IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received June 14, 2021, accepted July 23, 2021, date of publication July 27, 2021, date of current version August 5, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3100767

On the Trust and Trust Modeling for the Future Fully-Connected Digital World: A Comprehensive Study

HANNAH LIM JING TING, (Student Member, IEEE), XIN KANG, (Senior Member, IEEE), TIEYAN LI, (Member, IEEE), HAIGUANG WANG, (Senior Member, IEEE), AND CHENG-KANG CHU, (Member, IEEE)

Digital Identity and Trustworthiness Laboratory, Huawei Singapore, Singapore 138588

Corresponding author: Xin Kang (kang.xin@huawei.com)

ABSTRACT With the fast development of digital technologies, we are running into a digital world. The relationship among people and the connections among things become more and more complex, and new challenges arise. To tackle these challenges, trust — a soft security mechanism — is considered a promising technology. Thus, in this survey, we do a comprehensive study on trust and trust modelling for the future digital world. We revisit the definitions and properties of trust, analyse the trust theories, and discuss their impact on digital trust modelling. We analyse the digital world and its corresponding environment where people, things, and infrastructure connect with each other. We detail the challenges that require trust in these digital scenarios. Under our analysis of trust and the digital world, we define different types of trust relationships and find out the factors that are needed to ensure a fully representative model. Next, to meet the challenges of digital trust modelling, comprehensive trust model evaluation criteria are proposed, and potential security and privacy issues of trust modelling are analysed. Finally, we provide a wide-ranging analysis of different methodologies, mathematical theories, and how they can be applied to trust modelling.

INDEX TERMS Trust, trust modelling, digital trust, digital world, security and privacy.

The world of digital data and information transfer is growing. Devices, once only capable of communicating within homogeneous networks, can now transfer data between devices of varying background and capability. Technology continues to advance in this direction to allow devices, from sensors to smart phones, to communicate. As devices from diverse backgrounds connect, their physical and social environments become integrated with the digital world. Today, service providers — that aid people, businesses, and society — increasingly utilise digital technology. Information exchange supported by the digital world has a substantial impact on society. Thus, digital exchanges need to be safeguarded.

Integrating digital, social, and physical worlds, however, exposes the digital world to newer and more complex vulnerabilities. Growth in the number and variety of entities in the digital world means digital exchanges are subjective and situational, with entities behaving and prioritising differently. It also means hard security, which provides widespread authenticated access control, is now unfeasible [1]. As mere participants, malicious entities can infiltrate and cause disturbances in digital networks. Regulating behaviour to control the actions of entities and minimise their negative impact — a softer form of security — is now necessary to ensure digital communities are conducive [2].

Trust is a nuanced social concept instinctively used for interaction [3], [4]. Reflecting these social properties of trust into the digital world, allows digital entities to perceive others and choose their interactions, as is done in the real world. Therefore, trust, implemented as a soft security mechanism, provides much needed social management. Nevertheless, trust is hard to quantify; the perception mechanisms we use in the physical and social world are not available to implement in digital environments. Trust modelling is needed to mimic the evaluation and decision-making instinctively performed in real life. Efforts have been made in several digital environments and research continues to grow.

The associate editor coordinating the review of this manuscript and approving it for publication was Zesong Fei.

There have been several related trust survey papers. Shrikant and Sunilkumar performed a brief survey of trust models for Vehicular Ad hoc Networks (VANET) [5]. A more comprehensive trust survey for a broader range of trust scenarios was proposed by Yan Zhen in her survey of trust for Internet of Things (IoT) [6]. Ruan surveyed trust in a different digital environment: online social communities [7].

Each of these surveys discussed the unique characteristics of the digital environment. In the VANET and IoT survey, the unique characteristics of the environment was used to outline key, wide-ranging, sometimes practical objectives for trust management. Yan Zhen went a step further, discussing how each of these objectives had to be addressed in each of their defined IoT layers. Ruan's online social community trust survey considered trust management objectives differently, recognising that there were attacks specifically targeting trust management systems, that needed to be addressed.

This analysis of environment for trust management is valuable for the implementation of soft security mechanisms for the real world. However, the digital world is broad and consists of many digital environments that differ, even when abstracted. Since each of these surveys only discussed one type of digital environment, it is not sufficient for a representative analysis of the digital world.

One of the key purposes of a survey paper is to breakdown existing literature. Yan Zhen's paper classified different models based on their primary goal. Then, they evaluated the models based on whether the models met the outlined trust objectives for IoT. Ruan's survey for online social communities surveyed different methods of understanding, computing, and inferring trust. Then, Ruan evaluated each trust model based on its vulnerability to common attacks that undermine trust.

Categorising existing trust models into different methods and goals and evaluating them is useful in figuring out the appropriateness of different models. Existing survey papers categorise the different models in meaningful ways and evaluate models from meaningful perspectives. However, in these papers, it is not always clear whether it is the approach, or the methods typically used in each approach that is insufficient. If the approach is insufficient, the approach should change. If it is the method that is insufficient, methods can be tweaked and improved upon.

There were some survey papers that took an attack and security-oriented approach towards trust modelling. Wang's survey evaluated different service-types for their security requirements [1]. Different attacks and some models that addressed these attacks were discussed, though this discussion was brief. Hoffman performed a much more comprehensive attack and defence survey for reputation systems [8]. In their survey, they presented a framework for decomposing reputation systems. In this system, the different system components and design choices that were vulnerable to attacks were discussed. Lastly, which defence mechanisms were most appropriate and how they could be incorporated into reputation systems for attack-resilience was discussed.

In both the above surveys, discussions about different security requirements are particularly valuable in understanding the extent of the trust research problem. The evaluations were also useful in giving an idea of appropriate model design choices for attack-resilience. However, in both survey papers, the range of attacks considered were limited. For a diverse environment such as the digital world, this range may be insufficient for implementation of secure trust management. Furthermore, more technical aspects of design choices were rarely discussed or evaluated. Therefore, these survey papers did not offer a theoretical, technical baseline to expand on existing design choices to make them more suitable for defence. Instead, they only considered the engineering, system-design direction.

There were a few technical, method-oriented trust surveys. Guo classified different trust computation methods based on different design dimensions [9]. They summarised the advantages and disadvantages of each dimension and highlighted whether they were effective against malicious attacks, particularly for IoT systems. A machine learning-oriented survey was carried out for trust management by Wang *et al.* [10]. Covering different digital environments and rating methods, they discussed the machine learning methods that have been employed in different models. Each of the models were evaluated based on whether they could address the rubrics outlined.

Surveys that take the method-oriented approach are useful because they offer insight into suitable technical methods for trust management. However, Guo's survey was brief and machine learning is often not suitable in many digital environments. Other technical methods may be more suitable, but these were not discussed in Wang's survey on machine learning. A more detailed analysis of the theoretical and technical basis for trust modelling methods would be useful to the field. Furthermore, while methods are important, the factors considered are just as important. Many of these models did not consider, in depth, the appropriateness of factors and methods chosen to model specific factors. Hence, it is not known whether existing methods of modelling factors are appropriate and whether choice of factors is suitable for soft security.

The survey on computational trust and reputation by Diego discussed trust and reputation broadly from a computational, theoretical perspective [11]. In their survey Diego provided extensive definitions and concepts of trust and reputation. Then, they created a schematic to assess computational trust and reputation models. Finally, they analysed research directions taken by different models in the field. However, not all surveys can be examined via rubrics due to different standards and definitions in different models. Furthermore, the lack of mathematical analysis did not offer insight into the best possible methods for individual digital environments. So, while Diego's survey is useful in examining state of the art in trust, it does not give much insight into the direction in which trust modelling should move. Furthermore, the theoretical basis of models was rarely discussed which limits understanding of the most fitting methods for trust modelling.

In summary, the premises of many existing survey papers do not address a broad enough scope for the digital world. Moreover, they do not offer insights into appropriate factors, modelling and evaluation methods that will be suitable for the digital world. For methods, the gap in theoretical analysis is particularly stark. Hence, it is still hard to find the best mathematical tools for trust. To address these gaps, our survey contributions are as follows:

1) We analyse digital world environments and abstract them to obtain the different types of trust that they require. Each type of trust has further interpretations that influence how trust evaluation. To demonstrate our interpretation, we give examples from the digital world.
2) Based on the several types of trust and using our understanding of each digital environments, we propose sets of factors that can be used within the digital environment. These factors cover a broad range of types to offer a holistic perspective on trust for better, well-rounded evaluation.
3) We also use our understanding of digital environments to come up with a broad checklist for different trust models in different digital environments, along with trust-management attacks. Each model can be evaluated based on whether it can check of each of the boxes. A model that can pass the criterion is not only secure, but also usable.
4) Finally, we look at different modelling methods and how they are used. By offering technical details alongside relevant models, we offer insight into the usefulness and theoretical suitability for different digital environments.

The rest of our survey paper will proceed as follows. In Section I, we discuss key concepts relevant to trust and trust modelling, even discussing some social theories and how they relate to the digital world. In Section II, we outline the different digital environments, specifying the unique challenges they each face. Using our analysis of the digital world, we categorise trust and provide factors that best model each type of trust in Section III. Next, in Section VI, we discuss different mathematical methods, the models that have used them and how they can be used in a general trust modelling framework. Lastly, we conclude and propose some future research directions in Section VII.

I. TRUST DEFINITIONS AND PROPERTIES

Trust is understood and used differently in different fields. In the humanities, trust and society have long been of great interest. In this section, we outline the most notable social theories of trust while exploring how they each inform digital trust. Then, we discuss digital trust, specifically its definition, properties, and some crucial related concepts.

A. SOCIAL THEORIES AND THE DIGITAL WORLD

Social theories about trust were pioneered by Simmel who contributed to the field in two ways. First, Simmel identified the function of trust, describing it as a force that works for and through human association, to bring society together [12], [13]. Second, Simmel explained its source, describing trust as a combination of inductive knowledge and faith [13]. Later, Luhmann expanded on Simmel's theoretical foundations. Luhmann explained that performing any action, no matter how basic, involved uncertainty and risk. Therefore, trust was necessary to assume at least the more unlikely risks were negligible, so individuals could function normally [4].

Luhmann's and Simmel's ideas have been adapted by trust modelling. In the digital world, every exchange, no matter how basic, carries some form of risk. Like in the real world, trust is needed to simplify the substantial number of uncertainties so that necessary digital tasks can be performed. Digital trust uses evidence and implicit knowledge about the digital environment for reasoning and decision-making, like social trust, digital trust uses inductive knowledge and faith. Finally, when a digital environment implements trust, agents can utilise a basic social mechanism to interact. Trust is a synthetic force even in digital communities.

Later, Bernard Barber described how expectations form the foundation of interpersonal trust — expectations that social mechanisms functioned properly and others were willing and capable of fulfilling their roles [3]. This is highly relevant to the digital world. In the digital world, the ability and willingness to fulfil roles determines the success of interactions. Therefore, incorporating willingness and capability in trust modelling of digital agents would reflect the decision-making patterns from the real world.

More modern examinations of trust emphasise its ever-present necessity. Giddens discussed the emergence of social systems in the modern world and how these were founded on and helped sustained trust [14]. Francis Fukuyama discussed the importance of trust for today's economic activities [15]. Most recently, Piotr Sztompka discussed trust from multiple perspectives [16]. These discussions tell us that as we digitize more and more of our modern social and economic transactions, social interpretations of modern trust need to be reflected in its digital counterpart.

While modelling trust in the digital world may not directly depend on social theories, observations about society are still highly relevant to the network-like digital communities of today. Bearing social theories in mind allows more realistic trust modelling. Doing so, however, is challenging as the humanities only describe trust in its qualitative, vague, and complex form. Practical applications of trust require the digital world translate conceptual trust into a tangible quantity. We do this in the next section by describing digital understandings towards trust.

B. DIGITAL TRUST

Digital agents exchange digital services and/or information, during which honesty and capability is needed. Digital trust is defined as a ''measurable belief and/or confidence'' that is ''accumulated from past experiences'' and is an ''expecting value for the future'' [17]. To explain, this means trust quantifies one's certainty via sources of evidence, such as

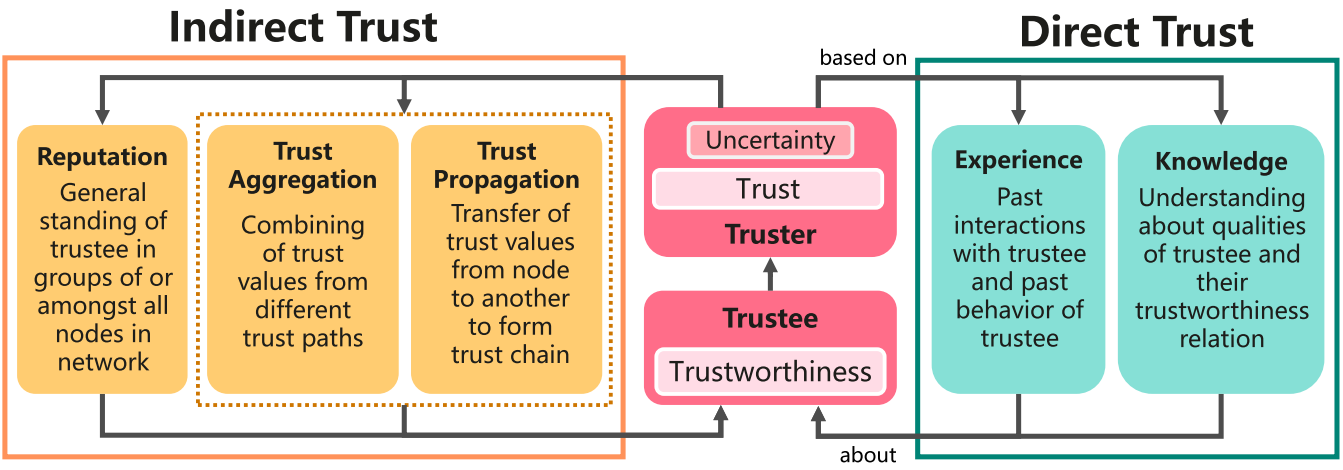


FIGURE 1. Flow chart of trust factors, concepts and agents.

experience. Evidence is accumulated and formulated into a prediction of future behaviour. In addition, there are several properties that need to be considered.

Subjective Trust levels differ between people [17]. Some digital entities are stricter, while others are laxer. Trust dispensed depends on each agent's own, potentially unobservable, preferences. Trust values and decision-boundary thresholds need to reflect individual preferences, not a universal standard.

Context-dependent Trust levels also vary with context [17]. Different digital environments invite entities with different, sometimes malicious, intentions. Within each digital environment, each interaction differs from the other in truster, trustee, purpose and other, observable, or unobservable, context features. Therefore, even within the most seemingly similar interactions, agents may behave differently.

Dynamic Trust tends to wane with time [17], [18]. As time passes, any existing knowledge becomes increasingly outdated. So, expectations about an agent's future actions become increasingly uncertain. Without overcompensating, trust models need to sufficiently reflect the decay of trust with time.

Transitive Trust is transferrable [17]. When a trusted individual offers recommendations, the truster's preceding trust in the recommender implies a trust in the recommendation. The transitivity of trust drives the formulation of indirect trust which will be discussed in Section I-B3. While trust may be transferrable, it should be noted that the extent of the transfer depends on the digital environment and individual agent.

Asymmetric Trust formed between truster and trustee is directed; the existence of trust in one direction does not imply existence in the other [17]. Therefore, a truster's belief in a trustee may not be reciprocated to the same degree or even at all. In some cases, asymmetry can be very stark. Where there is an imbalance of authority, certified authorities are likely more trustworthy but are unlikely to dispense trust as easily.

Easy to lose but hard to gain In all digital environments, caution is exercised to some extent, to ensure security. This caution means agents tend to choose interacting with trusted, familiar agents over strangers and when trust is betrayed, it is usually forfeited relatively quickly [18]. While greater caution naturally implies greater security, it also implies fewer risks. So, some opportunities with trustworthy individuals are naturally lost. A balance is needed to ensure the digital environment is safe but functional.

Pervasive Social theories from Section I-A indicate that trust is an inherent prerequisite to any interaction. Its necessity makes it pervasive. As social communities move into the digital realm, this pervasive nature is seen in digital communities as well. Digital agents will find it impossible to interact and function in the digital world, without trusting other digital agents. So, like in the social world, trust is pervasive.

In the next section, we define important terms and concepts. Figure 1 gives a broad overview of these sub-concepts associated with trust. Broadly, trust in a trustee is influenced by two factors: direct and indirect trust. Direct trust models personal knowledge about the trustee while indirect trust models the opinions of others. Both offer perspectives to form a more holistic view about the trustee. In the following sections, we will go into detail about each component.

1) TRUST AGENTS

In trust management, the modelled entity determines *what* is modelled. Typically, there are two entities of interest: the truster (the individual trusting) and the trustee (the individual being trusted) [17].

Seen in Figure 1, a truster dispenses trust to a trustee, based on direct and indirect trust, about their trustworthiness. For similar evidence, different trusters may decide to

trust differently. Trust propensity can be understood as the trustor's generalized expectation about the trustworthiness of trustees in general [17]. There are several influencing factors such as the level of security or degree of urgency. If an interaction were more important, for example when credit card information is being exchanged, higher trustworthiness would be needed from the trustee. If a particular service is needed urgently, trusters may lower their expectations. Trust propensity illustrates that trust is subjective. Contextual features are necessary to capture this subjective. Direct and indirect trust will be discussed in Sections I-B2 and I-B3.

Trust values assigned to trustees represent beliefs in the trustee's performance or behaviour. However, there may be uncertainty associated with the assigned value. This could be due to the reliability of evidence used; A trustee's past behaviour with the truster or with other recommenders does not guarantee their future behaviour. It is also possible that the amount of evidence is insufficient to be certain about the trust value of a trustee. A series of ten interactions provide greater certainty about trustworthiness than a single interaction. No method of modelling can fully capture trust. Trust is inherently subjective and vague. Representing trust quantitatively inevitably means missing some influencing variables. Trust values are therefore all inherently uncertain.

2) DIRECT TRUST

Trust values derived solely from truster's individual opinions are represented by direct trust. Such individual opinions are formulated from past experiences with the trustee which give information about their intentions and capabilities thus giving insight into their future actions. However, behaviour can fluctuate, intentionally or otherwise. Such fluctuations need to be accounted for in a non-misleading way using other knowledge about the trustee [18]–[23].

Experience includes past interactions and behaviour of a trustee [24]–[28]. Past interactions refer to direct interactions that have occurred between the truster and trustee and remove the need to rely on malicious recommenders. Typically, performance evaluation depends on interaction ratings, which are assumed to either be part of the environment or voluntarily provided by the truster [28]–[31]. Otherwise, binary successful and unsuccessful interactions are also used [18], [19], [27], [32], [33]. However, ratings may not be available in certain digital environments. Each interaction is also contextually different and potentially irrelevant. Contextual features help elaborate on the nature of the interaction so that its relevance to present day can be found and its contribution to direct trust adjusted.

Knowledge could refer to trustee features such as their communities, capabilities, and profiles. For example, the depth, detail, and content of a user's profile could help reveal their authenticity and intentions [24]. Device features, such as computational capability, have also been factored into evaluating trustworthiness of devices in networks [19]. Knowledge could also refer to contextual features. These include the nature and purpose of the interaction which indicate a trustee's incentive to perform well. Naturally, knowledge features need to be measurable, and any proxy would encounter the same issue of uncertainty that trust values do. Nevertheless, trustee features are a useful tool to build a more holistic view of the trustee, their capabilities, and intentions.

3) INDIRECT TRUST

For a truster to interact with an unfamiliar trustee, direct interactions are insufficient to draw reliable conclusions. Moreover, knowledge about trustees is not always available or accurate. Indirect trust is an added perspective to consider that instead, relies on the opinions of others.

Several trust models perform aggregate and propagate trust values throughout digital networks [18], [28]–[31], [34], [35]. Between any two nodes, there may be one or more intermediate nodes in which a directed path can be formed, where each node provides a trust value for the node after it. This path can then become a chain of reliable recommenders that results in input on the trustee, for the truster. This process of forming a trust chain is called *trust propagation*. Given that direct connections are not always available, this method of gathering information from surrounding, trusted nodes become useful to patch any insufficient information. After propagating trust values, each path's trust values are consolidated. *Trust aggregation* methods are needed here to consider which paths are trustworthy and to combine the different opinions.

Nevertheless, multiple opinions, numerous paths and the existence cycles make trust propagation and aggregation complex. When there are cycles or many paths, it becomes difficult for computational methods, iterating through the network and all paths, to converge. How to combine and infer the opinions of other nodes also depends on the application environment as recommenders may harbour ill-intention or unintentionally propagate inaccurate trust values. Some trust models have countered this by accounting for the quantity of evidence [21], [29], [32] and reliability of the advisor [19], [29]–[31], [36]. Path lengths are also a consideration as it is generally believed that the longer the path length, the more diluted the opinion [25], [28], [34], [35], [37].

Reputation is the general belief about a particular trustee. As in the social world, reputation can be derived globally, from all opinions [29] or from a select group of mutual acquaintances [30]. While methods of digital reputation may overlap with trust aggregation, rather than trust value inference, reputation is geared towards quantifying a node's surrounding structure and standing in a network — the more qualitative aspects that influence general opinion. For example, reputation covers network features such as clusters and connection types. Clusters of direct connections could indicate some latent connections between nodes; the large number of outgoing to incoming edges could indicate that an entity is randomly forming connections. Therefore, reputation adds an important perspective to indirect trust.

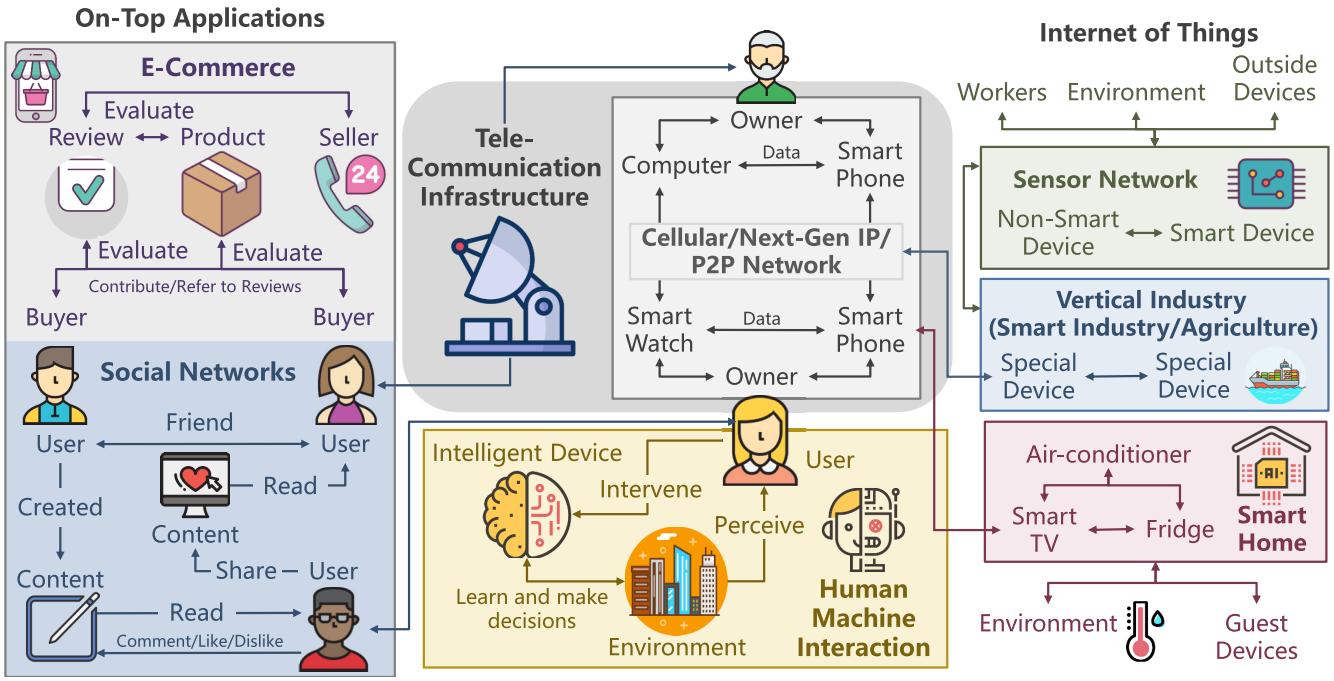


FIGURE 2. Outline of digital world applications and relationships.

II. DIGITAL ENVIRONMENTS IN THE DIGITAL WORLD

There are many diverse environments in the digital world, each with unique characteristics and needs. Figure 2 illustrates the most representative digital environments and highlights some possible intra and inter-environment relationships.

There are many diverse types of relationships, each with unique features, that occur within digital applications and span across digital environments. Each digital environment may also support interaction with non-digital entities from the social world and physical environments. In this section, we use Figure 2 to discuss the different digital environments, their relationships and the respective trust and security challenges they face. This discussion will help inform the requirements and criteria for environment-appropriate trust management as well as the different types of trust, how to model them and what factors to consider.

A. ON-TOP APPLICATIONS

On-Top Applications are upper layer type digital applications such as social networks and e-commerce platforms. In such free and open platforms, attackers can infiltrate and cause disturbances that make the network unconducive, driving users away [2].

For example, in social networks, content created or distributed could be misinformation [38], [39], spam [40] or harassment [41]. Harassment is extremely detrimental to the mental health of its victims. When spam is rampant on a platform, users get plagued with large amounts of irrelevant information. The spread of misinformation on social platforms may cause users to act on misinformation with real-life repercussions. Harm generated in social networks tend to be widespread as other users can absorb and continue to propagate such harmful information. Individuals need not even harbour malicious intent to help propagate damage. Trust is needed to manage interactions (sharing, liking, disliking, commenting and absorbing information) on social networks to reduce the impact of any harmful behaviour.

Harmful connections between digital users may also be formed. In some social networks, digital connections can access and receive updates about each other's content and personal information. Users may be granted privileges to directly converse with their connections. However, registration on most social networks do not require authentication so, agents can fake profile details. Malicious users can fake profiles to gain access to individual's private information or receive interacting privileges to harass or spread misinformation. It is then important for social networks to prevent these malicious relationships from being formed by deciding which users are trustworthy and which are vulnerable.

There are similar content-trustworthiness concerns on e-commerce platforms, particularly with reviews. Reviews that are dishonest prevent sellers from earning from their products and buyers from making good purchases. Such reviews could be written by competing consumers [36] or rival sellers to drive away competitors. While some reviews are clearly dishonest, there are more complicated contexts, with more grey areas. For example, reviewers may be disgruntled and or may not care about the quality of the review, resulting in partially true but potentially biased seller-ratings.

Reviews, however, are still important to prevent malicious and dishonest sellers from selling faulty or fake products. Therefore, reviewing, as well as buying and selling of products need to be protected by a trust mechanism. Trust measures are needed to disallow users from unfairly rating sellers or their products, without consequence. At the same time, reviews and ratings are also a trust mechanism to allow buyers and sellers to trust each other. In summary, trust management is useful for accurately reflecting product, seller, and reviewer quality, for all users to make good decisions.

Privacy is of particular concern in on top applications. User data is distributed on a large scale in social networks and often collected on e-commerce platforms. Distribution and collection of personal information on a large scale puts the digital environment at risk of violating the privacy of users and potentially leaking confidential information. The design of trust management systems thus needs to additionally consider safeguarding the transfer of large-scale personal data [17]. Concurrently, trust management schemes often rely on identity and behavioural data of agents. Information used for trust management must not infringe on user privacy and maintain confidentiality, while also ensuring security.

With cross-application communication, the effects of malicious attacks on social networks or e-commerce environments can extend to affect the functioning of and users in other networks. For example, in Figure 2 a user in some social network could communicate false information to someone in a smart city environment, which then negatively affects the functioning of smart cities.

B. CELLULAR, NEXT-GENERATION IP AND PEER-TO-PEER NETWORKS

Cellular, next-Generation IP and peer-to-peer Networks are some examples of communication networks supported by basic communication infrastructures. In such networks, communication may be less subjective and nuanced. However, other challenges exist. Devices vary in capability and service flows differ between applications.

A single owner could have their devices communicate with each other. For example, in Figure 2, a user could transfer video and audio files from their smart phone to their computer. During communication, each device needs to be sure that the data transferred has not been compromised. Uncompromised means the owner of the device indeed has control, has authorised the transfer of information and the information is not, by some mistake of the owner, harmful.

Data from one owner's device can also be transferred to that of another owner. In such situations, the devices have a different relationship than co-owner devices. Data can be sent by malicious attackers to negatively impact some unsuspecting owner. Such data needs to be differentiated from intentionally transferred data. Even recognised devices could also erroneously or maliciously send corrupted files. Trust management can consider the variety of application scenarios and adjust trust values to different needs for different situations.

In the above heterogenous networks, devices from a wide spectrum of capabilities can communicate. For example, in Figure 2, data is transferred between a computer and a smart watch, a smart watch and a smart phone. Trust management is needed to prevent the spread of potentially compromised data, regardless of the capabilities of the weakest device — the smartwatch. Trust management on networks need to be able to recognise malicious data and prevent its distribution, even if some devices involved are incapable of running large scale, computationally heavy algorithms [17].

C. INTERNET OF THINGS

Internet of Things (IoT) are networks of physical objects, integrated into information networks, to provide intelligent services. Physical objects include sensors, mobile devices, and monitors. They extract information from their surrounding users and environment [6]. In Figure 2, there are three primary digital environments: wireless sensor networks, vertical industries, and smart home ecosystems.

In IoT digital environments, devices communicate within networks. In a smart-home environment, air-conditioners, refrigerators, and other devices transfer data to facilitate intelligent home services. In vertical industries, special industrial equipment automatically communicates with each other to aid business operations. In wireless sensor networks, smart and non-smart devices communicate with each other to transfer collected data. Components within a network could be compromised when attackers launch malicious code to execute on IoT devices [42] or when devices have been physically tampered with, intentionally by an attacker or unintentionally due to environmental conditions. Primitive devices may be incompatible with newer technology and malfunction.

IoT networks are particularly prone to such attacks and accidental errors. In outdoor vertical industries and wireless sensor networks, IoT devices communicate with many device types, the surroundings, and personnel. When collecting data from the environment, its inherently complex nature makes it easy for devices to take erroneous readings. When devices communicate, data types and transfer modes may be incompatible and data packets are lost. Furthermore, exposure to outside persons makes it possible that the devices could be physically moved, or the data intercepted by malicious entities [43]. Trust is needed to (a) identify when a IoT device within a network has been compromised and (b) when information transmitted between devices is erroneous or malicious, due to for example, spoofing attacks [42].

Devices can also communicate outside their network via telecommunication infrastructures. Data is periodically transmitted from IoT networks to user devices so that users can for example, monitor their smart home ecosystems. Any attacks on such IoT networks can have several repercussions. If data transfer to outside networks is not well protected, wormhole attacks could steal personal information and device passwords [42]. When communicating with telecommunication infrastructure, denial of service attacks could disable

IoT networks or prevent their access to larger networks for necessary services [42]. Trust is needed for communication with outside devices to verify that devices will not steal information, reroute data packets or damage network functionality.

Challenges from managing different device types, mentioned in Section II-B, are particularly evident in IoT [17], [43]. Smarter, more capable devices consistently need to communicate with devices that are not primarily built to process data, such as sensors or refrigerators; Inter-environment communication means transmitted data tends to have largely different characteristics and requirements [43]. Trust management needs to consider such data compatibility issues to determine if a device is trustworthy by, for example, considering the device's capability of fending off attacks. At the same time, management schemes also need to ensure low computational overhead for particularly resource constrained devices like those seen in IoT networks [17].

Furthermore, the integration of inherently different physical, social and digital worlds pose unique security challenges [17]. Exposure to surroundings means IoT devices are more vulnerable to tampering, accidental and human errors. How IoT devices perceive information and whether data collected is reliable needs to be considered for trust. This means that besides malicious attackers compromising IoT and other digital devices, the presence of unreliable instructions from the social world and inaccurate data from the physical world also need to be considered.

One issue of great interest is the protection of personal data collected by IoT networks. In smart home ecosystems especially, IoT networks have gained entry to people's private homes and receive data from personal devices and information networks. Personal information needs to be kept confidential and privacy preserved, especially because information is widely distributed to other IoT networks and telecommunication infrastructures [43]. Simpler IoT devices, however, may not have the means to fend off against malicious interception of outgoing and incoming personal data. Entire IoT networks are then made vulnerable to privacy attacks. Trust management schemes therefore data flow. Simultaneously, trust management in IoT devices will require data about individuals, their profiles, and their actions. Trust management schemes need to consider from which devices data is collected, how it is collected and how it is transferred. Excessive collection of information will invade user privacy and make potential leaks much more damaging.

Challenges in trust management for IoT networks are wide ranging. Besides identifying damaging data packets, trust for IoT needs to consider device constraints and differences in data type; data collection and distribution within IoT networks and the trust management need to be compliant with confidentiality and privacy standards. At the same time, for a trust system to be functional, data packets still need to reach their relevant destinations to ensure the functioning of these IoT digital environments [6].

D. HUMAN MACHINE TRUST

Human-machine collaboration, where people interact with artificial intelligence and smart devices, is an emerging digital environment in trust. It has garnered interest for its usefulness in assisting decision-making and automating tasks and has already been employed in medicine, education, industry, and space exploration [44]. In human-machine collaboration, artificial intelligence in smart devices interact with people and the environment to take or propose appropriate actions [45], [46].

However, artificial intelligence is not always accurate and may make errors. Due to these errors, users may be tempted to distrust the machine and constantly override it, rendering the machine useless [45]. For a person to rely on machines and artificial intelligence, trust is needed. Trust allows people using machines to know when and under which conditions are machines trustworthy. This then allows the benefits from incorporating them into industries and service provision to be realised.

III. TRUST TYPES FOR THE DIGITAL WORLD

In Section II, digital environments, their need for trust and each of the unique sets of challenges they faced were discussed. Now, we consider how to formulate trust conceptually to best target each environment. By categorising trust into different types, we consider different scenarios, agent relationships and propose relevant factors. Figure 3 summarises the different types of trust with examples of their application scenarios and digital relationships.

A. PEOPLE TO PEOPLE TRUST

People to people trust is formed between any two people with some relationship. In such relationships, people may fake their identities, harass others, spread misinformation, spam, commit fraud, leak information, or infringe on privacy. With reference to Figure 3, to verify that trustees will not commit any of these acts, trusters need to *trust that a person is inherently good* or *trust that a person will not do harm*. This interpretation helps formulate factors that influence trust between people. They are outlined in Figure 4 along with the relevant security issues.

1) TRUST IN PERSON's INHERENT GOODNESS

People can trust that a person is inherently good. In this kind of person-centric trust, a truster believes that the identity and features a person presents themselves under is truthful and that they are inherently honest. In Figure 2 and Section II-A, we described how users makes ''friends''. To befriend others, users on online social networks need to trust the authenticity and features of a profile. On online job advertising platforms, employers and potential hires connect based on user profiles, features and identities. In such environments, trusters utilise general understandings to perceive trustees and form trust.

Knowledge about a trustee's traits and features describes them and who they are. This perception of trustees, combined with generalised expectations of people with similar traits,

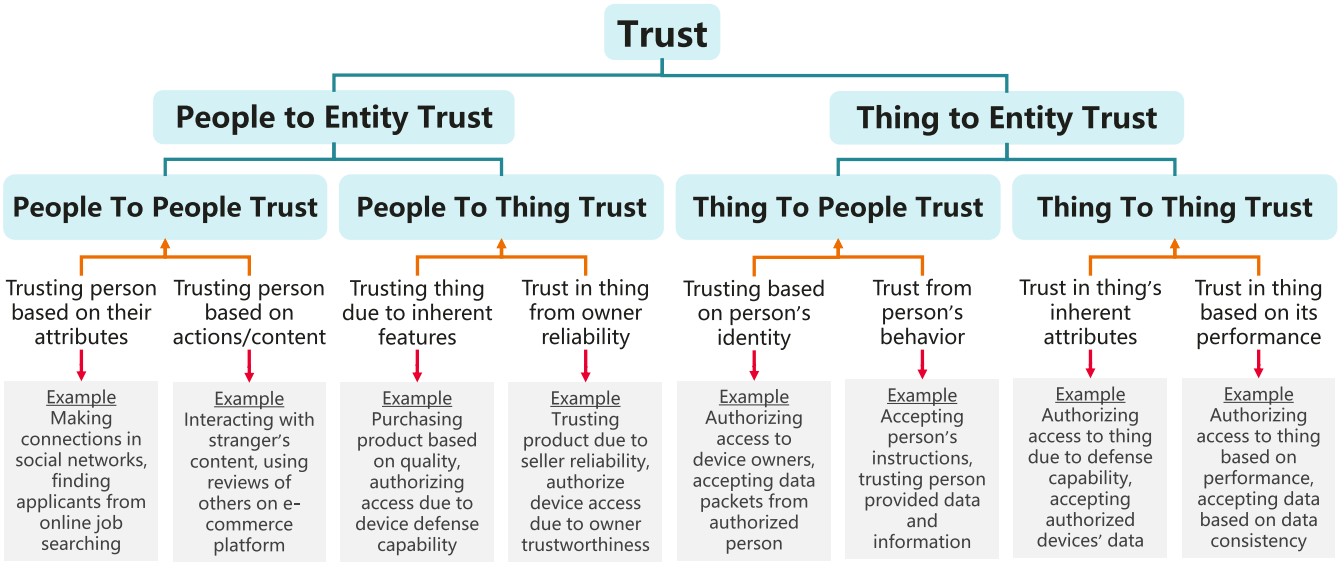


FIGURE 3. Organizational chart of different types of trust, how they are observed and examples.

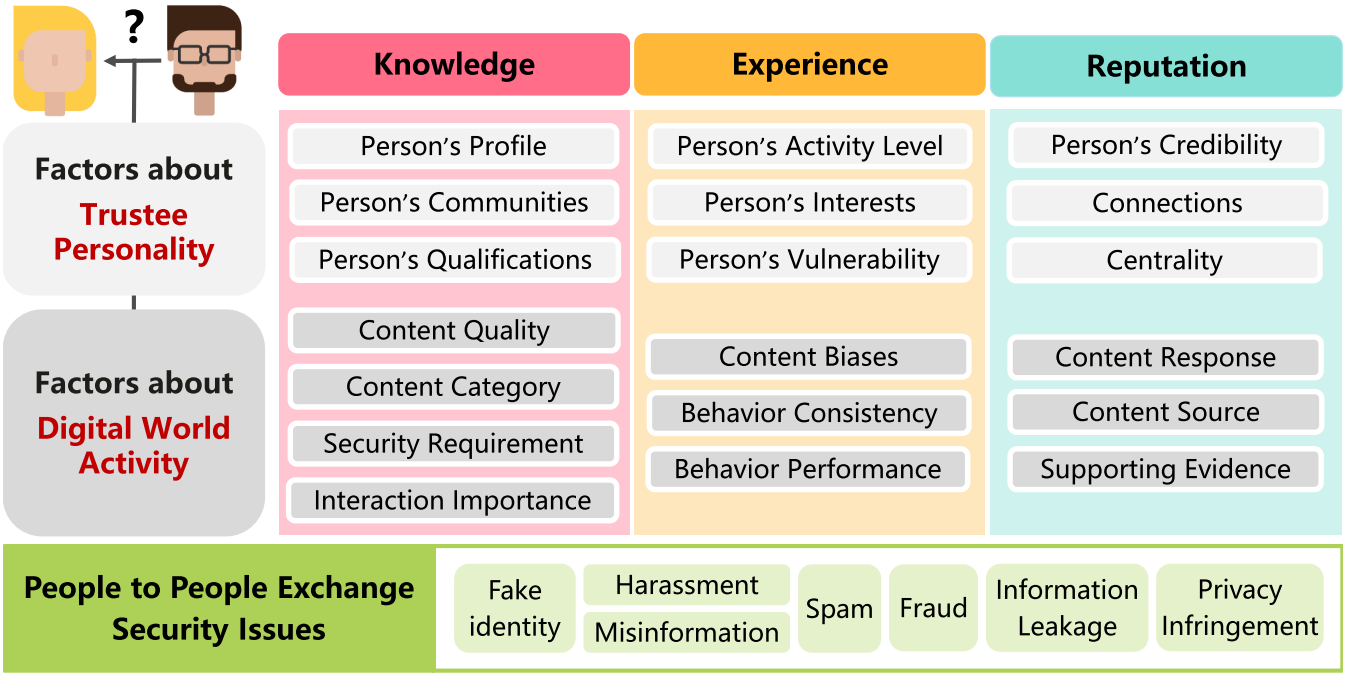


FIGURE 4. Overview of people to people trust factors and security issues.

help trusters understand whether the truster is inherently trustworthy.

Person's Profile On social networks, user profiles are widely available and can contain information such as a user's age, location, and write-ups about themselves. The quality of a profile can be very telling. Spam accounts lack incentive to invest much effort into profile creation so, the apparent amount of effort helps differentiate such accounts [24]. Profiles with many inconsistencies, suggest users are lying and dishonest. Profile features also partially describe trustees. Understanding trustee's personality and abilities gives insight into their trustworthiness.

Person's Communities Another way to describe a trustee is to use their communities and interests. Similar users are more likely to consider each other trustworthy [25], [47]. Knowing a trustee's communities and interests tells us which topics trustees are invested in, allowing them to

be paired with more like-minded trusters. For example, on social networks, if two users belong to the same networking group, they are more likely to share the same opinion. Therefore, they are more likely to find each other trustworthy.

Person's Qualifications In some cases, qualifications are relevant. A person's qualifications determines if a suitable connection can be formed for hiring on online job markets. On e-commerce sites, a reviewer can be particularly qualified to recommend certain products. With these qualifications, users directly perceive trustworthiness based on inherent characteristics. It should be noted that qualifications can be inherited from the real world, but they can also be implemented by trust systems based on environment-specific metrics.

To understand who a person is, trusters can also rely on past experiences with trustees. With experiences, trusters can gather evidence about a trustee's personality and reliability. This understanding of the trustee's personality directly determines their trustworthiness.

Person's Activity Level Activity levels indicate the level of investment in a digital community. Trustees that participate less, are less invested. A lack of emotional investment suggests users care less, so their activity could be more error prone. Activity patterns in different communities also inform which interests trustees are invested. Generalized expectations about people with certain interests form perceptions of trustees to different trusters.

Person's Interests Another way to describe personality is through interests. The topics that trustees respond to are an indicator of their level of interest in different topics. User interest can differentiate between trustworthy and untrustworthy trustees for specific trusters. For example, if users have a similar pattern of reviewing products on online markets or respond similarly to content on social networking sites, they likely have similar interests and so will share similar opinions [47].

Person's Vulnerability Trustees that are more exposed to or more trusting of damaging content are considered vulnerable. More vulnerable trustees have a higher likelihood of distributing and spreading harmful content. So, content passing through or originating from vulnerable nodes should be considered less trustworthy. At the same time, reducing a gullible node's trust propensity when they are trusters, reduces their exposure to damaging content. As trustees, they are then less vulnerable, thus they are more trustworthy. To measure vulnerability, the degree of absorption of malicious content can be measured, normalized or multiplied by a trustee's exposure.

Reputation factors quantify the opinions that other users have towards the trustee. With the opinions of others, trusters can gain a better insight into the trustee's personality, particularly their behaviour with other people other than the truster themselves. This patches any information the truster does not know about a trustee's personality.

Person's Credibility A user that is credible is a user with a good global reputation. Being credible can be measured by the structure and distributions of trust values surrounding a node. While specific methods to measure credibility may coincide with trust aggregation [25], [27], [28], [34], [35], [37], [47], global network-based indicators such as popularity or authority score, adapted from PageRank, have also been used [24], [25], [47], [48]. Rather than mere aggregation, PageRank-inspired popularity indicators consider the number and quality of incoming edges to measure credibility. Credibility thus ranks a user in relation to the entire network.

Connections Connections measure reputation using a similar underlying concept as credibility but restricted more locally. Connections of trustees and trusters are both of interest here. Connections a truster has is important because they form a group of people the truster believes. Therefore, a truster's community's opinion about a trustee would be important in helping a truster evaluate trustworthiness. A trustee's connections are also important because beliefs about a trustee's community are generalized to the trustee themselves.

Centrality Between a truster and a trustee, centrality is the degree to which a truster's network is central to the trustee or vice versa. For certain digital environments, if a truster's network is central to a trustee's, this suggests the trustee is important to the truster therefore, the trustee is more likely to behave favourably as their attention is less divided. Vice versa, if a trustee's network is central to the truster's, this could, in some digital environments, suggest that the trustee is already well-connected with people the truster is familiar with. Therefore, the trustee is already quite reputable amongst the truster's circle of friends.

2) TRUST IN PERSON's ACTIONS

Trusting that a person is inherently good may be too naïve in some cases. Sometimes, trust can only extend to believing that a person's current and future actions are not harmful. We have described content-type interactions between people in Figure 2 and Section II-A. On e-commerce platforms, users can evaluate reviews and decide that reviewers that have no incentive to leave harmful reviews. In social networking, users evaluate content to determine that content creators are objective. In both these cases, trusters interact with trustees because they believe trustees and their behaviour is not harmful.

To determine if a trustee is behaving in a non-harmful, objective, and fair manner, we require descriptive factors about a trustee's actions and how these related to a trustee's future trustworthiness. They are knowledge-type factors that describe a trustee's actions. In addition, we can also consider knowledge about trusters and their capacity to trust the actions of trustees.

Content Quality A trustworthy user's content reflects their quality. Quality can mean several things. First, it could mean factual accuracy. Comparing presented facts with known facts and counting the number of inaccuracies, we can determine if a user is objectively wrong. This method, however, naïvely assumes no grey areas. Second, quality could refer to presentation. In word-heavy digital communities, grammar, choice of words and punctuation reveal the amount of effort invested and emotions in a piece of writing. An excessively emotional piece could be biased, and writing riddled with grammatical errors and vulgarities are presentation-wise, like harmful content such as spam and harassment. In this sense, quality of presentation indicates content trustworthiness.

Content Category People's taste and preferences are indicative of their biases. The categories of content they typically engage with is indicative of this taste and preference. Understanding these potential biases, helps form an expectation of the trustee. We can then infer their potential biases in future related content. Then, without having to deterministically process any information, trusters can exercise caution with certain trustees about specific topics.

Security requirement Determining the trustworthiness of content on online social communities need not rely only on the specific trustee's activity. Online social communities can differ in policy and purpose, thereby attracting different kinds of people. For example, some online communities have stronger anonymity policies, reducing accountability and allowing users to be more irresponsible. Therefore, certain communities may see higher rates of malicious activity. Users in such communities need to exercise greater caution so, trust values should be universally lowered for tighter security.

Interaction importance Different users use digital services for different reasons. Some users may perform important transactions that have a large potential loss. For example, a user that is buying a big-ticket item on an online marketplace should be wearier of dubious sellers and dishonest recommendations than if they were buying something much cheaper. During such interactions, users need to be less trusting, meaning that trust values need to be lowered to fit the requirements of each interaction.

To evaluate if a trustee will or will not do harm, we can look at their past behaviour as a sign of their tendency to cause harm. A trustee that has been harmful in the past, has shown that they cannot be trusted to behave well in the future. To evaluate their potential future harmful actions, we discuss experience type factors.

Content Biases In online social communities, users lack incentive to be objective. For example, e-commerce reviewers who have had bad experiences may be inclined to give an exceptionally bad score, even if their experience was not objectively as bad. The pattern of behaviour, choice of words and manner of writing on online digital communities give an indication if a user tends to favour certain positions [47]. If a trustee tends to overreact, they are more likely biased so, they are likely less trustworthy.

Behaviour Consistency Consistency in a person's content reflects whether they behave in a consistent manner and thus, whether they can be trusted over time. A user that is more consistently good, provides a larger proportion of positive evidence, meaning their future behaviour is more certainly reliable. There are no instances of negative behaviour that may breed uncertainty in positive judgements. Alternatively, consistency can refer to informational consistency. If a user tends to contradict themselves within or between activities, this indicates the user is logically inconsistent, so they are untrustworthy.

Behaviour Performance While consistency determines if evidence about a trustworthy can give a certain outcome, the performance of a trustee determines what the outcome is. Obviously, if a trustee performs well, this performance supports that they are trustworthy. Otherwise, a user is untrustworthy. The magnitude of performance is particularly telling in people to people trust. Good content requires more time and effort. A more invested trustee is more trustworthy.

Finally, we use the opinions of others towards the action of a trustee to determine if the trustee's actions are harmful. By looking at how reputable the actions of a trustee are, we can determine if a trustee is likely to do be harmful. These are reputation factors.

Content Response Response to content is made up of reaction mechanisms such as liking, disliking, commenting and the pattern of propagation throughout the network. Such a method of evaluating content is beneficial as it borrows from the opinions of real people, who are better able to perceive nuances. As individuals respond, the manner of distribution, who the content is propagated to and who propagates the content, is recorded. Based on the similarity of distribution patterns, the nature of content can be inferred. By proxy, we can determine if the creator is trustworthy.

Content Source When content is distributed, the original source of information heavily implies the content's trustworthiness. If the originator of information is untrustworthy, it becomes highly likely the content is untrustworthy. That the trustee propagated untrustworthy information also reflects poorly on them.

Supporting Evidence Typically, to evaluate the factual accuracy of some piece of information, people will compare the information to sources they consider trustworthy. In trust, content can be directly compared to other sources. Ideally, these sources should be formal or highly regarded. Other sources of information act as supporting or disproving testimonies. The more

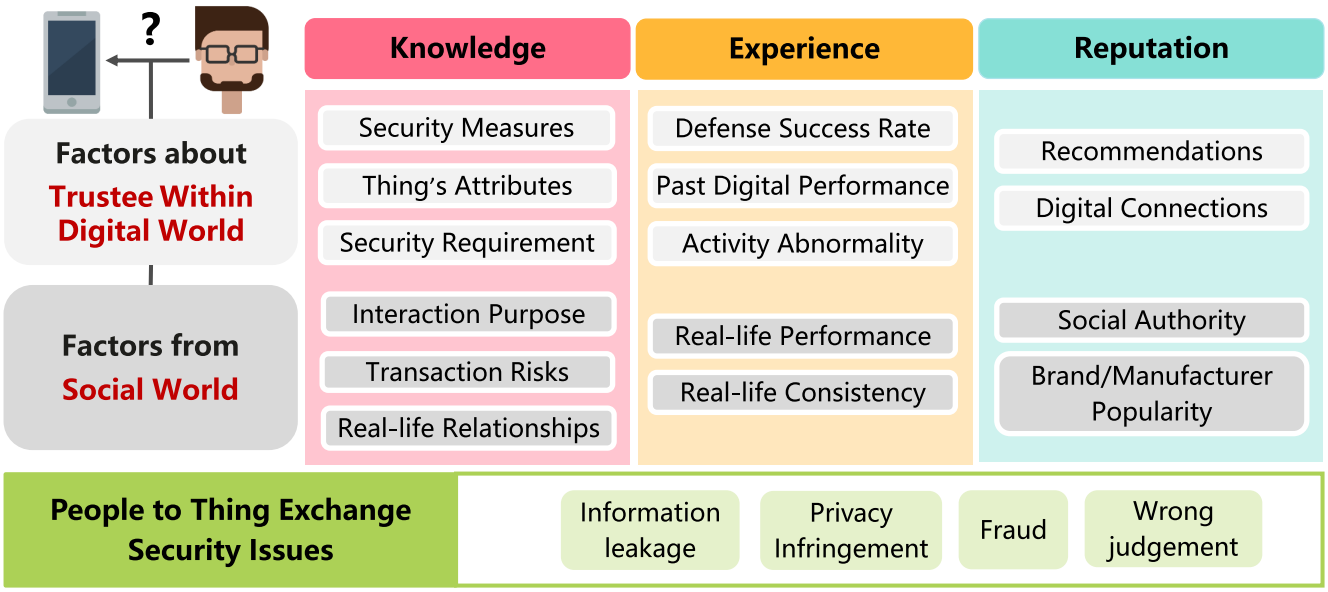


FIGURE 5. Overview of people to thing trust factors and security issues.

independent, trustworthy supporting testimonies there are, the more likely content is trustworthy.

B. PEOPLE TO THING TRUST

People to thing trust is established from people to non-sentient entities, such as devices, products, or information. People to thing trust crosses the divide between social and digital worlds; People trusters use social trust mechanisms to perceive trustee objects using only their characteristics presented within the digital world. In this dual-world, devices can leak information or infringe on the privacy of people; products can be fraudulent; devices can provide wrong judgements in human-machine interactions. To trust things, people can use their *digital world perception of the object* or *their real, social world perception associated to the object* to evaluate if the object is trustworthy. We use this understanding to propose some factors for modelling. These are outlined in Figure 5.

1) TRUST BASED ON DIGITAL WORLD PERCEPTION

Using their perception of the object in the digital environment, trusters can determine if it is of good quality. In Figure 2, we illustrated that a buyer in e-commerce markets needs to evaluate product trustworthiness. To do so, buyers evaluate the product based on its digital world characteristics. They may consider product features, brand, build and materials to figure out product quality; Users may use price to decide if a product is "too good to be true" or not worth the price, making purchase decisions based on cost. In peer-to-peer networks or smart home ecosystems, people interact with devices. Users may allow devices with strong defence capability to decide whether to accept incoming data. These demonstrate that people can trust in things using only the object's digital characteristics.

Knowledge factors can tell us if an object in the digital world have characteristics that make them trustworthy. This is also influenced by other characteristics of the digital world that may influence people's perception of the digital object.

Security measures Security measures refer to mechanisms deployed to fight malicious attacks. Measures whether deployed by trusters or trustees are both important. Trusters with capable security mechanisms can exercise less caution. In cellular, next-Generation IP and peer-to-peer networks (discussed in Section II-B), devices that have anti-virus software installed can trust other devices more readily. In e-commerce networks, a buyer can readily purchase products if they have buyer's insurance policies. Trustee's security measures are also important. Trustees that are not well-protected lack capability; capability in the sense that they are less able to provide non-harmful services to trusters with high success rate. For example, a device with no anti-virus software is more likely to transfer compromised files. Trustee devices should be less inclined to receive files from poorly protected devices.

Thing's attributes Obviously, thing features are important to determine trustworthiness. Trustworthiness entails that the thing can function as needed, without causing harm. Features considered differ depending on environment. In people to device environments, we can consider a device's build, brand, and computational capability to determine if they are truly capable of performing necessary tasks, without corrupting files. Whether a product is trustworthy, is determined by their quality. In people to product digital environments, the amount of information

available about the product, product-specific specifications, pictures, and materials give indication of authenticity and reliability.

Security requirement Since people to thing interactions can have very real implications, users generally have scenario-specific requirements to ensure an ideal set of outcomes for themselves. These requirements determine trust propensity. If users are engaging in a high-risk or high-cost activity, like exchanging substantial amounts of personal information or buying an expensive product, they are more cautious and less trusting. In scenarios where digital environments are ridden with dishonest users, trusters should trust fewer trustees to reduce their likelihood of meeting malicious agents. In this case, security requirements are higher so, trust propensity is lower.

Experience about digital world activity tends to apply more to device behaviour. Understanding the past behaviour of a device helps formulate a person's perception of how trustworthy and reliable a device is in the digital world. Digital devices that do not perform well are likely old, too primitive or compromised.

Defence success rate Devices with good defence success rate is resistant to malicious attacks. Being resistant means, it is unlikely the device has been compromised before. Therefore, when interacting with the device, users can be more confident there is no existing malicious software in the device that can steal information or cause harm. A good defence rate is also evidence that a device can deliver under malicious threat. Therefore, when interacting, users can be more confident the interaction will be successful even if there were any attempts at a malicious attack on the device.

Past digital performance Past digital performance of a device gives an indication of a device's ability to perform tasks and any potential malicious intent. Devices with good past digital performance means they have the computational capability to handle their assigned tasks, thus far. That a device can fully provide necessary services also implies it has not been compromised by malicious attackers to cause disturbances in the network. Therefore, good digital performance is indication the device is trustworthy.

Activity abnormality Finally, without external interception, devices should typically function as per normal. Abnormal behaviour in devices is indication that devices have been compromised. Compromised devices are more likely to cause harm and fail at providing services. Therefore, they should be considered less trustworthy until they are fixed. Abnormal behaviour can be measured by deviation in the device's behaviour from what is normal. For example, if an owner notices unusual power consumption, this could mean the existence of added malicious background software, especially if nothing was done to trigger such power consumption.

Lastly, we can use the opinions of other devices to adjust our perception of an object as a digital world entity. These opinions tell us which devices regard the device in what way. Based on other device perception, people can determine if a device is likely to have been compromised.

Recommendations Recommendations are useful in people to thing digital environments where users are not always familiar with all the devices and products they interact with. Having recommendations, users are better able to determine if a product or device is trustworthy based on the opinions of others. For example, in e-commerce markets, users might not buy the same product multiple times. Recommendations by other users about products then help users make better decisions about which products are more reliable. This has been discussed in detail in Section III-A.

Digital connections Devices and products can possess connections to other devices and products. These connections allow one to infer about the trustworthiness of the target trustee. For example, if a product is associated - by seller, brand, manufacturer, or any other feature — to a disreputable product, buyers should be less inclined to buy the target product less they possess the same issues as their connections. A device that has connected with compromised devices have been exposed to malicious or damaged devices. Therefore, there is a higher chance the target trustee device has been compromised. Again, users should be less inclined to trust target trustees with disreputable connections.

2) TRUST IN THING BASED ON SOCIAL WORLD RELATION

It is not always possible to evaluate an object based purely on its characteristics. However, there are often people intermediaries between trustees (things) and trusters (people). Trust in a thing can be inherited from existing trust in people. Examples are illustrated in Figure 2. On e-commerce platforms, the quality of a product cannot be fully verified. However, users can still choose to buy products anyway because a seller can be trusted to deliver authentic, true-to-picture and decent quality products. In the case of smart homes, guest devices can connect to home networks if the guest device owner and homeowner know each other. In both these cases, people to thing trust can be established because trusters have some form of guarantee. In the case of smart homes, this guarantee is not observable in the digital world.

Social world relationships influence people's decisions. However, social variables are latent from the digital perspective. We can use knowledge factors to understand device owners and their social world relationships.

Interaction purpose Interactions are generally undertaken with some goal in mind. Interaction goals aid trust modelling in two ways. A trustee's goal when taking part in interactions tell us how motivated they are to perform well. If users stand to benefit from an interaction, they are more likely to cooperate. For example,

trustee device owners have more incentive to behave cooperatively if the interactions will significantly up the reputation score of the trustee. Therefore, the probability of success increases. Second, purpose of interaction informs the level of access trustees should have access to. For example, to provide services, devices typically only require access to some, and not all, personal information. If devices request for say, full access, this should arouse suspicions. Otherwise, understanding the purpose of any interaction helps tailor the trust threshold for decision-making.

Transaction risk Some interactions carry more risk than others. In high-risk transactions, trusters should exercise more caution. For example, when buying expensive products online, trusters should exercise more caution in ensuring the quality of the product and the seller honesty before putting money down. Trust evaluation needs to be more stringent in high-risk cases to reduce the odds of large losses.

Real-life relationships Since social world relationships heavily influence people's trust in related devices, the existence and nature of real-life relationships naturally influence whether a person truster trusts a device owner. For example, when a device requests to connect to the network in a person's smart home ecosystem, it is ill-advised to simply accept a stranger device's request. However, if the two owners know each other, even if digitally, the truster is a stranger to the device, real-life relationships between the device owner and truster allows a trust relationship to be formed. Such real-life relationships should be considered wherever possible as they are influencing variables that cannot be seen in the digital world.

Like in people to people trust, experiences with social intermediaries are evidence of their trustworthiness. This trustworthiness is inherited by the associated thing as good experience with the social intermediary offers guarantee when interacting with the associated thing.

Real-life Performance Based on past experiences with an associated person, trustee things can inherit trustworthiness from their associated social world agents. Being confident in associated entities offers greater guarantee for success when interacting with the thing. For example, in e-commerce networks, trusters may be familiar with a particular seller, knowing the seller often sells high-quality and authentic products. They can use this familiarity to evaluate product reliability, even if it cannot be directly verified.

Real-life Consistency Consistency is relevant to experiences with intermediaries. If a trustee intermediary exhibits good but inconsistent performance, there is a greater level of uncertainty associated with their trustworthiness. Therefore, less trust should be assigned to the intermediary so, less trust is passed on to the object. This can be seen in online marketplaces. If a seller is inconsistent in delivering good products, trusters will be more cautious about buying from them, less the truster is unlucky and the seller fails to perform. Decisions to trust the product by buying it are then less likely.

Finally, products and devices are made in the context of the real, social world. Within the social world, things have attributes with social reputations tied to them. Using these related reputation factors, thing trustworthiness can be determined. We discuss them here.

Social authority If an object's intermediary has a good reputation, their increased social authority implies they are more trustworthy. This trustworthiness is inherited by the truster. For example, if a device owner is a verified government personnel, by formal standards, it is likely devices deployed by such personnel are more trustworthy. Therefore, even if the device is new to the network, they inherit social authority to become more powerful within the digital network.

Brand/Manufacturer popularity Trustee things may also possess attributes related to the social world. Trusters use their generalized expectations to form an understanding of the thing based on its attributes. For devices or products, the most relevant attribute would be brand or manufacturing popularity. People form expectations about brands and manufacturers based on hearsay and their own experiences. If a brand is known for producing high quality goods, so long as the goods are truly from the seller, trusters are more likely to purchase the branded goods as quality is, to the truster, guaranteed. If devices produced by certain manufacturers are known for being secure, users may use this understanding to choose devices by such manufacturers. Associated real world understandings of intermediaries thus influence object trustworthiness.

C. THING TO PEOPLE TRUST

In an increasingly digital society, people become agents in device networks. Devices receive information — instructions or data — from people and must make trustworthiness evaluations about whether the information is harmful i.e., whether it is misinformation, virus, or spam. Devices also must determine if users will infringe on privacy by excessively accessing private information stored on the device or leaking any information they access. It should be noted that devices are less able to perceive nuance and cannot rely on intuitive, social notions of trust. However, users still leave a digital trail in device networks with trends and patterns. Devices can use this to determine if people are trustworthy in that *they are authorised* and if they *are causing or mean to cause harm*. Figure 6 gives an overview of factors that we considered based on these two perspectives along with the above security issues.

1) TRUST IN THE PERSON's AUTHORISATION

First, devices need to verify if users are authorised. In Section II-B and Figure 2, we discussed that owners have

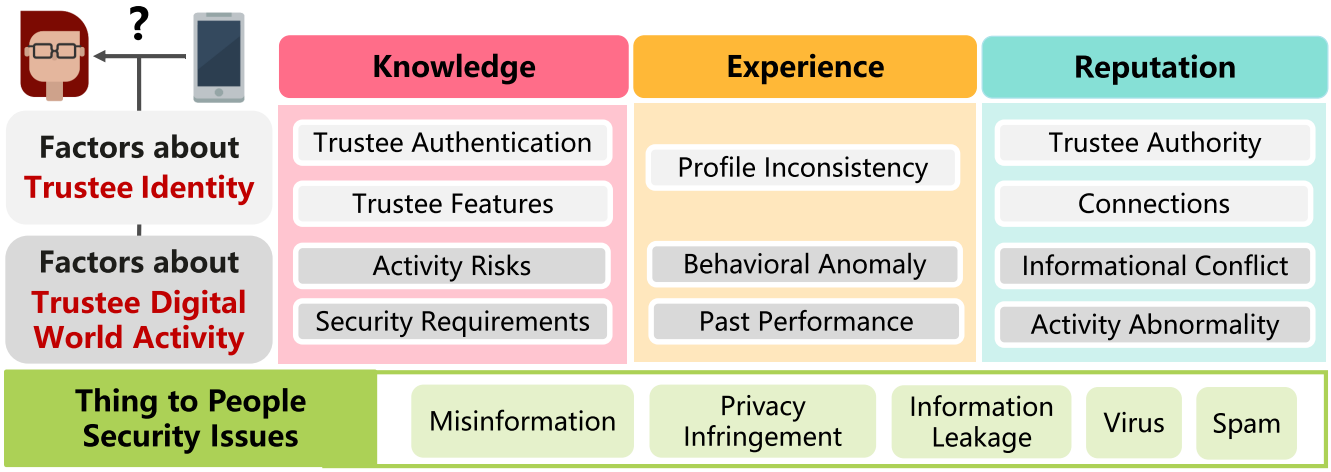


FIGURE 6. Overview of thing to people trust factors and security issues.

usage relationships with their devices, while in vertical industries, personnel interact with devices. Only non-malicious and real users should be able to pass authentication to gain access. However, passwords can be leaked, and users can become malicious. Trust management can help devices detect if a user is malicious in nature. If a user has a malicious identity, they should not be given access.

Knowledge factors elaborate on a truster's identity. Using this understanding of the trustee's identity, devices can identify if the features of the user are unusual or harmful. Unusual or harmful behaviour indicates the user is in fact, not authorised to use the device.

Trustee authentication A first obvious step to determine trustworthiness is to require authentication. If a user cannot pass authentication, they are not authorised to access the device because their identity is inauthentic, or they have malicious intentions.

Trustee features Trustee features possessed by device users can also give clues about whether a user is unreliable. These are features tied to the identity of the user. For example, if personnel in a vertical industry network happens to have low rank within the plant but is requesting a unusually high level of access, this could suggest the person trustee has gained access to the device for malicious purposes.

Every user has identifying features in the way they use devices and what information they store. Experience factors determine if identifying features follow an unusual pattern.

Profile inconsistency A user typically has identity-related features stored within a device. However, through usage, additional data is collected. This data could be inconsistent with previously stored data. Confirmation would be useful here to determine if the inconsistency was malicious or accidental or well-intentioned. If a user's identifying features are inconsistent with before, the same level of authorisation cannot be granted less the user has changed.

Reputation factors describe how people trustees are connected to the rest of the digital world. These connections inform if the trustee has a malicious identity in the digital world based on their connections with other devices in the network.

Trustee authority Trustee authority is important in digital environments like vertical industries where multiple users may access a single device. Trustees of different authority level have different levels of access within a network. A trustee that attempts to access a level beyond their workplace authority should be flagged so their behaviour can be accounted for with other potentially suspicious behaviour.

Connections Trustees are socially connected in the real world. These connections may be used to authorise access to devices. For example, a friend may unlock their personal devices for their friends to use temporarily; Low level personnel may be given access to devices temporarily by other workers. These situations should be accounted for to avoid wrongfully writing users as untrustworthy.

2) TRUST IN THE PERSON's BEHAVIOUR

Devices should trust users based on their behaviour, especially the typicality of behaviour and harmfulness of actions. For example, in vertical industries illustrated in Figure 2, devices can receive malicious instructions to perform harmful actions, or they can receive harmful or incorrect data. Devices that receive malicious instructions could damage infrastructure; misinformation could affect device's computation and output; harmful information, like spam, jams the network and service-systems. The more harmful a user's actions are, the less likely they are trustworthy. It is possible the current user is not the real device owner, has been given access but has malicious intentions or the device owner is making a mistake. Devices should defer from executing unusual instructions until further confirmation.

Devices can determine if user's behaviour is atypical or harmful using knowledge factors. Knowledge factors aid experience evaluation to enhance trustworthiness evaluation. Understanding the environment and activities undertaken in the environment, give insight into how potential harmful or atypical behavioural patterns of trustees are.

Activity risks When determining if the actions taken by a trustee are trustworthy, it is important to consider the level of concern required. Some actions have bigger impact than others. For example, in smart agriculture, if an exceptionally large amount of water is to be released into the plantation, this presents a large risk of flooding the entire plantation and damaging all the plants. With such extreme and harmful actions, greater caution is required so the device should require additional verification to make sure the action is well-intentioned or defer the action completely.

Security requirements In some digital environments, the general level of trustworthiness could be very low. The sample space of users has a larger number of untrustworthy users, so the probability of encountering an untrustworthy user is higher. Higher security requirements should be standard for all trustees to reduce the chances of an untrustworthy person gaining access to the device network.

Trust in a user's behaviour requires evaluation of their activity. Experience factors are necessary here to evaluate based on experience if a user will or is behaving harmfully or unusually. This helps verify their identity and that their identity is non-malicious.

Behavioural anomaly Behavioural anomaly is an important factor to consider. Some devices store activity history and can track behavioural patterns of users. If a user begins to exhibit anomalous behaviour, this suggests the device has changed hands. If there is no reason for this change, the authenticity of the user should come under suspicion. For example, if devices record sudden traffic to highly unusual, particularly dangerous sites, this suggests a malicious user has hacked into the device. This is also true of vertical industries where multiple personnel can interact with devices. If personnel make an anomalous request that could damage the device network, caution should be taken to ensure personnel are not malicious or making errors. Alternatively, since devices can also record data from people, if data deviates from trend, the trustee is potentially untrustworthy. For example, if a device receives data about traffic congestion from a user that is a significant distance away from the reported location or the reported traffic congestion occurs at a highly unusual timing, the data and trustee should be unreliable.

Past performance Like in other types of trust, performance also plays a role in determining trustworthiness. Performance determines how capable and willing a user is to handle devices properly. For example, if a user tends to download viruses easily, it shows the user is incapable of identifying and avoiding bad links or using the device maliciously. If personnel often log incorrect data, them, and any future data they log could be flagged as untrustworthy.

When users release information into device networks, information from within the device network can also give information about how harmful and incorrect the information is. This sort of evaluation relies on reputation factors about the trustee's behaviour.

Informational conflict In digital environments where information is collected from multiple users, data about an observation or instructions at any time can come from multiple people. If one person's information is conflicts with that of a large majority of other people, this information and trustee should be flagged. Alternatively, if data is corroborated by a reliable person or device, this indicates the information is highly reliable so, the trustee demonstrates good performance and is reliable.

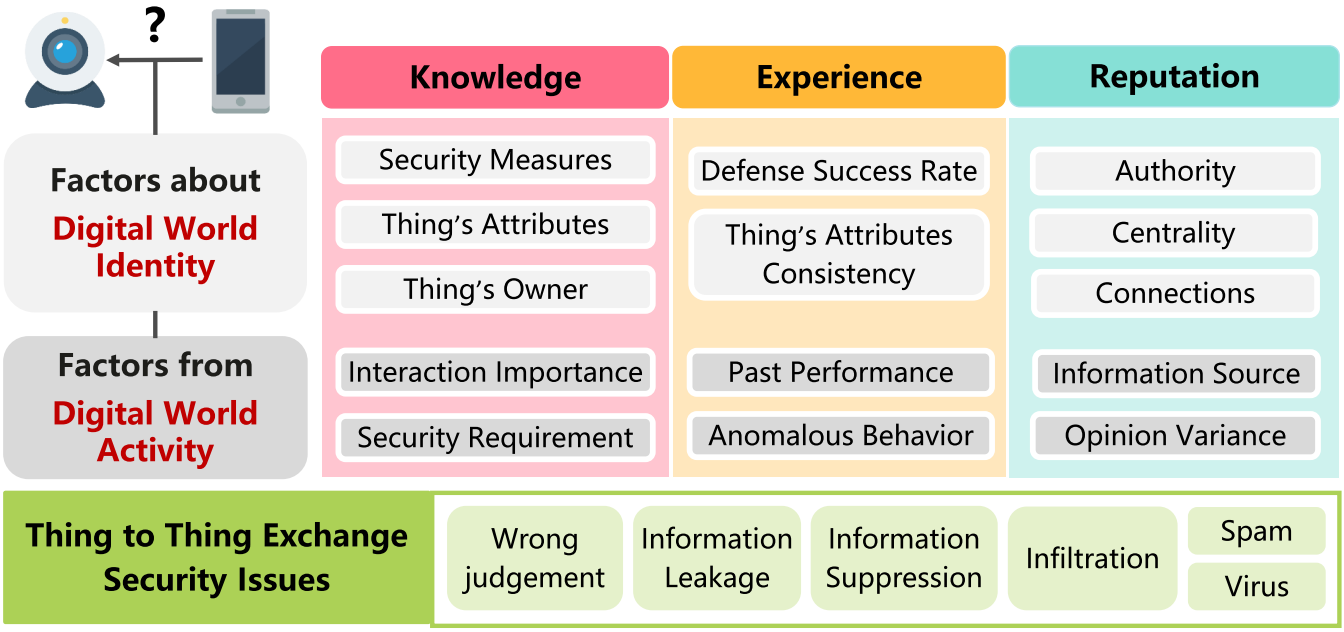
Activity abnormality Activity abnormality uses generalised information about user behaviour to determine if a particular user's behaviour is suspicious. For example, if a user downloads software from an unusual site where users typically do not download such software, devices can flag this behaviour as unusual. The device can then look out for more signs that may suggest the user is untrustworthy and intends to behave in a malicious manner.

D. THING TO THING TRUST

In many digital networks, devices pass information on to other devices to form a large network of automated communication. In these networks, devices can transfer poor computational output to other devices, leak personal information to other devices, fail to pass information on to other devices or pass on harmful information such as spam or virus. To do any of these things, malicious devices infiltrate the network. Thing to thing trust is trust between two non-sentient entities. To address security challenges in device networks, thing to thing trust is needed. Devices need to believe the trustee thing is *inherently trustworthy* or is *at least trustworthy within the impending interaction*. Factors for each of these instances is outlined in Figure.

1) THING's INHERENT TRUSTWORTHINESS

Devices can evaluate if trustee things are inherently trustworthy. In Section II-B, Section II-C and Figure 2, devices supported by widespread telecommunication infrastructure and open device networks, can receive data from unfamiliar devices. Malicious devices can exploit this openness to infiltrate networks and cause disturbances. For example, they can send a virus-infected file to a node in a wireless sensor network or use communication infrastructure to randomly attack personal devices in peer-to-peer networks. Truster devices thus need to ensure that trustees are inherently trustworthy and do not have intentions to carry out malicious attacks.



To determine if a device's identity is trustworthy, we can use knowledge factors. Knowledge factors give insight into the device and its surrounding social or physical environment. These give insight into the purpose of deploying the device and in turn, its future actions. In addition, knowledge about the digital environment also provides insight into the likely identity of the device so we consider those as well.

Security measures Whether a device is secure against malicious attacks determines its defence capability against malicious attacks and incidents that compromise its performance. What security measures are employed determine this security. For example, a device that has anti-virus software should be able to defend against viruses transmitted from other devices. Therefore, they are less likely to pass on the virus to other devices when transferring files if they have never been compromised. This goes both ways. If a device lacks the appropriates software to defend against malicious attacks, the device should set higher requirements for trustworthiness values to exercise more caution.

Thing's attributes A device's characteristics also give information about whether it is capable enough to be considered trustworthy. A device that is older and more primitive might be more prone to mistakes and vulnerable to attacks, particularly if its software is not regularly updated. Therefore, the information from the device is less reliable, making the device less trustworthy.

Thing's owner A thing can also be connected to an owner. If two owners know each other, even the two devices have never interacted, a relationship can still be established between the two devices. For example, if two device owners wish to share an Internet connection, the two devices can connect via tethering even if the two devices have never connected before. That the two device owners are friends is a characteristic that influences mutual trustworthiness of both devices.

Experience factors provide evidence about the pattern of behaviour, any anomalies and exposure history of trustees. Using this information, trusters can infer the identity of a device — whether it is capable or willing to cause harm. Inferring this provides information about device's future actions and thus, their trustworthiness.

Defence success rate Despite its best efforts, devices may find that they still encounter malicious attacks. These attacks could compromise the device making it such that the actions of the device affect other devices it interacts with. Then, the success rate of the device and whether the device has been compromised in the past becomes relevant before choosing to interact with it. For example, if a device has experienced information leakage before, a device that wishes to pass information through that node may choose not to do so; if the cause of the information leakage is still present, it will leak the truster device's information.

Thing's attribute consistency Another experience feature about device identity is whether the attributes of the device remain consistent with time. If a device has a particular characteristic, this characteristic needs to be demonstrated with time and its various actions. Otherwise, the device is behaving unusually. For example, a device such as a laptop has high computational power. However, if it often fails at simple processing tasks, this could suggest the laptop has been compromised so, it is behaving oddly.

Finally, we look at reputation features that give insight into the intentions of the device. By relying on reputation, devices overall network behaviour acts as more evidence in

case current experience is insufficient to reliable infer about a device's intentions.

Authority In some device networks, selected devices have greater authority. They may be large professionally deployed communication infrastructure, government deployed units or officially authorised nodes in the network. Since these devices are verified, the information they distribute is considered instantly and definitively trustworthy. For example, in vehicular networks, vehicle nodes can directly trust road-side units (RSU) because they are officially deployed to disseminate important information about traffic conditions.

Centrality Centrality is important due to the network-nature of device networks. Between two devices, there are common devices that both truster and trustee have interacted with. Depending on the trust values and the proportion the common to total acquittance devices, how information flows between truster and trustee device can be visualised. If there is a sizeable proportion of common trusted acquaintances, the indirectly flow of reliable information between the devices is significant. So, just like in people to people trust, centrality also matters.

Connections Connections influences the formation of trust between devices in two ways. First, since trust is transitive, a truster can use a trusted node to exchange with a trustee, forming an indirect connection for information exchange. Second, if a device has formed many disreputable connections, they are likely to have been compromised on multiple occasions or are also a malicious device. Therefore, the device is unwilling or incapable of performing well in an exchange, making it untrustworthy.

2) TRUST IN BEHAVIOUR HARMFULNESS

Devices often do not have sufficient information about the inherent trustworthiness of a device based of its identity. An alternative would be simply to evaluate if a device's incoming action is trustworthy. This is analogous to determining if a trustee will do harm and requires evaluation of the trustee's past and current actions. In Section II-C, it is hard to determine if a device in an outdoor wireless sensor network is inherently untrustworthy. Even if the device is originally trustworthy, at any time the device can topple over and record data wrongly or be physically manipulated by a malicious entity to send inaccurate data. In such environments, no device can be inherently trustworthy. Instead, truster devices can only evaluate if the incoming data and interaction with the device is non-harmful.

To carry out such evaluation, knowledge factors about the interaction and environment are useful. Devices can infer the likelihood of a harmful interaction and to what extent the interaction would be harmful. This aids the decision to continue with an interaction.

Interaction importance Whether an interaction is important significantly affects the degree of caution the truster should employ. If a truster device is exchanging sensitive data, the trust threshold should be much higher to ensure that trustee nodes do not either fail to pass on necessary information or pass on misinformation. Data with a substantial impact on the real world can also be considered important. For example, if a device such as a vehicular node releases information about traffic conditions, this information will direct traffic along different roads. If the information is wrong, intentionally, or otherwise, this could cause significant traffic congestion. Therefore, only highly trusted nodes should be allowed to release such high-impact information.

Security requirement If there are many untrustworthy nodes in a network, truster devices should also exercise greater caution. This applies like discussed before since most device networks are a type of network in some form or another. Therefore, each node interacts with other nodes in the network. If the network has a large proportion of malicious nodes, it is more naïvely probable that a randomly chosen node is malicious. Therefore, extra caution needs to be taken during any interaction.

To determine if a current interaction will be successful, it is useful to gather past evidence about the trustee device. This requires experience factors which indicate whether the device has been compromised and whether it will behave harmfully.

Past performance Naturally, like in all types of trust, the performance of a trustee is important. A trustee's performance directly reflects their willingness and capability to cooperate in exchanges. If a device often supplies erroneous information, it is likely the device is either compromised or too old so, it can no longer be trusted. It would thus be better for the entire network to simply ignore the device as its past performance does not bode well for the device's future performance.

Anomalous behaviour Since "things" refer to devices, which while not static, cannot interpret nuanced, subjective information, humans rely based on the trend of behaviour to determine if a device can be considered suspicious. If a device typically behaves in a certain way, any behaviour that does not follow this trend suggests a change in its surroundings or tampering by some party. If there are no events that may suggest this change, this should arouse suspicious that the device has been compromised.

Finally, to evaluate if actions from a device are harmful, we can rely on the behaviour and information being transmitted. In particular, the reputation of the action within the network is very telling in determining if the incoming action is trustworthy, even if not much can be directly known about the trustee or their behaviour.

Information source Since devices form parts of a network, the information typically travels in paths and cycles. This means that information can be tracked to its original location. If the original source of the information is unreliable, this could suggest that the information is,

by proxy, unreliable. Furthermore, the act of propagating this information to the truster suggests the distributing device has poor connections, reducing their trustworthiness.

Opinion variance Finally, trusters can receive information from multiple devices in a device network. Potentially, the device may receive similar information from multiple sources or if a group of devices all manage the same network, the data collected from all devices need to be logically consistent. This data collected as a group acts as a type of corroborating evidence. If the data collected by the trustee device deviates too greatly from what is expected, this suggests that the information and thus, the device is untrustworthy.

IV. CRITERIA FOR DIGITAL TRUST MODELLING

In Section II, we laid out the different digital environments and mentioned some trust modelling challenges. In this section, we discuss different rubrics and criteria to address basic trust modelling targets and these additional challenges. They are outlined in Figure 7.

A. SECURITY

A good trust model is secure meaning it is safe for use and can defends against threats, attacks, errors, and back doors. There are many requirements for a trust model to be secure.

Privacy A secure trust model preserves privacy [1], [6], [10]. This means it allows users to select the type, method and who can access their personal information. Trust management schemes must be able to prevent the invasion of privacy by malicious attackers while itself minimising the use of personal information while managing trust. Entities within the network *and* trust models should not be able to retrieve user's information without permission of the user. Privacy is necessary to the security of the trust model as lack of privacy could make users feel uncomfortable, resulting in harm, even if the company has no intention of misusing the data. That the user cannot confirm such misuse will not occur is sufficient to cause distress.

Confidentiality Confidentiality is a concept highly related to privacy and is too, necessary in trust management [5], [6]. Confidentiality involves the prevention of excessive collection and leakage of personal data. Like in privacy, this means that both the digital service *and* the trust model should keep confidentiality. Maintaining confidentiality means trust management should only collect relevant data and cannot leak data to any entities outside those that absolutely need it. Personal data leakages are harmful to users as they could reveal sensitive information, causing real life harmful effects on the user.

Availability A trust model is secure if it is available whenever needed [5]. This means that attackers are not able to cause disturbances to the network such that users are not able to access its services. Trust is necessary here to filter interactions that try to jam and reduce availability, such as spam and virus attacks. The process of trust modelling also cannot compromise the performance and capacity of the network.

Transparency Trust management should be transparent such that its processes and the information collected are open to each user. The system also needs to be accountable for any failure to protect digital services and errors made during the trust management process. Transparency also means that there should not be any back doors. A lack of transparency in managing trust makes users wary of digital environments.

Integrity Integrity is an obvious requirement of any trust management system [5], [6]. A trust management that has integrity can prevent malicious or harmful interactions and behaviour. Trust is needed to decide and measure when an entity is malicious. Then, these entities need to be punished or removed accordingly. Integrity protects the digital environment and its interactions from meeting bad interactions and excessive harm.

Non-repudiation Non-repudiation refers to holding all users accountable for their actions [5]. So, when malicious or harmful interactions occur in the network, the relevant perpetrators can be tracked and punished for the interaction. Trust is needed for this as the trust degree should track these harmful behaviours and accurately reflect them in numerical values. Entities with low trust values should not be able to interact in the environment.

Authenticity Non-repudiation and its implementation is intricately linked to authenticity. Authenticity means that user's identities are verifiable. Being able to authenticate users prevents them from being able to take on multiple identities to take the blame for their harmful actions in the network. Authenticity of the network ensures that trust values and the management system is in fact effective. If users were able to create false identities when their trust values fell too low, the trust values would have no meaning. Therefore, authenticity is a necessary requirement for all trust models [5].

B. COMPREHENSIVENESS

Most trust models only partially consider comprehensiveness. A comprehensive trust model considers all the different dimensions and aspects of trust modelling and can adjust to account for these aspects. Comprehensive trust models should fulfil the following requirements.

Dynamicity Dynamic trust models are models that consider the time-varying nature of trust and are necessary as trust is itself, dynamic [7]. Many models are dynamic by considering the age of evidence, recommendations, and trust values. Older evidence is less reliable as it is less reflective of the entity's future behaviour. However, writing off old evidence can result in a lack of evidence.

Certainty Trust, being an inherently complicated concept, carries an inherent uncertainty, as discussed

Security	Comprehensiveness	Usability	Functionality	Robustness
Privacy	Dynamicity	Computationally Efficient	Corresponding Access	Network Disturbance
Confidentiality	Certainty	Data Usability	Service Oriented Access Control	Propagation Errors
Availability	Trust Degree (Trust/Neutral/No Trust/Distrust)	Considers Node Diversity	Continuity of Access Rights	Cold Start Problem
Transparency	Context-Aware	Usable in Different Networks		
Integrity	Subjective			
Non-repudiation				
Authenticity				

FIGURE 7. Table of rubrics categorised into general concepts.

in Section I-B. Models that are comprehensive should consider this uncertainty in its trust values and decisions. This means that when evidence is lacking or contradictory, the metrics used — be it the trust value itself or a separate certainty indicator — need to reflect this and factor it into trust decision-making.

Trust Degree A comprehensive trust model obviously requires some representation of trust [7], be it binary, discrete, or continuous. However, since trust is complicated this trust degree must also reflect nuanced understandings of trust. Trust models should also be able to account for the differences between ''no trust'' and ''neutral'' and reflect them accordingly in the model. Moreover, ''no trust'', ''neutral'' and ''distrust'' are not necessarily the same. ''Distrust'' implies that the user is repelled by the node while ''no trust'' or ''neutral'' could simply be that there is insufficient information or conflicting information respectively about the user's trustworthiness.

Context Aware Since trust is by nature a highly situational concept, trust models need to consider the different situations and how they affect trust [6], [10]. Not only do they have to consider the different situations, but they also must do so appropriately, choosing suitable contextual features that inform a node's trust the most. A trust model that can consider this is holistic and sufficiently comprehensive for implementation in real life.

Subjective As mentioned, trust differs from person to person. Truster's requirements need to be holistically considered, factoring in all the relevant truster features to decide the most fitting trust value and decision for the truster. A trust model that can adjust to fit its users' needs is sufficiently comprehensive for all users [7], [10].

C. USABILITY

Many trust models today claim to accurately detect malicious users and attacks; however, few models consider whether they are usable and implementable in real-life digital environments. We break down how trust models can become usable for the real world.

Computationally Efficient As mentioned in Sections II-B and II-C, computational efficiency is a huge limiting factor in trust model performance. Therefore, trust management schemes need to carefully manage time and storage complexity as well as the convergence of algorithms [5], [8], [10]. However, complex, and time-consuming algorithms are still usable in the right digital environments. Networks that have primitive devices cannot utilise high computational consumption algorithms but trust models for online social networks, that utilise large, central servers and processes might be able to.

Data Usability As mentioned in Section I-B2, ratings, trust values and other indicators are commonly assumed to exist in digital environments. This is not necessarily true. Furthermore, feature based knowledge modelling in Section I-B2 may not be able to access the necessary inputs due to unavailability of data or privacy concerns. To be implementable, trust models need to consider what data is available and if their factors can be computed. Proxies or alternatives need to be found for incomputable factors.

Considers Node Diversity For trust models to be usable they also need to be able to account for individual nodes. This is related to the cold start problem. Trust models that rely on factors such as experience, face the issue where some nodes lack sufficient experience to have high direct trust values. Such new nodes find themselves unable to interact but unable to perform the necessary exchanges to raise their trust values sufficiently. Trust models should find alternative methods to calculate trust that can account for all the possible different users in the network.

Usable in Different Networks Trust models need to be usable in different types of networks. This means that trust management and its calculation methods need to be able to work in networks, even if they have

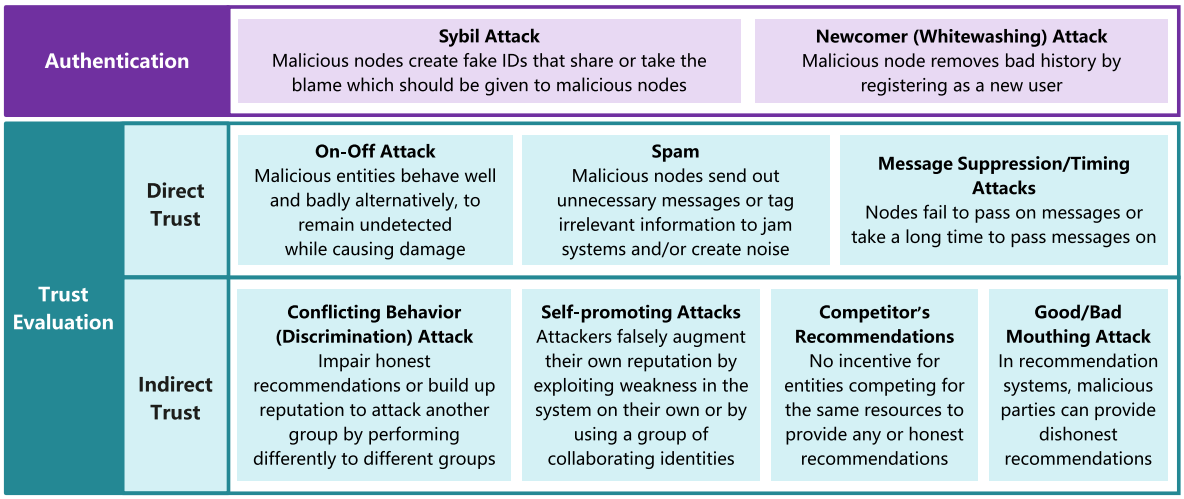


FIGURE 8. Table of standard trust security threats.

cycles [28], are sparse or have other problematic graph characteristics. Particularly, graph theory related methods and path-finding algorithms may not converge or collapse whenever cycles or sparse networks are involved. Alternative modelling methods need to be considered accordingly.

D. FUNCTIONALITY

A trust model must be functional in that it supplies a suitable decision-making framework. Many trust models often aim to output a trust value. While this is important, the trust values should be interpreted into **corresponding access** for the trustee. In other words, trust models need to provide corresponding access to the different trust values and levels. There are several problems access control and functionality must address.

Service-oriented Access Control Corresponding access control needs to be service oriented [1]. This means that the number of thresholds, how narrow the bands are and the strictness for different access levels need to be tailored according to the service. Service here should additionally consider provision-context, the user and device to which the service is provided.

Continuity of Access Rights Access control also needs to identity when a user is not trustworthy enough to continue their access rights [1]. Their rights then need to be revoked accordingly. Whether a user is trustworthy enough depends on the application, even if a single malicious act were sufficient to revoke access rights. Therefore, even for trustworthy agents, their trust values should be tracked, constantly evaluating trustees.

E. ROBUSTNESS

Robustness has to do with the trust management system's ability to function and provide appropriate trust values despite **network disturbances** and system-errors. Network disturbances may occur in soft-security distributed digital environments. Therefore, in the transmission of trust related data, trust management may find themselves vulnerable to disturbances where for example, nodes propagate wrong trust values, or a malicious attack occurs. Trust management schemes need to be robust enough such that these trust values are still available and the digital environment functioning, no matter these disturbances. Trust management faces two key types of disturbances.

Propagation Errors Errors can also include propagation errors such as natural errors made in the transmission of information between nodes. Trust systems should be able to account for potential transmission errors of trust values and adjust the trust value or certainty accordingly. This ensures constant security and availability for the digital environment.

Cold Start Problem The cold start problem mentioned in Section IV-C is relevant here. A trust management system is robust if it can counter instances of insufficient information about the trustworthiness of a node and still produce a good, usable trust value. It means the system is comprehensive enough to be robust even under problematic circumstances.

V. ATTACKS ON TRUST MODELS

Known attacks inform security protocols on trust management systems. A secure trust management system is able to resist the attacks seen in Figure 8. We describe each attack, potential repercussions and how preventing them helps secure trust models.

Sybil attack In Sybil attacks, malicious nodes create fake IDs that share or take the blame which should be given to malicious nodes [1], [5], [7], [8]. Malicious nodes carry out Sybil attacks by creating many fake identities that then each attack honest nodes. Instead of the single node taking on all the blame for the collection of attacks,

the malicious nodes, all from the same user, share the blame so that their individual trust values drop more slowly. This way, the collection of malicious nodes is able to launch more attacks before being individually detected. A system that is ensures authenticity of users, non-repudiation and integrity can resist Sybil attacks. Users are authenticated so that multiple identities cannot be created without at least being able to trace all nodes back to the same identity. The authenticated identity is then held responsible for attacks of any attacks by its malicious nodes and removed before other nodes can launch attacks.

Newcomer attack In newcomer attacks, malicious nodes remove bad history by registering as a new user [1], [8]. The newcomer attack is slightly different from the Sybil attack. Malicious attackers that have accumulated bad reputations on the system are able to leave the system and return, erasing history of their behaviour so they can launch attacks on honest nodes again. Like the Sybil attack, authentication, non-repudiation, and integrity help resist newcomer attacks. Authentication traces the identity of each newly created node to its original creator and can carry the bad reputation from the earlier node to this new node. The attacker is thus unable to launch attacks despite its new identity.

On-off attack In on-off attacks, malicious entities alternate between behaving well and badly, to remain undetected while causing damage [1]. On-off attacks are attacks launched via direct interactions. Malicious nodes perform well for a while to gain users trust before attacking the user by performing poorly. They may repeat this again to gain back the users trust before attacking, remaining undetected to maintain trust values above a certain level. A trust model that maintains non-repudiation and integrity will be able to identify malicious nodes performing on-off attacks, punish them and prevent them from attacking.

Spam Spam occurs when malicious nodes send out unnecessary messages or tag irrelevant information to jam systems and/or create noise. Spam is another type of attack that occurs in direct interactions. Such spam can be seen in digital environments with tagging systems [40]. Spam is harmful as it distributes and introduces large amounts of irrelevant, if not harmful, content into networks. Non-repudiation and integrity requirements target this attack so that trust management systems that are secure can identify when a node is distributing large amounts of irrelevant content and remove the node and their content as needed.

Message Suppression/Timing Attacks In message suppression attacks, nodes fail to pass on messages and in timing attacks, nodes take a long time to pass messages on [5], [8]. Message suppression and timing attacks are particularly damaging to time sensitive and distribution reliant networks such as vehicular networks or wireless sensor networks. When attackers launch message suppression and timing attacks, they do not or take a long time to pass messages along. This could result in failures to inform target users about necessary information in time or at all, causing harm to the functioning of the entire network.

Conflicting Behaviour Attack In conflicting behaviour attacks, malicious nodes impair honest recommendations or build up reputation to attack another group by performing differently to different groups [1]. Conflicting behaviour attacks are damaging to reputation systems. Users destroy the reputations of honest users by behaving positively only with them. By doing so, they build up a good reputation with this select group of users who then recommend the malicious nodes to other honest nodes. This results in a) the reputation or recommendation nodes being tarnished and b) the truster nodes being harmed from interactions with malicious nodes due to poor recommendations.

Self-promoting Attack In self-promoting attacks, attackers falsely augment their own reputation by exploiting weakness in the system on their own or by using a group of collaborating identities [8]. Self-promoting attacks can occur in poorly designed and poorly authenticated recommendation systems. In systems that are poorly designed, malicious users can either falsely augment their own reputation by recommending themselves or create and collaborate with other users to augment their own reputation. By doing so, malicious users can increase their trust values via indirect trust to perform harmful actions.

Competitor's Recommendation In competitive environments, there is no incentive for entities competing for the same resources to provide any or honest recommendations [1]. Competing recommendations are particularly problematic in e-commerce platforms where buyers compete for limited products and sellers compete for limited buyers. When there are limited resources, buyers and sellers have no incentive to supply good and honest recommendations [36]. Giving false recommendations results in entities making poor decisions while not giving any recommendations results in users having to act on insufficient information or not acting at all. The lack of incentive to recommend could result in the collapse or stalemate of systems where there are insufficient direct interactions, particularly in new digital environments. Integrity and non-repudiated systems can counter poor recommendations while systems that meet the availability requirement are able to prevent collapse due to lack of recommendations.

Good/Bad Mouthing Attack Good/Bad mouthing attacks are attacks where malicious entities provide dishonest recommendations [10]. Bad mouthing attacks makes it difficult for trusters to find interactions to serve their needs and trustees to gain the trust of trusters. Good mouthing attacks increase the indirect trust values

of malicious nodes and cause honest trusters to interact with malicious entities. Recommendation systems that are flooded with such attacks will drive users away or cause users to no longer rely on recommendations. Systems that meet integrity and non-repudiation can resist such attacks.

Collusion Attack Lastly, collusion attacks are attacks that use a combination of the above attacks [7]. This allows users to apply maximum damage to the digital environments. A fully secure trust system must also ensure that the digital environment is not vulnerable to a combination of attacks and that defence against one attack does create vulnerabilities to another.

VI. MODELLING METHODS

The criteria discussed can be met by employing the right methods for modelling. In this section we discuss existing methods used in trust modelling, their theories and how they have been used. First, we present an overview of each of the steps in trust modelling. For each of the steps, the existing methods that have been used are illustrated and coloured by the mathematical field they are under.

A. BASIC METHODS

Simple trust models are models that employ basic mathematical constructions. Heuristics are employed and introduced in these equations so that their trends represent trust-related processes in digital environments. There is no fixed way to construct equations, some are adapted from known and common basic constructions.

1) WEIGHTED AVERAGE

The key component of many simple models are usually weighted averages. Weighted averages, like averages, combine values of some set and normalises this sum by the number of elements to achieve a representation of the set that considers all its elements. In weighted averages, however, different values are weighted differently, to provide a more reflective combination. Weighted averages are calculated

$$\bar{x} = \frac{\sum_{i=1}^{n} \omega_i x_i}{\sum_{i=1}^{n} \omega_i} \tag{1}$$

where ω_i are weights for each i-th factor, x_i. There are n factors being considered. Being able to apply heuristics to weights in trust modelling is advantageous in considering factors that do not have fixed mathematical constructions to represent their meanings. This is useful to trust as the actual magnitudes of ratings tend to be subjective.

a: COMBINING TRUST FACTOR

In [18], [19], [22], [29], [31]–[33], [48], weighted averages were used to balance factors to output a trust value. Most trust models do not define weights or leave it to implementers or device owners to decide the weight of each factor [19], [22], [29], [31]–[33], [48]. This allows users to personalise their trust model according to their needs and makes them responsible for weighting the factors based on what they know about themselves.

However, some models may instead define weights and/or additionally adjust the weights provided by users. For example, in [18], dynamic confidence factors were used to weight direct and indirect trust such that with sufficient interactions, greater, if not all, weight would be given to direct trust values. Where N_{ij}^T is the number of direct interactions between users i and j, the dynamic confidence factor is

$$\alpha_{ij} = \frac{N_{ij}^T}{N_{ij}^T + c} \quad \text{or} \quad \alpha_{ij} = 1 - \beta^{N_{ij}^T} \tag{2}$$

where $c > 0$ and $0 < \beta < 1$ are parameters that can be adjusted by the user. These user-defined parameters help describe how reliant on direct interactions the user wants to be. The two equations are both monotonically increasing with limit at infinity 1 but have different trends. The dynamic confidence factor can be chosen based on the needs of the digital environment.

In [33], confidence and time help adjust the user-defined weights for indirect trust. Confidence is modelled such that the more interactions and recommenders there are, the higher the confidence. This increase in confidence can be modelled with any monotonic increasing function. To account for time, [33] considered the number of time intervals passed since recommendations were received from each recommender. The longer the time lapsed, the smaller the weight assigned.

b: AGGREGATE EXPERIENCES

Weighted averages are also used to aggregate experiences [22], [29], [31], [36]. This helps collect all the evidence (from interactions) and combines them into a single representative value. These values can then be used as part of more complex methods. In [29], x_i in Eq. 1 would be the number of positive interactions at each time window, N_{pos}. Time-based weighting was carried out by simply using a *forgetting factor*, $0 < \lambda < 1$, exponentiating using time intervals so that older intervals would have smaller weight. The weighted average representing positive interactions was defined

$$\bar{N}_{pos} = \sum_{i=1}^{N} N_{pos} \cdot \lambda^{i-1} \tag{3}$$

where i is each time-window. Alternatively, if time intervals are not appropriate, time passed t_i after each interaction N_i can also be used to as a weight for each i-th interaction [31]

$$\omega_i = e^{-\lambda t_i} \tag{4}$$

so that as time passes, the weight given to older experiences will decrease exponentially. Contextual features have also been used as weights. The weights measure how similar the i-th experience is to the current experience and would take on ω_i in equation 1.

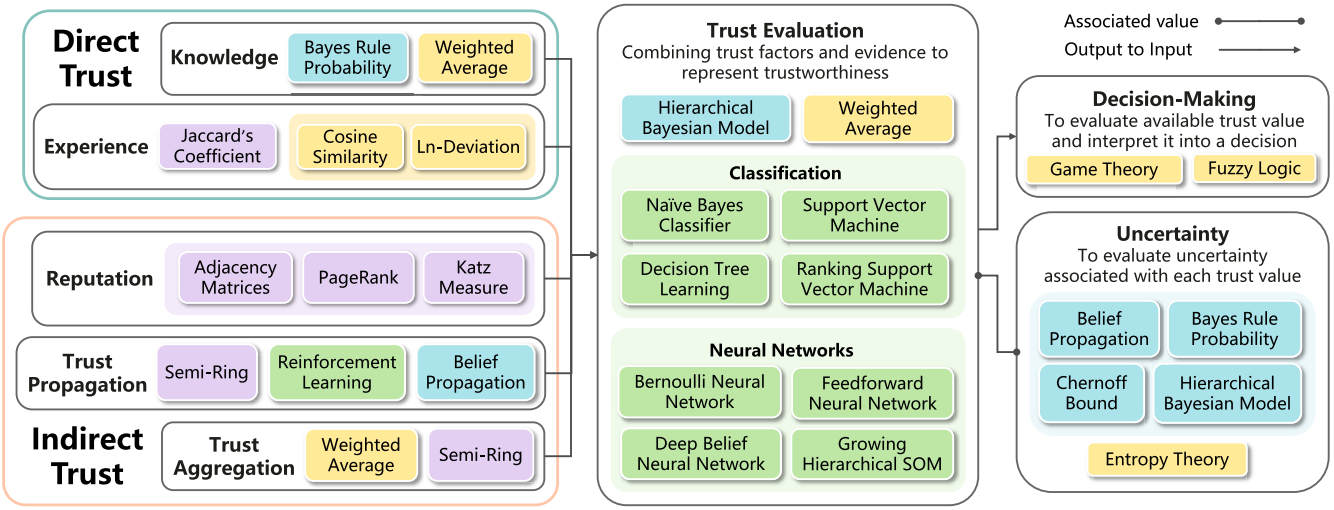


FIGURE 9. Overview of trust modelling process and methods in each step.

c: AGGREGATE TRUST VALUES

Weighted averages are also used to aggregate trust values [19], [27], [30], [49], [50]. Trust values can be that describing paths or individual nodes in a network. Typically, they are weighted using the trust values of intermediaries so that ω_i is the trust value about the intermediary(advisor) and the factor is the advisor's opinion about some other intermediary or the trustee.

2) RELATIONAL MEASUREMENT

Relational measures include similarity measures, deviation, correlation measures and other simple constructions. and are simple constructions meant to represent how far two variables, are related, whether positively or negatively. This relation can give insight into whether these two users are likely to trust each other.

a: SIMILARITY MEASURES

Similarity measures are useful to measure how similar two users are in terms of their opinions. It is believed that the more similar two users are the more likely they are to trust each other. Cosine similarity was used in [25], [47] to measure how similar two ratings were. When ratings are involved, such as in review systems on e-commerce platforms, the similarity between the ratings of two individuals about the same objects can reflect the level of trust in each other. For example, if two users rated the same set of n products and the ratings were collected into vectors $\mathbf{a}$ and $\mathbf{b}$ for each user respectively, the cosine similarity would be

$$\cos(\theta) = \frac{\mathbf{a} \cdot \mathbf{b}}{\|\mathbf{a}\| \cdot \|\mathbf{b}\|} \tag{5}$$

This way the actual magnitude of the scores is recorded and only the relative differences between the two users' ratings will be considered. For example, a user could have rated two products with the vector (2, 5) while someone else rated those same products (4, 10). The reviews may have very different magnitudes but their opinion towards the products is similar (that the first one is worse than the second). Cosine similarity captures this similarity in rating patterns rather than the total magnitudes of the ratings.

b: DEVIATION MEASURE

ln-deviation is also used to measure how unusual some variable for a particular node is [24], [48]. ln-deviation is adapted from mean-log deviation used in income-inequality which is like standard deviation except that ln is applied to reduce the range of very wide-ranging features. For some variable $X(i)$ for user i, ln-deviation is given as

$$lndev(i) = -\ln\left(\frac{X(i)+1}{\max_{j\in N}(X(j))+1}\right) \tag{6}$$

where N is the total number of nodes in the network. The degree to which the variable for the node deviates from some standard (in this case the maximum) could be an indicator of the impact of their actions [48] or the abnormality of responses or behaviour in the network [24], [48] and thus, an indicator of trustworthiness.

Deviation was also used in Zeinab's trustworthiness reputation scheme [36]. The trustworthiness of an advisor is measured by the competency and willingness of the advisor to provide good recommendations. Competency of the advisor was measured using a standard deviation method. The reliability of ratings was computed using the evidence-based certainty equation introduced by Wang in [51]. Then the standard deviation of the credibility of ratings for different sellers was computed to represent the uncertainty in the seller's ratings. This helps account for the reliability of the advisor despite sellers that discriminate in their behaviour by aggregating the general rating behaviour of the advisor for common sellers.

This deviation was multiplied by the dishonesty of the advisor. This dishonesty was also computed using standard

deviation and was represented by the successful and unsuccessful interactions with the seller that the advisor has had. Beta probability described in Section VI-C1 was used to measure while accounting for uncertainty the expected rating of the seller. The output is then formulated as the expected rating by the advisor or buyer about the seller. The standard deviation was used on this expectation for common sellers to reflect whether ratings provided by the advisor align with those experienced by the buyer. If the ratings do not align, the standard deviation would be high, reflecting a high level of dishonesty.

Basic difference between ratings was also used and modified using heuristics to compute the willingness of an advisor to provide good ratings. The difference in the rating by the advisor and all other ratings over time was iteratively computed over different time intervals. This was coupled with integration over the exponentiated form (to give greater weight to larger rating differences). The final integration reflects the rating behaviour of the advisor relative to the buyer. A large, accumulated difference in weighting reflects that the advisor is less willing to provide accurate recommendations. This difference was also weighted by the ratio of good recommendations provided by the buyer to the recommender previously that were truthful. This captures the intuition that the advisor will react in kind to how the advisor was treated by the buyer in the past.

3) FUZZY LOGIC

Unlike Boolean logic where ''true'' and ''false'' are the only values allowed, fuzzy logic allows multiple values to be assigned to better represent situations where a statement cannot be determined absolutely true or absolutely false. These intermediate logical values take the form of fuzzy sets such as ''not very true'', ''somewhat false'' and so on. Translating observable real-life quantities to numeric values can be done via the membership function. These membership functions output the degree to which the quantity is belongs to some defined fuzzy set. Fuzzy logic operators and rules act on values attached to fuzzy sets. These rules help map quantities to a fuzzy set output. The fuzzy output can be translated back into a numerical value if necessary [52].

The entire process described has been implemented by Song for evaluating trust in grid computing based on numerical type features [53]. In their trust management model, numerical values of features defence capability and job success rate were fuzzified. Using membership functions, the membership degree of the self-defence capability feature in the fuzzy sets ''low'', ''medium'' and ''high'' was determined; the membership degree of the job success rate in the fuzzy sets ''very low'', ''low'', ''medium'', ''high'', ''very high'' was determined. Based on intuition, fuzzy rules and logical operators were defined and used to map the fuzzy set values to fuzzy set values describing trust. The possible trust fuzzy sets were ''very low'', ''low'', ''medium'', ''high'' and ''very high'', and the fuzzy rules and operators help obtain membership degrees for trust value in each of these fuzzy sets. Using the membership degrees and the inverse membership function, a numerical trust value can be obtained by aggregation.

The concept of membership functions was implemented in [54], [55]. The decision tree, explained in Section VI-D1.c, is trained by iterating through the different splitting conditions, using a greedy algorithm, to determine the best condition that minimizes the error between derived trust values and actual trust values. These splitting conditions can be used to classify users into trustworthy and untrustworthy classes. However, as highlighted in the motivations for fuzzy logic, continuous attributes are rarely completely in one (intermediate) category or another. Fang addressed this by using membership functions to partially allocate users into each category for each continuous attribute. By iterating through different thresholds in the training process the decision tree can be trained to be more flexible by determining the best threshold to partially allocate users in different categories.

Wu employed a similar method to train a fuzzy neural network [55]. By pre-processing continuous inputs, Wu categorised continuous inputs into sets to reflect human interpretation on whether a particular value is ''low'', ''just so-so'', ''high'' or ''very high''. Therefore, the neural network can process instead the categories and the degree to which the user belongs in a particular attribute instead of the raw values themselves. This allows human interpretation within the neural network. The fuzzy operators and rules were replaced with the neural network. This process is described in Section VI-D4.b.

4) GAME THEORY

The iterated prisoners' dilemma problem in game theory has been applied to trust modelling for reputation systems in e-commerce markets by Zeinab [36]. In the prisoners' dilemma game, two players must decide whether to cooperate or defect. If both players cooperate, the reward is R; if both players defect, the punishment is P; if only of the players defect, the player that defected receives reward T while the other receives punishment S. These payoffs satisfy $T > R > P > S$ and $2R > T + S$ [56]. In the iterated prisoners' dilemma game, the two players repeat the prisoner's dilemma multiple times and can remember and react to past actions [56].

In e-commerce markets, buyers must balance between competition due to limited resources and not being able to discover good sellers if they report untruthfully. Zeinab adapted this iterated prisoners' dilemma game to address this aspect of reputation systems in e-commerce markets [36]. In the reputation system, an advisor can choose to cooperate or defect by providing or not providing good recommendations about sellers. The rewards and payoffs from the prisoner's dilemma can be defined analogously here. Using the defined payoffs, an expected payoff from continuing the pattern of cooperation and defection over multiple interactions can be defined. This long-term payoff computation considers the time remaining in the e-commerce network and the

trustworthiness value of the advisor. This trustworthiness value of the advisor is determined by the competency and willingness of the advisor to provide good recommendations [36]. Standard deviations and difference values were used and are described in Section VI-A2.b.

5) ENTROPY THEORY

Entropy theory aims to quantify the uncertainty associated with a random variable. To do this, entropy theory quantifies information gained or uncertainty as $I(p) = -\log p$ because [57]

- $I(p)$ is monotonically decreasing. This reflects that as the probability of an event occurring increases, we are less uncertain of the possible outcome so, uncertainty decreases.
- $I(p) \geq 0$ which reflects that uncertainty is always non-negative.
- $I(1) = 0$ which reflects that events that almost surely occur do not have any uncertainty.
- $I(p_1, p_2) = I(p_1) + I(p_2)$ which reflects that the uncertainty of two independent events is the sum of the uncertainty from either event.

Then, the amount of entropy associated $H(X)$ with a random variable X is the expected amount of uncertainty with the variable based on all the outcomes and the probabilities of those outcomes. Therefore, entropy is defined [57]

$$H(X) = \mathbb{E}\left[I\left(p\right)\right] = -\sum_{i=1}^{n} P\left(x_i\right) \log P\left(x_i\right) \tag{7}$$

where the base of log depends on the application. In trust, base 2 is usually used. Then, given the random variable X that an agent cooperates with probability p, the entropy of the agent cooperating is useful in representing the degree of uncertainty in the agent. Explicitly, this entropy is

$$H(p) = -p \log_2 p - (1-p) \log_2 (1-p) \tag{8}$$

using Equation 7. Entropy has been used as a trust value [49], [50] and as a weight [58].

a: ENTROPY AS A TRUST VALUE

Trust is often understood as the probability of an agent behaving cooperatively. However, trust may not increase linearly with probability of cooperation. Using entropy as a trust value, probability-based uncertainty associated with the agent is used to represent trust. The trust value is usually defined [49], [50]

$$T(p) = \begin{cases} 1 - H(p) & \text{if } 0.5 \leq p \leq 1 \\ H(p) - 1 & \text{if } 0 \leq p < 0.5 \end{cases} \tag{9}$$

so that for more extreme probabilities of cooperation (extremely high or extremely low probability), the rate of change of certainty is very much higher than if the probability is not as extreme. At less extreme probabilities (around 0.5), the rate of change in certainty is much slower. Notice that the entropy is flipped in this definition to represent certainty instead of uncertainty. Trust using this definition of certainty models a truster who

- For extremely low probabilities of cooperation, trust drops rapidly. If an agent already has a low probability of 0.2 of cooperating and they drop further to 0.1, their trust value drops to a more disproportionate extent to show that the truster is disproportionately more certain the more extremely low the probability of cooperation.
- For mid-value probabilities, certainty in the agent, and therefore trust, changes slower. So, if an agent's probability of cooperation increases from 0.5 to 0.6 (0.4 to 0.3), the probability is still fairly low (not that low) and the difference to the truster between the two values might not be very large. So, the 0.1 increase (decrease) results in a disproportionately smaller increase (decrease) in trust.
- For extremely high probabilities of cooperation, trust increases rapidly. If an agent with very good probability of performing well can further improve their performance, this continued improvement reflects very well on them. Therefore, the increase in certainty in the agent increases disproportionate.

b: ENTROPY AS WEIGHT

Entropy has also been used as a weight in trust modelling [58]. Using entropy from Equation 8 as a weight, more weight is given to interactions that have less extreme probability values. For each interaction, there is a probability of the interaction outcome being positive. If by entropy, the interaction carries more information, more weight should be given to the interaction when aggregating all experiences.

Jayasinghe did this in his model for IoT devices [58]. Adapting his method, suppose a truster has had $c_1, \ldots, c_n$ interactions with a trustee, for each interaction c_m there is a probability p_m of the interaction succeeding. Using Equation 8, the uncertainty-weighted performance of the trustee denoted CFD is

$$CFD = \sum_{m=1}^{n} \frac{c_m}{t_m} H(p_m) \tag{10}$$

where $\frac{c_m}{t_m}$ is the fraction of total time spent by the trustee for each interaction and helps to further weight the interaction based on its duration. In CFD, if an interaction had roughly 50/50 chance of succeeding, its success was more uncertain and so any information gained from the interaction would be greater, by simplifying some of the uncertainty. Therefore, more weight is given to the interaction outcome and duration.

B. GRAPH METHODS

In graph methods, we discuss the available methods to capture the structure of a network, globally and locally for each node. These graph methods are particularly useful because many digital networks can be simplified into graphs with nodes and edges. The existing relationships and transfer of any information offers large amounts of information that can

be modelled into features for trust. Otherwise, graph methods are useful for inferring trust values about other nodes based on the trust values of surrounding nodes or along paths.

1) NETWORK FEATURES

First, in graph methods for trust modelling, of key interest is how to describe a node's structural position in a network and relationships to other nodes. We call these network features and discuss how graph methods can and have been used for modelling of such network structural features.

a: NETWORK OVERLAP

The degree to which two individuals in a network have common ''friends'' is an indicator of how significant a role the other plays in each other's network. The larger the role, the more significant the role and so the more likely there will be trust between the two users. There are two representations in trust modelling: centrality and Jaccard's coefficient. Centrality is measured by a simple mathematical concept in several models [19], [58]. Centrality of trustor i and trustee j is calculated as

$$c_{ij} = \frac{|K_{ij}|}{|N_j|} \tag{11}$$

where K_{ij} is the set of common friends and N_j is the set of friends that j has. Then, c_{ij} represents the degree to which i and j are indirectly connected. N_j is necessary for normalization. For the same $|K_{ij}|$, a larger N_j means that the number of common friends is proportionally smaller. Thus, the overlap in connections is not as significant.

Jaccard's coefficient is a similar measure but normalized with the total number of friends [47]. Where $I(i)$ is the set of i's friends, Jaccard's coefficient for trust between i and j is usually

$$J_{ij} = \frac{|I(i) \cap I(j)|}{|I(i) \cup I(j)|} \tag{12}$$

Then, Jaccard's coefficient represents the proportion of the friend group considering both users. This means that if either user has a large group of friends, the union set would be large and so the number of shared friends would not be proportionally large enough to warrant a large amount of trust between the two, based on common friends.

b: GLOBAL REPUTATION

A more global relational measure, adapted from PageRank, measures the importance of a node in relation to the entire network based on its links. PageRank is an algorithm typically used to rank webpages on search engines. In trust modelling, PageRank is modified so that the rank given to the page is analogous to the authority or popularity of a particular node in the network [24], [25], [47], [48]. This reputation score is given by

$$R(i) = \frac{1-d}{n} + d \sum_{j \in M(i)} \frac{R(j)}{L(j)} \tag{13}$$

where n is the total number of nodes, $M(i)$ is the number of people who trust i and $L(j)$ is the number of nodes that j trusts. d is the damping factor, representing in the case of trust, when the user continues trusting others within the network, instead of not trusting anyone. PageRank algorithm for trust outputs a ranking of nodes in the network that considers number and trustworthiness of the truster nodes for any trustee node. It is assumed that the trustworthiness of a trustee node is indicated by the number of nodes that trust the trustee.

c: ADJACENCY MATRICES

Adjacency matrices are square matrices that represent finite graphs. The ij-th entry in the matrix indicates a relationship between node i and node j in the graph. In a weighted graph, like those typically used in trust, the weights will be used directly as entries in the adjacency matrix. Otherwise, the adjacency matrix will be binary with a 1 entry indicating the presence of a relationship and a 0 representing no relationship. For trust, the adjacency matrix is most useful for illustrating, in matrix form, the total path weights (for weighted graphs) or total number of paths (for non-weighted graphs) from any one node to another. This representation allows an entry-by-entry visualisation of the type of relationship each node has with every other node. This representation can be further expanded into other metrics.

d: KATZ MEASURE

Katz measure is useful in measuring the degree of influence of a node in a network based on all the paths going to the node [59]. Katz measure was used in [25] as an input feature to a neural network and in [28] for the indirect trust value. Katz measure of user i is given

$$X_{katz}(i) = \sum_{k=1}^{\infty} \sum_{j=1}^{n} \alpha^k (A^k)_{ji} \tag{14}$$

where A is the adjacency matrix of the underlying graph. Then, $(A^k)_{ji}$ counts all the paths of length k from user j to i and does so for all users $j = 1$ to $j = n$. This is repeated for paths of all lengths from $k = 1$ to $k = \infty$. The ''attenuation factor'' $0 <\leq \alpha \leq 1$ then weights the paths such that the longer the path, the less weight assigned to it. This reflects how trust diffuses the more intermediate recommenders there are along a path. Computing Equation 14 is equivalent to finding the column sums for the following matrix [28], [59]

$$\boldsymbol{X}_{katz} = (\boldsymbol{I} - \alpha \boldsymbol{A})^{-1} - \boldsymbol{I} \tag{15}$$

where $\boldsymbol{I}$ is the identity matrix. When α is less than the reciprocal of the largest characteristic root of $\boldsymbol{A}$, computations can be simplified significantly so that there is no need to compute powers of matrices [59]. Otherwise, an iterative method proposed by [28] also offers a more efficient method of computation.

2) SEMI-RINGS

Semi-rings are algebraic structures defined by a tuple $(A, \oplus, \otimes, ⓪, ①)$ such that for all elements a, b, c in the non-empty set A and for the elements $⓪, ① \in A$, the following conditions hold true [35], [60]

- $\oplus$ is commutative, associative and ⓪ is the additive neutral element:

$$a \oplus b = b \oplus a \tag{16}$$
$$(a \oplus b) \oplus c = a \oplus (b \oplus c) \tag{17}$$
$$a \oplus ⓪ = a \tag{18}$$

- $\otimes$ is associative, ① is the neutral element and ⓪ is the absorbing element:

$$(a \otimes b) \otimes c = a \otimes (b \otimes c) \tag{19}$$
$$a \otimes ① = ① \otimes a = a \tag{20}$$
$$a \otimes ⓪ = ⓪ \otimes a = ⓪ \tag{21}$$

- $\otimes$ distributes over $\oplus$:

$$(a \oplus b) \otimes c = (a \otimes c) \oplus (a \otimes b) \tag{22}$$
$$a \times (b \oplus c) = (a \otimes b) \oplus (a \otimes c) \tag{23}$$

The properties of the semi-ring can be interpreted to correspond to human interpretations of trust. These interpretations are described in [60] and [61] and demonstrate the usefulness of semi-rings for trust modelling.

a: SEMI-RINGS FOR GRAPHS

Semi-rings have been proposed by [60] for trust-modelling between nodes in graphs.

Typically, the set A in semi-rings used for trust modelling are Cartesian planes of trust and confidence values. Each point in the Cartesian plane is denoted (t, c) for trust and confidence, respectively. The semi-ring operator $\oplus$ serves to combine trust values of nodes along a path and while $\otimes$ serves to aggregate the trust values across different paths.

In [60], two semi-ring constructions were proposed. In both semi-ring definitions, the additional properties were imposed to reflect properties of trust. These properties are

- $a \otimes b \leq a, b$ to reflect that opinions propagated along a path are limited by the trust values of nodes along the graph and
- $a \oplus b \geq a, b$ to reflect that opinions aggregated across paths are more information rich so they should be greater than individual opinion values.

The first construction is the path semi-ring. In this semi-ring, trust, and confidence values each come from the domain [0, 1]. Therefore, the trust semi-ring would be $S = ([0, 1], [0, 1], \oplus, \otimes)$ and the operators are defined [60]

$$\otimes: \quad \left(t_{ik}, c_{kj}\right) \otimes \left(t_{kj}, c_{kj}\right) = \left(t_{ik} t_{kj}, c_{ik} c_{kj}\right) \tag{24}$$
$$\oplus: \quad \left(t_{ij}^{p1}, c_{ij}^{p1}\right) \oplus \left(t_{ij}^{p2}, c_{ij}^{p2}\right) \tag{25}$$
$$= \begin{cases} \left(t_{ij}^{p1}, c_{ij}^{p1}\right), & \text{if } c_{ij}^{p1} > c_{ij}^{p2} \\ \left(t_{ij}^{p2}, c_{ij}^{p2}\right), & \text{if } c_{ij}^{p2} > c_{ij}^{p1} \\ \left(\max(t_{ij}^{p1}, t_{ij}^{p1}), c_{ij}^{p1}\right), & \text{if } c_{ij}^{p2} = c_{ij}^{p1} \end{cases} \tag{26}$$

where t_{ij} indicates the trust from i to j and c_{ij} indicates the confidence in this trust value. The superscript $p_k p$ indicates that the corresponding value refers to that for a path k. In this construction, only the trust value of one path is used at the end to decide the trustworthiness of the user.

[60] also proposed an alternative distance semi-ring construction. The domain of trust values is $[0, \infty]$ and confidence values if $[0, 1]$. Therefore, the semi-ring is $S = ([0, \infty], [0, 1], \oplus, \otimes)$. The operators were defined

$$\otimes: \quad \left(t_{ik}, c_{kj}\right) \otimes \left(t_{kj}, c_{kj}\right) = \left(\frac{1}{\frac{1}{t_{ik}} + \frac{1}{t_{kj}}}, c_{ik} c_{kj}\right) \tag{27}$$

$$\oplus: \quad \left(t_{ij}^{p1}, c_{ij}^{p1}\right) \oplus \left(t_{ij}^{p2}, c_{ij}^{p2}\right) = \left(\frac{c_{ij}^{p1} + c_{ij}^{p2}}{\frac{c_{ij}^{p1}}{t_{ij}^{p1}} + \frac{c_{ij}^{p2}}{t_{ij}^{p2}}}, c_{ij}^{p1} + c_{ij}^{p1}\right) \tag{28}$$

so that the trust and confidence values are considered for multiple trust paths. Both semi-rings behave differently as demonstrated in [60] and should be chosen based on the digital environment.

When considering more than two nodes along the path, the above semi-ring operators proposed by [60] can be extended to iterative computation so that the trust value between the source s and target t is given

$$\left(t_{st}^{p_k}, c_{st}^{p_k}\right) = \bigotimes_{e_{ij} \in p_k} (t_{ij}, c_{ij}) \tag{29}$$

where e_{ij} is an edge carrying the trust and confidence values from i to j and i and j are intermediate nodes that from s to t in the path p_k.

Similarly, with more than two paths, the final path aggregated trust value between source s and target t is

$$(t_{st}, c_{st}) = \bigoplus_{p_k \in \text{paths}} \left(t_{ij}^{pk}, c_{ij}^{pk}\right) \tag{30}$$

where p_k is the k-th path in the set of possible paths from s to t.

In [35], it is noted that confidence values are not immediately available to implement the semi-rings proposed by [60]. To obtain certainty values, [35] proposes two c_{ik} functions, that takes in the edge e_{ik} between two directly connected nodes i to k. The first function reduces the confidence value as paths get longer by defining c_{ik} to be

$$c_{ik}(e_{ik}) = \alpha, \quad 0 < \alpha < 1 \tag{31}$$

using a constant α as a decaying factor. Then clearly, by Equation 24 and 27, the confidence between two indirectly connected nodes decreases the more edges there are between

two the nodes [35]. This reflects the intuitive understanding that the more intermediate recommenders are required, the more likely information is diluted by different opinions and so the less reliable the propagated information.

Alternatively, [35] also proposes a distance-based confidence function defined linearly by

$$c_{ik}(e_{ik}) = \min(\beta + \gamma d_i, 1) \tag{32}$$

or exponentially by

$$c_{ik}(e_{ik}) = 1 - \eta^{d_i} \tag{33}$$

where d_i is the degree of node i in the edge from i to k and β, γ, η are tunable parameters. Note that $0 < \eta < 1$. The underlying intuition is that the degree of certainty of the trust value given by i about k is determined by how reliable i is and the more edges i has, the more reliable i is. Therefore, the greater the degree of i, the greater the confidence in the trust value provided by i about k.

b: SEMI-RINGS FOR HYPER-GRAPHS

Hyper-graphs differ from graphs in that each edge can point from one node to multiple nodes. Hyper-graphs are useful for representing relationships in digital environments where trust can be towards a group of individuals rather than a single entity.

In the trust hyper-graphs proposed by [61], trust relationships can be established between one trustor and a group of trustees. The direct edges typically found in a normal graph can be grouped to form an ''AND connector''. The trust-confidence tuple attached to the edges can be combined to form a single tuple describing the relationship the trustor has with multiple trustees. To form an indirect trust relationship between a trustor and a target group of trustees, the trustor, through intermediate groups, to the target groups. This is called an ''AND tree''. To compute the trust and confidence values attached to the ''AND tree'', the semi-ring from Equation 26 in [60] was proposed.

C. BAYESIAN METHODS

Interactions in digital environments depend on many observable and unobservable variables that play into people's decision making and so, the outcome of any interaction can be perceived as a random variable. Trusting another agent prior to an interaction can then be measured by the believed probability that some interaction in a trust environment would result in a positive outcome. This interpretation of trust allows the use of probability theory and inference for trust modelling and decision making. Bayesian probability is frequently applied [27], [29], [32], [38], [49], [50], [62]–[65]. Just as humans combine their knowledge and observations to make decisions, Bayesian probability, which is founded on Bayes rule, combines data and a priori knowledge to produce evidence-based probabilities.

1) BAYES RULE PROBABILITY MODEL

The most basic Bayesian inference model that will be described in this section has been widely applied in trust [27], [29], [32], [49], [50], [65]. Bayes rule is given by

$$p(\theta|\text{data}) \propto p(\text{data}|\theta)\, p(\theta) \tag{34}$$

which tells us that the distribution of some parameter or random variable given some observations can be given by some combination of a likelihood ($p(\text{data}|\theta)$) and prior knowledge ($p(\theta)$).

Let trustworthiness be the random variable T and there have already been s successful interactions and n negative interactions with the target agent. Bayesian probability can determine the probability of trust given the collection of past interactions as evidence. This is the posterior. In the context of trust, T will be the belief in the trustworthiness of an agent and (s, f) will constitute evidence of successful and failed interactions with the agent, respectively. By Bayes rule, [27], [65]

$$P(T|s,f) = \frac{P(s,f|T)P(T)}{\text{Normalization}}. \tag{35}$$

The prior and likelihood functions are not fixed. In trust, [27], [29], [32], [49], [50], [65] have all used binomial distributions for the likelihood function and Beta distributions for the prior. Then, the posterior can be computed

$$P(T|s,f) = \frac{Binomial(s+f, T) \cdot Beta(\alpha, \beta)}{\text{Normalization}} \tag{36}$$

$$= Beta(s+\alpha, f+\beta) \tag{37}$$

The binomial distribution, $Binomial(s+f, T)$ is the probability mass function of the number of successes in a series of $s+f$ independent interactions. Each interaction has probability T of success since the agent's trustworthiness reflects the likelihood of successful interactions. $Beta(\alpha, \beta)$ represents the prior knowledge about trustworthiness of the trustee which can range from 0 to 1. α and β are the beta distribution parameters and determine the shape of initial distribution of the likely value of T. For trust, α can be interpreted to be the prior expectation of the number of successful interactions and β will be the prior expectation of the number of unsuccessful interactions.

Suppose there is evidence of (s, f) past interactions and suppose not enough is known to form a prior expectation so we set $Beta(1, 1) = Uniform(0, 1)$. Then the distribution for the trustworthiness value of the target would be, by Equation 36, $Beta(s+1, f+1)$ [65]. Then, when there are additional s' and f' successful and unsuccessful interactions, we can let the previous posterior $Beta(s+1, f+1)$ be the new prior and the likelihood function be $Binomial(s'+f', T)$. Then, the final posterior would be $Beta(s'+s+1, f'+f+1)$. For convenience, we denote total positive interactions to be $N_{pos} = s' + s$ and negative interactions $N_{neg} = f + f'$. The expected trustworthiness value is then the statistical

expectation of $Beta(N_{pos}+1, N_{neg}+1)$ which is

$$\mathbb{E}\left[T|N_{pos}, N_{neg}\right] = \frac{N_{pos}+1}{N_{pos}+N_{neg}+2} \tag{38}$$

which is a single consolidated trustworthiness value.

2) CHERNOFF-HOEFFDING BOUND

In its simplest form, Chernoff-Hoeffding Bound provides probability bounds for the sum of independent random variables. It states that for $X_1, X_2, \ldots, X_m$ independent random variables where $0 \leq X_i \leq 1$ for $i = 1, \ldots, n$, then for $0 < \varepsilon < 1 - \mu$,

$$P\left[\bar{X} - \mu \geq \varepsilon\right] \leq e^{-2m\varepsilon^2} \tag{39}$$

where $\mu = \mathbb{E}[\bar{X}]$ [66]. Note that μ is equivalent to the population mean. By definition of variables, we can interpret this inequality as an upper bound for the minimum difference between the sample mean and population mean.

Mohtashemi, Zhang and Zeinab all applied this bound directly to their respective trust models [27], [29], [36]. In trust, Chernoff-Hoeffding bound is typically applied to determine the necessary number encounters (the value of m) to achieve the desired level of confidence. Mohtashemi, Zhang and Zeinab all defined ratings for interactions between a trustor and a trustee to be binary. In other words, each random variable X_i can have a value of 1 meaning the i-th interaction was successful or 0 meaning the i-th interaction was unsuccessful.

By definition, we can see that $X_1, X_2, \ldots, X_m$ is in fact a series of m Bernoulli random variables. Let trust be the probability of success of each interaction (i.e., of each Bernoulli random variable) and denote this θ. Since X_i is a Bernoulli random variable, the population mean $\mathbb{E} = \theta$ is the expected number of successful interactions in the long run. In Mohtashemi, Zhang and Zeinab, $\hat{\theta}$, equivalent to the sample mean of interactions, acts as the estimator of trust, θ, between the trustor and the trustee. Applying Bound 39 tells us that for the m past interactions,

$$P\left[|\theta - \hat{\theta}| \geq \varepsilon\right] \leq 2e^{-2m\varepsilon^2} \leq \delta \tag{40}$$

where $\varepsilon > 0$ is a trustee set constant representing the maximum tolerable error between the actual trust value and estimator. The trustee should also define the maximum level uncertainty allowed for error value, denoted δ. For example, a trustee that is only willing to accept a trust value error of 0.05 and must be 95 percent certain that the error is within acceptable range will set $\varepsilon = 0.05$ and $\delta = 1 - 0.95 = 0.5$. We can then manipulate the second inequality in Bound 40 to be

$$m \geq -\frac{1}{2\varepsilon^2} \ln\left(\frac{\delta}{2}\right) \tag{41}$$

which tells us the minimum number of interactions to achieve the desired level of trust value accuracy with the desired level of confidence. Using the example above, we would need at least 277 interactions before the desired level of accuracy and confidence is achieved.

3) HIERARCHICAL BAYESIAN MODELS

In hierarchical Bayesian models, the relationship between a random parameter and its observations is extended to multiple layers, where each random parameter forms a layer which can have theoretically infinite layers of random parameters above it (hyperparameters). Each parameter layer influences the parameter below it in the same way θ influences data in the posterior of Equation 34. Bayes rule given by Equation 34 is extended to multiple layers of random variables to compute the posteriors - the distribution of all random parameters given the observed data in the lowest level [67].

Suppose the random variable θ is of interest and there are multiple instances of this random variable, denoted θ_j for the j-th random variable. Suppose also that each θ_j has produced a finite vector of observations $\boldsymbol{y_j} = (\ldots, y_{jk}, \ldots)$. Let the distribution in which the data is observed be Q. Then, it can be said that $y_{jk} \sim Q(\theta_j)$. Further, suppose that each θ_g in fact rises from a common distribution W described by the parameter γ. Then $\theta_j \sim W(\gamma)$. This process of defining distribution parameters by other random variables can be repeated as many layers as is needed. In this case, suppose the distribution of γ is roughly known (e.g., uniform distribution with known start and end distributions), then by Bayes rule, the posterior for all unknown parameters is

$$p(\gamma, \theta|\mathbf{y}) \propto p(\mathbf{y}|\theta, \gamma)\, p(\theta|\gamma)\, p(\gamma) \tag{42}$$

which gives us the distributions of the parameters of interest. It is more likely that the overarching random parameter that determines the distributions of all the parameters is of interest [67]. Therefore, the marginal distribution of γ can be computed

$$p(\gamma, \theta|\mathbf{y}) \propto \int p(\mathbf{y}|\theta, \gamma)\, p(\theta|\gamma)\, p(\gamma) d\theta \tag{43}$$

when θ is a continuous random variable. The discrete case is analogously defined. Computations are typically performed stochastically using methods such as Markov Chain Monte Carlo (MCMC) methods [67].

This hierarchical Bayesian model can be immediately and simply applied. For example, $theta_j$ describes the random variable where agent j is cooperative and $\mathbf{y}_j$ is the past experiences that have been had with agent j. In this case, γ would determine agent behaviour in general, being the parameter in the distribution of agent behaviour *theta*. The posterior of random variable $\gamma|\mathbf{y}$, computed using Equation VI-C3 would thus tell us the distribution of the behaviour of agents given past experiences. One issue with Bayesian models is that the distribution itself is not defined, even if its parameters can be encoded as random variables. Nevertheless, hierarchical Bayesian models have been used for trust modelling.

A more complicated hierarchical Bayesian model, BLADE, was applied by Regan for e-commerce and reputation systems [64]. In BLADE, ratings by advisors and buyers about different sellers are random variables influenced by random variable representing the features of the respective sellers. Since the distribution parameters of ratings

and features are unknown, their distributions are determined further by random variables, denoted θ by Regan, that determine the respective distributions of each feature or rating random variable. Dirichlet distributions were used to describe each of the individual random variables. Time is incorporated into this model by dropping the parameters of the Dirichlet distribution at every time step by a constant factor.

HABIT, a hierarchical Bayesian model proposed by Teacy, takes a different approach with the hierarchy of parameters and chosen distributions [62]. In HABIT, two different but related models - the confidence and reputation models - are defined. The confidence model describes the likely outcome of a particular interaction between a specific trustor and trustee. The reputation model describes the trustee's global reputation. In the confidence model, the outcome of an interaction between trustor i and trustee j is a random variable, O_{ij}. To determine the distribution of the outcome random variable, a random distribution parameter for each outcome random variable, denoted θ_{ij}, is defined. This random variable is determined by a random vector variable, $\theta_{\cdot j}$ which collects all opinions about trustee j. This random vector parameter has an additional random variable that determines its distribution parameters, denoted ϕ. These variables $\theta_{\cdot j}$ and ϕ form the reputation model.

For the confidence model, a vector describing probability of the likely value of the outcome of the interaction was used as the likelihood function. A Dirichlet distribution was assigned as the conjugate prior distribution. Hyperparameters for each Dirichlet distribution determined the shape of the distribution θ_{ij} and were updated whenever with each direct interaction. For the reputation model, a non-parametric Dirichlet process (not Dirichlet distribution) model. The prior for ϕ was fully described by a constant and the set of trustees $\{\theta_{\cdot j}\}_{j=1}^{n}$. The posterior for this prior was defined along similar terms. Bayes rule could be used and there would be a final closed form solution. Alternatively, a Gaussian reputation model could be implemented which requires the use of MCMC methods to achieve a closed form solution. In the Gaussian reputation model, ϕ is the vector containing all means and covariances which represents general trustee behaviour and how informative reputations sources are [62]. The likelihood function is then defined using the standard Gaussian probability density function. The prior is then selected to be the normal-inverse-Wishart distribution.

4) DEMPSTER-SHAFFER THEORY

Dempster-Shaffer Theory (DS Theory) is a generalisation of Bayesian probability that maintains the conditioning on observed data and summarising state of belief but removing the need for a global probability distribution assignment [68]. In DS Theory, probabilities are assigned to sets of events rather than each individual mutually exclusive event [69], [70]. This means that evidence collected can be associated with a set of events and assumptions need not be made about the single events within this evidential set [70]. For a set of mutually exclusive and exhaustive single events, DS Theory is interested in its power set (set of all subsets) and assigns a mass $m \in [0, 1]$ to each element in the power set, via the *basic probability assignment* function (bpa) [69], [70]. It should be noted that the mass for the null set should be 0 and all the masses should sum to 1 [69]. Formally,

$$m : \mathcal{P}(X) \rightarrow [0, 1], \tag{44}$$

$$m(\emptyset) = 0, \tag{45}$$

$$\sum_{A \in \mathcal{P}(X)} = 1 \tag{46}$$

where X is the universal set of events and $\mathcal{P}(X)$ is the power set.

There are three functions, besides bpa, that are of interest in trust. They are the belief in event of interest $A \in \mathcal{P}(X)$, the disbelief in A and the uncertainty in A. The belief, disbelief and uncertainty functions take as input the basic probability assignment and are denoted $b(A)$, $d(A)$ and $u(A)$ respectively. The belief in set A represents the total belief about A when all evidence bearing on A has been pooled [69]. Disbelief represents the total belief that A does not occur and uncertainty represents the level of uncertainty in the occurrence of event A or $\neg A$ [71]. The following definitions then follow

$$b(A) = \sum_{B|B \subseteq A} m(B), \tag{47}$$

$$d(A) = \sum_{B|B \cap A = \emptyset} m(B), \tag{48}$$

$$u(A) = \sum_{B \left| \begin{smallmatrix} B \cap A \neq \emptyset \\ B \not\subseteq A \end{smallmatrix} \right.} m(B). \tag{49}$$

and demonstrate the method DS Theory takes to generalise Bayesian probability in the assignment of beliefs based on evidence.

Another way in which DS Theory generalises Bayesian probability is in its combination of multiple sources of evidence. Combination rules in DS Theory aggregates these multiple sources of evidence to provide a single meaningful value summarising belief in the events of interest. These multiple sources of evidence provide different belief assessments for the events in the universal set and DS Theory assumes that these sources are independent [70].

Dempster's rule of combination combines evidence using the aggregation of basic assignment values, m_1 and m_2, in the following way

$$m_{12}(A) \tag{50}$$

$$= \begin{cases} \dfrac{1}{1-K} \sum_{B \cap C = A} m_1(B) \cdot m_2(C), & \text{when } A \neq \emptyset \\ m_{12}(\emptyset) = 0, & \text{when } A = \emptyset \end{cases} \tag{51}$$

where $K = \sum_{B \cap C = \emptyset} m_1(B) m_2(C)$ [70]. K represents the basic probability mass associated with conflict. $1 - K$ is a normalization factor that serves to completely ignore conflict and attributing the associated probability mass to the null set [70]. Combination rules have been shown to provide

counterintuitive results [72]. For this reason, some other types of combinations rules have been proposed [70].

In trust, DS Theory has been used in conjunction with several other methods. Wang used DS Theory as a precursor to a neural network [47]. In the DS Theory portion of Wang's neural network, a set of representative features were selected from the set of inducing factors. This set of representative features was used as evidence. Then basic belief assignment was carried out using degree to which each input factor belongs to trust and distrust classes. The masses for each class and evidence were combined using a mass combining unit that applied Dempster's rule of combination. The combined masses were then used as input to the fusing units that formed the neural network. How the fusing layers of the neural network work is discussed in Section VI-D4.c.

An alternative application of DS Theory combined with Beta distributions, as discussed in Section VI-C1, was applied by Zhang and Zhou to their binary trust reputation systems [21], [29]. The possible atomic outcomes defined by Zhang and Zhou were that the trustee was either trustworthy, event denoted T, or untrustworthy, event denoted $\neg T$. This means that the power set of interest would simply be $\mathcal{P}(\{T, \neg T\}) = \{\{T\}, \{\neg T\}, \{T, \neg T\}, \emptyset\}$. From general DS Theory, the aim now would then be to assign belief functions to the subset of interest, $\{T\}$.

With the definitions of belief, disbelief, and uncertainty, and because the way the set of outcomes is so simply defined, the belief, disbelief and uncertainty functions could be simply given by each of their first equalities in Equations 52. Then, to assign the mass functions which are necessary to compute belief, disbelief, and uncertainty, bpa values must be assigned. This was done using the number of successful interactions, denoted r, and unsuccessful interactions, denoted s, as evidence. Borrowing from Beta distributions discussed in Section VI-C1, the expectations of successful and unsuccessful interactions were adapted to define the mass functions. Therefore, the belief, disbelief, and uncertainty are

$$b(T) = m(\{T\}) = \frac{r}{r+s+2}, \tag{52}$$

$$d(T) = m(\{\neg T\}) = \frac{s}{r+s+2}, \tag{53}$$

$$u(T) = m(\{T, \neg T\}) = \frac{2}{r+s+2} \tag{54}$$

which were then used by Zhang as a computational tool for reputation scores [29] and by Zhou as a decision-making condition in their trust process [21].

5) BELIEF PROPAGATION

Belief Propagation (BP) algorithms coupled with k-Nearest Neighbours Graph (k-NNG), discussed in Section VI-D2.b, has been used in trust modelling of online content distribution [73]. Trust modelling using belief propagation requires labels for at least a small number of known entities in the network so that information can be propagated throughout some graph.

Belief propagation is a likelihood updating algorithm that uses Bayes Theorem to propagate the effect of new evidence throughout a directed, acyclic graph called a belief network [74], [75]. Using the directed graph obtained from the k-NNG algorithm, the relationship between nodes Ⓐ → Ⓑ will be that the trustworthiness of A implies the trustworthiness of B. Their edge is quantified by the conditional probability $P(B_j|A_i)$, where A_i and B_j are the states of variables A and B [74], [75]. In the context of trustworthiness, $A, B \in \{0, 1\}$ are the trustworthiness variables defined by the indicator random variable in Equation 72. The edge weight can be interpreted in this context as the probability that B_j is in trustworthy or untrustworthy given that A_i is of trustworthy or untrustworthy.

For a single connected network - there is at most 1 undirected path between any two nodes – Pearl describes an BP algorithm that updates probability values at for each node (variable) in one pass and produces probabilities that are consistent with the axioms of probability theory [75]. Following the general fragment of a singly connected graph in Pearl's paper [75], suppose Ⓐ has parents Ⓑ and Ⓒ. Since the graph is acyclic, the graph above Ⓐ can be partitioned into the subgraph of nodes connected (directly or indirectly) to Ⓑ and the subgraph of nodes connected to Ⓐ. Also suppose that Ⓐ has children Ⓧ and Ⓨ. Like with parent nodes of Ⓐ, the subgraph that is a child to Ⓐ can also be partitioned into a subgraph of nodes connected to Ⓧ and a subgraph of nodes connected to Ⓨ.

The data contained in the subgraph of Ⓑ and Ⓒ is denoted D^+_{BA} and D^+_{CA} respectively and the data contained in the subgraph of Ⓧ and Ⓨ is denoted D^-_{AX} and D^-_{AY} respectively. Node Ⓐ separates Ⓑ, Ⓒ and each of its connected subgraphs from Ⓧ and Ⓨ and each of its connected subgraphs. Therefore, the effect of D^+_{BA} and D^+_{CA} and A_i trustworthiness data on the children subgraphs is summarised by A_i and we can write [74], [75]

$$P(D^-_{AX}, D^-_{AY}|A_i, D^+_{BA}, D^+_{CA}) = P(D^-_{AX}|A_i)P(D^-_{AY}|A_i). \tag{55}$$

The goal of belief propagation in our context would be to find out the probability that A is trustworthy or untrustworthy given all the available data. We can then denote belief for node and state A_i as $\mathrm{BEL}(A_i)$ and define this belief using conditional probability to be

$$\mathrm{BEL}(A_i) \triangleq P(A_i|D^-_{AX}, D^-_{AY}, D^+_{BA}, D^+_{CA}). \tag{56}$$

Following the derivation in [75], obtain

$$\mathrm{BEL}(A_i) = \alpha P(D^-_{AX}|A_i)P(D^-_{AY}|A_i) \cdot \left[\sum_{jk} P(A_i|B_j, C_k)P(B_j|D^+_{BA})P(C_k|D^+_{CA})\right] \tag{57}$$

where α is a normalization constant. Equation 57 demonstrates how the belief of the trustworthiness of A is determined by causal data from the parent subgraphs, diagnostic data from children subgraphs and the fixed conditional probability

matrix that determines how A is affected by its immediate causes: B and C.

It can be seen that the children Ⓧ and Ⓨ then need to propagate $P(D_{AX}^{-}|A_i)$ and $P(D_{AY}^{-}|A_i)$ information to Ⓐ and parent nodes Ⓑ and Ⓒ need to propagate $P(B_j|D_{BA}^{+})$ and $P(C_k|D_{CA}^{+})$ to Ⓐ. According to Pearl, the information propagated from parent nodes is denoted [74], [75]

$$\lambda_X(A_i) = P(D_{AX}^{-}|A_i), \quad \lambda_Y(A_i) = P(D_{AY}^{-}|A_i) \tag{58}$$

and that propagated from children nodes is denoted

$$\pi_A(B_j) = P(B_j|D_{BA}^{+}), \quad \pi_A(C_k) = P(C_k|D_{CA}^{+}) \tag{59}$$

for ease of defining the updating equations.

When some node, for example Ⓐ, receives new information, it needs to update its parent nodes and children nodes. As derived by Pearl, the updating function for propagation by Ⓐ to each of its respective parents is [75]

$$\lambda_A(B_i) = \lambda \sum_j \left[\pi_A(C_j) \sum_k \lambda_X(A_k)\lambda(A_k)P(A_k|B_i, C_j) \right] \tag{60}$$

and the updating function for propagation to each of its children is

$$\pi_X(A_i) = \alpha\lambda_Y(A_j) \left[\sum_j kP(A_i|B_i, C_k)\pi_A(B_j)\pi_A(C_k) \right]. \tag{61}$$

Belief propagation is touted by its creators to mimic the way people make decisions [75]. It is also guaranteed to achieve equilibrium in time proportional to the network diameter [75] compared to other machine learning methods that may not reach convergence. The calculations necessary for each node is also simple and so is hardware implementable [75] and can address the issues raised in Section II.

It should however be noted that the above calculations only work for singly connected graphs. This is highly unlikely in the context of trust. Between any two people there can be multiple trust relationships. Methods to extend the method for singly connected graphs to multiply connected graphs have been proposed briefly by Pearl [75]. Relatively faster variants of the belief propagation algorithm have also been proposed such the Fast Belief Propagation algorithm proposed in by Gisel [73].

In Fast Belief Propagation [76], the final belief of nodes is approximated by solving for $\boldsymbol{b}_h$ in the linear system

$$\left[\mathbf{I} + a\mathbf{D} - c'\mathbf{A}\right]\mathbf{b}_h = \boldsymbol{\phi}_h \tag{62}$$

where $\boldsymbol{\phi}_h$ is a vector containing the prior beliefs about nodes, $\boldsymbol{D}$ is a diagonal matrix, $\boldsymbol{A}$ is the adjacency matrix and $\boldsymbol{I}$ is the identity matrix. $a = \frac{4h_h^2}{1-4h_h^2}$ and $c' = \frac{2h_h}{1-4h_h^2}$ are constants defined to account for h_h which is the degree of similarity between two nodes. h_h is chosen using conditions in [76] before solving to ensure convergence.

D. MACHINE LEARNING METHODS

Machine learning models are useful for learning different trust factors and assigning weights to them based on data. Machine learning methods vary, each with their unique benefits. The main benefit of machine learning is the construction of trust models based off on data rather than human understandings of trust which may be non-representative and too complex to model.

1) CLASSIFICATION

Classification in trust modelling is used to gather multiple factors and use them to classify nodes as trustworthy or untrustworthy. There are several different methods each with their individual benefits.

a: SUPPORT VECTOR MACHINE (SVM)

SVM is a classification method used in trust modelling [58], [77]. Data was obtained from online communities by Liu *et al.* [77] and an IoT network from a convention by Jayasinghe *et al.* [58]. Decisions to interact with reviews on online communities were used as indicators for trust decision in Liu's paper [77]. In [58], the trust values were created using k-means clustering so that while there was no explicit trust value in the data set, the effectiveness of the selected features could still be tested to an extent.

In the trust model by Liu variables available in the data set were used directly [77]. However, Jayasinghe described features that were thought to described trust, such as past experience and centrality, using simple mathematical constructions [58]. In both cases, the selected or modelled features were mapped onto a feature space and SVM applied.

In SVM, the goal of the algorithm is to find a hyperplane that correctly divides data points into trustworthy (represented as 1) and untrustworthy (represented as -1) with maximum margin between the two classes. Where the trust features are represented in the vector $\mathbf{x}$ and the trust decision is $y \in \{-1, 1\}$, the set of n trust data points will be

$$(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \ldots, (\mathbf{x}_n, y_n). \tag{63}$$

To prevent data points from falling into the margins of the hyperplane and being incorrectly classified, the hyperplane is restricted (for the linear form) such that [78]

$$y_i\left(\mathbf{w}^T\mathbf{x}_i - b\right) \geq 1, \quad \text{for all } 1 \leq i \leq n. \tag{64}$$

To maximise the split between the data points, the distance between the hyperplanes, given by $\frac{2}{\|\mathbf{w}\|}$, should be maximised and so, $\|\mathbf{w}\|$ needs to be minimized. Finally, in linear form, the optimal hyperplane that divides the data points correctly with maximal margin will be [78]

$$\mathbf{w^T x} - \mathbf{b} = \mathbf{0}. \tag{65}$$

By changing the dot product of vectors $\mathbf{u}$ and $\mathbf{v}$ to a nonlinear kernel $K(\mathbf{u}, \mathbf{v})$, the SVM algorithm can be modified to best suit the digital environment [78]. The radial basis

function kernel [78]

$$K(\mathbf{u}, \mathbf{v}) = \exp\left(-\frac{\|\mathbf{u} - \mathbf{v}\|^2}{\sigma^2}\right) \tag{66}$$

was used by Liu [77]. Other knowledge about the digital environment and the way trust decisions are made can also be implemented within the kernel function to make for a more accurate trust model [78].

b: RANKING SVM (RSVM)

RSVM is a variant of SVM that was used by [26] for establishing trustworthiness of different users on social networks. This variant of SVM was used so that users could be accurately ranked according to features and this ranking used to establish corresponding continuous trust values. The trust values can then be said to be consistent with the rankings established by the features in the trust model.

RSVM for trust modelling can be performed as follows. Suppose for some trustor j in a data set, there are n trustees to be ranked where each of the trustees have known ranks r_j^*. Then, RSVM should aim to find the ranking function $f_{\mathbf{w}}(j)$ such that the resulting rank $r_{f_{\mathbf{w}}(j)}$ and r^* have as few contradicting ordered pairs as possible (discordant pairs) [79]. $\mathbf{w}$ is learned by RSVM and determines $f_{\mathbf{w}}(j)$ which then determines the rank of trustees by the projection of data points in the feature space onto $\mathbf{w}$ [79]. The RSVM optimization problem aims to maximize the number of pairs such that [79]

$$\forall j \left(\forall (d_x, d_y) \in r_j^* : (d_x, d_y) \in f_{\mathbf{w}}(j)\right). \tag{67}$$

This is NP-hard. The solution is approximated by introducing non-negative, slack variables $\xi_{i,j,k}$ and SVM margin maximisation [79] to instead find $\mathbf{w}$ that minimizes [26]

$$V(\mathbf{w}, \xi) = \frac{1}{2}\|\mathbf{w}\|^2 + C\sum \xi_{i,j,k} \tag{68}$$

subjected to

$$\forall (j > k) : \mathbf{w}^T \mathbf{x}_{ij} - \mathbf{w}^T \mathbf{x}_{ik} \geq 1 - \xi_{i,j,k}, \quad \xi_{j,k} \geq 0 \tag{69}$$

where $\mathbf{x}_{ij}$ is the normalized feature vector between users i and j.

c: DECISION TREE LEARNING

Decision tree learning has been used by Zhang and Fang to classify users into trustworthy and untrustworthy classes [20], [54]. Given a data set with known trust values, the decision tree training algorithm finds the splitting condition that divides the data subset the best for the particular level, at the particular node. This recursive splitting stops when no other splitting condition can add value to the prediction.

Decision trees are useful in considering factors regardless of whether they are binary, discrete, or continuous. They are also useful because they allow variables to split in a hierarchical manner, considering certain variables only after some other variable has been considered. The end product is a series of splitting conditions and an order in which to perform this splitting condition that will likely determine if a trustee is trustworthy with the highest accuracy.

d: Naïve BAYES CLASSIFIER

Naïve Bayes classifier has been used to classify trustees based on their features using Bayesian probability [24], [41], [77], [80]. Suppose there are features, $x_1, x_2, \ldots, x_k$ to be used to classify a trustee, Bayes rule is used to compute the probability that the trustee is trustworthy given the available feature evidence. This is given by

$$p\,(T|x_1, x_2, \ldots, x_k) = \frac{p(T)p(x_1, x_2, \ldots, x_k|T)}{p(x_1, x_2, \ldots, x_k)} \tag{70}$$

$$= \frac{p\,(x_1, x_2, \ldots, px_k, T)}{p(x_1, x_2, \ldots, x_k)} \tag{71}$$

where 71 is achieved due to the definition of conditional probability and T is the indicator random variable for the trustworthiness of the trustee, defined

$$T = \begin{cases} 1, & \text{if trustworthy} \\ 0, & \text{if untrustworthy.} \end{cases} \tag{72}$$

The denominator is just a normalization factor, so we ignore it. To compute the numerator easily, it is naïvely assumed that all features are mutually independent, conditioned on T. Therefore, the posterior is given

$$p\,(T|x_1, x_2, \ldots, x_k) \propto p(T) \cdot \prod_{i=1}^{k} p\,(x_i|T)\,. \tag{73}$$

Finally, the classifier will provide the trust decision, $\hat{T}$,

$$\hat{T} = \begin{cases} 1, & \text{if } p\,(1|\mathbf{x}) \geq p\,(0|\mathbf{x}) \\ 0, & \text{if } p\,(1|\mathbf{x}) < p\,(0|\mathbf{x}) \end{cases} \tag{74}$$

where $\mathbf{x} = (x_1, x_2, \ldots, x_k)$ is the vector of features [80].

Since trust modelling is a form of social control and modelling where agents act in a highly interdependent manner, the mutual independence assumption is likely too strong. Nevertheless, the classifier and use of Bayes Rule is still applicable. In fact, the use of Bayes rule is beneficial as it allows for a priori knowledge, evidence, and the strength of evidence to be accounted for.

2) CLUSTERING

While clustering may not allow for classification of trustees, clustering is still a useful tool for combining and identifying similar agents and entities based on defined criteria for further analysis.

a: K-MEANS CLUSTERING

K-means clustering is a common clustering algorithm used to group similar data by their numerical features [81]. Each cluster of data is represented by its centroid, typically a weighted average of all the points within the cluster. The number of clusters, conventionally denoted K, is user-defined. The algorithm works by first selecting K initial centroids. Then, the K clusters are formed by assigning each data point to its closest centroid. The centroid of each cluster is recomputed, and the assignment process is repeated until the centroid does not change.

Jayasinghe created numerical features describing interactions and connections between trustee and trustor agents in an IoT setting [58]. Data however typically does not come with trust value tagged to each interaction. To perform further classification, Jayasinghe first performed k-means clustering on the data to group similar interactions into two to three clusters. Doing so helps to distinguish between better and worse performing interactions with respect to the features. Given that the features reasonable represent trust, the clusters then help indicate which interactions were more likely trustworthy or untrustworthy, despite the lack of explicitly recorded data. If there were three clusters, one of them could be used to indicate neutrality about trustworthiness.

Obtaining the clusters created a set of labelled data which was then used in SVM for classification. The use of SVM for classification has been discussed in Section VI-D1.a. Briefly, SVM helped to find a boundary of features that could distinguish between clusters. This boundary then offers a standard for features to determine if a future interaction should reasonably belong to one cluster or the other.

b: K-NEAREST NEIGHBOUR GRAPHS

Another way to group users together borrows from k-NN classification to build a graph. This graph can function as a method to perform propagation or other trust evaluation methods. Given a set of data, the nearest neighbour of $v_i \in V$ is a point $v_j \in V, j \neq i$, with minimum distance from v_i. Generally, Euclidean distance is used and to ensure uniqueness of the nearest neighbour, the maximum index when there are ties is used. The edges can then be defined $e(v_i) = \langle v_i, v_j \rangle \in E$ and the nearest-neighbour graph would be the tuple $G = (V, E)$ [82]. The k-NNG graph is simply the nearest-neighbour graph with k edges instead of just one edge.

k-NNG coupled with Belief Propagation (BP) algorithms, discussed in Section VI-C5, has been used in trust modelling of online content distribution [73]. The idea behind using k-NNG was primarily to group similar entities into a k-NNG so that known labels could be propagated over the rest of the network. This is then useful to determine the trustworthiness of entities even if direct data about them is not explicitly available.

In [73], the words in online articles were gathered and passed through tensor decomposition to group similar articles. The articles were then represented as nodes in the graph, and which were grouped into a k-NNG. Articles that were sufficiently similar to known fake news articles were labelled as fake news. To generate a k-NNG graph, metrics for distance are needed. Besides what was done in [73], it is also possible to incorporate trust features, such as those described in Section III, within a distance metric like Euclidean distance. The k-NNG graph is then a directed graph $G = (V, E)$ where for each agent $v \in V$, there are k edges pointing to the k most similar agents denoted $u_i \in V$. It is then known the agents that are feature-wise most similar to each other. This similarity can then be used to make trust decisions, even if data such as quality of direct interactions, is not available.

3) REINFORCEMENT LEARNING

Reinforcement learning is a type of machine learning method where an agent interacts with the environment in a discrete series of time steps to achieve some goal [83]. In reinforcement learning, the agent may be in *states*, take *actions* and use the received *rewards* to evaluate the quality of their choices [83]. The agent uses the states, actions, and rewards to formulate a *policy* which maps states to actions [83], [84]. Using this policy, the *return* is the expected future rewards that the agent should aim to maximise [83]. The policy has *value functions* which assigns to states or state-action pairs the expected return if the agent follows the policy [83], [84].

One of the prevalent methods of performing reinforcement learning is Q-learning where the expected value of each action in different states is stored and incrementally updated [84]. The policy is formed from executing the action with the highest expected value [84]. The value function used by Q-learning is a function of the immediate reward and the expected reward based on the new state [84]. This is expressed with

$$Q(\boldsymbol{x}_t, \boldsymbol{u}_t) = (1-\alpha)Q(\boldsymbol{x}_t, \boldsymbol{u}_t) + \alpha(R + \gamma Q(\boldsymbol{x}_{t+1}, \boldsymbol{u}_{t+1})) \quad (75)$$

where Q is the expected value of performing action $u \in \boldsymbol{u}$ in state $x \in \boldsymbol{x}$, R is the reward, α is the learning rate and γ is the discounting factor [84]. The learning rate determines the weight given to new information and the discounting factor determines how the emphasis the agent places on future rewards.

a: NETWORK EXPLORATION

Reinforcement learning has been used to learn the trustworthiness of target agents based on the strength of trust paths leading into the target agent [34]. This is done by first initializing a trust graph such that each node represents a user, and each edge represents the relationship between two users. Each edge is weighted with the direct trust value τ between the two users.

Two methods for Q-learning were proposed in [34]. For both methods, the agent starts from the source node and selects one of its neighbours using ε-policy.

In the min-max aggregation approach, upon choosing the next node v_j from the current node u_i, the agent receives a reward $r_{strength}$

$$r_{strength}(ui, v_j) \quad (76)$$

$$= \begin{cases} -1, & \text{if } v_j \text{ has already been visited} \\ \tau(u_i, v_j), & \text{otherwise} \end{cases} \quad (77)$$

which rewards the agent based on the trust outcome of the node choice. The -1 punishes the agent for visiting existing nodes to avoid the formation of cycles. The agent can then learn from this reward and update the expected return from

taking this particular trust edge with

$$Q^t_{strength}(u_i, v_j) \tag{78}$$
$$= (1-\alpha)\cdot Q^{t-1}_{strength}(u_i) \tag{79}$$
$$+\alpha_t \cdot \min\left(r_{strength}(u_i, v_j), Q^{t-1}_{strength}(v_j, v^*)\right) \tag{80}$$

which defines the terms for Q-learning in Equation 75 for the context of trust. The expected future reward is determined by the node's trustworthiness or the benefit of following the optimal policy for the subsequent action [34]. This learning process is repeated until one of the neighbouring nodes of the target is reached. A path to the target node is now available. Using the learned expected edge rewards, nodes are chosen using the optimal policy

$$\pi^*(u_i) = \underset{v_j \in \text{Neighbour}(u_i)}{\arg\max} Q_{strength}(u_i, v_j) \tag{81}$$

starting from the source node to reach the target node. The trust strength of the target is determined by the trust strength of the path, optimised by reinforcement learning, which is defined to be the minimum value of the trust edge.

A more complex method proposed by [34] for indirect trust computation is the weighted mean aggregation method which attempts to find the path strength to all neighbouring nodes. Here the reward for each node is redefined to be

$$r_{path}(u_i, v_j) \tag{82}$$
$$= \begin{cases} -1, & \text{if } v_j \text{ has already been visited} \\ 1, & \text{if target's neighbour node is reached} \\ 0, & \text{otherwise} \end{cases} \tag{83}$$

to learn whether or node reaching a particular node would result in a cycle being formed and avoid such a situation. The learning proceeds via the Q-learning equation

$$Q^t_{path}(u_i, v_j) = (1-\alpha)Q^{t-1}_{path}(u_i, v_j) + \alpha(r_{path}(u_i, v_j) + \gamma Q^*_{path}(v_j, v*)) \tag{84}$$

which is similar in the way it employs Q^* value like in Equation 78 but uses an averaging method instead like that seen in Equation 75. In the weighted average method, instead of moving on to the next node and repeating the learning process, the agent must first repeat visiting each node from the current node. This time it visits nodes that have a $Q^t_{path} > 0$ to avoid nodes that are likely to form cycles. After doing so, the agent can learn the strength of the trust path using another Q-learning equation

$$V^t_{strength} \tag{85}$$
$$= (1-\alpha)V^{t-1}_{strength}(u_i) \tag{86}$$
$$+\alpha\left(\frac{\sum_{v_j \in C(u_i)} r_{strength}(u_i, v_j)V^{t-1}_{strength}(v_j)}{\sum_{v_j \in C(u_i)} r_{strength}(u_i, v_j)}\right) \tag{87}$$

where $C(u_i)$ is the set of neighbours u_i trusts. Here, the immediate reward is defined

$$r_{strength}(u_i, v_j) = \tau(u_i, v_j) \tag{88}$$

until the node v_j is reached. At the target node, the final $r_{strength}$ is defined to be 1 and $V_{strength}(v_j) = \tau(u_i, v_j)$.

4) ARTIFICIAL NEURAL NETWORKS

Artificial neural networks are a machine learning paradigm that centres around mimicking the way the human brain learns. An artificial neural network consists of simple processing units called *neurons* and *weighted connections* between those neurons. In neural networks, there may be multiple layers of neurons, such as the input layer, hidden layer, and output layer, that are connected to each other and within the layer depending on the type of neural network. Between two neurons (i, j), the weight is defined by a function $\omega((i, j))$ [85]. Each neuron receives either inputs to the network or the output of other neurons and processes these as inputs to the *propagation function* [85]. The output of the propagation function and the previous *activation state* of the neuron acts as input to the *activation function* which will output the new activation state of the neuron [85]. Finally, the activation acts as input to the *output function* which gives the data output for other neurons [85].

a: BASIC NEURAL NETWORK TRUST MODEL

The most basic neural network has been used for trust modelling in [20], [25], [47], [55]. In [25], different features were modelled using the methods in Section VI-A2 and other heuristics. This generated a set of features which then underwent exponential time-based weighting to reduce the weight that the older feature values have. These feature values were then used as input to a neural network forming the input layer. This neural network then outputs which nodes with no previous trust relationship is likely to have a trust relationship and the strength of this newly formed trust.

In Zhang's paper, the neural network is implemented in context of the vehicular network digital environment to determine if the vehicle is trustworthy based on the receiving and delivery time of the messages by the vehicle [20]. A simple ratio of Euclidean distances which represents the time taken for a particular vehicular node to deliver a particular message it receives was used as a proxy for trust values. Since there may be transmission errors, the computed trust value may not always be consistent with the expected trust values. Therefore, the Euclidean distances are adjusted using a neural network to obtain a more reflective trust value.

The message receiving vehicular nodes was used as the neurons for the input layer to the neural network. These took in the numerator Euclidean distance (representing the time take for the receiving vehicular node to receive message) as inputs. The next layer - the hidden layer - consisted of the message forwarding nodes that take in the denominator Euclidean distance (representing the time taken for the message to be forwarded after it is first delivered by the previous sender). In the output layer, the nodes receive the ratio, which is the trust value. When the trust values that were expected and the actual trust values are inconsistent, back propagation is performed, and the Euclidean distances are adjusted by

the respective nodes in the hidden and input layer. This way, a more representative trust value is obtained and stored for future interactions.

b: FUZZY LOGIC AND NEURAL NETWORKS

Neural networks can be used in conjunction with other computational methods. In Wu's paper, features obtained directly from data sets of a user were used as input into a fuzzy logic module [55]. In this module, fuzzy logic was used to determine the degree to which each particular feature was considered ''low'', ''average'', ''high'' or ''very high''. This was done using membership functions for each category which produced the degree of membership for each feature in each of the categories. How this is done is further discussed in Section VI-A3.

These membership degrees were stored in the neurons in the input layer of the neural network. This neural network fused these inputs with a rule layer which consists of 45 neurons to cover all the combinations of inputs. The rule layer acts as input to the output layer which only has four neurons. Each neuron in the final output layer represents the membership function of the trustworthiness of the user in each of the four categories - ''low trustworthiness'', ''average trustworthiness'' and so on. The trustworthiness of the user is the category in which the user has the highest membership degree. Training of this neural network was done with the gradient descent method and back propagation algorithm.

c: DEMPSTER SHAFFER THEORY AND NEURAL NETWORKS

In [47], features were constructed using functions from Section VI-A2 and heuristics about a user. These features were used as input to a Dempster Shaffer Theory module. In the module, mass functions were assigned to each feature to obtain an evidence prototype. Dempster's rule of combination was then used to combine the different sources of evidence to derive a joint mass function for each feature value. More details of Dempster Shaffer Theory will be discussed in Section VI-C4.

The joint mass function for each feature value formed a single neuron in the input layer of the neural network. In the next layer, the local fusing layer, the neurons are trained by the data set based on the joint mass functions from the inputs. The best outputs from the local fusing layer form the masses which are trained in the neurons in the global fusing layer. In each node of the fusing layer, a logistic sigmoid activation function was used in each node. Finally, the output layer gives the mass functions for the event that the user is trustworthy and the event that the user is untrustworthy. The trustworthiness is thus determined by the most likely trustworthiness event. This neural network was also learned using the standard back propagation with gradient descent approach.

d: BERNOULLI NEURAL NETWORK

Besides the most basic neural network, more complicated constructions have been implemented in trust management. One such example is the Bernoulli neural network implemented by [86]. In the Bernoulli neural network, there are three layers - the input, hidden and output layer. The defining characteristic of the Bernoulli neural networks is that the hidden layer uses Bernoulli polynomials as activation functions [87]. The $n-1$-th hidden layer neuron can be computed recursively with

$$\phi_{n-1}(x) = x^{n-1} - \sum_{k=0}^{n-2} \binom{n}{k} \phi_k(x)/n \tag{89}$$

which is the recursive form of the Bernoulli polynomial. The input and output layer each have one neuron, both activated by a simple linear function $f(x) = x$. The weight between the input neuron and the hidden layer neurons is set to be one. The weights between the hidden layer and the output layer neuron are set to be ω_j where $j = 0, 1, \ldots, n-1$ which should be decided or adjusted. Let the input into the network be x and the output be y. From the structure of the entire neural network, the output of the network is

$$y = \omega_0\phi_0(x) + \omega_1\phi_1(x) + \cdots + \omega_{n-1}\phi_{n-1}(x). \tag{90}$$

A Bernoulli neural network is implemented as part of the trust model in [86]. In the model, advisor agents train models individually, using contextual features, based on past interactions to output a predicted conditional probability about the trustworthiness of the trustee. This recommendation acts as evidence which is aggregated together with the truster's own first-hand evidence using the Bernoulli neural network. The neural network trains its weights using gradient descent back propagation with a cross-entropy loss function.

e: DEEP BELIEF NEURAL NETWORK

Deep Belief Neural Networks(DBN) are neural networks with many hidden layers to perform a deep hierarchical representation of the input data [88]. One type of DBN uses the Restricted Boltzmann machines(RBM) in each layer of the neural network [88], [89]. RBMs is a stochastic network that consists of two layers of nodes - a hidden layer and a visible layer [89]. Each node in one layer is connected to all the nodes in the other layer with weights and vice versa so that the values in one layer affects the values of the other. Nodes take on a value of either 0 or 1 at different time intervals with probability conditioned on the nodes in the other layer [89]. RBM learns by unsupervised learning by presenting training patterns to the visible nodes [88]. The weights of connections and the biases in each layer is adjusted to minimize the energy of the network [88].

DBN uses RBMs in each of its layers by constructing the neural network such that top hidden layer of one RBM acts as visible bottom layer of the RBM layer above it [89]. Training of the DBN is done by first iteratively training each RBM layer using unsupervised learning to obtain ideal parameters for extracting features from the data [89]. Then supervised learning classification using methods such as back

propagation with gradient descent is performed to fine tune the weights throughout the whole network [89].

This DBN training method has been used in trust modelling to achieve context aware trust values [21]. In Zhou's paper, the DBN model is trained to link the features of a situation - features of the trustor, trustee, and context of interaction - with trust values [21]. When there is insufficient evidence for a specific interaction - the specific trustor, trustee, and context features - the DBN neural network can still derive a trust value even with little information by using incompletely related past interactions.

Zhou achieved this using a DBN with an input layer, three hidden layers and a label layer [21]. The input layer takes in normalized feature values and passes them into the first hidden layer. The hierarchical nature of DBN is used here to progressively filter more important features at each layer. At the first hidden layer, there are nodes corresponding to the total number of features. At the second hidden layer, only more significant features are selected by restricting the total number of nodes to a fraction of the total number of features. At the last hidden layer, the nodes are fused to decide a trust value for the specific context. The final label layer outputs a single value representing the trust value for the current interaction.

f: GROWING HIERARCHICAL SELF-ORGANISING MAP

Self-organising maps(SOM) are an unsupervised learning, neural network model that preserves the topological in the input space into the output space [90], [91]. This topological preservation means that the similarity of the input data is mirrored to a very large extent in geographical vicinity within the representation space [92]. In SOMs, neurons are organised in a two-dimensional rectangular or hexagonal grid and each neuron is assigned a weight vector of the same dimension as the input vector [91]. At each iteration, SOM finds the weight vector that is closest to the input vector in the data [91]. The SOM algorithm updates the weight of this vector and that of its surrounding nodes while maintaining the connections from the original grid [91]. This is repeated until the map converges

Growing Hierarchical Self-Organising Map (GH-SOM) is a variant of SOM that independently determines the topological space (which needs to be decided prior to training in SOM) and mirrors the hierarchical relations in the data [92]. This is done by a hierarchical structure of multiple layers where each layer is several independent SOMs. The first layer contains one SOM and for every neuron in the SOM, an SOM can be added to form the next layer. This expansion helps represent the subset of data at the specified level of granularity [93].

Capua utilised GH-SOM to classify content on social networks as harassment or non-harassment. By collecting different features about the content, [93] collected the features into an input vector. These input vectors representing features of a specific content piece are used as input data to a GH-SOM. The GH-SOM then groups in a hierarchical manner, independently, the different content based on their features.

VII. FUTURE WORK AND CONCLUSION

In our survey paper, we covered the basic definitions and properties of trust, analysed social theories of trust while discussing their impact on digital trust. We analysed a broad range of environments in the digital world, including how these environments connect. Then using our understanding of the digital world, we illustrated how trust was needed as a soft security mechanism. Based on the established understanding of the digital world, we defined different types of trust relationships and discussed the factors needed to make a complete, representative model. To address the challenges of trust modelling, we came up with an evaluation criteria for trust models. Finally, we explained different trust modelling methods and how they have been used, including their theoretical basis and practical usefulness. In writing this survey, we hope to have offered a well-rounded survey of trust management from analysis of application environment to actual modelling methodology.

There are several areas for improvement. First, did not manage to align our digital environment analysis and factors with modelling methodology. An additional step in our survey would have been to tie in which factors have been modelled and how they can be modelled using existing mathematical methods. However, we were not able to do so as the scope of the survey would have been too big. Another aspect of trust that we could have explored was the suitability of each method for security. While we covered the theoretical basis of each model, we did not manage to perform any analysis on the suitability of each method and how each method would evaluate trust in the face of security attacks. Combinations of methods — whether they complement or conflict each other — were also not considered. Further analysis can be performed in the future to determine how sensitive each method is to different trends and behaviour in the system. This pushes research in the direction of the most suitable technical method for trust management.

The lack of real-world data about trust makes research challenging. While some real-world data is available about some digital environments, it is rare to have actual data with corresponding trust evaluation, much less data sets that are recent. In the future, methods to expand on data sets or to meaningfully make use of existing data sets should be analysed. Alternatively, using real world data, sample and simulation data can be created. Finally, from our derivation of different factors based on the digital world, we find that there are many tangentially related fields that can aid trust management. Future surveys in trust can consider looking into behavioural management, anomaly-detection, and risk management models to borrow relevant theories for trust.

ACKNOWLEDGMENT

The authors thank the editor and anonymous referees of this journal whose comments substantially improved this article.

REFERENCES

[1] D. Wang, T. Muller, Y. Liu, and J. Zhang, "Towards robust and effective trust management for security: A survey," in *Proc. IEEE 13th Int. Conf. Trust, Secur. Privacy Comput. Commun.*, Sep. 2014, pp. 511–518.

[2] A. Jøsang, "Robustness of trust and reputation systems: Does it matter," in *Proc. 6th Int. Conf. Trust Manage. (TM)* in Trust Management VI, vols. AICT–374, T. Dimitrakos, R. Moona, D. Patel, and D. H. McKnight, Eds. Surat, India: Springer, May 2012, pp. 253–262. [Online]. Available: https://hal.inria.fr/hal-01517648

[3] R. A. Heineman, "The logic and limits of trust. By Bernard barber," *Amer. Political Sci. Rev.*, vol. 78, no. 1, pp. 209–210, 1984.

[4] N. Luhmann, "Trust and power," *Stud. Sov. Thought*, vol. 23, no. 3, pp. 266–270, 1982.

[5] S. S. Tangade and S. S. Manvi, "A survey on attacks, security and trust management solutions in VANETs," in *Proc. 4th Int. Conf. Comput., Commun. Netw. Technol. (ICCCNT)*, Jul. 2013, pp. 1–6.

[6] Z. Yan, P. Zhang, and A. V. Vasilakos, "A survey on trust management for Internet of Things," *J. Netw. Comput. Appl.*, vol. 42, pp. 120–134, Jun. 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1084804514000575

[7] Y. Ruan and A. Durresi, "A survey of trust management systems for online social communities—Trust modeling, trust inference and attacks," *Knowl.-Based Syst.*, vol. 106, pp. 150–163, Aug. 2016, doi: 10.1016/j.knosys.2016.05.042.

[8] K. Hoffman, D. Zage, and C. Nita-Rotaru, "A survey of attack and defense techniques for reputation systems," *ACM Comput. Surveys*, vol. 42, no. 1, pp. 1–31, Dec. 2009, doi: 10.1145/1592451.1592452.

[9] J. Guo and I.-R. Chen, "A classification of trust computation models for service-oriented Internet of Things systems," in *Proc. IEEE Int. Conf. Services Comput.*, Jun. 2015, pp. 324–331.

[10] J. Wang, X. Jing, Z. Yan, Y. Fu, W. Pedrycz, and L. T. Yang, "A survey on trust evaluation based on machine learning," *ACM Comput. Surv.*, vol. 53, no. 5, pp. 1–36, Oct. 2020, doi: 10.1145/3408292.

[11] D. D. S. Braga, M. Niemann, B. Hellingrath, and F. B. D. L. Neto, "Survey on computational trust and reputation models," *ACM Comput. Surv.*, vol. 51, no. 5, pp. 1–40, Jan. 2019, doi: 10.1145/3236008.

[12] S. P. Altmann, "Simmel's philosophy of money," *Amer. J. Sociol.*, vol. 9, no. 1, pp. 46–68, 1903. [Online]. Available: http://www.jstor.org/stable/2762310

[13] G. Möllering, "The nature of trust: From Georg Simmel to a theory of expectation, interpretation and suspension," *Sociology*, vol. 35, no. 2, pp. 403–420, 2001. [Online]. Available: http://www.jstor.org/stable/42856292

[14] A. Giddens, *The Consequences Modernity*. Stanford, CA, USA: Stanford Univ. Press, 1990.

[15] F. Fukuyama, *Trust: Social Virtues Creation Prosperity*. New York, NY, USA: Free Press, 1995.

[16] E. W. Lehman and P. Sztompka, "Trust: A sociological theory," *Contemp. Sociol.*, vol. 30, no. 4, p. 418, Jul. 2001.

[17] *Series Y: Global Information Infrastructure, Internet Protocol Aspects, Next-Generation Networks, Internet of Things and Smart Cities*, International Telecommunication Union Telecommunication Standardization (ITU-T), Geneva, Switzerland, 2017.

[18] X. Kang and Y. Wu, "A trust-based pollution attack prevention scheme in peer-to-peer streaming networks," 2014, *arXiv:1408.0726*. [Online]. Available: http://arxiv.org/abs/1408.0726

[19] M. Nitti, R. Girau, L. Atzori, A. Iera, and G. Morabito, "A subjective model for trustworthiness evaluation in the social Internet of Things," in *Proc. IEEE 23rd Int. Symp. Pers., Indoor Mobile Radio Commun. (PIMRC)*, Sep. 2012, pp. 18–23.

[20] D. Zhang, F. R. Yu, and R. Yang, "A machine learning approach for software-defined vehicular ad hoc networks with trust management," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2018, pp. 1–6.

[21] P. Zhou, X. Gu, J. Zhang, and M. Fei, "A priori trust inference with context-aware stereotypical deep learning," *Knowl.-Based Syst.*, vol. 88, pp. 97–106, Nov. 2015, doi: 10.1016/j.knosys.2015.08.003.

[22] E. Mokhtari, Z. Noorian, B. T. Ladani, and M. A. Nematbakhsh, "A context-aware reputation-based model of trust for open multi-agent environments," in *Proc. 24th Can. Conf. Adv. Artif. Intell.* Berlin, Germany: Springer, 2011, pp. 301–312.

[23] A. M. Aref and T. T. Tran, "A decentralized trustworthiness estimation model for open, multiagent systems (DTMAS)," *J. Trust Manage.*, vol. 2, no. 1, p. 3, Dec. 2015. [Online]. Available: https://app.dimensions.ai/details/publication/pub.1046152003 and https://journalof trustmanagement.springeropen.com/track/pdf/10.1186/s40493-0%15-0014-4

[24] X. Chen, Y. Yuan, L. Lu, and J. Yang, "A multidimensional trust evaluation framework for online social networks based on machine learning," *IEEE Access*, vol. 7, pp. 175499–175513, 2019.

[25] K. Zolfaghar and A. Aghaie, "Evolution of trust networks in social web applications using supervised learning," *Procedia Comput. Sci.*, vol. 3, pp. 833–839, Jan. 2011. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1877050910005120

[26] X. Li, H. Fang, Q. Yang, and J. Zhang, "Who is your best friend?: Ranking social network friends according to trust relationship," in *Proc. 26th Conf. User Modeling, Adaptation Personalization*, Jul. 2018, pp. 301–309, doi: 10.1145/3209219.3209243.

[27] L. Mui, M. Mohtashemi, and A. Halberstadt, "A computational model of trust and reputation," in *Proc. 35th Annu. Hawaii Int. Conf. Syst. Sci.*, Jan. 2002, pp. 2431–2439.

[28] F. E. Walter, S. Battiston, and F. Schweitzer, "Personalised and dynamic trust in social networks," 2009, *arXiv:0902.1475*. [Online]. Available: http://arxiv.org/abs/0902.1475

[29] J. Zhang and R. Cohen, "Evaluating the trustworthiness of advice about seller agents in e-marketplaces: A personalized approach," *Electron. Commerce Res. Appl.*, vol. 7, no. 3, pp. 330–340, 2008.

[30] J. Sabater and C. Sierra, "REGRET: Reputation in gregarious societies," in *Proc. 5th Int. Conf. Auto. Agents (AGENTS)*, 2001, pp. 194–195, doi: 10.1145/375735.376110.

[31] T. D. Huynh, N. R. Jennings, and N. R. Shadbolt, "An integrated trust and reputation model for open multi-agent systems," *Auton. Agents Multi-Agent Syst.*, vol. 13, no. 2, pp. 119–154, 2006. [Online]. Available: https://eprints.soton.ac.uk/262593/

[32] S. Che, R. Feng, X. Liang, and X. Wang, "A lightweight trust management based on Bayesian and entropy for wireless sensor networks," *Secur. Commun. Netw.*, vol. 8, no. 2, pp. 168–175, 2015. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/sec.969

[33] H. Jameel, L. Xuan Hung, U. Kalim, A. Sajjad, S. Lee, and Y.-K. Lee, "A trust model for ubiquitous systems based on vectors of trust values," in *Proc. 7th IEEE Int. Symp. Multimedia (ISM)*, Dec. 2005, p. 6.

[34] Y. A. Kim and H. S. Song, "Strategies for predicting local trust based on trust propagation in social networks," *Knowl.-Based Syst.*, vol. 24, no. 8, pp. 1360–1371, Dec. 2011. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0950705111001249

[35] P. Gao, H. Miao, J. S. Baras, and J. Golbeck, "STAR: Semiring trust inference for trust-aware social recommenders," in *Proc. 10th ACM Conf. Recommender Syst.*, Sep. 2016, pp. 301–308, doi: 10.1145/2959100.2959148.

[36] Z. Noorian, J. Zhang, Y. Liu, S. Marsh, and M. Fleming, "Trust-oriented buyer strategies for seller reporting and selection in competitive electronic marketplaces," *Auto. Agents Multi-Agent Syst.*, vol. 28, no. 6, pp. 896–933, Nov. 2014.

[37] S. Al-Oufi, H.-N. Kim, and A. El Saddik, "A group trust metric for identifying people of trust in online social networks," *Expert Syst. Appl.*, vol. 39, no. 18, pp. 13173–13181, Dec. 2012, doi: 10.1016/j.eswa.2012.05.084.

[38] X. L. Dong, E. Gabrilovich, K. Murphy, V. Dang, W. Horn, C. Lugaresi, S. Sun, and W. Zhang, "Knowledge-based trust: Estimating the trustworthiness of web sources," *Proc. VLDB Endowment*, vol. 8, no. 9, pp. 938–949, Feb. 2015, doi: 10.14778/2777598.2777603.

[39] S. Kumar and N. Shah, "False information on web and social media: A survey," 2018, *arXiv:1804.08559*. [Online]. Available: http://arxiv.org/abs/1804.08559

[40] B. Markines, C. Cattuto, and F. Menczer, "Social spam detection," in *Proc. 5th Int. Workshop Adversarial Inf. Retr. Web (AIRWeb)*, 2009, pp. 41–48, doi: 10.1145/1531914.1531924.

[41] H. Hosseinmardi, R. I. Rafiq, R. Han, Q. Lv, and S. Mishra, "Prediction of cyberbullying incidents in a media-based social network," in *Proc. IEEE/ACM Int. Conf. Adv. Social Netw. Anal. Mining (ASONAM)*, Aug. 2016, pp. 186–192.

[42] M. Nawir, A. Amir, N. Yaakob, and O. B. Lynn, "Internet of Things (IoT): Taxonomy of security attacks," in *Proc. 3rd Int. Conf. Electron. Design (ICED)*, Jul. 2016, pp. 321–326.

[43] R. Mahmoud, T. Yousuf, F. Aloul, and I. Zualkernan, "Internet of Things (IoT) security: Current status, challenges and prospective measures," in *Proc. 10th Int. Conf. Internet Technol. Secured Trans. (ICITST)*, Dec. 2015, p. 336.

[44] B. Chandrasekaran and J. M. Conrad, "Human-robot collaboration: A survey," in *Proc. SoutheastCon*, Apr. 2015, pp. 1–8.

[45] W.-L. Hu, K. Akash, T. Reid, and N. Jain, "Computational modeling of the dynamics of human trust during human–machine interactions," *IEEE Trans. Human-Mach. Syst.*, vol. 49, no. 6, pp. 485–497, Dec. 2019.

[46] D. Kaur, S. Uslu, A. Durresi, G. Mohler, and J. G. Carter, "Trust-based human-machine collaboration mechanism for predicting crimes," in *Advanced Information Networking and Applications*, L. Barolli, F. Amato, F. Moscato, T. Enokido, and M. Takizawa, Eds. Cham, Switzerland: Springer, 2020, pp. 603–616.

[47] X. Wang, Y. Wang, and H. Sun, "Exploring the combination of Dempster-Shafer theory and neural network for predicting trust and distrust," *Comput. Intell. Neurosci.*, vol. 2016, pp. 1–12, Jan. 2016.

[48] Y. Gao, X. Li, J. Li, Y. Gao, and S. Y. Philip, "Info-trust: A multi-criteria and adaptive trustworthiness calculation mechanism for information sources," *IEEE Access*, vol. 7, pp. 13999–14012, 2019.

[49] Y. L. Sun, W. Yu, Z. Han, and K. J. R. Liu, "Information theoretic framework of trust modeling and evaluation for ad hoc networks," *IEEE J. Sel. Areas Commun.*, vol. 24, no. 2, pp. 305–317, Feb. 2006.

[50] H. Luo, J. Tao, and Y. Sun, "Entropy-based trust management for data collection in wireless sensor networks," in *Proc. 5th Int. Conf. Wireless Commun., Netw. Mobile Comput.*, Sep. 2009, pp. 3171–3174.

[51] Y. Wang and M. P. Singh, "Formal trust model for multiagent systems," in *Proc. 20th Int. Joint Conf. Artif. Intell.* San Francisco, CA, USA: Morgan Kaufmann, 2007, pp. 1551–1556.

[52] N. A. Bansod, V. Kulkarni, and S. Patil, *Soft Computing—A Fuzzy Logic Approach*. Dhankhavadi, India: Allied Publishers, Bharati Vidyapeeth College of Engineering, 2005.

[53] S. Song, K. Hwang, and M. Macwan, "Fuzzy trust integration for security enforcement in grid computing," in *Proc. IFIP Int. Conf. Netw. Parallel Comput., (NPC)*. Berlin, Germany: Springer, 2004, pp. 9–21.

[54] H. Fang, J. Zhang, M. Sensoy, and N. M. Thalmann, "A generalized stereotypical trust model," in *Proc. IEEE 11th Int. Conf. Trust, Secur. Privacy Comput. Commun.*, Jun. 2012, pp. 698–705.

[55] Y. Wu, "Research of trust degree evaluation for C2C E-commerce based on fuzzy neural network," in *Proc. 2nd Int. Conf. E-business Inf. Syst. Secur.*, May 2010, pp. 1–4.

[56] C. Hilbe, A. Traulsen, and K. Sigmund, "Partners or rivals? Strategies for the iterated prisoner's dilemma," *Games Econ. Behav.*, vol. 92, pp. 41–52, Jul. 2015. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0899825615000822

[57] T. Carter, J. Haldane, A. Einstein, and E. Gibbon, *An Introduction to Information Theory and Entropy*. Santa Fe, NM, USA: Complex Systems Summer School, Jun. 2000.

[58] U. Jayasinghe, G. M. Lee, T.-W. Um, and Q. Shi, "Machine learning based trust computational model for IoT services," *IEEE Trans. Sustain. Comput.*, vol. 4, no. 1, pp. 39–52, Jan. 2019.

[59] L. Katz, "A new status index derived from sociometric analysis," *Psychometrika*, vol. 18, no. 1, pp. 39–43, 1953, doi: 10.1007/BF02289026.

[60] G. Theodorakopoulos and J. S. Baras, "On trust models and trust evaluation metrics for ad hoc networks," *IEEE J. Sel. Areas Commun.*, vol. 24, no. 2, pp. 318–328, Feb. 2006.

[61] S. Bistarelli, S. N. Foley, B. O'Sullivan, and F. Santini, "Semiring-based frameworks for trust propagation in small-world networks and coalition formation criteria," *Secur. Commun. Netw.*, vol. 3, no. 6, pp. 595–610, Nov. 2010. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/sec.252

[62] W. T. L. Teacy, M. Luck, A. Rogers, and N. R. Jennings, "An efficient and versatile approach to trust and reputation using hierarchical Bayesian modelling," *Artif. Intell.*, vol. 193, pp. 149–185, Dec. 2012. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0004370212001075

[63] D. Melaye and Y. Demazeau, "Bayesian dynamic trust model," in *Proc. 4th Int. Central Eastern Eur. Conf. Multi-Agent Syst. Appl. (CEEMAS)*. Berlin, Germany: Springer, 2005, pp. 480–489.

[64] K. Regan, P. Poupart, and R. Cohen, "Bayesian reputation modeling in e-marketplaces sensitive to subjectivity, deception and change," in *Proc. 21st Nat. Conf. Artif. Intell.*, vol. 2, 2006, pp. 1206–1212.

[65] S. Ganeriwal, L. K. Balzano, and M. B. Srivastava, "Reputation-based framework for high integrity sensor networks," *ACM Trans. Sensor Netw.*, vol. 4, no. 3, pp. 1–37, May 2008, doi: 10.1145/1362542.1362546.

[66] W. Hoeffding, *Probability Inequalities for Sums of Bounded Random Variables*. New York, NY, USA: Springer, 1994, pp. 409–426, doi: 10.1007/978-1-4612-0865-5_26.

[67] S. M. Lynch, *Introduction to Hierarchical Models*. New York, NY, USA: Springer, 2007, pp. 231–269, doi: 10.1007/978-0-387-71265-9_9.

[68] A. P. Dempster, *A Generalization Bayesian Inference*. Berlin, Germany: Springer, 2008, pp. 73–104, doi: 10.1007/978-3-540-44792-4_4.

[69] J. Gordon and E. H. Shortliffe, *The Dempster-Shafer Theory of Evidence*. San Francisco, CA, USA: Morgan Kaufmann, 1990, pp. 529–539.

[70] K. Sentz and S. Ferson. (Apr. 2002). *Combination of Evidence in Dempster-Shafer Theory*. [Online]. Available: https://www.osti.gov/biblio/800792

[71] A. Jøsang, "A logic for uncertain probabilities," *Int. J. Uncertainty, Fuzziness Knowl.-Based Syst.*, vol. 9, no. 3, pp. 279–311, Jun. 2001, doi: 10.1142/S0218488501000831.

[72] R. Haenni, "Shedding new light on Zadeh's criticism of dempster's rule of combination," in *Proc. 7th Int. Conf. Inf. Fusion*, vol. 2, Aug. 2005, p. 6.

[73] G. B. Guacho, S. Abdali, N. Shah, and E. E. Papalexakis, "Semi-supervised content-based detection of misinformation via tensor embeddings," 2018, *arXiv:1804.09088*. [Online]. Available: http://arxiv.org/abs/1804.09088

[74] J. Pearl, "Reverend Bayes on inference engines: A distributed hierarchical approach," in *Proc. 2nd AAAI Conf. Artif. Intell.*, 1982, pp. 133–136.

[75] J. Pearl, "Fusion, propagation, and structuring in belief networks," *Artif. Intell.*, vol. 29, no. 3, pp. 241–288, Sep. 1986, doi: 10.1016/0004-3702(86)90072-X.

[76] D. Koutra, T.-Y. Ke, U. Kang, D. H. P. Chau, H.-K. K. Pao, and C. Faloutsos, "Unifying guilt-by-association approaches: Theorems and fast algorithms," in *Proc. Eur. Conf. Mach. Learn. Knowl. Discovery Databases*. Berlin, Germany: Springer, 2011, pp. 245–260, doi: 10.1007/978-3-642-23783-6_16.

[77] H. Liu, E.-P. Lim, H. W. Lauw, M.-T. Le, A. Sun, J. Srivastava, and Y. A. Kim, "Predicting trusts among users of online communities: An epinions case study," in *Proc. 9th ACM Conf. Electron. Commerce (EC)*, 2008, pp. 310–319, doi: 10.1145/1386790.1386838.

[78] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, Sep. 1995, doi: 10.1023/A:1022627411411.

[79] T. Joachims, "Optimizing search engines using clickthrough data," in *Proc. 8th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, 2002, p. 133, doi: 10.1145/775047.775067.

[80] W. Yuan, D. Guan, and S. Lee, "A dynamic trust model based on naive Bayes classifier for ubiquitous environments," in *Proc. 2nd Int. Conf. High Perform. Comput. Commun.*, Sep. 2006, pp. 562–571.

[81] P. Rai and S. Singh, "A survey of clustering techniques," *Int. J. Comput. Appl.*, vol. 7, no. 12, pp. 1–5, Oct. 2010.

[82] D. Eppstein, M. S. Paterson, and F. F. Yao, "On nearest-neighbor graphs," *Discrete Comput. Geometry*, vol. 17, no. 3, pp. 263–282, 1997.

[83] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge, MA, USA: A Bradford Book, 2018.

[84] C. Gaskett, D. Wettergreen, and A. Zelinsky, "Q-learning in continuous state and action spaces," in *Proc. 12th Austral. Joint Conf. Artif. Intell., Adv. Topics Artif. Intell.* Berlin, Germany: Springer, 1999, pp. 417–428.

[85] D. Kriesel. (2007). *A Brief Introduction to Neural Networks*. [Online]. Available: http://www.dkriesel.com

[86] L. Zeynalvand, T. Luo, and J. Zhang, "COBRA: Context-aware Bernoulli neural networks for reputation assessment," 2019, *arXiv:1912.08446*. [Online]. Available: http://arxiv.org/abs/1912.08446

[87] Y. Zhang and G. Ruan, "Bernoulli neural network with weights directly determined and with the number of hidden—Layer neurons automatically determined," in *Proc. 6th Int. Symp. Neural Netw. Adv. Neural Netw. (ISNN)*. Berlin, Germany: Springer, 2009, pp. 36–45, doi: 10.1007/978-3-642-01507-6_5.

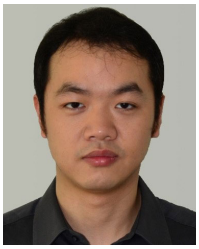
[88] V. Golovko, A. Kroshchanka, U. Rubanau, and S. Jankowski, "A learning technique for deep belief neural networks," in *Neural Networks and Artificial Intelligence*, V. Golovko and A. Imada, Eds. Cham, Switzerland: Springer, 2014, pp. 136–146.

[89] Y. Hua, J. Guo, and H. Zhao, "Deep belief networks and deep learning," in *Proc. Int. Conf. Intell. Comput. Internet Things*, Jan. 2015, pp. 1–4.

[90] T. Villmann, R. Der, M. Herrmann, and T. Martinetz, "Topology preservation in self-organizing feature maps: General definition and efficient measurement," in *Fuzzy Logik*, B. Reusch, Ed. Berlin, Germany: Springer, 1994, pp. 159–166.

[91] H. Yin, "The self-organizing maps: Background, theories, extensions and applications," in *Computational Intelligence: A Compendium*. Berlin, Germany: Springer, 2008.
[92] M. Dittenbach, D. Merkl, and A. Rauber, "The growing hierarchical self-organizing map," in *Proc. IEEE-INNS-ENNS Int. Joint Conf. Neural Netw. (IJCNN) Neural Comput., New Challenges Perspect. New Millennium*, vol. 6, Jul. 2000, pp. 15–19.
[93] M. Di Capua, E. Di Nardo, and A. Petrosino, "Unsupervised cyber bullying detection in social networks," in *Proc. 23rd Int. Conf. Pattern Recognit. (ICPR)*, Dec. 2016, pp. 432–437.

HANNAH LIM JING TING (Student Member, IEEE) is currently pursuing the B.S. degree in mathematical sciences with Nanyang Technological University, Singapore (NTU). For nine months, she interned at the Digital Identity and Trustworthiness Laboratory, Huawei Singapore, where she researched on trust modeling and its related methods. Her current research interests include mathematical modeling, trust modeling, digital security, and cryptography. She plans to pursue further studies and learn more about mathematical modeling for the environment and/or digital security.



XIN KANG (Senior Member, IEEE) received the B.Eng. degree in electrical engineering from Xi'an Jiaotong University, China, in 2005, and the Ph.D. degree in electrical and computer engineering from the National University of Singapore, Singapore, in 2011. From 2011 to 2014, he was a Research Scientist with the Institute for Infocomm Research, A*STAR, Singapore. After that, he joined Shield Laboratory, Huawei Singapore Research Center, as a Senior Researcher. He is currently a Senior Researcher with TTE Lab, Huawei Singapore Research Center. He has published more than 30 journal articles on IEEE top journals, and more than ten of them are listed as SCI highly cited research articles. He has also published more than 30 conference papers on first-tier IEEE conferences. After joining Huawei, he has filed more than 40 patents on security protocol designs. Besides, he is also very active in standardization. Up to now, he has contributed more than 30 technical proposals to 3GPP SA3, and 17 of his proposals have been accepted by 3GPP SA3 TR 33. 899. He is one of the key contributors to the newly published ITU-T standard X.1365, and newly established work item X.ztd-iot. He is also one of the key contributors to Huawei 5G security white articles. He has more than ten years of research experience. His research interests include trust modeling, digital identity, blockchain, network security, security protocol design, and applied cryptography. He has received the Best Paper Award from IEEE ICC 2017, and Best 50 Papers Award from IEEE GlobeCom 2014.

TIEYAN LI (Member, IEEE) received the Ph.D. degree in computer science from the National University of Singapore. He is currently leading research on digital trust-building the trust infrastructure for future digital world, and previously on mobile security, the IoT security, and AI security with Shield Laboratory, Singapore Research Center, Huawei Technologies. From that on, he was a Security Scientist with the Institute for Infocomm Research, I2R Singapore. He is an expert on security and applied cryptography, and a technology generalist on applications, systems and networks. He has more than 20 years of experience and is proficient in security design, architect, innovation, and practical development. He was also active in academic security fields with tens of publications and patents. His research interests include trustworthy AI, trustworthy computing, trustworthy identity, and network infrastructure. He has served as the PC members for many security conferences, and is an influential speaker in industrial security forums.

HAIGUANG WANG (Senior Member, IEEE) received the bachelor's degree from Peking University, in 1996, and the Ph.D. degree in computer engineering from the National University of Singapore, in 2009. From 2001 to 2013, he was a Research Engineer/Scientist with the Institute for Infocomm Research (I2R), Singapore, and doing research on communication and network protocol design, innovation, and practical development. He joined Huawei International, in 2013, and doing research on security area since then. He is an expert on communication network security and identity management and access control, and a technology generalist on systems, communications and networks. He is currently doing research on digital identity and trust management, security automation, and network infrastructure security for future digital world, and previously on 5G communication network security. He has published/filed more than 60 research articles and patents together. He has been actively contributed to various standards, including IEEE 802.11, 3GPP SA3, ITU-T SG-17, and IETF.

CHENG-KANG CHU (Member, IEEE) received the Ph.D. degree in computer science from the National Chiao Tung University, Taiwan. He is currently a Senior Researcher with Huawei International, Singapore. Before joining Huawei, he was a Research Scientist with the Cryptography and Security Department, Institute for Infocomm Research (I2R), Singapore. He has had a long-term interest in the development of new technologies in applied cryptography, cloud computing security, and the IoT security. He has published many research articles in major conferences and journals, such as PKC, CT-RSA, AsiaCCS, IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, and IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY. His research interests include mobile security, the IoT security, and decentralized digital identity. He received the Best Student Paper Award in ISC 2007. He also served as the Vice Chair for the IEEE CCNC 2012 and on the program committee of many international conferences.