulates second-hand evidence using ML models, and makes accurate trust predictions using a Bernoulli Neural Network (BNN) which fuses both first-hand and second-hand evidence, where the valuable context information was preserved by the ML models. Hence the discussion with regards to the computational complexity of COBRA is three folds:

- First, in evidence encapsulation, each agent trains a ML model using its past interaction records with other agents in different contexts. COBRA does not restrict the choice of ML models and this is up to the application and the agents. For example, agents hosted on mobile phones can choose simple models such as decision trees and Naive Bayes, while those on desktop computers or in the cloud can use more complex models such as random forests and XGBoost. Hence in COBRA the ML models used in evidence encapsulation can be heterogeneous. For example, in part of our evaluation of COBRA in Section 4.4.2, in a hybrid approach called COBRA-Hyb,

decision tree is used for half of the agents while Gaussian Naive Bayes is used for the rest. Hence, the computational complexity of evidence encapsulation for each agent depends on the agent's choice of ML model. For example, the train-time complexity of Support Vector Machine (SVM) is $O(n^2)$ while that of Naive Bayes is $O(n.d)$ where n is the number of first-hand evidence, and d is the number of context features.

- Second, as explained in Section 4.3.4, Algorithms 2-4 use the ML models to prepare the training data for the BNN. This training data is initialized once using Algorithm 2 and updated *vertically* upon acquiring new first-hand evidence using Algorithm 3 and updated *horizontally* upon acquiring a new model using Algorithm 4. The computational complexity of Algorithm 2 is $O(|E_a| \times |X_a|)$ while that of Algorithms 3-4 respectively are $O(|X_a|)$ and $O(|E_a|)$, where $|E_a|$ is the number of first-hand evidence of an arbitrary agent a and $|X_a|$ is the number of its advisers.
- Third, this training data is then used to train our proposed BNN. In our evaluation of COBRA using a real-world dataset, we compare the training time of our proposed BNN to that of a normal fully-connected neural network (see Figure 4.2(e)). The results indicate that our proposed BNN architecture significantly reduces the time required for training.

Moreover, we do not restrict COBRA to offline assessment. Particularly, our proposed BNN in Section 4.3 with stochastic gradient descent is an online model. Also, our proposed evidence encapsulation in Section 4.2 leaves the choice of the ML models to each IoT agent, and hence the agents can choose to use online ML models if its desired. For example, if an agent chooses to use decision trees (DT) for evidence encapsulation, then online versions of DT shall be used for online assessment. A classic example of such algorithms is proposed by the authors in [Crawford, 1989] where they extend the CART decision tree algorithm to incorporate the training data in an incremental and online fashion. A more recent example of an online decision tree algorithm is proposed by the authors in [Manapragada et al., 2018] where they modify the MOA implementation of Hoeffding Tree.

4.4 Evaluation

We evaluate COBRA using both experiments and simulations.

4.4.1 Experiment setup

Dataset. We use a public dataset obtained from [Zheng et al., 2014] which contains the response-time values of 4,532 web services invoked by 142 service agents over 64 time slices. The dataset contains 30,287,611 records of data in total, which translates to a data sparsity of 26.5%. Following [Nielsen, 1994], we assume a standard SLA which specifies that 1 second is the limit that keeps a user’s flow of thought uninterrupted. Hence, response time above 1 second is considered violation of SLA and assigned a *False* label, while response time below or equal to 1 second is assigned a *True* label which indicates that the SLA is met. We consider non-Boolean outcomes a future research perspective as we discuss in Chapter 6.

Platform. All measurements are conducted using the same Linux workstation with 12 CPU cores and 32GB of RAM. The functional API of `Keras` is used for the implementation of the neural network architectures on top of `TensorFlow` backend while `scikit-learn` is used for the implementation of Gaussian process, decision tree, and Gaussian Naive Bayes models.

Benchmark methods. We use the following benchmarks for comparison:

- *Trust and Reputation using Hierarchical Bayesian Modelling (HABIT)* : This probabilistic trust model is proposed by [Teacy et al., 2012] and uses Bayesian modelling in a hierarchical fashion to infer the probability of trustworthiness based on direct and third-party information and outperforms other existing probabilistic trust models.
- *Trust Management in Social IoT (TMSIoT)*: This model is proposed by [Nitti et al., 2014], in which the trustworthiness of a service provider is a weighted sum of a node’s own experience and the opinions of other nodes that have interacted with the service provider.
- *Beta Reputation System (BRS)*: This well-known model as proposed by [Josang and Ismail, 2002] uses the beta probability density function to combine feedback from various agents to calculate a trust score.

Figure 4.2: Experiment results: (a,b) ACC and RMSE of different approaches; (c,d) Moving average of prediction and training time for BNN compared to Dense; (e,f) Convergence of validation accuracy and loss for BNN compared to Dense.

Evaluation metrics. We employ two commonly used metrics. One is the accuracy defined as

$$ACC = \frac{TP + TN}{TP + TN + FP + FN}$$

where TP = True Positive, FP = False Positive, TN = True Negative, and FN = False Negative. The other metric is the root mean squared error (RMSE) defined by

$$RMSE(T, \hat{T}) = \sqrt{\frac{1}{m} \sum_{i=1}^m (T_i - \hat{T}_i)^2}$$

Where T is the ground-truth trustworthiness and $\hat{T}$ is the predicted probability of trustworthiness and m is the total number of predictions.

4.4.2 Experiment procedure and results

We run COBRA for each of the 142 web-service clients to predict whether a web-service provider z can be trusted to meet SLA, given a context ζ which is the time slice during which the service was consumed. We experiment on 800,000 random samples of the dataset due to two main considerations: (1) COBRA is a multi-agent approach but in the experiment we build all the models and BNNs on one machine, (2) the significantly high time and space complexity of the Gaussian process used in HABIT restricts us to work with a sample of the dataset. We employ 10-fold cross validation and compare the performance of COBRA with the benchmark methods described in Section 5.3.1. In COBRA-DT, decision tree is used for model encapsulation for all 142 agents, in COBRA-GNB, Gaussian Naive Bayes is used for the encapsulation for all 142 agents, and in a hybrid approach, COBRA-Hyb, decision tree is used for 71 randomly selected agents while Gaussian Naive Bayes is used for the rest. In HABIT the reputation model is instantiated using Gaussian process with a combination of *dot product* + *white kernel* co-variance functions. In COBRA-DT/GNB/Hyb-B, our proposed neural network architecture in Section 4.3 (BNN) is used, while in COBRA-DT/GNB/Hyb-D, a fully connected feed-forward architecture (Dense) is used instead. For fair evaluation, BNN and Dense are configured similarly, and no hyper-parameter tuning² is conducted as the common standard learning rate of 0.01 and Glorot (also called Xavier) uniform initializer (Keras defaults) fit well the complexity of the target task. The choice of values for depth, width, activation functions, etc., for DNN is set to match that of BNN which is explained in Section 4.3.2 and Section 4.3.3.

The results, as illustrated in Fig. 4.2(a) and Fig. 4.2(b), indicate that all the versions of COBRA with Bernoulli neural engine outperform the benchmark methods, while without our proposed Bernoulli neural architecture, HABIT is competent to Dense version of COBRA-GNB. The choice of the encapsulation model only slightly affects the performance in hybrid mode, which suggests that the performance of COBRA is stable.

Furthermore, we present the moving average of prediction and training time for BNN versions of COBRA compared to Dense versions of COBRA respectively in Fig. 4.2(f) and Fig. 4.2(e). The results indicate that our proposed BNN architecture significantly reduces the time required for training and making predictions.

Moreover, as illustrated in Fig. 4.2(c), the divergence between training accuracy and validation accuracy of BNN is significantly smaller than that of Dense. Similarly, Fig.

²Tuning was attempted during our experiments, no significant change was observed in the results.

4.2(d) depicts a smaller divergence between training loss and validation loss of BNN compared to that of Dense. These results indicate that Dense is more prone to overfitting as the epochs increase.

4.4.3 Simulation setup

For a more extensive evaluation of COBRA, especially with respect to extreme scenarios which may not be observed often in the real world, we also conduct simulations.

We simulate a multi-agent system with 51 malicious agents and 49 legitimate agents, in consideration of the 51 percent attack. The condition where 51 percent (majority) of the advisers are malicious is a cliché example of where classical trust assessment methods such as BRS fail and in such methods one commonly accepted assumption is that the majority is not malicious. The attack model used for malicious agents consists of misleading testimonies which is a common attack in TRM systems. Specifically, a model shared by a malicious agent provides opposite prediction of the trustworthiness probability of a target agent, i.e., it outputs $1 - p$ when the model would predict p if it were not malicious.

In order to keep the simulation conclusive we avoid cherry-picking simulation scenarios and simulate 100 different scenarios which cover all possible extremities as well as non-extremities. This is made possible by using a random variable with beta distribution as the probability that two arbitrary agents does not interact with each other. Denote by Φ the probability that an arbitrary agent does not interact with an arbitrary target agent, is a random variable with beta distribution parameterized by α and β . We run 100 simulations each with a different distribution of Φ . The probability distribution function of Φ for some example values of α and β is illustrated in Fig. 4.3(c). For example, $\alpha = \beta = 0.5$ means that one group of agents interact with the target agent frequently while another group seldom interact with the target agent; $\alpha = \beta = 2$ means that most of the agents have half chance to interact with the target agent; $\alpha = 1$ and $\beta = 3$ means that most of the agents interact with the target agent frequently, while $\alpha = 5, \beta = 1$ means that most of the agents seldom interact with the target agent.

Since context features are those of the interaction (e.g. time, location, etc.,) not the inter-actors, they are independent of whether the actor is good or bad. Hence, to avoid bias, we randomly generate features for interactions with both good and bad agents. Specifically, we use 4 synthesized context features randomly distributed in the range $[-1, 1]$, and generate 100 different target agents that violates SLA with a probability following the normal

Figure 4.3: Simulation results: (a) Accuracy of COBRA (in percentage) versus different distributions of Φ parameterized by α and β (in log scale). (b) Contour plot of (a); The color spectrum from *dark* to *light* represents accuracy from *low* to *high*; (c) The Beta distribution function (adapted from [Horas, 2018])

distribution on condition of each context feature.

4.4.4 Simulation results

The simulation results are shown in Fig. 4.3(a)-4.3(b), where the key observations are:

- COBRA is able to predict accurate trust scores (probability of being trustworthiness) for the majority of the cases. Particularly, in 90 out of 100 simulated distributions of Φ an accuracy greater than or equal to 85% is achieved.
- Specifically, COBRA works best when α is low and β is high. This is because low α and high β means that given any arbitrary target agent of interest, most of the agents

interact with the target agent frequently. As a result the evidence space from which the agents derive their ML models is dense and hence, yields more accurate models.

- COBRA does not require that a model be equally accurate in all contexts nor it requires all models to be accurate. In contrast, COBRA has high tolerance for inaccurate models. The reason for this is that our proposed BNN, enables COBRA to learn which models are more credible. This is evident based on two observations:
 1. The results in Fig. 4.3(a)-4.3(b) are achieved when 51% of the agents are malicious and intentionally disseminate inaccurate models. This shows that COBRA is resistant to the 51 percent attack.
 2. COBRA works accurately even with a setting of $\alpha = \beta = 0.5$ which is when one group of agents interact with an arbitrary target agent frequently and hence are able to build accurate ML models while another group seldom interact with the target agent and as a result are not able to build accurate ML models.
- However, a sparse evidence space such as that of a simulation scenario with high α and low β limits the effectiveness of COBRA. This is because a high α and low β means that the agents seldom interact with one another. As a result agents are not able to build accurate models about any other agent since the evidence from which they derive the ML models is too sparse to yield adequate accuracy. In Chapter 5 we expand with great details on this issue, conduct more detailed evaluation, and propose a different approach to tackle it.

4.5 Summary

This chapter proposes COBRA, a context-aware trust assessment framework for large-scale online environments (e.g., MAS and IoT) without a trusted intermediary. The main issue it addresses is an accuracy-privacy dilemma. Specifically, COBRA uses model encapsulation to preserve privacy that could otherwise be exposed by second-hand evidence, and in the meantime to retain context information as well. It then uses our proposed Bernoulli neural network (BNN) to aggregate the encapsulated models and first-hand evidence to make an accurate prediction of the trustworthiness of a target agent. Our experiments and simulations demonstrate that COBRA achieves higher prediction accuracy than state-of-the-art TRM systems, and is robust to 51 percent attack in which the majority of agents

are malicious. It is also shown that the proposed BNN trains much faster than a standard fully-connected feed-forward neural network, and is less prone to overfitting.

Chapter 5

Trust and Sparse Evidence Space

5.1 Problem Statement

User-generated Big Data with the prevalence of e-commerce applications poses new trust challenges that render traditional TRM approaches ineffective, especially in mitigating and discouraging trust attacks such as whitewashing and Sybil attacks. Specifically, there are three challenges. First, user-generated data is increasingly becoming sparse, making the derived trust models not robust to attacks. Second, the cost of attacks has significantly decreased over time due to the widespread presence of bots in e-commerce applications, and thus the traditional assumption of majority users being honest often no longer holds. This also further exacerbates the previous challenge. Third, e-commerce applications involve publicity (user-generated data originating from influencers and paid users), which is not formulated in existing trust models.

The focus of Chapter 5 is sparse evidence space. As explained earlier in this Section, sparse evidence space manifests best in User-generated Big Data such as that of e-commerce. Hence, we first motivate the research problems explained above from the perspective of User-generated Big Data using several discussions as follows:

- The proliferation of web-based and mobile user-applications has led to the formation of multiple disjoint sets of TRM data. For instance, a merchant can list its products on both Amazon and Alibaba, and hence, may have some user feedbacks on Amazon and some on Alibaba; in fact, users are free to leave their feedbacks on many social media as well, such as Facebook, Twitter, YouTube, LinkedIn, and many blogging sites. In such circumstances, each individual set of data is often insufficient for reliable Big Data analytics. For example, new marketplaces do not have as many users as a

mature marketplace such as Amazon or Alibaba, and hence the TRM data that they have often does not constitute Big Data. The research problem that arise is: How to connect and synchronise multiple disjoint sources of TRM data to form Big Data that can be processed for TRM. Our proposed blockchain-enabled TRM system in Chapter 5 provides a solution to this problem and enables existing TRMs to benefit from the feedbacks about common entities by synchronising with the blockchain to form an abstract global TRM.

- The sparsity of evidence space in User-generated Big Data has adverse consequences. The most important among such consequences is that the traditional assumption that the majority of users in TRM are honest does not always hold [Sigala and Gretzel, 2017]. This makes the TRM within such applications extremely vulnerable to *white-washing* and *Sybil attacks* [Levine et al., 2006]. Chapter 5 provides a solution to this problem as explained with details in Section 5.2.2.
- An important consideration in Big Data analytics is the approach to data processing. Parallelization of the computations is the dominant approach which is commonly pursued using either a programming paradigm called map-reduce or more recently using multi-agent approaches [Erl et al., 2016]. The techniques proposed in Chapter 5, in particular Eq. 5.1-8, are summation based. This fits well the map-reduce programming paradigm, and hence a desired form of computation in Big Data analytics. We discuss this further with full details in the relevant section of the computational complexity.

Although the focus of Chapter 5 is sparse evidence space and this kind of evidence space manifests best in User-generated Big Data such as that of e-commerce. However, the techniques proposed in Chapter 5 can also be used in newly formed IoT environments. This is because when an IoT environment is created, initially the evidence space is empty and it will gradually become dense. Hence, the solutions that Chapter 5 provides also relate to this problem with respect to IoT. However, as discussed earlier, in this thesis we explore two categories of Big Data, one of which relates to IoT, and hence not every Chapter in the thesis is about IoT. This also explains the reason why IoT is not in the title of the thesis.

In this chapter, we propose a new TRM framework¹ called BEQA. It uses Blockchain

¹The work in this chapter is published as Leonit Zeynalvand, Tony T Luo, Ewa Andrejczuk, Dusit Niyato, Teo SinGee, Jie Zhang. *A Blockchain-enabled Quantitative Approach to Trust and Reputation Management with Sparse Evidence (Extended Abstract)*. In the Proceedings of AAMAS 2021.

to transform multiple disjoint and sparse sets of user-generated data into a single dense-enough dataset, and formulates the cost of Sybil attacks using Blockchain transaction fees. In particular, publicity is formulated as a *whitewashing deposit* such that a higher level of publicity will impose higher cost on the Sybil attack. To evaluate the performance of BEQA, we conduct experiments using three real datasets, and additional simulations for more extensive scenarios. Our results show that BEQA outperforms the state-of-the-art TRM models. For instance, BEQA yields more accurate trustworthiness assessment and more rapid mitigation on Sybil attacks.

5.2 Design of BEQA

BEQA uses Blockchain transaction fees to quantify the cost of whitewashing and Sybil attacks. First, BEQA proposes a heuristic method that identifies publicity of an entity, which is the set of feedbacks submitted by sources with significantly more experience than others. Second, BEQA establishes a mathematical relationship between the weight of a submitted feedback and the costs incurred by the feedback source, where the weight grows over time with an unknown growth rate (determined later). Third, BEQA quantifies the cost of a Sybil attack as a function of the growth rate. Finally, BEQA calculates the growth rate in such a way that the cost of a Sybil attack is controlled by the aggregate costs that the entity’s influencers incur. This way, BEQA leverages the costs incurred by the influencers of an entity as a whitewashing deposit for that entity, which discourages both whitewashing and Sybil attacks.

Why Blockchain? Blockchain provides a tamper resistant record of time stamped non-gratuitous participation history. To the best of our knowledge, we are the first to incorporate this property of the blockchain into a new TRM model which contributes meaningfully to the state-of-the-art in TRM. It is crucial to note that BEQA does not use the blockchain as a second-hand evidence database, instead what is a design requirement, is to guarantee that the second-hand evidence (regardless of where it is stored) is Write Once Read Many (WORM) compliant and the producer of each evidence bears the cost of WORM compliance. To meet this particular requirement, BEQA uses blockchain. This serves three purposes:

Firstly, it decouples database from tamper resistance. This is of great importance because a database is managed by a database management system (DBMS). Its key job is to index data and facilitate various data queries such as SQL/no-SQL. The blockchain in

our view however, is not a database, instead it is a tamper resistant ledger; It is a WORM compliant ledger. Hence, it is more efficient to read from this ledger and store in a local database so that the data can be then easily indexed and queried upon using the state-of-the-art database technologies while still complying with the WORM requirement of BEQA.

Second, the producer of each WORM compliant evidence bears the cost of the WORM compliance which is the blockchain transaction fee. This is the basis of BEQA's TRM model and is used to deal with white washing and Sybil trust attacks as explained with details in Section 5.2.2. As for P2P alternative designs, to the best of our knowledge there is no alternative WORM compliant P2P framework which meets BEQA's design requirement explained above. In Chapter 2 we discussed that P2P WORM compliance is in fact what the blockchain has been used for since 2009 in Bitcoin and hence battle-tested for more than 10 years without any successful tampering attempts.

Third, the transaction fees in BEQA are not paid to the BEQA framework or any of its related entities such as the buyers or the sellers. That is, if researchers choose to implement BEQA and more precisely the feedback fees outside a blockchain (e.g. using digital cash or micro-payment), they need to answer an important question. Who will collect the fees? A trusted third-party? The buyers? The sellers? These fees are the cost of what? Those who collect the fees, have done what to deserve being compensated with these fees? Will this encourage them to act maliciously? In BEQA however the fees are the cost of writing to a WORM compliant ledger. Those who write to this ledger pay the fees and those who maintain this ledger (i.e. the miners) collect these fees. BEQA uses public blockchain where the miners have strong competing incentives not to tamper with the ledger.

5.2.1 Process Overview

In this section, we provide an overview of the BEQA process. All the main notations are defined in Table 5.1.

feedback. We use the term *feedback* to refer to the information shared by users about the entities. A feedback i can contain qualitative information such as user reviews or comments, or quantitative information such as user rating. In the remainder of this chapter, we denote the qualitative information by d_i and quantitative information by v_i .

Transaction fee. Storing a single feedback i on the Blockchain incurs the source a fee equal to f_i . We refer to this as the *transaction fee*. This fee depends on the amount of stored data and is a parameter of the Blockchain not a setting of BEQA. Hence, we can use any

Table 5.1: Definitions and Notations

Notation	Definition	First Ref.
d_i	Qualitative information such as a review included in an arbitrary feedback i .	Eq. (5.8)
e	An entity whose trustworthiness is of interest.	Eq. (5.2)
f_i	The fee of storing feedback i .	Eq. (5.2)
t_i	The time-stamp of feedback i .	Eq. (5.2)
v_i	The numeric value of feedback i representing a quantitative assessment such as rating.	Eq. (5.8)
A_e	The set of all available feedbacks about an entity e whose trustworthiness is of interest.	Eq. (5.8)
B_i	The set of all the feedbacks (about any entity) submitted not later than feedback i and submitted by the same user who submitted feedback i .	Eq. (5.2)
S_i	The set of all the feedbacks (about any entity) submitted by the same user who submitted feedback i .	Eq. (5.1)

Blockchain in which the transaction fees are not zero to implement BEQA.

We illustrate an overview of the main processes in BEQA in Fig. 5.1. On one hand, as illustrated in interaction 1 of Fig. 5.1, to submit feedback $i = (u_i, v_i, d_i, e)$ about an entity e , a user with a public-key u_i first signs the feedback with its own private key. This makes BEQA robust against non-repudiation attacks as in BEQA each user has a randomly generated key pair associated with its identity. Next, the user submits the signed feedback to Blockchain using a wallet². This will incur the user a transaction fee f_i and when the transaction is confirmed on Blockchain, it is assigned a time-stamp t_i .

On the other hand, as illustrated in Fig. 5.1, interaction 2, when a user enquires about the trustworthiness of some entities $e_1, e_2, \dots, e_n$, BEQA fetches all the existing feedbacks for those entities, and the corresponding fees and timestamps from a database. This database constantly synchronizes itself with the Blockchain and crawls and indexes only the Blockchain transactions relevant to BEQA framework. An example of database technologies capable of targeted crawling and indexing of the Blockchain is proposed by [corp, 2019]. Next, BEQA verifies the signatures of the feedbacks of entities $e_1, e_2, \dots, e_n$. This prevents identity attacks in which an attacker associates its malicious feedbacks with a legitimate user's identity. To offset the computational overhead of signature verification, the SBD application caches the previously verified feedbacks. Finally, BEQA's trustworthiness

²A wallet is a software used to perform transactions on Blockchain.

Figure 5.1: Sequence diagram of the main interactions in BEQA. Interaction 1 illustrates the feedback submission process. Interaction 2 illustrates trustworthiness enquiry process.

analysis model uses Blockchain transaction fees to quantify the cost of whitewashing and Sybil attacks and provide a robust assessment of the trustworthiness of entities $e_1, e_2, \dots, e_n$. In the following, we explain the details of BEQA's trustworthiness analysis.

5.2.2 Trustworthiness Analysis

This section provides details of BEQA's trust model.

Experience of a source. The more feedbacks a source submits over time, the more experienced it becomes. Thus, we use the experience of a source to refer to the history of the source's previous assessments. Since each feedback i incurs the source a transaction fee f_i , we can measure the experience of the source of feedback i by calculating the transaction fee expenditure of the source using:

$$x_i = \sum_{j \in B_i} f_j$$

where B_i is the set of all the feedbacks (about any entity) submitted by the source of i and

Figure 5.2: Publicity filter (vertical dashed line) applied to various distributions of transaction fee expenditure: (a) Normal distribution with $\mu = 100$ and $\sigma = 10$; (b) Lognormal distribution with $\mu = 0$ and $\sigma = 1$; (c) Chi-Square distribution with 5 degrees of freedom; (d) Beta distribution with $\alpha = 4$ and $\beta = 2$; (e) Mixture of Gaussian distributions with $\mu_1 = 15, \mu_2 = 35, \mu_3 = 120$ and $\sigma_1 = \sigma_2 = \sigma_3 = 4$ and $w_1 = 0.6, w_2 = 0.2, w_3 = 0.2$; (f) Mixture of Gaussian distributions with $\mu_1 = 15, \mu_2 = 100, \mu_3 = 120$ and $\sigma_1 = \sigma_2 = \sigma_3 = 4$ and $w_1 = 0.6, w_2 = 0.2, w_3 = 0.2$; (g) Mixture of Gaussian distributions with $\mu_1 = 30, \mu_2 = 150$ and $\sigma_1 = \sigma_2 = 5$ and $w_1 = 0.66, w_2 = 0.33$; (h) Beta distribution with $\alpha = 0.1$ and $\beta = 0.1$.

submitted not later than i .

Publicity. Ideally, the more experienced a source is, the more credible its feedbacks are. However, if the source of a feedback is outstandingly more experienced than others (take social media influencers as an example), the credibility of feedback may be questionable since the influencers usually have personal interests and economical incentives when submitting data. We refer to such source as *publicity influencer*, and we refer to its assessment as *publicity*. BEQA needs to be able to distinguish the publicity of an entity from the rest of its feedbacks. Traditionally, statistical filtering methods are used to remove noise from signals [Garnett et al., 2005, Simoncelli and Adelson, 1996]. However, one major difference between publicity and noise is that publicity can form a significant portion of the feedbacks. Many new services and products try to populate their customer reviews with paid feedbacks. Another important difference is that the publicity of an entity does not follow the random nature of noise. That is, usually, publicity influencers are chosen from influential figures in social media. As a consequence, we cannot use conventional outlier detection or noise removal mechanisms for this purpose. Moreover, we refrain from using more sophisticated techniques such as deep learning due to their high computational complexity and poor performance with sparse data [Crocetta, 2019]. Therefore, we propose a heuristic statistical filtering mechanism to separate the publicity of an entity from other feedbacks. Because our proposed mechanism does not require training, it has significantly lower computational overhead and it retains high accuracy even with sparse data.

Formally, we obtain the publicity of an entity e from (5.1) where μ_x , σ_x^2 , $\mathcal{S}_x$, and $\mathcal{K}_x$ are the first four statistical moments³ of the set $\{x_i = \sum_{j \in B_i} f_j \mid i \in A_e\}$ as follows:

$$P_e = \left\{ i \in A_e \mid x_i \geq \mu_x + \sigma_x \left(1 - \frac{1}{\mathcal{K}_x} \right) \left(\frac{2}{\pi} \tan^{-1}(-|\mathcal{S}_x|) + 3 \right) \right\} \quad (5.1)$$

³Moments are statistical parameters used to measure a distribution. The first four moments of $\{x_i = \sum_{j \in B_i} f_j \mid i \in A_e\}$ are:

Mean (1st raw moment): $\mu_x = \frac{1}{|A_e|} \sum_{i \in A_e} x_i$

Variance (2nd central moment): $\sigma_x^2 = \frac{1}{|A_e|} \sum_{i \in A_e} (x_i - \mu_x)^2$

Skewness (3rd normalized moment): $\mathcal{S}_x = \frac{1}{|A_e| \sigma_x^3} \sum_{i \in A_e} (x_i - \mu_x)^3$

Kurtosis (4th normalized moment): $\mathcal{K}_x = \frac{1}{|A_e| \sigma_x^4} \sum_{i \in A_e} (x_i - \mu_x)^4$

Thanks to (5.1), BEQA selects the data points deviating significantly from the mean in a normal or skew-normal distribution, as illustrated in Fig. 5.2(a)-(d). This is a common practice in widely used outlier detection mechanisms such as *z-score*. However, if the distribution is not normal, skewness and kurtosis enable (5.1) to select the divergence ahead as a portion of the distribution which is more likely to contain publicity as depicted in Fig. 5.2(e)-(h).

Weighting feedbacks. BEQA assigns a weight of zero to feedbacks from publicity influencers. BEQA weights the remaining of the feedbacks proportionately to the costs incurred by the data source. This costs are grown over time using an exponential growth function with an unknown growth rate. This way, BEQA quantifies the weight of a feedback as a function of this growth rate.

Formally, let B_i be the set of all feedbacks submitted not later than feedback i by the same user who submitted feedback i . We define the weight of a feedback i about an entity e as w_i as follows:

$$w_i = \begin{cases} \sum_{j \in B_i} f_j (1 + r_e)^{t_i - t_j} & i \notin P_e \\ 0 & i \in P_e \end{cases} \quad (5.2)$$

where t_i and t_j are the timestamps of feedback i and feedback j respectively, P_e is the set containing the *publicity* of entity e , and r_e in (5.2) is a growth rate that BEQA calculates for each entity dynamically. Note that r_e is a critical parameter, and its calculation is explained in detail as follows.

Computing the growth rate r_e . The computation of r_e plays a decisive role in determining the trustworthiness of an entity. This is because we want BEQA to calculate r_e for each entity to disincentivize not only source-centric attacks such as *Sybil* but also less addressed entity-centric attacks such as *whitewashing* (see Section 1.2.3 for the definitions of these attacks).

For an entity e , we denote M and N as the sets of malicious and nonmalicious feedbacks respectively such that $N \cap P_e = M \cap P_e = \emptyset$. We say a *Sybil* attack is *successful*, if the total weight of malicious feedbacks is greater than or equal to that of nonmalicious feedbacks. If we weight each feedback using (5.2), then for these attacks to be successful the inequality

in (5.3) should hold.

$$\frac{\sum_{i \in N} \left(\sum_{j \in B_i} f_j (1 + r_e)^{t_i - t_j} \right)}{\sum_{i' \in M} \left(\sum_{j' \in B_{i'}} f_{j'} (1 + r_e)^{t_{i'} - t_{j'}} \right)} \leq 1 \quad (5.3)$$

Hence, the cost of an **immediate** Sybil attack (i.e. when $B_{i'} = \{i'\}$) according to (5.3) is:

$$C_e = \sum_{i' \in M} f_{i'} \geq \sum_{i \in N} \left(\sum_{j \in B_i} f_j (1 + r_e)^{t_i - t_j} \right) \quad (5.4)$$

Proposition 5.2.1 *The range of C_e in (5.4) is unbounded from above, given that $r_e \in [0, +\infty)$ (see Section 5.2.3 for proof).*

And a consequence of proposition 5.2.1 is:

Corollary 5.2.1.1 *Given an attack budget Y and the assumption of $\exists i \in N, \exists j \in B_i \mid i \neq j$, the cost of an immediate Sybil attack can be arranged to be higher than Y to disincentivize the attacker.*

Our goal is to not only disincentivize attackers from attacking the reputation of entity e , but also to disincentivize the entity e from whitewashing its own bad reputation. Thus, BEQA prevents whitewashing by leveraging the *publicity expenditure* of entity e , which is the total cost incurred by the influencers of entity e , in the calculation of r_e . Specifically, we calculate r_e such that the cost of an immediate attack on the reputation of entity e is always less than its publicity expenditure (see inequality (5.5) for details). This disincentivizes whitewashing as with each attempt, the entity loses its publicity. Without publicity and abundant feedbacks the cost of an immediate attack drops to extremely low values, which renders the trust assessment unreliable.

Formally, given a publicity set P_e , we calculate the maximum growth rate r_e such that both inequalities in (5.4) and (5.5) hold.

$$C_e < \sum_{i \in P_e} \left(\sum_{j \in B_i} f_j \right) \quad (5.5)$$

If (5.4) and (5.5) hold, then (5.6) holds.

$$\sum_{i \in N} \left(\sum_{j \in B_i} f_j (1 + r_e)^{t_i - t_j} \right) < \sum_{i \in P_e} \left(\sum_{j \in B_i} f_j \right) \quad (5.6)$$

With regards to monotonicity:

Proposition 5.2.2 *The polynomial on the left hand side of the inequality in (5.6) is monotonic and increasing on the interval $r_e \in [0, +\infty)$ (see Section 5.2.3 for proof).*

And a consequence of proposition 5.2.2 is:

Corollary 5.2.2.1 *The feasible solution region for (5.6) (if exists) consists of $r_e \in [0, R_{P_e}^{max})$ where $R_{P_e}^{max}$ is the **minimum** rate using which the inequality in (5.6) does **not** hold for an entity e .*

However, if a solution region does not exist for (5.6), we define $R_{P_e}^{max} = 0$.

Now, given a set of entities E , we adopt the notion of Kaldor-Hicks improvement from [Kaldor, 1939, Hicks, 1939] and obtain r_e for $e \in E$ from (5.7) where R_0^{max} is $\min_{e \in E}(R_{P_e}^{max})$ to ensure allocative efficiency.

$$r_e = \frac{\sum_{i \in P_e} \left(\sum_{j \in B_i} f_j \right)}{\max_{e' \in E} \left(\sum_{i \in P_{e'}} \left(\sum_{j \in B_i} f_j \right) \right)} \times R_0^{max} \quad (5.7)$$

(5.7) calculates r_e for $e \in E$ such that the entity with more publicity expenditure retains a higher growth rate. This leads to a higher cost of both an **immediate** Sybil attack (as per (5.4)) and whitewashing attack (the attacker loses its publicity with each whitewashing attempt). Recall from (5.5) that a low publicity results in a low cost of attack and trust assessments which have a low cost of attack are deemed unreliable.

Trust computation. Having obtained r_e for $e \in E$ from (5.7), we can compute the trustworthiness of the entities in a set of interest $\{e \in E \mid C_e > Threshold\}$ using (5.8) where *Threshold* is application specific. In addition to (5.8), we can use (5.2) for sorting, filtering, and/or any other projection of the qualitative information such as reviews included in the feedbacks.

$$T_e = \frac{\sum_{i \in A_e} v_i \cdot w_i}{\sum_{i \in A_e} w_i} \quad (5.8)$$

$$I_e = \frac{\sum_{i \in P_e} v_i}{|A_e|} \quad (5.9)$$

We also compute a complementary score for publicity using (5.9). This publicity score accounts for the feedbacks which are filtered by (5.2). If we assume that the majority of the influencers are trustworthy, then the publicity score is a reliable score. However, if the majority of the influencers are not trustworthy, then techniques such as sentiment analysis, natural language processing, social network analysis, etc., should be used if keen to detect the malicious influencers which is not the focus of TRM and this thesis. Hence, we suggest relying on the trust score obtained from (5.8) instead which is agnostic to the potential malevolence of the influencers.

This does not mean that we discount publicity, instead we make good use of publicity as a guarantor against white washing attacks. The explicit calculation of the weights is among the strengths of our model and is a crucial element in understanding our contribution with regards to publicity which jointly addresses whitewashing and Sybil attacks. We calculate the weights using (5.2). The other unknown in (5.2) is r_e which we calculate using (5.6-7). These equations formulate publicity as a whitewashing deposit such that a higher level of publicity will impose higher cost on Sybil attacks. This is an important contribution in Chapter 5 and highlights the fact that publicity (influencers), are indeed an important part of our model. However it is their presence (i.e. spent transaction fees) that matters to our proposed model not their opinions. Hence, while we make good use of publicity and the fees associated with it as a guarantor against white washing attacks, we refrain from mixing such content with normal feedbacks of ordinary users to avoid misleading conclusions. Instead we dedicate a separate (optional) score to publicity and present it separately as explained in the previous paragraph.

Computational complexity: In BEQA, with each new second-hand evidence, (5.1) needs to be updated. The sequential complexity of this update is $O(|A_e|)$ where $|A_e|$ is the number of second-hand evidence. However, (5.1), is a summation based formula where there is no dependency between the summands. This is referred to as a *perfectly parallel* workload [Herlihy and Shavit, 2012]. That is, (5.1) can be separated into a desired number n of separate tasks where $n \leq |A_e|$. This fits well the map-reduce programming paradigm, and hence a desired form of computation in Big Data analytics. Similarly, the sequential complexity of (5.2) is $O(|B_i|)$ where $|B_i|$ is the number of first-hand evidence. The workload of (5.2) is also a *perfectly parallel* workload. Hence the overall sequential complexity of BEQA which is that of (5.8) is $O(|A_e| \times |B_i|)$ and it is a *perfectly parallel* workload, suitable for Big Data analytics.

5.2.3 Proofs

Next, we prove propositions 5.2.1 and 5.2.2 formulated in Section 5.2.2.

Proposition 5.2.1: The range of C_e in (5.4) is unbounded from above, given that the rate $r_e \in [0, +\infty)$.

Proof 5.2.1 *Given that $\exists i \in N, \exists j \in B_i \mid i \neq j$ then*

$$\lim_{r_e \rightarrow +\infty} \sum_{i \in N} \left(\sum_{j \in B_i} f_j (1 + r_e)^{t_i - t_j} \right) = +\infty$$

therefore

$$\forall Y \geq 0, \exists r_e > 0 \implies |C_e| > Y$$

therefore C_e is unbounded from above.

Proposition 5.2.2: The polynomial on the left hand side of the inequality in (5.6) is monotonic and increasing on the interval $r_e \in [0, +\infty)$.

Proof 5.2.2 *Let $g(r_e) := \sum_{i \in N} \left(\sum_{j \in B_i} f_j (1 + r_e)^{t_i - t_j} \right)$, by definition t_i and t_j are non-negative integers such that $t_i \geq t_j$. Therefore:*

$$\frac{d}{dr_e} g(r_e) = \sum_{i \in N} \left(\sum_{j \in B_i} (t_i - t_j) f_j (1 + r_e)^{t_i - t_j - 1} \right)$$

We know that $r_e \geq 0$ by definition. Therefore:

$$\forall r_e \in [0, +\infty) \quad \frac{d}{dr_e} g(r_e) \geq 0$$

Hence, $g(r_e)$ is monotonic and increasing on the interval $[0, +\infty)$.

5.3 Evaluation

We evaluate BEQA using experiments with real datasets as well as using extensive simulations.

5.3.1 Experiment setup

Datasets. We use three public Yelp⁴ datasets. Specifically we use the following datasets:

⁴Yelp is a public company which publishes crowd-sourced reviews about different businesses.

1. **YelpChi-R** dataset obtained by [Mukherjee et al., 2013] contains 61,542 feedbacks about restaurants in the Chicago area of which 13.2% are fake feedbacks.
2. **YelpChi-H** dataset obtained by [Mukherjee et al., 2013] contains 5,855 feedbacks about hotels in the Chicago area of which 7.5% are fake feedbacks.
3. **YelpNYC-R** dataset obtained by [Rayana and Akoglu, 2015] contains 359,052 feedbacks about restaurants in the New York City area of which 10.2% are fake feedbacks.

Benchmark methods. We use the following benchmarks for comparison:

- *Trust and Reputation using Hierarchical Bayesian Modelling (HABIT)*: [Teacy et al., 2012] takes a statistical approach and uses hierarchical Bayesian modelling to infer the probability of trustworthiness from first-hand and second-hand (i.e. shared by others) observations. HABIT outperforms existing statistical and probabilistic trust models.
- *Context-aware Bernoulli Neural Networks for Reputation Assessment (COBRA)*: [Zeynalvand et al., 2020] proposes a novel neural network to predict the trustworthiness of an entity given a context. COBRA uses the reputation models shared by other users and is resilient to 51 percent attacks in which the attackers form the majority of the system.
- *Beta Reputation System (BRS)*: [Josang and Ismail, 2002] aggregates the feedbacks from various users into one trust score using the beta probability density function.

Platform. We perform the experiments using a Linux workstation with 8 CPU cores and 12GB of RAM. We use the functional API of Keras [Chollet, 2015] and TensorFlow backend [Abadi et al., 2015] for the implementation of the neural network architectures. while `scikit-learn` is used for the implementation of Gaussian process, decision tree, and Gaussian Naive Bayes models.

Evaluation metrics. We employ three commonly used metrics. That is, root mean squared error (RMSE), mean absolute error (MAE), and mean squared logarithmic error (MSLE) respectively defined as

$$RMSE(T, \hat{T}) = \sqrt{\frac{1}{m} \sum_{i=1}^m (T_i - \hat{T}_i)^2}$$

$$MAE(T, \hat{T}) = \frac{1}{m} \sum_{i=1}^m |T_i - \hat{T}_i|$$

Figure 5.3: Performance of BEQA compared to benchmark methods on different datasets using: (a) RMSE evaluation metric; (b) MAE evaluation metric; (c) MSLE evaluation metric.

$$MSLE(T, \hat{T}) = \frac{1}{m} \sum_{i=1}^m (\ln(1 + T_i) - \ln(1 + \hat{T}_i))^2$$

where T is the ground-truth trustworthiness, $\hat{T}$ is the predicted trustworthiness score of BEQA using (5.8), and m is the total number of predictions.

5.3.2 Experiment Procedure and Results

In all the datasets introduced in Section 5.3.1, each review is accompanied by a score (i.e. rating). To ensure consistency in our evaluations, we scale all the scores to the interval $[0, 1]$. In each of the datasets, we refer to the reviews which are not marked as fake by Yelp as *genuine reviews* and the corresponding trustworthiness scores as *genuine scores*. We use BEQA as well as the benchmark methods introduced in Section 5.3.1 to predict the genuine scores submitted by each user. Then we compare the predicted scores with the ground truth scores and measure the RMSE, MAE, and MSLE (see Section 5.3.1) of the predictions across all datasets.

We show the results in Fig. 5.3(a)-(c). We see that BEQA outperforms the benchmark methods in all three datasets. Specifically, BEQA achieves on average 50% lower RMSE than other methods (see Fig. 5.3(a)), 58% lower MAE (see Fig. 5.3(b)), and 69% lower MSLE (see Fig. 5.3(c)). The advantage of BEQA over BRS comes from the fact that BRS is a simple model that does not consider the credibility of the reviews. Thus, it weights all reviews equally. COBRA and HABIT consider the credibility of the reviews but their trust measure becomes inaccurate when the data set is sparse. The poor performance of COBRA and HABIT verifies our claim that the sparsity of SBD negatively affects the derived trustworthiness models. Mainly, probabilistic models which rely on prior distributions to derive an accurate posterior are very vulnerable to a sparse user-entity matrix.

Moreover, compared to MAE, RMSE is a more punitive measure with respect to larger errors. That is, RMSE assigns a relatively higher weight to large errors compared to MAE. Hence, comparing Fig. 5.3(a) with Fig. 5.3(b) reveals that the prediction errors of BEQA are relatively smaller than that of the benchmark methods. As a consequence, we observe that RMSE of BEQA is 50% lower than that of other methods and MAE of BEQA is 58% lower than that of other methods.

Furthermore, RMSE and MAE cannot distinguish between an overestimation versus an underestimation. In contrast, MSLE has an asymmetric error curve subject to the ratio of predicted score to the ground truth. Therefore, Fig. 5.3(c) reveals that the datasets are biased towards high scores which often leads to overestimation in benchmark methods. However, BEQA proves to be more resilient to bias compared to the benchmark methods.

5.3.3 Simulation procedure and results

To further evaluate BEQA and demonstrate its step-by-step computational aspects, we also conduct extensive simulations. We simulate a marketplace since profit seeking adversaries often choose their victims from e-commerce. There are three fundamental facts that we consider in this simulation. First, an important indicator of marketplace stability is *buyer-to-seller* ratio. That is, the ratio of the total number of buyers in the marketplace to the total number of sellers in the marketplace. As a rule of thumb, a ratio of 3:1 is a commonly accepted indicator of a stable marketplace [rul, 2001, www.meilleursagents.com, 2014]. Hence in this simulation we also adopt this rule-of-thumb with respect to the ratio of buyers to sellers.

Second, research suggests that significant portion of the online reviews are fake. In

many cases, fake reviews account for almost half of the total reviews [Sigala and Gretzel, 2017]. In extreme cases, more than 90 percent of the reviews are fake [www.fakespot.com, 2019]. In this simulation we adopt the worst case scenario mainly because this simulation is not our only evaluation of BEQA and we also evaluate the performance of BEQA compared to the state-of-the-art TRM models using three public data-sets in Section 5.3.1 and Section 5.3.2. These data-sets do not reflect a worst case scenario with respect to the ratio of malicious users to legitimate ones, and hence for the sake of the completeness of evaluation it is also beneficial to have an evaluation under a worst case scenario. Therefore, in this simulation we adopt a scenario where the ratio of malicious buyers to legitimate ones is 9:1. Note that the buyers are the users who submit feedback about the sellers. Also, the goal of the TRM system is to assess the trustworthiness of the sellers based on the feedbacks submitted by the buyers.

Third, the task of trust assessment is harder when the ratio of trustworthy sellers to untrustworthy ones is closer to 1:1. This is because, in a prediction problem, the distribution of the label space affects the accuracy of the predictions. For example, to predict whether it will snow today in Singapore a trivial prediction model can just output *No* by default all the time and have an accuracy close to 100%. However, same strategy yields an accuracy of about 50% if the prediction is with respect to rain, given that the definition of rain in Singapore is a rainfall of 0.2mm or more for that day [weather.gov.sg, 2021]. Hence, in this simulation the ratio of trustworthy sellers to untrustworthy ones is equal to 1:1 to ensure that the distribution of the labels in the label space is not biased towards any specific label.

For the purpose of further justification let us denote the number of the trustworthy sellers in this simulation as x , and the number of untrustworthy sellers as y . From what we explained above we have $x = y$. Moreover, we denote the number of malicious buyers as z , and the number of legitimate buyers as w . From what we explained above, we have $z = 9 \times w$. Furthermore, given the 3:1 ratio of buyers to sellers we have $w = 3 \times (x + y)$. Hence, we denote the total number of active entities in this simulation by n and calculate its value as follows:

$$n = x + y + w + z$$

since $z = 9 \times w$, and $w = 3 \times (x + y)$, therefore:

$$n = x + y + 3 \times (x + y) + 9 \times (3 \times (x + y))$$

and because $x=y$, so:

$$n = 2x + 6x + 54x = 62x$$

On a separate note, we need to repeat the simulation until the results converge. For this simulation, the convergence to a confidence level of 99% is achieved after 10,000 repetitions. That is, our simulation emulates $10,000 \times n$ entities in total. Given that as explained in the setup, we perform the experiments and the simulations using a Linux workstation with 8 CPU cores and 12GB of RAM, we can emulate around 3,000,000 entities in a reasonable time. Hence we have:

$$10,000 \times n < 3,000,000$$

thus for this simulation to finish in a reasonable time on our platform we have:

$$62x < 300 \implies x < 4.84$$

We round up and adopt $x = 5$ in this simulation. Consequently, we have $y = 5$, $w = 30$, and $z = 300$. Hence, we use the following marketplace setup in our simulations:

- 30 honest buyers,
- 300 Sybil attackers,
- 5 trustworthy entities (trustworthy sellers),
- 5 untrustworthy entities (untrustworthy sellers).

We adopt the distribution of *advertising expenditure* from [Singh, 2015]. That is, entities from different industries have different advertisement expenditure in contrast to entities within same industry which have similar expenditure. For example, travel agencies spend more on advertisement and publicity than books stores. Hence, in our simulations the maximum publicity expenditure of each entity regardless of whether it is trustworthy or not follows a mixture of uniform distributions with different means of 100, 200, and 400 and equal mixture weights. This results in having three clusters of entities with an average whitewashing deposit of 100, 200, and 400 respectively because as mentioned in Section 5.2.2, the publicity expenditure of each entity is its whitewashing deposit in BEQA.

At each time-step of the simulation, each entity interacts with a randomly chosen honest buyer. Then, this user submits their assessment of the entity's trustworthiness as a binary value. That is, the feedback submitted by a user u about an entity e , indicates whether u

Figure 5.4: Simulation results: (a) Cost of a successful Sybil attack for entities with different publicity deposits at each time-step of the simulation; (b) r_e for entities with different publicity deposits at each time-step of the simulation; (c) RMSE for entities with different publicity deposits at each time-step of the simulation.

found e trustworthy in the recent interaction or not. Submission of each feedback incurs a unit cost. Next, we present three experiments we conducted to show the effectiveness of our method.

Experiment 1

In the first experiment, we want to use BEQA to measure the costs of Sybil and whitewashing attacks. At each time-step of the simulation t , we measure the minimum immediate cost of a successful Sybil attack (using Eq. (5.4)) for different clusters of entities. Note that entities from different clusters have different publicity deposits which is a hedge against white washing. That is, if the entity attempts to white wash its bad reputation by acquiring a new identity, it loses the publicity deposit associated with its previous identity. Thus, the bad reputation whitewashing cost is tantamount to the aggregated fees of the publicity

Figure 5.5: RMSE of BEQA compared to benchmark methods for different distributions of Φ : (a) Φ follows log-normal distribution with a mean of $\mu = 0$ and standard deviation of $\sigma = 1$; (b) Φ follows log-normal distribution with a mean of $\mu = 0$ and standard deviation of $\sigma = 1.5$; (c) Φ follows log-normal distribution with a mean of $\mu = 0$ and standard deviation of $\sigma = 2$.

to which we refer as publicity deposit and hence is different between entities from distinct clusters. We present the experiment results in Fig. 5.4(a). The results show that in general (most of the time) a higher average whitewashing (publicity) deposit leads to a higher Sybil attack cost, which is a desired property. That is, the more an entity invests in its publicity to ensure that it will not perform whitewashing in future, the costlier it gets for an attacker to attack the trustworthiness of such entity. Hence, BEQA quantifies the cost of whitewashing and Sybil attacks and promotes publicity by incorporating the whitewashing cost (which depends on publicity) into the data aggregation mechanism that determines the Sybil cost.

Experiment 2

In the second experiment, we want to measure how long it takes for clusters of entities to mitigate an immediate attack. The results obtained in Experiment 1 (Section 5.3.3) indicate that the cost of Sybil attack in the simulation scenario does not exceed 300 units during the time-steps of our simulation. Thus, in the second experiment, we simulate a Sybil attack with that budget, i.e. 300 cost units which we distribute uniformly among the attackers. Attackers carry out the Sybil attack at $t = 0$. First, we perform this experiment by calculating the compounding rate r_e for each entity (using Eq. (5.7)) at each time step of simulation. Fig. 5.4(b) indicates that r_e converges to zero after 25 simulation time-steps. This validates that our simulation duration is enough to evaluate the dynamics of BEQA.

Second, at the beginning of the simulation, the set of feedbacks is sparser and inadequate. Thus, all of the curves in Fig. 5.4(b) have larger gradients (as they have steeper slopes). This accelerates the mitigation of the Sybil attacks.

Third, we use r_e to calculate the trustworthiness of all entities (using Eq. (5.8)) at each time step of simulation t . We calculate the root mean squared error (RMSE) of the obtained trustworthiness values for each entity cluster. We present the results in Fig. 5.4(c). Similarly to the observation of r_e , RMSE decreases with a faster rate in the beginning which confirms that the attack mitigation is faster in the beginning of the simulation.

Another observation from Fig. 5.4(b) and Fig. 5.4(c) is that the curves that represent entities with larger whitewashing deposits have steeper slopes. That is, entities with larger average whitewashing deposits mitigate the Sybil attacks faster due to the rapid adjustment of the compounding rate of r_e . This result further supports our aim to encourage entities to invest in publicity as a good faith deposit against whitewashing.

Experiment 3

In this experiment, we compare the performance of BEQA with that of the benchmark methods. We adopt the attack budget from Experiment 2 (Section 5.3.3). We define the share of each attacker from the budget as $\lfloor \Phi \rfloor$ where Φ is a log-normal random variable. This is because Blockchain transaction fees are calculated per byte of data and the length of the review in an arbitrary feedback follows a log-normal distribution [Sobkiewicz et al., 2013]. In this experiment, we consider three different parameter settings for Φ . Specifically we consider the following distributions:

1. Φ follows log-normal distribution with a mean of $\mu = 0$ and standard deviation of $\sigma = 1$ in Fig. 5.5(a).
2. Φ follows log-normal distribution with a mean of $\mu = 0$ and standard deviation of $\sigma = 1.5$ in Fig. 5.5(b).
3. Φ follows log-normal distribution with a mean of $\mu = 0$ and standard deviation of $\sigma = 2$ in Fig. 5.5(c).

We simulate an immediate Sybil attack using each of the above attack budget distributions. We measure how different trust models mitigate the attack. That is, we simulate three scenarios in which we calculate the trustworthiness of the entities using BEQA at each time step of simulation as well as using benchmark methods (introduced in subsection 5.3.1). Finally, we calculate RMSE of the obtained trustworthiness values for all entities.

As we show in Fig. 5.5(a)-(c), BEQA outperforms the benchmark methods shortly after the commencement of the simulation. We also observe that more uneven distributions of the attack budget among the attackers (i.e. Φ with larger σ) trigger a swifter mitigation in BEQA while it has less significant effect on other methods. This can be attributed to the publicity quantification mechanism in BEQA. That is, publicity quantification mechanism in BEQA filters the malicious feedbacks from the attackers with a significantly large share of the total attack budget. Hence, such malicious feedbacks do not only retain a zero weight in BEQA, but also increase the cost of the Sybil attack. This is because, as shown in 5.3.3, an increase in publicity increases the cost of Sybil attack.

5.4 Summary

This chapter proposes BEQA, a Blockchain enabled trust framework. The main issues it addresses are sparsity of evidence, publicity, and trust attacks (in particular, whitewashing and Sybil attacks). Specifically, BEQA uses the Blockchain technology to integrate multiple sparse sets of evidence into a single dense set. It then identifies publicity and quantifies the cost of Sybil attacks using Blockchain transaction fees. BEQA promotes publicity as a whitewashing deposit by ensuring that increasing publicity increases the cost of a Sybil attack. Our experiments with real datasets demonstrate that BEQA outperforms state-of-the-art trust models. Moreover, our extensive simulations validate the computational aspects of BEQA and demonstrate that BEQA yields more rapid mitigation on Sybil attacks in comparison with the state-of-the-art trust models.

Chapter 6

Conclusion and Future Work

6.1 Summary

This dissertation addresses issues within TRM systems in the emerging Big Data environments such as IoT and e-commerce. Particularly, issues concerning the subjectivity of the evidence space in such environments, the density of IoT-generated data, and the sparsity of user-generated data are identified and addressed.

Hence, in this dissertation we first propose a multi-agent subjectivity alignment (MASA) mechanism, which models the subjectivity using a regression technique and exchanges the models among agents as the input to an alignment process. This mechanism substantially counteracts biases incurred by different agents and improves the accuracy of second-hand information fusion as demonstrated by our simulations. In addition, we also conduct experiments using a real-world dataset which further validates the efficacy of MASA.

Second, we propose COBRA, a context-aware trust assessment framework for IoT-generated data. The main issue it addresses is an accuracy-privacy dilemma which is the consequence of the dense evidence space in IoT. Specifically, COBRA uses model encapsulation to preserve privacy that could otherwise be exposed by second-hand evidence, and in the meantime to retain context information as well. It then uses our proposed Bernoulli neural network (BNN) to aggregate the encapsulated models and first-hand evidence to make an accurate prediction of the trustworthiness of a target agent. Our experiments and simulations demonstrate that COBRA achieves higher prediction accuracy than state-of-the-art TRM systems, and is robust to 51 percent attack in which the majority of agents are malicious. It is also shown that the proposed BNN trains much faster than a standard fully-connected feed-forward neural network, and is less prone to overfitting.

Third, we propose BEQA, a Blockchain enabled trust framework. The main issues it address are sparsity of evidence space, publicity, and trust attacks (in particular, whitewashing and Sybil attacks). Specifically, BEQA uses the Blockchain technology to integrate multiple sparse sets of evidence into a single dense dataset. It then identifies publicity and quantifies the cost of Sybil attacks using Blockchain transaction fees. BEQA promotes publicity as a whitewashing deposit by ensuring that increasing publicity increases the cost of a Sybil attack. Our experiments with real datasets demonstrate that BEQA outperforms state-of-the-art trust models. Moreover, our extensive simulations validate the computational aspects of BEQA and demonstrate that BEQA yields more rapid mitigation on Sybil attacks in comparison with the state-of-the-art trust models.

6.2 Real-world Datasets

With respect to evaluation methodology, we want to highlight that every chapter of this thesis is extensively evaluated using both **real-world public datasets** and simulations. The full details of all experiments and simulations are provided in the respective sections in each chapter. To highlight the extent of our evaluations particularly with regards to real-world datasets, we provide a conclusive summary of the experiments in this thesis with real-world datasets as follows.

Chapter 3: Specifically, in Section 3.4.3 we use a subset of the public MovieLens dataset [Harper and Konstan, 2015] which contains 25,790 user ratings and we augment its feature space with additional real-world data which we crawl from IMDB website. Our augmentation extends the feature space of the resulting real-world dataset to also contain:

- *Language*. That is the language of the movie.
- *Duration*. That is the duration of the movie expressed in hours.
- *MPAA content rate*. That is an alphanumeric symbol assigned to a movie by the the Motion Picture Association of America (MPAA) indication the fitness of the movie contents for different age groups. Symbols such as R, PG13, and G are examples of such content rate.
- *Primary genre*. That is the main genre of the movie.
- *Secondary genre*. That is the sub-genre of the movie (if any).

- *Third genre*. That is the auxiliary genre of the movie (if any).
- *Release year*. That indicates the year when the movie was released.

We explain the experiment setup with full details in Section 3.4.3. Then, we explain the experiment procedure and analyse the results in the remaining of Section 3.4.3.

Chapter 4: Particularly, we explain that in Section 4.4.1, we use a public dataset of IoT-generated Big Data obtained from [Zheng et al., 2014] which contains the response-time values of 4,532 web services provider agents invoked by 142 service agents over 64 time slices. The dataset contains 30,287,611 records of data in total and retains a high degree of evidence density equal to 74.5%. We explain the experiment setup with full details in Section 4.4.1. We explain the experiment procedure and analyse the results in Section 4.4.2. We depict the results in 6 figures (Figure 4.2).

Chapter 5: Specifically with respect to the evaluation using real-world datasets, in Section 5.3.1, we explain that we use **three** public Yelp¹ real-world datasets. Specifically we use the following datasets:

1. **YelpChi-R** dataset obtained by [Mukherjee et al., 2013] contains 61,542 feedbacks about restaurants in the Chicago area of which 13.2% are fake feedbacks.
2. **YelpChi-H** dataset obtained by [Mukherjee et al., 2013] contains 5,855 feedbacks about hotels in the Chicago area of which 7.5% are fake feedbacks.
3. **YelpNYC-R** dataset obtained by [Rayana and Akoglu, 2015] contains 359,052 feedbacks about restaurants in the New York City area of which 10.2% are fake feedbacks.

We explain the experiment setup with full details in Section 5.3.1. We explain the experiment procedure and analyse the results in Section 5.3.2.

6.3 Suggestions for Future Research

The problems which we tackle in this dissertation have potential for future work. In this section we identify four future research directions as follows.

The issue of discrimination in TRM systems

This is one of the hardest problems to address in trust research [Jøsang et al., 2007]. The reason is that the boundaries between being discriminated against and having a subjective

¹Yelp is a public company which publishes crowd-sourced reviews about different businesses.

opinion is often not clear to a third party. For instance, it is not trivial for a third party to decide solely based on some minority negative second-hand evidence whether the seller is discriminating against some users or those users simply provide subjective evidence.

In Chapter 3 we tackle the problem of subjectivity in trust, and propose a mechanism to align the second-hand evidence with the subjective perspective of each user. We believe our proposed approach (MASA) has the potential to be further extended to address the issue of discrimination in TRM systems. This is because MASA enables a third party to remove the subjective bias from a set of second-hand evidence. As a result, discrimination is no longer indistinguishable from subjectivity.

Entropy-based weight initialization and non-Boolean outcomes to augment COBRA

Gradient based learning is commonly used to compute the weights of a neural network and the process is more generally referred to as *neural network training*. However, the weights need to be initialized since gradient based learning, being a non-convex optimization problem, is sensitive to the initial values of the weights. Numerous existing literature [LeCun et al., 2015, Glorot and Bengio, 2010, Dolezel et al., 2016, de Sousa, 2016] point out the importance of weight initialization and its impact on the performance of the neural network. Some research [LeCun et al., 2015] even suggest that the revival of feed-forward deep networks is partly because of novel application-specific schemes for weight initialization. The extent to which the performance of a feed-forward neural network can be affected by the weight initialization justifies the use of more noble initialization schemes instead of simple uniform random initialization which is the current method in the vast majority of applications.

In Chapter 4 we propose Bernoulli neural networks (BNN) for reputation assessment and we show that BNN trains faster than a fully connected (DENSE) alternative. This is because we make use of information gain and entropy to determine node interconnections. We believe there is also potential to exploit the entropy of the first-hand evidence for weight initialization. That is, we suggest to explore the potential of Eq. (4.2) in Chapter 4 to replace the uniform distribution $U[-\frac{1}{\sqrt{n}}, \frac{1}{\sqrt{n}}]$ which heuristically is being used to initialize the weights provided that n is the layer size. However, this requires careful consideration of the possibility and thorough formal verification.

On a separate vein, in addition to improving the performance of COBRA with entropy-based weight initialization, the application of COBRA can also be extended. In Chapter 4

we consider Boolean events however we believe that we can extend COBRA’s application to more domains by allowing for non-Boolean events and outcomes. This deserves thorough study and research attention.

Federated learning in TRM

Privacy is increasingly becoming an important issue in TRM systems as well as other data-rich environments. Regulations regarding the protection of data and privacy are rapidly being adopted by the international community. As a result there is a growing interest in the use of federated learning methods. Federated learning refers to the set of methods which enable the use of machine learning and data mining to extract insights and meaningful patterns from multiple data sources without requiring each source to share the data with others or with any third party.

In Chapter 4 we explained that in our proposed approach (COBRA), each agent first builds individual ML models of the outcome of its previous interactions with other agents. Second, the agents share these models with each other instead of sharing the raw evidence which may contain private information. Third, each agent uses our proposed Bernoulli neural network to predict the target probability of trustworthiness for a given context and agent of interest. We believe that COBRA can be further extended to allow for federated learning in TRM. The reason is that in COBRA, model parameters are exchanged instead of training data. Although at its current form COBRA is a multi-agent trust and reputation prediction approach. However, thanks to its design and to the fact that only model parameters are exchanged instead of the training data, COBRA has potential to allow for federated trust and reputation management.

UX and pilot study for TRM systems

UX (user experience) research is crucial for extending a design process with realistic and practical enhancements. It helps to assess the level of user acceptance or reluctance towards novel mechanisms and paradigm shifts. For example, in Chapter 5 we propose such novel approach called BEQA. It uses Blockchain to formulate the cost of Sybil attacks using Blockchain transaction fees. In particular, publicity is formulated as a *whitewashing deposit* such that a higher level of publicity will impose higher cost on the Sybil attack. Although our experiments demonstrate that BEQA outperforms state-of-the-art trust models and yields more rapid mitigation on Sybil attacks in comparison with the state-of-the-art

trust models, these achievements come with a UX cost.

Hence, we believe common UX research methods can be very insightful for the commercialization of our proposed approaches in this thesis. Although UX is not the focus of this thesis and we believe that researchers from other disciplines are more fit to conduct UX studies, but to highlight the potential of such studies and the insights that they can provide, we conduct a UX study as follows.

First, we prepare a questionnaire [Zeynalvand, 2020] to identify the major concerns that TRM users had with respect to their previous experience with various TRM systems. Then we augment the questionnaire to perform a feasibility analysis on the UX cost, particularly with regards to dealing with Sybil attacks. Next, we extend the questionnaire to capture insights about the density of User-generated evidence compared to IoT-generated evidence. The wording of the questionnaire is leveled down to fit that of the average user. The questions are as follows:

- What is your major concern with respect to Video on Demand (VoD) reputation platforms such as IMDB, MovieLense, Netflix, etc?
 - A) I submit so many feedbacks everyday to an extent that I am afraid others may figure out my location or other sensitive contextual information by analysing the timestamps of my submitted feedbacks.
 - B) The ratings are often biased and the feedbacks are subjective. So the scores are often not so helpful.
 - C) I feel that most ratings and feedbacks are fake and are submitted by fake user accounts to boost the score of certain items.
 - D) Some items with many bad ratings often disappear and then reappear as new listing with no rating or a few good ratings.
- What is your major concern with respect to e-commerce reputation platforms such as those in Amazon, booking.com, TaoBao, etc?
 - A) I submit so many feedbacks everyday to an extent that I am afraid others may figure out my location or other sensitive contextual information by analysing the timestamps of my submitted feedbacks.
 - B) The ratings are often biased and the reviews are subjective. So the scores are often not so helpful.

- C) I often spot ratings and reviews that seem fake and are probably submitted by fake user accounts to boost the rating of certain item or merchant.
- D) Some items with many bad ratings often disappear and then reappear as new listing with no rating or a few good ratings.
- What is your major concern with respect to Internet of Things (IoT) devices such as smart watch, fitbit, TraceTogether, etc., that exchange information on your behalf in the background?

A) I suspect that such devices very frequently share information in the background and I am afraid that I wouldn't know what information is being shared about me or that others might extract sensitive contextual information about me such as my location by analysing the timestamps of the information that my device frequently shared on my behalf.

B) IoT devices are of different qualities. The accuracy and the precision of the information is subject to the device quality and capabilities. I wouldn't want my device blindly rely on feedbacks obtained from low quality devices which might not be precise or suitable for my device.

C) I feel that many of such information and feedbacks are fake and are submitted by fake IoT devices to boost the rating of certain devices.

D) Some devices with bad reputation often disappear and then reappear as new device with no reputation or good reputation.
 - How much is each second of your time worth in USD? For example if you enter 0.001 it means that your time is worth 3.6 USD per hour (We will not ask you to fill in any information about your identity, however for true anonymity you can consider using the anonymity tool of your choice such as a vpn to fill in the survey).
 - One of the reasons why there can be many fake feedbacks on YouTube is because it is free to report a video on YouTube. If there was a mechanism to eliminate all fake reports but that mechanism would cost you every time you report something, how much (max) would you be willing to pay for each report that you submit? For example if you enter 0.001 it means that you are willing to pay maximum 0.001 USD each time you hit report button on a YouTube video.
 - On average how many seconds it takes you to skip a YouTube advertisement?

Figure 6.1: UX Survey Results: (a) The polar bar plot of users' major concerns w.r.t different Big Data environments; (b) The box plot of the statistical distribution of how much each user's time is worth and how much they would pay to record a feedback, if they indicated Sybil as one of their concerns; (c) The violin plot of the density of the Iot-generated evidence and that of the user-generated evidence (y-axis is in logarithmic scale).

- On average, as a user how many online interactions you have a day? Ordering food with an app/website, booking a hotel, restaurant, ride, etc., or even posting something online are all examples of user interactions.
- How many exchanges does your trace-together token/app do per day on average? This information is written in the homepage of your trace-together app, stated as “x total exchanges today”. (best to check at the end of the day) Kindly note that this is not the number of nearby devices and is written in the next line.
- Would you consider it rational to pay some amount (not zero) to be able to skip an advertisement immediately?

Overall, 28 users participated in this user study. We illustrate the results in Fig. 6.1(a)-(c). With respect to IoT-generated evidence the main concern from the perspective of the

users is privacy. This can have two reasons: 1) In IoT, data is frequently shared which makes it a better candidate for machine learning and data mining. 2) Users are often more sensitive when a device is sharing data with their permission compared to when the users themselves are directly sharing the data.

With respect to Sybil and white washing attacks, Fig. 6.1(a) shows two key outcomes. Firstly, Sybil attacks are a bigger concern for the users compared to whitewashing. This can be attributed to the fact that white washing attacks are naturally very stealthy and are not supposed to get noticed by the users. Second, user-generated evidence from e-commerce applications are where these type of attacks manifest best. This validates our choice of public dataset in Chapter 5 of this thesis.

With respect to subjectivity, Fig. 6.1(a) shows that it is a concern in all types of Big Data, however it manifests best in user-generated evidence from Video-on-Demand (VoD) applications. This validates our choice of public dataset in Chapter 3 of this thesis.

In a separate vein, Fig. 6.1(b) provides two important insights. First, the 25 – *th* percentile of the chart indicating how much the users would consider paying for each feedback is much higher than existing blockchain transaction fees [Blockchain, 2020]. This shows that the mechanism presented in Chapter 5 is feasible. Second, the charts in Fig. 6.1(b) also indicate that the users value 1 second of their time significantly more than what they would consider paying for each feedback. That is, the UX cost associated with our proposed mechanism in Chapter 5 is significantly less than the captcha-based Sybil defence mechanisms which require the user to spend significant (more than 1 second) amount of time to solve a captcha to prove that they are a real user.

Moreover, Fig. 6.1(b) provides a good comparison on the density of user-generated evidence and that of IoT-generated evidence. It indicates that IoT-generated evidence from a single IoT application is orders of magnitude more dense than that of user-generated evidence. This also coincides with the assumptions that we made about these two types of evidence in Chapter 1.

Appendix A

List of Publications

- Leonit Zeynalvand, Jie Zhang, Tony T Luo, Shuo Chen. *MASA: Multi-agent subjectivity alignment for trustworthy Internet of Things*. In the Proceedings of 21st International Conference on Information Fusion (FUSION), pp. 2013-2020. IEEE, 2018.
- Tie Luo, Leonit Zeynalvand. *Reshaping mobile crowd sensing using cross validation to improve data credibility*. In the Proceedings of 2017 IEEE Global Communications Conference (GLOBECOM 2017), pp. 1-7. IEEE 2017.
- Leonit Zeynalvand, Tony T Luo, Jie Zhang. *COBRA: Context-Aware Bernoulli Neural Networks for Reputation Assessment*. In the Proceedings of AAAI, pp. 7317-7324. AAAI, 2020.
- Leonit Zeynalvand, Tony T Luo, Ewa Andrejczuk, Dusit Niyato, Teo SinGee, Jie Zhang. *A Blockchain-enabled Quantitative Approach to Trust and Reputation Management with Sparse Evidence (Extended Abstract)*. In the Proceedings of AAMAS 2021.

Bibliography

- [rul, 2001] (2001). *The Moscow Times Business Review*. Number v. 9, nos. 3-10. ZAO Independent Press.
- [Abadi et al., 2015] Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., and Zheng, X. (2015). TensorFlow: Large-scale machine learning on heterogeneous systems. Software available from tensorflow.org.
- [Abdul-Rahman and Hailes, 1998] Abdul-Rahman, A. and Hailes, S. (1998). A distributed trust model. In *Proceedings of the 1997 workshop on New security paradigms*, pages 48–60.
- [Abdul-Rahman and Hailes, 2000] Abdul-Rahman, A. and Hailes, S. (2000). Supporting trust in virtual communities. In *Proceedings of the 33rd annual Hawaii international conference on system sciences*, pages 9–pp. IEEE.
- [Akinci et al., 2007] Akinci, S., Kaynak, E., Atilgan, E., and Aksoy, Ş. (2007). Where does the logistic regression analysis stand in marketing literature? *European Journal of Marketing*.
- [Al-Hamadi et al., 2019] Al-Hamadi, H., Chen, R., and Cho, J.-H. (2019). Trust management of smart service communities. *IEEE Access*, 7:26362–26378.
- [Alieyan et al., 2020] Alieyan, K., Almomani, A., Abdullah, R., Almutairi, B., and Alauthman, M. (2020). Botnet and internet of things (iots): A definition, taxonomy, challenges,

- and future directions. In *Security, Privacy, and Forensics Issues in Big Data*, pages 304–316. IGI Global.
- [Allcott and Gentzkow, 2017] Allcott, H. and Gentzkow, M. (2017). Social media and fake news in the 2016 election. *Journal of economic perspectives*, 31(2):211–36.
- [AmazonGo, 2016] AmazonGo (2016). The supermarket with no checkouts. <https://www.amazon.com/b?ie=UTF8&node=16008589011/>. (Accessed on 20 Dec 2017).
- [Arun and Jabasheela, 2014] Arun, K. and Jabasheela, D. L. (2014). Big data: review, classification and analysis survey. *International Journal of Innovative Research in Information Security (IJIRIS)*, 1(3):17–23.
- [Atzori et al., 2011] Atzori, L., Iera, A., and Morabito, G. (2011). Siot: Giving a social structure to the internet of things. *IEEE communications letters*, 15(11):1193–1195.
- [Audrezet et al., 2018] Audrezet, A., De Kerviler, G., and Moulard, J. G. (2018). Authenticity under threat: When social media influencers need to go beyond self-presentation. *Journal of Business Research*.
- [Bartosik-Purgat et al., 2016] Bartosik-Purgat, M. et al. (2016). Social media as a source of information about products and services in the light of cross-cultural research in china, poland and united states. *Prace Naukowe Uniwersytetu Ekonomicznego we Wrocławiu*, (447):22–31.
- [Belghache et al., 2016] Belghache, E., Georgé, J.-P., and Gleizes, M.-P. (2016). Towards an adaptive multi-agent system for dynamic big data analytics. In *2016 Intl IEEE Conferences on Ubiquitous Intelligence & Computing, Advanced and Trusted Computing, Scalable Computing and Communications, Cloud and Big Data Computing, Internet of People, and Smart World Congress (UIC/ATC/ScalCom/CBDCCom/IoP/SmartWorld)*, pages 753–758. IEEE.
- [Biswas and Muth, 2016] Biswas, K. and Muth, V. (2016). Securing smart cities using blockchain technology. In *Proceedings of IEEE 18th international conference on high performance computing and communications; (HPCC-16)*, pages 1392–1393.

- [Blockchair, 2020] Blockchair (2020). Average blockchain transaction fee. <https://blockchair.com/bitcoin-sv/charts/average-transaction-fee-usd>. (Accessed on 20 Jan 2021).
- [Bondi, 2000] Bondi, A. B. (2000). Characteristics of scalability and their impact on performance. In *Proceedings of the 2nd international workshop on Software and performance*, pages 195–203.
- [Boukerche and Ren, 2008] Boukerche, A. and Ren, Y. (2008). A trust-based security system for ubiquitous and pervasive computing environments. *Computer communications*, 31(18):4343–4351.
- [Braga et al., 2018] Braga, D. D. S., Niemann, M., Hellingrath, B., and Neto, F. B. D. L. (2018). Survey on computational trust and reputation models. *ACM Computing Surveys (CSUR)*, 51(5):1–40.
- [Bu et al., 2010] Bu, J., Tan, S., Chen, C., Wang, C., Wu, H., Zhang, L., and He, X. (2010). Music recommendation by unified hypergraph: combining social media information and music content. In *Proceedings of the 18th ACM international conference on Multimedia*, pages 391–400.
- [Buist, 2006] Buist, K. (2006). Definition of trust. <http://www.teamtechnology.co.uk/trustworthiness/definition> [Accessed: May 2019].
- [Chang et al., 2006] Chang, E., Hussain, F. K., and Dillon, T. (2006). Reputation ontology for reputation systems. In *OTM Confederated International Conferences” On the Move to Meaningful Internet Systems”*, pages 1724–1733. Springer.
- [Chen et al., 2014] Chen, L., Zhou, Y., and Chiu, D. M. (2014). A study of user behavior in online vod services. *Computer Communications*, 46:66–75.
- [Chen et al., 2016a] Chen, R., Bao, F., and Guo, J. (2016a). Trust-based service management for social internet of things systems. *IEEE transactions on dependable and secure computing*, 13(6):684–696.
- [Chen et al., 2016b] Chen, R., Guo, J., and Bao, F. (2016b). Trust management for soa-based iot and its application to service composition. *IEEE Transactions on Services Computing*, 9(3):482–495.

- [Chen et al., 2018] Chen, R., Guo, J., Wang, D.-C., Tsai, J. J., Al-Hamadi, H., and You, I. (2018). Trust-based service management for mobile cloud iot systems. *IEEE transactions on network and service management*, 16(1):246–263.
- [Chollet, 2015] Chollet, F. (2015). keras. <https://github.com/fchollet/keras>.
- [corp, 2019] corp, P. (2019). Planaria: crawl and index the bitcoin blockchain.
- [Corson and Macker, 1999] Corson, S. and Macker, J. (1999). Rfc2501: Mobile ad hoc networking (manet): Routing protocol performance issues and evaluation considerations.
- [Crawford, 1989] Crawford, S. L. (1989). Extensions to the cart algorithm. *International Journal of Man-Machine Studies*, 31(2):197–217.
- [Crocetta, 2019] Crocetta, C. (2019). *Theoretical and Applied Statistics*. Springer.
- [crypto51.app, 2020] crypto51.app (2020). Cost of 51 percent attack for different public blockchains.
- [Dasgupta, 2000] Dasgupta, P. (2000). Trust as a commodity. *Trust: Making and breaking cooperative relations*, 4:49–72.
- [de Sousa, 2016] de Sousa, C. A. (2016). An overview on weight initialization methods for feedforward neural networks. In *(IJCNN), 2016 International Joint Conference on Neural Networks*, pages 52–59. IEEE.
- [Deutsch, 1973] Deutsch, M. (1973). *The resolution of conflict: Constructive and destructive processes*. Yale University Press.
- [Deutsch, 1985] Deutsch, M. (1985). Distributive justice: A social-psychological perspective.
- [Di Pietro et al., 2018] Di Pietro, R., Salleras, X., Signorini, M., and Waisbard, E. (2018). A blockchain-based trust system for the internet of things. In *Proceedings of the 23rd ACM on Symposium on Access Control Models and Technologies*, pages 77–83.
- [Dolezel et al., 2016] Dolezel, P., Skrabanek, P., and Gago, L. (2016). Weight initialization possibilities for feedforward neural network with linear saturated activation functions.

- IFAC-PapersOnLine*, 49(25):49 – 54. 14th IFAC Conference on Programmable Devices and Embedded Systems PDES 2016.
- [Duggan and Brenner, 2013] Duggan, M. and Brenner, J. (2013). *The demographics of social media users, 2012*, volume 14. Pew Research Center’s Internet & American Life Project Washington, DC.
- [Earle, 2010] Earle, T. C. (2010). Trust in risk management: A model-based review of empirical research. *Risk Analysis: An International Journal*, 30(4):541–574.
- [Edemacu and Bulega, 2014] Edemacu, K. and Bulega, T. (2014). Resource sharing between m2m and h2h traffic under time-controlled scheduling scheme in lte networks. In *2014 8th International Conference on Telecommunication Systems Services and Applications (TSSA)*, pages 1–6. IEEE.
- [Erl et al., 2016] Erl, T., Khattak, W., and Buhler, P. (2016). *Big data fundamentals: concepts, drivers & techniques*. Prentice Hall Press.
- [Esfandiari and Chandrasekharan, 2001] Esfandiari, B. and Chandrasekharan, S. (2001). On how agents make friends: Mechanisms for trust acquisition. In *Proceedings of the fourth workshop on deception, fraud and trust in agent societies*, volume 222, page 19.
- [Fang et al., 2012] Fang, H., Zhang, J., ensoy, M., and Thalmann, N. M. (2012). Sarc: subjectivity alignment for reputation computation. In *AAMAS*, pages 1365–1366.
- [Fang et al., 2014] Fang, H., Zhang, J., and Magnenat Thalmann, N. (2014). Subjectivity grouping: learning from users’ rating behavior. In *AAMAS*, pages 1241–1248.
- [Fiedler and Meissner, 2013] Fiedler, M. and Meissner, S. (2013). Iot in practice: examples: Iot in logistics and health. In *Enabling Things to Talk*, pages 27–36. Springer, Berlin, Heidelberg.
- [Folger and Cropanzano, 2001] Folger, R. and Cropanzano, R. (2001). Fairness theory: Justice as accountability. *Advances in organizational justice*, 1(1-55):12.
- [Gambetta et al., 2000] Gambetta, D. et al. (2000). Can we trust trust. *Trust: Making and breaking cooperative relations*, 13:213–237.
- [Garnett et al., 2005] Garnett, R., Huegerich, T., Chui, C., and He, W. (2005). A universal noise removal algorithm with an impulse detector. *IEEE Transactions on image processing*, 14(11):1747–1754.

- [Gervais et al., 2016] Gervais, A., Karame, G. O., Wüst, K., Glykantzis, V., Ritzdorf, H., and Capkun, S. (2016). On the security and performance of proof of work blockchains. In *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, pages 3–16.
- [Ghani et al., 2019] Ghani, N. A., Hamid, S., Hashem, I. A. T., and Ahmed, E. (2019). Social media big data analytics: A survey. *Computers in Human Behavior*, 101:417–428.
- [Gibs and Bruich, 2010] Gibs, J. and Bruich, S. (2010). Nielsen/facebook report: The value of social media ad impressions. *Nielsen Blog*, 20.
- [Glorot and Bengio, 2010] Glorot, X. and Bengio, Y. (2010). Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 249–256.
- [Golbeck, 2005] Golbeck, J. (2005). A definition of trust for computing with social networks. <http://www.mindswap.org/papers/TrustDef.doc> [Accessed: May 2019].
- [Google, 2018a] Google (2018a). *Statistical trend analysis of e-commerce from 2004 to present*. <https://trends.google.com/trends/explore?date=all&q=e-commerce> [Accessed: March 2018].
- [Google, 2018b] Google (2018b). *Statistical trend analysis of IoT from 2004 to present*. <https://trends.google.com/trends/explore?date=all&q=IoT> [Accessed: March 2018].
- [Greenhow and Lewin, 2016] Greenhow, C. and Lewin, C. (2016). Social media and education: Reconceptualizing the boundaries of formal and informal learning. *Learning, media and technology*, 41(1):6–30.
- [Grishchenko, 1999] Grishchenko, V. S. (1999). “redefining web-of-trust: reputation, recommendations, responsibility and trust among peers”, ural state university.
- [Grishchenko, 2004] Grishchenko, V. S. (2004). A fuzzy model for context-dependent reputation. In *Proceedings of the 2004 International Conference on Trust, Security, and Reputation on the Semantic Web-Volume 127*, pages 72–77. CEUR-WS. org.

- [Gubbi et al., 2013] Gubbi, J., Buyya, R., Marusic, S., and Palaniswami, M. (2013). Internet of things (iot): A vision, architectural elements, and future directions. *Future generation computer systems*, 29(7):1645–1660.
- [Guha et al., 2004] Guha, R., Kumar, R., Raghavan, P., and Tomkins, A. (2004). Propagation of trust and distrust. In *Proceedings of the 13th international conference on World Wide Web*, pages 403–412.
- [Guo et al., 2005] Guo, W., Shi, X., Cao, L., and Yang, K. (2005). Trust in knowledge flow networks. In *2005 First International Conference on Semantics, Knowledge and Grid*, pages 35–35. IEEE.
- [Gurajala et al., 2015] Gurajala, S., White, J. S., Hudson, B., and Matthews, J. N. (2015). Fake twitter accounts: profile characteristics obtained using an activity-based pattern detection approach. In *Proceedings of the 2015 International Conference on Social Media & Society*, pages 1–7.
- [Gurney, 1997] Gurney, K. (1997). *An introduction to neural networks*. CRC press.
- [Hammi et al., 2018] Hammi, M. T., Hammi, B., Bellot, P., and Serhrouchni, A. (2018). Bubbles of trust: A decentralized blockchain-based authentication system for iot. *Computers & Security*, 78:126–142.
- [Hao et al., 2018] Hao, Y., Li, Y., Dong, X., Fang, L., and Chen, P. (2018). Performance analysis of consensus algorithm in private blockchain. In *Proceedings of the 2018 IEEE Intelligent Vehicles Symposium (IV)*, pages 280–285.
- [Harper and Konstan, 2015] Harper, F. M. and Konstan, J. A. (2015). The movielens datasets: History and context. *Acm transactions on interactive intelligent systems (tiis)*, 5(4):1–19.
- [Hassani et al., 2018] Hassani, H., Huang, X., and Silva, E. (2018). Big-crypto: Big data, blockchain and cryptocurrency. *Big Data and Cognitive Computing*, 2(4):34.
- [Haykin, 1994] Haykin, S. (1994). *Neural networks: a comprehensive foundation*. Prentice Hall PTR.
- [Heaton, 2008] Heaton, J. (2008). *Introduction to neural networks with Java*. Heaton Research, Inc.

- [Hendrikx et al., 2015] Hendrikx, F., Bubendorfer, K., and Chard, R. (2015). Reputation systems: A survey and taxonomy. *Journal of Parallel and Distributed Computing*, 75:184–197.
- [Herlihy and Shavit, 2012] Herlihy, M. and Shavit, N. (2012). *The Art of Multiprocessor Programming, revised first edition*. Morgan Kaufmann.
- [Hicks, 1939] Hicks, J. R. (1939). The foundations of welfare economics. *The Economic Journal*, 49(196):696–712.
- [Highfield and Leaver, 2016] Highfield, T. and Leaver, T. (2016). Instagrammatics and digital methods: Studying visual social media, from selfies and gifs to memes and emoji. *Communication Research and Practice*, 2(1):47–62.
- [Hoffman et al., 2009] Hoffman, K., Zage, D., and Nita-Rotaru, C. (2009). A survey of attack and defense techniques for reputation systems. *ACM Computing Surveys (CSUR)*, 42(1):1–31.
- [Hopkins and Hawking, 2018] Hopkins, J. and Hawking, P. (2018). Big data analytics and iot in logistics: A case study. *The International Journal of Logistics Management*.
- [Horas, 2018] Horas, K. (2018). Probability density function for the beta distribution. created using gnuplot. https://en.wikipedia.org/wiki/File:Beta_distribution_pdf.svg. (Accessed on 6 Nov 2018).
- [Hossai et al., 2017] Hossai, M. R. T., Shahjalal, M. A., and Nuri, N. F. (2017). Design of an iot based autonomous vehicle with the aid of computer vision. In *2017 International Conference on Electrical, Computer and Communication Engineering (ECCE)*, pages 752–756. IEEE.
- [Hussain and Chang, 2007] Hussain, F. K. and Chang, E. (2007). An overview of the interpretations of trust and reputation. In *The Third Advanced International Conference on Telecommunications (AICT’07)*, pages 30–30. IEEE.
- [IDC, 2018] IDC (2018). Worldwide data sphere. <https://www.seagate.com/files/www-content/our-story/trends/files/idc-seagate-dataage-whitepaper.pdf>. (Accessed on 20 July 2020).
- [Jøsang, 1996] Jøsang, A. (1996). The right type of trust for distributed systems. In *Proceedings of the 1996 workshop on New security paradigms*, pages 119–131.

- [Josang and Haller, 2007] Josang, A. and Haller, J. (2007). Dirichlet reputation systems. In *Proceedings of the The Second International Conference on Availability, Reliability and Security, ARES '07*, pages 112–119, Washington, DC, USA. IEEE Computer Society.
- [Josang and Ismail, 2002] Josang, A. and Ismail, R. (2002). The beta reputation system. In *Proceedings of the 15th bled electronic commerce conference*, volume 5, pages 2502–2511.
- [Jøsang et al., 2007] Jøsang, A., Ismail, R., and Boyd, C. (2007). A survey of trust and reputation systems for online service provision. *Decision support systems*, 43(2):618–644.
- [Jøsang and Ismail, 2002] Jøsang, A. and Ismail, R. (2002). The beta reputation system. In *In Proceedings of the 15th Bled Conference on Electronic Commerce*.
- [Kaldor, 1939] Kaldor, N. (1939). Welfare propositions of economics and interpersonal comparisons of utility. *The Economic Journal*, pages 549–552.
- [Kastidou et al., 2009] Kastidou, G., Larson, K., and Cohen, R. (2009). Exchanging reputation information between communities: A payment-function approach. In *Twenty-First International Joint Conference on Artificial Intelligence*.
- [Kleinbaum et al., 2002] Kleinbaum, D. G., Dietz, K., Gail, M., Klein, M., and Klein, M. (2002). *Logistic regression*. Springer.
- [Kolias et al., 2017a] Kolias, C., Kambourakis, G., Stavrou, A., and Voas, J. (2017a). Ddos in the iot: Mirai and other botnets. *Computer*, 50(7):80–84.
- [Kolias et al., 2017b] Kolias, C., Kambourakis, G., Stavrou, A., and Voas, J. (2017b). Ddos in the iot: Mirai and other botnets. *Computer*, 50(7):80–84.
- [Kouicem et al., 2020] Kouicem, D. E., Imine, Y., Bouabdallah, A., and Lakhlef, H. (2020). A decentralized blockchain-based trust management protocol for the internet of things. *IEEE Transactions on Dependable and Secure Computing*.
- [Kreps and Wilson, 1982] Kreps, D. M. and Wilson, R. (1982). Reputation and imperfect information. *Journal of economic theory*, 27(2):253–279.
- [Kumar et al., 2020] Kumar, P., Gupta, G. P., and Tripathi, R. (2020). Tp2sf: A trustworthy privacy-preserving secured framework for sustainable smart cities by leveraging blockchain and machine learning. *Journal of Systems Architecture*, page 101954.

- [Labrinidis and Jagadish, 2012] Labrinidis, A. and Jagadish, H. V. (2012). Challenges and opportunities with big data. *Proceedings of the VLDB Endowment*, 5(12):2032–2033.
- [LeCun et al., 2015] LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *nature*, 521(7553):436.
- [Levine et al., 2006] Levine, B. N., Shields, C., and Margolin, N. B. (2006). A survey of solutions to the sybil attack. *University of Massachusetts Amherst, Amherst, MA*, 7:224.
- [Lim et al., 2017] Lim, X. J., Radzol, A. M., Cheah, J., and Wong, M. (2017). The impact of social media influencers on purchase intention and the mediation effect of customer attitude. *Asian Journal of Business Research*, 7(2):19–36.
- [Lin et al., 2017] Lin, J., Shen, Z., and Miao, C. (2017). Using blockchain technology to build trust in sharing lorawan iot. In *Proceedings of the 2nd International Conference on Crowd Science and Engineering*, pages 38–43.
- [Lin and Dong, 2017] Lin, Z. and Dong, L. (2017). Clarifying trust in social internet of things. *IEEE Transactions on Knowledge and Data Engineering*, 30(2):234–248.
- [Liu et al., 2011] Liu, S., Zhang, J., Miao, C., Theng, Y.-L., and Kot, A. C. (2011). iclub: An integrated clustering-based approach to improve the robustness of reputation systems. In *The 10th International Conference on Autonomous Agents and Multiagent Systems-Volume 3*, pages 1151–1152. International Foundation for Autonomous Agents and Multiagent Systems.
- [Liu et al., 2013] Liu, X., Datta, A., and Rzađca, K. (2013). Trust beyond reputation: A computational trust model based on stereotypes. *Electron. Commer. Rec. Appl.*, 12(1):24–39.
- [Liu et al., 2017] Liu, Y., Liang, C., Chiclana, F., and Wu, J. (2017). A trust induced recommendation mechanism for reaching consensus in group decision making. *Knowledge-Based Systems*, 119:221–231.
- [Liu and Zhou, 2012] Liu, Y. and Zhou, G. (2012). Key technologies and applications of internet of things. In *2012 Fifth International Conference on Intelligent Computation Technology and Automation*, pages 197–200. IEEE.
- [Lu et al., 2018a] Lu, Z., Liu, W., Wang, Q., Qu, G., and Liu, Z. (2018a). A privacy-preserving trust model based on blockchain for vanets. *IEEE Access*, 6:45655–45664.

- [Lu et al., 2018b] Lu, Z., Wang, Q., Qu, G., and Liu, Z. (2018b). Bars: a blockchain-based anonymous reputation system for trust management in vanets. In *Proceedings of the 17th IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom-2018)*, pages 98–103.
- [Luhmann, 2000] Luhmann, N. (2000). Familiarity, confidence, trust: Problems and alternatives. *Trust: Making and breaking cooperative relations*, 6(1):94–107.
- [Luo and Zeynalvand, 2017] Luo, T. and Zeynalvand, L. (2017). Reshaping mobile crowd sensing using cross validation to improve data credibility. In *IEEE GLOBECOM*.
- [Luo and Yang, 2014] Luo, Y. and Yang, F. (2014). Deep learning with noise. [hp://www.andrew.cmu.edu/user/fanyang1/deep-learning-with-noise.pdf](http://www.andrew.cmu.edu/user/fanyang1/deep-learning-with-noise.pdf).
- [Madić and Marinković, 2010] Madić, M. J. and Marinković, V. J. (2010). Assessing the sensitivity of the artificial neural network to experimental noise: a case study. *FME Transactions*, 38(4):189–195.
- [Madrigal, 2017] Madrigal, A. C. (2017). What facebook did to american democracy. *The Atlantic*, 12.
- [Malbon, 2013] Malbon, J. (2013). Taking fake online consumer reviews seriously. *Journal of Consumer Policy*, 36(2):139–157.
- [Malik et al., 2019] Malik, N., Nanda, P., He, X., and Liu, R. (2019). Trust and reputation in vehicular networks: A smart contract-based approach. In *2019 18th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/13th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE)*, pages 34–41. IEEE.
- [Malik and Bouguettaya, 2009] Malik, Z. and Bouguettaya, A. (2009). Rateweb: Reputation assessment for trust establishment among web services. *The VLDB Journal*, 18(4):885–911.
- [Manapragada et al., 2018] Manapragada, C., Webb, G. I., and Salehi, M. (2018). Extremely fast decision tree. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1953–1962.

- [Mansour, 2018] Mansour, E. (2018). The adoption and use of social media as a source of information by egyptian government journalists. *Journal of Librarianship and Information Science*, 50(1):48–67.
- [Marjani et al., 2017] Marjani, M., Nasaruddin, F., Gani, A., Karim, A., Hashem, I. A. T., Siddiqua, A., and Yaqoob, I. (2017). Big iot data analytics: architecture, opportunities, and open research challenges. *IEEE Access*, 5:5247–5261.
- [Marti et al., 2000] Marti, S., Giuli, T. J., Lai, K., and Baker, M. (2000). Mitigating routing misbehavior in mobile ad hoc networks. In *Proceedings of the 6th annual international conference on Mobile computing and networking*, pages 255–265.
- [Marzano et al., 2018] Marzano, A., Alexander, D., Fonseca, O., Fazzion, E., Hoepers, C., Steding-Jessen, K., Chaves, M. H., Cunha, Í., Guedes, D., and Meira, W. (2018). The evolution of bashlite and mirai iot botnets. In *2018 IEEE Symposium on Computers and Communications (ISCC)*, pages 00813–00818. IEEE.
- [Mezzetti, 2003] Mezzetti, N. (2003). Towards a model for trust relationships in virtual enterprises. In *14th International Workshop on Database and Expert Systems Applications, 2003. Proceedings.*, pages 420–424. IEEE.
- [Mezzetti, 2004] Mezzetti, N. (2004). A socially inspired reputation model. In *European Public Key Infrastructure Workshop*, pages 191–204. Springer.
- [Mhetre et al., 2016] Mhetre, N. A., Deshpande, A. V., and Mahalle, P. N. (2016). Trust management model based on fuzzy approach for ubiquitous computing. *International Journal of Ambient Computing and Intelligence (IJACI)*, 7(2):33–46.
- [Michiardi and Molva, 2002] Michiardi, P. and Molva, R. (2002). Core: a collaborative reputation mechanism to enforce node cooperation in mobile ad hoc networks. In *Advanced communications and multimedia security*, pages 107–121. Springer.
- [Mingxiao et al., 2017] Mingxiao, D., Xiaofeng, M., Zhe, Z., Xiangwei, W., and Qijun, C. (2017). A review on consensus algorithm of blockchain. In *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics (SMC-2017)*, pages 2567–2572.

- [Moinet et al., 2017] Moinet, A., Darties, B., and Baril, J.-L. (2017). Blockchain based trust & authentication for decentralized sensor networks. *arXiv preprint arXiv:1706.01730*.
- [Mui et al., 2002a] Mui, L., Halberstadt, A., and Mohtashemi, M. (2002a). Evaluating reputation in multi-agents systems. In *Workshop on Deception, Fraud and Trust in Agent Societies*, pages 123–137. Springer.
- [Mui et al., 2002b] Mui, L., Mohtashemi, M., and Halberstadt, A. (2002b). A computational model of trust and reputation for e-businesses. In *Proceedings of the 35th Annual Hawaii International Conference on System Sciences (HICSS'02)-Volume 7 - Volume 7*, HICSS '02, pages 188–, Washington, DC, USA. IEEE Computer Society.
- [Mui et al., 2002c] Mui, L., Mohtashemi, M., and Halberstadt, A. (2002c). Notions of reputation in multi-agents systems: a review. In *Proceedings of the first international joint conference on Autonomous agents and multiagent systems: part 1*, pages 280–287.
- [Mukherjee et al., 2013] Mukherjee, A., Venkataraman, V., Liu, B., and Glance, N. (2013). What yelp fake review filter might be doing? In *Proceedings of the AAAI-2013*.
- [Nagy and Simon, 2018] Nagy, A. M. and Simon, V. (2018). Survey on traffic prediction in smart cities. *Pervasive and Mobile Computing*, 50:148–163.
- [Nielsen, 1994] Nielsen, J. (1994). *Usability engineering*. Elsevier.
- [Nielsen, 2015] Nielsen, M. A. (2015). *Neural networks and deep learning*, volume 2018. Determination press San Francisco, CA, USA:.
- [Nielsen and Schröder, 2014] Nielsen, R. K. and Schröder, K. C. (2014). The relative importance of social media for accessing, finding, and engaging with news: An eight-country cross-media comparison. *Digital journalism*, 2(4):472–489.
- [Nitti et al., 2014] Nitti, M., Girau, R., and Atzori, L. (2014). Trustworthiness management in the social internet of things. *IEEE Transactions on knowledge and data engineering*, 26(5):1253–1266.
- [Nitti et al., 2014] Nitti, M., Girau, R., and Atzori, L. (2014). Trustworthiness management in the social internet of things. *IEEE Transactions on Knowledge and Data Engineering*, 26(5):1253–1266.

- [Papadimitriou, 2003] Papadimitriou, C. H. (2003). *Computational complexity*. John Wiley and Sons Ltd.
- [Parkvall et al., 2017] Parkvall, S., Dahlman, E., Furuskar, A., and Frenne, M. (2017). Nr: The new 5g radio access technology. *IEEE Communications Standards Magazine*, 1(4):24–30.
- [Patel and Jhaveri, 2015] Patel, N. J. and Jhaveri, R. H. (2015). Trust based approaches for secure routing in vanet: A survey. *Procedia Computer Science*, 45:592–601.
- [Patti and Acquaviva, 2016] Patti, E. and Acquaviva, A. (2016). Iot platform for smart cities: Requirements and implementation case studies. In *2016 IEEE 2nd International Forum on Research and Technologies for Society and Industry Leveraging a better tomorrow (RTSI)*, pages 1–6. IEEE.
- [Perner, 2015] Perner, P. (2015). Decision tree induction methods and their application to big data. In *Modeling and Processing for Next-Generation Big-Data Technologies*, pages 57–88. Springer.
- [Pissinou et al., 2004] Pissinou, N., Ghosh, T., and Makki, K. (2004). Collaborative trust-based secure routing in multihop ad hoc networks. In *International Conference on Research in Networking*, pages 1446–1451. Springer.
- [Provost and Fawcett, 2013] Provost, F. and Fawcett, T. (2013). Data science and its relationship to big data and data-driven decision making. *Big data*, 1(1):51–59.
- [Qiu et al., 2016] Qiu, J., Wu, Q., Ding, G., Xu, Y., and Feng, S. (2016). A survey of machine learning for big data processing. *EURASIP Journal on Advances in Signal Processing*, 2016(1):67.
- [Qiu et al., 2013] Qiu, W., Zheng, Z., Wang, X., Yang, X., and Lyu, M. R. (2013). Reputation-aware qos value prediction of web services. In *2013 IEEE International Conference on Services Computing*, pages 41–48. IEEE.
- [Qualman, 2012] Qualman, E. (2012). *Socialnomics: How social media transforms the way we live and do business*. John Wiley & Sons.
- [Rayana and Akoglu, 2015] Rayana, S. and Akoglu, L. (2015). Collective opinion spam detection: Bridging review networks and metadata. In *Proceedings of the 21th ACM*

- SigKDD International Conference on Knowledge Discovery and Data Mining*, pages 985–994.
- [Regan et al., 2006] Regan, K., Poupart, P., and Cohen, R. (2006). Bayesian reputation modeling in e-marketplaces sensitive to subjectivity, deception and change. In *AAAI*, volume 21, page 1206.
- [Resnick et al., 2000] Resnick, P., Kuwabara, K., Zeckhauser, R., and Friedman, E. (2000). Reputation systems. *Communications of the ACM*, 43(12):45–48.
- [Resnick and Zeckhauser, 2002a] Resnick, P. and Zeckhauser, R. (2002a). Trust among strangers in internet transactions: Empirical analysis of ebay’s reputation system.
- [Resnick and Zeckhauser, 2002b] Resnick, P. and Zeckhauser, R. (2002b). Trust among strangers in internet transactions: Empirical analysis of ebay’s reputation system.
- [Ripley, 2007] Ripley, B. D. (2007). *Pattern recognition and neural networks*. Cambridge university press.
- [Ross, 2005] Ross, T. J. (2005). *Fuzzy logic with engineering applications*. John Wiley & Sons.
- [Ruder, 2016] Ruder, S. (2016). An overview of gradient descent optimization algorithms. *arXiv preprint arXiv:1609.04747*.
- [Rydning, 2018] Rydning, D. R.-J. G.-J. (2018). The digitization of the world from edge to core. *Framingham: International Data Corporation*.
- [Ryzko, 2020] Ryzko, D. (2020). *Modern Big Data Architectures: A Multi-agent Systems Perspective*. John Wiley & Sons.
- [Sabater, 2004] Sabater, J. (2004). Evaluating the regret system. *Applied Artificial Intelligence*, 18(9-10):797–813.
- [Sabater and Sierra, 2001a] Sabater, J. and Sierra, C. (2001a). Regret: reputation in gregarious societies. In *Proceedings of the fifth international conference on Autonomous agents*, pages 194–195.
- [Sabater and Sierra, 2001b] Sabater, J. and Sierra, C. (2001b). Regret: Reputation in gregarious societies. In *Proceedings of AGENTS ’01, AGENTS ’01*, pages 194–195, New York, NY, USA. ACM.

- [Sabater and Sierra, 2001c] Sabater, J. and Sierra, C. (2001c). Social regret, a reputation model based on social relations. *ACM SIGecom Exchanges*, 3(1):44–56.
- [Sabater and Sierra, 2002a] Sabater, J. and Sierra, C. (2002a). Reputation and social network analysis in multi-agent systems. In *Proceedings of the first international joint conference on Autonomous agents and multiagent systems: Part 1*, pages 475–482.
- [Sabater and Sierra, 2002b] Sabater, J. and Sierra, C. (2002b). Reputation and social network analysis in multi-agent systems. In *Proceedings of the First International Joint Conference on Autonomous Agents and Multiagent Systems: Part 1*, AAMAS '02, pages 475–482, New York, NY, USA. ACM.
- [Salam et al., 2005] Salam, A. F., Iyer, L., Palvia, P., and Singh, R. (2005). Trust in e-commerce. *Communications of the ACM*, 48(2):72–77.
- [Savelyev, 2017] Savelyev, A. (2017). Contract law 2.0: ‘smart’ contracts as the beginning of the end of classic contract law. *Information & Communications Technology Law*, 26(2):116–134.
- [Schneider et al., 2000] Schneider, J., Kortuem, G., Jager, J., Fickas, S., and Segall, Z. (2000). Disseminating trust information in wearable communities. *Personal Technologies*, 4(4):245–248.
- [Sigala and Gretzel, 2017] Sigala, M. and Gretzel, U. (2017). *Advances in social media for travel, tourism and hospitality: New perspectives, practice and cases*. Routledge.
- [Simoncelli and Adelson, 1996] Simoncelli, E. P. and Adelson, E. H. (1996). Noise removal via bayesian wavelet coring. In *Proceedings of the 3rd IEEE International Conference on Image Processing*, volume 1, pages 379–382.
- [Simplify.Inc., 2018] Simplify.Inc. (2018). Simplify: An app that brings sharing economy to the iot space. <https://simplify.network/mobile> [Last accessed July 7, 2020].
- [Singh et al., 2020] Singh, P. K., Singh, R., Nandi, S. K., Ghafoor, K. Z., Rawat, D. B., and Nandi, S. (2020). Blockchain-based adaptive trust management in internet of vehicles using smart contract. *IEEE Transactions on Intelligent Transportation Systems*.

- [Singh, 2015] Singh, S. (2015). The impact of marketing variables on business performance: An analysis of fmcg, consumer durables and textile industries. *IUP Journal of Business Strategy*, 12(2).
- [Small, 2006] Small, H. (2006). Tracking and predicting growth areas in science. *Scientometrics*, 68(3):595–610.
- [Sobkowicz et al., 2013] Sobkowicz, P., Thelwall, M., Buckley, K., Paltoglou, G., and Sobkowicz, A. (2013). Lognormal distributions of user post lengths in internet discussions-a consequence of the weber-fechner law? *EPJ Data Science*, 2(1):2.
- [Song and Ying, 2015] Song, Y.-Y. and Ying, L. (2015). Decision tree methods: applications for classification and prediction. *Shanghai archives of psychiatry*, 27(2):130.
- [Spiekermann et al., 2015] Spiekermann, S., Acquisti, A., Böhme, R., and Hui, K.-L. (2015). The challenges of personal data markets and privacy. *Electronic markets*, 25(2):161–167.
- [Sprinthall and Fisk, 1990] Sprinthall, R. C. and Fisk, S. T. (1990). *Basic statistical analysis*. Prentice Hall Englewood Cliffs, NJ.
- [Stieglitz and Dang-Xuan, 2013] Stieglitz, S. and Dang-Xuan, L. (2013). Social media and political communication: a social media analytics framework. *Social network analysis and mining*, 3(4):1277–1291.
- [Streitfeld, 2012] Streitfeld, D. (2012). The best book reviews money can buy. *The New York Times*, 25(08).
- [Strobl et al., 2009] Strobl, C., Malley, J., and Tutz, G. (2009). An introduction to recursive partitioning: Rationale, application, and characteristics of classification and regression trees, bagging, and random forests. *Psychological methods*, 14:323–48.
- [Sztompka, 1999] Sztompka, P. (1999). *Trust: A sociological theory*. Cambridge University Press.
- [Tariq et al., 2019] Tariq, N., Asim, M., Al-Obeidat, F., Zubair Farooqi, M., Baker, T., Hammoudeh, M., and Ghafir, I. (2019). The security of big data in fog-enabled iot applications including blockchain: a survey. *Sensors*, 19(8):1788.

- [Teacy et al., 2012] Teacy, W. L., Luck, M., Rogers, A., and Jennings, N. R. (2012). An efficient and versatile approach to trust and reputation using hierarchical bayesian modelling. *Artificial Intelligence*, 193:149–185.
- [Teacy et al., 2006] Teacy, W. T. L., Patel, J., Jennings, N. R., and Luck, M. (2006). Travos: Trust and reputation in the context of inaccurate information sources. *Autonomous Agents and Multi-Agent Systems*, 12(2):183–198.
- [Tian and Yang, 2011] Tian, C. and Yang, B. (2011). R2trust, a reputation and risk based trust management framework for large-scale, fully decentralized overlay networks. *Future Generation Computer Systems*, 27(8):1135–1141.
- [Trcek and Brodnik, 2013] Trcek, D. and Brodnik, A. (2013). Hard and soft security provisioning for computationally weak pervasive computing systems in e-health. *IEEE wireless communications*, 20(4):22–29.
- [Twardowski and Ryzko, 2014] Twardowski, B. and Ryzko, D. (2014). Multi-agent architecture for real-time big data processing. In *2014 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT)*, volume 3, pages 333–337. IEEE.
- [Urena et al., 2019] Urena, R., Kou, G., Dong, Y., Chiclana, F., and Herrera-Viedma, E. (2019). A review on trust propagation and opinion dynamics in social networks and group decision making frameworks. *Information Sciences*, 478:461–475.
- [Van Nee, 2011] Van Nee, R. (2011). Breaking the gigabit-per-second barrier with 802.11 ac. *IEEE Wireless Communications*, 18(2):4–4.
- [Victor et al., 2009] Victor, P., Cornelis, C., De Cock, M., and Da Silva, P. P. (2009). Gradual trust and distrust in recommender systems. *Fuzzy Sets and Systems*, 160(10):1367–1382.
- [Wang et al., 2005] Wang, B., Soltani, S., Shapiro, J. K., and Tan, P.-N. (2005). Local detection of selfish routing behavior in ad hoc networks. In *8th International Symposium on Parallel Architectures, Algorithms and Networks (ISPAN’05)*, pages 8–pp. IEEE.
- [Wang and Vassileva, 2003a] Wang, Y. and Vassileva, J. (2003a). Bayesian network-based trust model. In *Proceedings IEEE/WIC International Conference on Web Intelligence (WI 2003)*, pages 372–378. IEEE.

- [Wang and Vassileva, 2003b] Wang, Y. and Vassileva, J. (2003b). Trust and reputation model in peer-to-peer networks. In *Proceedings Third International Conference on Peer-to-Peer Computing (P2P2003)*, pages 150–157. IEEE.
- [weather.gov.sg, 2021] weather.gov.sg (2021). Climate of singapore. <http://www.weather.gov.sg/climate-climate-of-singapore>.
- [Weng et al., 2006] Weng, J., Miao, C., and Goh, A. (2006). An entropy-based approach to protecting rating systems from unfair testimonies. 89-D:2502–2511.
- [Wessel et al., 2016] Wessel, M., Thies, F., and Benlian, A. (2016). The emergence and effects of fake social information: Evidence from crowdfunding. *Decision Support Systems*, 90:75–85.
- [Wu et al., 2015] Wu, J., Chiclana, F., and Herrera-Viedma, E. (2015). Trust based consensus model for social network in an incomplete linguistic information context. *Applied Soft Computing*, 35:827–839.
- [Wu et al., 2018a] Wu, J., Dai, L., Chiclana, F., Fujita, H., and Herrera-Viedma, E. (2018a). A minimum adjustment cost feedback mechanism based consensus model for group decision making under social network with distributed linguistic trust. *Information Fusion*, 41:232–242.
- [Wu et al., 2018b] Wu, J., Dai, L., Chiclana, F., Fujita, H., and Herrera-Viedma, E. (2018b). A new consensus model for social network group decision making based on a minimum adjustment feedback mechanism and distributed linguistic trust. *Information Fusion*, 41:232–242.
- [Wu et al., 2016] Wu, J., Xiong, R., and Chiclana, F. (2016). Uninorm trust propagation and aggregation methods for group decision making in social network with four tuple information. *Knowledge-Based Systems*, 96:29–39.
- [www.fakespot.com, 2019] www.fakespot.com (2019). Fakespot, a review analysis service. <https://www.fakespot.com/>.
- [www.meilleursagents.com, 2014] www.meilleursagents.com (2014). Mapping out the buyer/seller ratio. <https://www.meilleursagents.com/>.

- [Yan et al., 2015] Yan, S.-R., Zheng, X.-L., Wang, Y., Song, W. W., and Zhang, W.-Y. (2015). A graph-based comprehensive reputation model: Exploiting the social context of opinions to enhance trust in social commerce. *Information Sciences*, 318:51–72.
- [Yan et al., 2014] Yan, Z., Zhang, P., and Vasilakos, A. V. (2014). A survey on trust management for internet of things. *Journal of network and computer applications*, 42:120–134.
- [Yang et al., 2018] Yang, Z., Yang, K., Lei, L., Zheng, K., and Leung, V. C. (2018). Blockchain-based decentralized trust management in vehicular networks. *IEEE Internet of Things Journal*, 6(2):1495–1505.
- [Yousefpour et al., 2019] Yousefpour, A., Fung, C., Nguyen, T., Kadiyala, K., Jalali, F., Niakanlahiji, A., Kong, J., and Jue, J. P. (2019). All one needs to know about fog computing and related edge computing paradigms: A complete survey. *Journal of Systems Architecture*.
- [Yu et al., 2018] Yu, B., Wright, J., Nepal, S., Zhu, L., Liu, J., and Ranjan, R. (2018). Iotchain: Establishing trust in the internet of things ecosystem using blockchain. *IEEE Cloud Computing*, 5(4):12–23.
- [Yu et al., 2011] Yu, H., Liu, S., Kot, A. C., Miao, C., and Leung, C. (2011). Dynamic witness selection for trustworthy distributed cooperative sensing in cognitive radio networks. In *Communication Technology (ICCT), 2011 IEEE 13th International Conference on*, pages 1–6. IEEE.
- [Yu et al., 2013] Yu, H., Shen, Z., Leung, C., Miao, C., and Lesser, V. R. (2013). A survey of multi-agent trust management systems. *IEEE Access*, 1:35–50.
- [Yu et al., 2010] Yu, H., Shen, Z., Miao, C., Leung, C., and Niyato, D. (2010). A survey of trust and reputation management systems in wireless communications. *Proceedings of the IEEE*, 98(10):1755–1772.
- [Zeynalvand, 2020] Zeynalvand, L. (2020). Ux survey study for trust and reputation management in big data. <https://myprototype.online/ux> [Last accessed July 7, 2020].
- [Zeynalvand et al., 2020] Zeynalvand, L., Luo, T., and Zhang, J. (2020). Cobra: Context-aware bernoulli neural networks for reputation assessment. In *AAAI*.

- [Zhang et al., 2019] Zhang, R., Xue, R., and Liu, L. (2019). Security and privacy on blockchain. *ACM Computing Surveys (CSUR)*, 52(3):1–34.
- [Zhang, 2010] Zhang, W. (2010). *Computational ecology: artificial neural networks and their applications*. World Scientific.
- [Zheng et al., 2014] Zheng, Z., Zhang, Y., and Lyu, M. R. (2014). Investigating qos of real-world web services. *IEEE Trans. Serv. Comput.*, 7(1):32–39.
- [Zhou et al., 2015] Zhou, P., Gu, X., Zhang, J., and Fei, M. (2015). A priori trust inference with context-aware stereotypical deep learning. *Knowledge-Based Systems*, 88:97–106.
- [Zhu et al., 2017] Zhu, C., Cheng, G., and Wang, K. (2017). Big data analytics for program popularity prediction in broadcast tv industries. *IEEE Access*, 5:24593–24601.
- [Zouridaki et al., 2006] Zouridaki, C., Mark, B. L., Hejmo, M., and Thomas, R. K. (2006). Robust cooperative trust establishment for manets. In *Proceedings of the fourth ACM workshop on Security of ad hoc and sensor networks*, pages 23–34.