Computing Massive Trust Analytics for Twitter using Apache Spark with Account Self-assessment

Georgios Drakopoulos[1], Andreas Kanavos[2,3], Konstantinos Paximadis[3],
Aristidis Ilias[2], Christos Makris[2] and Phivos Mylonas[1]
[1]*Department of Informatics, Ionian University, Corfu, Greece*
[2]*Computer Engineering and Informatics Department, University of Patras, Patras, Greece*
[3]*Hellenic Open University, Patras, Greece*

Keywords: Higher Order Metrics, Web Trust, Distributed Classification, MLlib, Apache Spark, PySpark, Twitter.

Abstract: Although trust is predominantly a human trait, it has been carried over to the Web almost since its very inception. Given the rapid Web evolution to a true melting pot of human activity, trust plays a central role since there is a massive number of parties interested in interacting in a multitude of ways but have little or even no reason to trust *a priori* each other. This has led to schemes for evaluating Web trust in contexts such as e-commerce, social media, recommender systems, and e-banking. Of particular interest in social networks are classification methods relying on network-dependent attributes pertaining to the past online behavior of an account. Since the deployment of such methods takes place at Internet scale, it makes perfect sense to rely on distributed processing platforms like Apache Spark. An added benefit of distributed platforms is paving the way algorithmically and computationally for higher order Web trust metrics. Here a Web trust classifier in MLlib, the machine learning library for Apache Spark, is presented. It relies on both the account activity but also on that of similar accounts. Three datasets obtained from topic sampling regarding trending Twitter topics serve as benchmarks. Based on the experimental results best practice recommendations are given.

1 INTRODUCTION

Social networks are already teenagers and they are still expanding at an impressive rate both in terms of their respective account base as well as of the variety of applications they natively support. Moreover, there is a definite shift towards multimodal account interaction where accounts share messages, images (often in the form of memes), voice clips, and geolocation information, especially in conjunction with smartphones. Additionally, social networks recently tend to cooperate by providing social login services across sites. For instance, now ResearchGate allows both LinkedIn and Google login credentials in addition to its own. Despite that certain groups tend to migrate among social networks, this is more than compensated by the creation of new accounts on a daily basis from people across all generational cohorts. Clearly, this rapidly increasing account base requires a set of rules in order to interact appropriately. In turn, they rely on Web trust, namely the set of mechanisms and procedures ensuring up to a reasonable degree that the entity behind an account is actually who (in the case of netizens) or what (in the case of groups and companies) it claims to be.

The need for Web trust actually predates social networks. In the proto-Internet Web trust was originally not considered a primary issue because of its small size and little digital value. That was mainstream mentality until the Morris worm in 1989 almost incapacitated ARPANET (Furnell and Spafford, 2019; Obimbo et al., 2018). Since then the Morris worm has been depicted in numerous stories including the 1995 film *Hackers*[1], which has already attained cult status. Similarly the *Markovian parallax denigrent* incident of 1996, where a considerable number of Usenet fora were massively spammed with automatically generated messages whose language was very close to real world written English in terms of syntax, proved beyond a shadow of a doubt that certain cyberattacks could rely on machine intelligence for social engineering (Baldwin, 2017; Salahdine and Kaabouch, 2019). Therefore, cyberdefenses of any kind should include Web trust as a major com-

[1] https://www.imdb.com/title/tt0113243

Drakopoulos, G., Kanavos, A., Paximadis, K., Ilias, A., Makris, C. and Mylonas, P.
Computing Massive Trust Analytics for Twitter using Apache Spark with Account Self-assessment.
DOI: 10.5220/0010214104030414
In *Proceedings of the 16th International Conference on Web Information Systems and Technologies (WEBIST 2020)*, pages 403-414
ISBN: 978-989-758-478-7

ponent in order to reliably and efficiently answer in the Web, essentially in the digital domain, the fundamental human question “Who are you?”.

Currently trust in the Web translates to a computational problem, typically in the form of digital reputation. This happens because trust has to be established between parties who have no *a priori* reason to trust each other, mostly for one time interactions. Other manifestations include the solution certificate to an NP-hard problem such as the key in asymmetric cryptography systems of the various proofs in blockchain environments. This in fact constitutes the motivation behind this conference paper. Several computational mechanisms are in place to determine Web trust level:

- In e-commerce applications a generic or more specialized feedback mechanism is typically in effect. For instance, eBay relies heavily on detailed transaction feedback which breaks down to rating in categorical scales both the overall experience (positive, negative, or neutral) as well as a multitude of individual aspects such as buyer-seller communication, item description accuracy, item condition, and dispatch time. Moreover, feedback is incomplete without comments. Feedback score is not only public but is in fact shown in a very prominent position right next to the account name, connecting it thus with its score.
- In job related portals trust tends to mimic actual candidate verification procedures. For instance, in LinkedIn trust is expressed by a combination of recommendation letters from former employers or peers, skill endorsements, and links to third party degree or qualification verification sites.
- In social media trust is frequently expressed through conact- and communication related attributes. Trust in Facebook has close connections to friendship as friends are supposed to be trusted. To this end Facebook enforced a policy mandating that accounts must have the true name of their owner (Haimson and Hoffmann, 2016). Twitter relies on administrative tools for cross-checking account profiles[2]. However, because of the enormous number of Twitter base, only a small fraction has been verified.
- Recently, blockchain technology along with its competition like IOTA[3] promise to safeguard data of any kind through distributed ledger technologies. Identity verification as well as any reward claims in blockchains require distributed verification from a large number of independent parties, rendering phony claims difficult to make.

The primary research contribution of this conference paper is twofold. First, a generic model for Web trust is proposed. It is based on two components, namely on online account activity attributes as well as on the activity of similar accounts. This has the advantage that features pertaining to both the micro and macro activity level can be combined. Second, said model has been specialized for Twitter and a realistic Spark implementation thereof has been made. The results on the benchmark datasets obtained from topic sampling of trending issues are encouraging.

The rest of the paper is structured as follows. Section 2 presents background topics in sentiment analysis and community detection. Section 3 presents the proposed technique in its general form and analyses trust in Twitter. Section 4 contains implementation details, benchmark dataset synopses, results analysis, and best practice recommendations. Finally, section 5 concludes the work and presents directions for future research. Technical acronyms are explained the first time they are encountered. The popular term *netizen* denotes a social media user. Finally, the notation for this conference paper is summarized in table 1.

Table 1: Paper Notation.

Symbol	Meaning
$\triangleq$	Definition or equality by definition
$\{s_1,\ldots,s_n\}$	Set with elements $s_1,\ldots,s_n$
$\lvert S\rvert$	Set cardinality
$\Phi(t_k)$	Set of followers of account t_k
$\Psi(t_k)$	Set of followees of account t_k

2 RELATED WORK

Trust is paramount in establishing a wide array of short- or long-lasting and properly functional relationships over the Web (Buskens, 2002). A prime application is e-commerce (Hallikainen and Laukkanen, 2018), particularly as shown in the meta-analysis of e-commerce relationships examined in (Kim and Peterson, 2017) or in the scale examined at (Oliveira et al., 2017) where consumer behavior is analyzed in terms of trust. Moreover Web trust is essential in candidate recruiting as explained in (Drakopoulos et al., 2020) where open LinkedIn attributes determined admissible candidates for startup teams. Along a similar line of reasoning in (Sharma and Sharma, 2017) the importance of trusted candidates for HR departments is analyzed. Further applications include recommender systems (Massa and Bhattacharjee, 2004; O’Donovan

[2] https://help.twitter.com/en/managing-your-account/about-twitter-verified-accounts

[3] www.iota.org

and Smyth, 2005; Alexopoulos et al., 2020), e-voting (Garg et al., 2019; Risnanto et al., 2019), and e-mail sender verification where the pretty good privacy (PGP) web of trust plays a central role for a long number of years now (Attia et al., 2016). Trust metrics based on online interactions between accounts are presented among others in (Lumbreras and Gavaldà, 2012) where trust takes the form of recommendations, in (Richardson et al., 2003) where Semantic Web features are exploited, and in (Kamvar et al., 2003) where the EigenTrust method is developed for peer-to-peer (P2P) networks. Recently, the advent of blockchains has delegated trust to a widely distributed computation of an NP hard problem (Liang et al., 2018). Most blockchain implementations offer a high level of security (Li et al., 2020) including resistance to distributed attacks (Sengupta et al., 2020). For an overview of proof systems see (Drakopoulos et al., 2019a).

Beyond the limits of its formal definition, it should be noted that Web trust ultimately relies heavily on myriads of human decisions and actions (Papaoikonomou et al., 2013). In turn, the latter depend on emotions, which are widely considered the main human motivators (Albanie et al., 2018). Affective computing takes this psychological fact into consideration for various objectives like efficient human-computer interfaces (Davis et al., 2020). Human emotion can be estimated through voice as described in (Drakopoulos et al., 2019b), physiological signals as in (Egger et al., 2019), face expression as for instance explained in (Jain et al., 2018) and in (Wang et al., 2018), gait as shown among others in (Chiu et al., 2018) and (Xue et al., 2019), or any combination thereof (Schirmer and Adolphs, 2017). Recently deep learning architectures have been used to estimate human emotional state from the fusion of multiple modalities (Ranganathan et al., 2016).

In social networks Web trust may well take different forms. For instance, in small world networks trust issues are examined in (Gray et al., 2003). More recently, trust in social networks is inherently tied to whether an account refrains from spreading fake news (Buntain and Golbeck, 2017). In the overwhelming majority of cases it depends on digital account activity (Adali et al., 2010). A notable exception is Twitter account verification (Kyriazidou et al., 2019) as accounts may optionally report their real identity including phone number and Web site. Additionally, digital influence may be considered as an indirect confirmation of trust (Drakopoulos et al., 2017). The broad picture of the Web of trust in Twitter is explored in (Tavakolifard et al., 2013), while various trust attributes for Twitter are discussed in (Castillo et al., 2011). Additionally, various recurrent- (RNN) and convolutional neural network (CNN) architectures for discovering fake news on Twitter are proposed in (Ajao et al., 2018)

The recent advent of massive distributed frameworks such as Apache Spark has enabled an in-depth analysis of Web trust (Ventocilla, 2019). Principles for the development of distributed trust analytics are given in (Ciordas-Hertel et al., 2019). Computational aspects of the Web trust when seen as a data intensive problem are examined in (Terzi et al., 2017). In (Adib et al., 2017) is discussed the reduction of Web trust to a large scale computational problem which can be efficiently solved in Spark. Moreover, trusted product recommendation where Spark computes trustworhiness levels is the focus of (Patil et al., 2017).

3 PROPOSED METHODOLOGY

3.1 Trust Directions

In the contemporary digital sphere the notion of Web trust is literally everywhere, although each site may well have its own interpretation in an effort to provide high quality services. Web trust has the following three basic directions, which albeit distinct may well overlap in certain sites:

- **Site-to-Account:** A site should have in place efficient and transparent rules, policies, and procedures in order to order to verify both the fact that a given piece of material was indeed posted by the account it claimed to have done so but also the identity of an account owner if the need arises and depending on the nature of the site. Frequently two- or multi-factor authentication (2FA/MFA) are part of the trust protocols, as so recently are machine- or deep learning (ML/DL) techniques. This direction is the focus of the proposed methodology presented here.
- **Account-to-Site:** An account should be able to trust the content of material and messages posted by authenticated site administrators or moderators. This is often accomplished by non-repudiation techniques, the majority of which rely in turn on advanced asymmetric encryption schemes such as RSA or the El Gamal algorithm. An extra layer of trust comes from using PGP signed e-mails. Although there may be a significant gap between the parties involved in terms of computing power, cryptographic protocols ensure a considerable degree of parity.

- **Account-to-Account:** In contrary to the previous two cases, this is a peer problem in the sense that the two parties are usually equal in terms of computing power and of access level. Depending on the nature of the site and the general goals of the parties involved, the two accounts can trust each other for a single transaction or action item, maintain a trusted connection until a set of objectives is accomplished, or they can establish a permanent trust relationship.

At this point it should be noted that the level and intensity of malicious online activity such as false rumour circulation from fake news farms and trolling through inflammatory comments and irony along with social engineering and old school frauds have forced EU to issue the publicly available guide titled *Fake news and disinformation online*.

3.2 Web Trust Model

In this subsection the most general form of proposed methodology is described, whereas its specific from for Twitter is given in later subsections. The fundamental observation underlying it is the following:

Observation 1. *Trust is eventually a human trait. Therefore, its computational aspects should rely on elements of human activity in order to estimate it.*

Analyzing the above observation further, it can be deduced that the trust level of an account has at least two main components:

- Trustworthiness related to the individual activity of the particular account. This is estimated by network-specific attributes. Nonetheless, the generality of the proposed methodology is not affected by the nature of this attribute set.
- Trustworthiness derived from accounts similar to the particular account. To this end group-specific attributes as well as an account similarity metric must be defined. In this work the term *group* denotes the set of similar accounts.

The proposed methodology is also shown in figure 1.

One way to mathematically express the preceding analysis is through the following model. Specifically, let L be the level of Web trust which ultimately depends on two attribute sets, one containing n_i features $\left\{t_k^i\right\}$ with $1 \leq k \leq n_i$ pertaining to the individual online behavior of an account and one consisting of n_g attributes $\left\{t_k^g\right\}$ with $1 \leq k \leq n_g$ of the group this account belongs to. Assuming that there are n accounts in total, let column vector $\mathbf{f_i}$ contain the normalized scores from an appropriate metric or ML model which relies on $\left\{t^i\right\}$ as shown in equation (1):

$$\mathbf{f_i} \stackrel{\triangle}{=} \left[f_i[1] \quad f_i[2] \quad \ldots \quad f_i[n]\right]^T \tag{1}$$

As a shorthand, let $\mathbf{f_i}[\neg j]$ denote $\mathbf{f_i}$ without its j-th element, resulting in a column vector of length $n-1$. Since relative trust scores are more important than raw ones and in order to keep the scores at the same scale with all the weights involved in the scheme, the raw scores $\tilde{\mathbf{f}}_\mathbf{i}[j]$ have been normalized by computing the softmax score as follows as in equation (2):

$$\mathbf{f_i}[j] \stackrel{\triangle}{=} \frac{e^{\tilde{\mathbf{f}}_\mathbf{i}[j]}}{\sum_{j=1}^{n} e^{\tilde{\mathbf{f}}_\mathbf{i}[j]}} \quad \in U \stackrel{\triangle}{=} [0,1] \tag{2}$$

Along the same line of reasoning let matrix $\mathbf{F_g}$ contain the normalized scores obtained from an account similarity metric or ML model. Clearly its diagonal contains only ones, which is the maximum possible similarity value. Also let $\mathbf{F_g}[\neg j; j]$ be the column vector of length $n-1$ resulting from keeping the j-th column and removing the element in the j-th row. The final trust score for the j-th account is determined by the formula in equation (3):

$$L[j] \stackrel{\triangle}{=} T\left(w_i \mathbf{f_i}[j] + w_b \mathbf{F_g}[\neg j; j]^T \mathbf{f_i}[\neg j]\right) \tag{3}$$

The Web trust scores are collected to a vector $\mathbf{L}$ which may consequently used for ranking, account recommendation, or any other social media analytics.

In the preceding equation the weighted sum consists of the trust for the j-th account and that of accounts similar to it. This sum is in turn the input of a non-linear kernel $T[\cdot]$ which yields the final Web trust score. The respective weights w_i and w_g obey the constraint of equation (4):

$$w_i + w_g = 1, \qquad 0 < w_i, w_g < 1 \tag{4}$$

This scheme works if both models $f_i(\cdot)$ and $f_g(\cdot)$ yield results in the same range. For the purposes of this work the weights were set to the values:

$$w_i \stackrel{\triangle}{=} \frac{n_i}{n_i + n_g}, \quad w_g \stackrel{\triangle}{=} \frac{n_g}{n_i + n_g} \tag{5}$$

The kernel $T(\cdot)$ may well be anything which appropriately expresses Web trust in the underlying domain. It may even be the identity function, if the weighted sum suffices. The reason a non-linear kernel is proposed is that typically the latter can differentiate between two trust scores on the linear scale. The modified range of the maximum possible to the lowest possible yields the discrimination power of the kernel shown in equation (6):

$$\frac{\max_j L[j]}{\max\left\{\beta_0, \min_j L[j]\right\}} \tag{6}$$

In the above equation β_0 is a least nominal trust for an account when the computed trust level is below it. The main reason for assigning a known minimum trust score to an account is to help new accounts boost their trustworthiness. Moreover, it gives a second chance to accounts involved in serious incidents to make a clean start in terms of Web trust.

Although it is not a strict requirement, the non-linear kernel $T : U \to V$ should be differentiable in order to ensure a smooth mapping of its domain U to its range V.

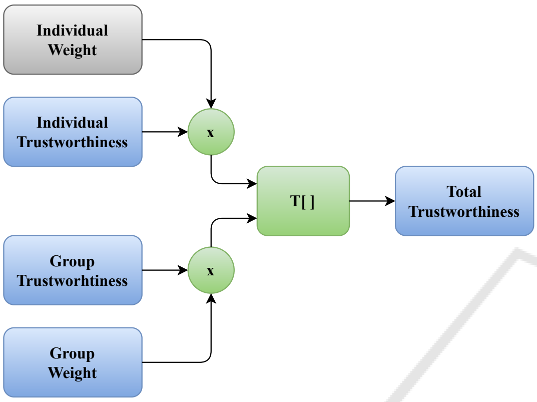

Figure 1: Proposed general scheme.

The proposed scheme includes certain metrics found in the literature. In Twitter trust can be tied to the digital influence of an account with the rationale that humans know almost immediately which accounts to trust based on their real world experience. One intuitive way to evaluate the influence $P[j]$ of the j-th account is the logarithm of the followers-to-followees ratio as defined in equation (7):

$$P[j] \triangleq \ln\left(1+\frac{|\Phi(j)|}{\max\{1,|\Psi(j)|\}}\right) \qquad (7)$$

Notice that the definition of equation (7) is a first order metric in the sense that only attributes concerning account j are needed in order to compute its digital influence. Thus, computations can be done in parallel, allowing the influence of a massive number of accounts to be determined. However, this approach says little about the influence of an account compared to others, especially accounts with similar online behavior. To this end, a higher order, non-linear extension $Q[\cdot]$ of $P[\cdot]$ could lead to the following recursive definition of trust as shown in equation (8):

$$Q[j] \triangleq \ln\left(1+\frac{\sum_{j'\in\Phi(j)} Q[j']}{\max\left\{1,\sum_{j'\in\Psi(j)} Q[j']\right\}}\right) \qquad (8)$$

The above equation can be also expressed by the general model of equation (3). This can be seen by expanding the above recursive definition of $Q[\cdot]$ to a non-linear equation system.

3.3 Trust in Twitter

Twitter is perhaps the prime microblogging platform where conversations of various kinds take place on a daily basis. Recent figures reveal the extent of its account base: There are 330 million monthly active accounts and 152 million daily ones. 500 million tweets are posted per day, the 80% of them from mobile devices. Since there is a low upper limit of 280 characters per tweet (in fact only 140 until 2018), conversations tend to be apophthegmatic in nature, although threads are progressively becoming common. In turn this may occasionally lead to emotionally charged conversations, especially in comparison to other social media, e-mail, or even ordinary texting.

Twitter has to preserve trust in order to keep malicious accounts at bay, especially if they attempt to consolidate a considerable follower base and becoming thus potential influencers. In this context trust may be defined as the possibility to authenticate who (for persons) or what (for organizations) the account claims to be. An additional question is whether tweets from a given account are accurate or are they fraudulent (e.g. promoting low quality products) or even fake (and even worse straight from a fake news farm). Therefore, there are two primary trust axes of trust:

- **Account Axis:** This is the stronger axis since trusting an account also covers their respective tweets. Typically metrics such as digital influence or verification are used. Higher order metrics are better in discovering fraudulent accounts at the expense of increased computational cost.
- **Tweet Axis:** Trust is weaker as only tweets of various accounts are deemed as trustworthy or not. Frequently natural language processing (NLP) and affective computing methodologies uncover any tweet inconsistencies. However, occasionally these schemes can be misled by honest mistakes made by accounts, unconventionally formulated tweets, or irony, a very common trait in social media conversations which cannot be easily detected even by humans as it is culture-specific.

Obviously malicious parties can create fake Twitter accounts and spread out through strategically planned tweets rumors in order to cause harm somebody, provoke doubts, or achieve other questionable objectives. Before the advent of massive distributed platforms it was almost impossible to algorithmically authenticate every logged in user and check every uploaded tweet for trustworthy content. From the perspective of a human account owner trust may rely on the following characteristics:

- The bio and photo of a Twitter profile have to be clear with no uncertainties or inconsistencies. Although many sophisticated image processing algorithms can discover objects, detect whether an image has been tampered with, or perform scene analysis, few can actually understand convoluted visual semantics (Lewis et al., 2019).
- The account is verified. In contrast to most of the features in this list, this can be easily verified algorithmically with access to Twitter application interface (API) and the appropriate method.
- The account corresponds to a well-known real world person or entity. This can be easily verified by any netizen, especially by someone with a considerable online activity. On the contrary, a classifier might need to consult online databases for the same purpose, unless of course the account in question is verified by Twitter.
- Tweets are precise, with a clear meaning, and correct in terms of grammar and syntax. NLP techniques can very efficiently parse tweets but their semantic analysis is a different story as irony, innuendos, and mentions to previous tweets are usually hard to discover.
- Any references to persons, products, events, or technology are accurate. The same algorithmic constraints described earlier apply.
- Content delivery is done through trusted links to sites with secure protocols like https and transport layer security (TLS). Here an algorithmic classifier may in fact fare better than an human since the former may employ a Web crawler and quickly collect information about content and site quality.
- Tweets are posted regularly. Also this is somewhat fuzzy for an algorithm to check, although probabilistic methods can determine whether time intervals between tweets follow a certain distribution or progressively construct an empirical distribution and check for outliers.

From the above list it follows that these criteria may be easily estimated by a human but not algorithmically. This can be primarily attributed to the heavy reliance on semantics as well as on fuzzy quantities. For instance how can a regular tweet rate be defined? Humans, partly because of their analog thinking and the completion principle (Boselie and Wouterlood, 1989), can make sense of degraded or irregular information.

3.4 Individual Attributes for Twitter

The attributes given as input to the MLlib classifiers are very closely tied to the online activity of an account and they have been frequently mentioned in the relevant scientific literature. Following the analysis presented earlier, this list contains a combination of tweet trust axis features (the first six) and account trust axis features (the remaining six). Observe how this list differs from the preceding one.

- **Tweet Number of Characters (Numerical):** Tweet size. Tweets too long or too short may signal an abnormal situation.
- **Tweet Number of Words (Numerical):** Number of words. A consistent number of many words may be an attempt to shift the topic of a conversation. Alternatively, tweets may have been algorithmically generated. Conversely, very short tweets may be the result of trolling, but they may also well be clever use of English in an argument.
- **Question Mark (Binary):** True if the tweet contains a question mark. This is a relatively weak indicator for a single tweet, yet trolling or fake accounts are known to answer questions with questions to evade direct answers towards them. Also they may attempt to shift attention from them by asking carefully engineered questions to change the subject or subtly accuse others. Therefore, untrustworthy accounts are expected to have a long string of tweets with questions.
- **Exclamation Point (Binary):** True if the tweet contains an exclamation point. Depending on culture, an exclamation point may indicate a variety of emotions ranging from happiness and surprise to anger and disgust. Also it is common in certain cases of irony. Despite this, a single use is not always a red flag by itself. Instead, frequent apperances may indicate an account trying to increase the emotional potential of a reply or an entire conversation, perhaps trying to shift its focus.
- **Number of Capital Letters (Numerical):** Number of capital letters. A very low number relative to the total tweet characters may indicate a tweet with improper syntax or a machine generated tweet. Moreover, a very high number of capital letters hint at frequent tantrums, either real or fake. At any rate, this should be flagged as a potentially abnormal situation.
- **More than one "?" or "!" (Binary):** A large number of these punctuation marks may constitute a sign of irony or passive aggressive behavior, especially if this is a recurring phenomenon.

However, context should also be taken into account for better results.

- **Number of Tweets (Numerical):** How many tweets this account has posted. This is an overall account attribute, meaning it influences all tweets even indirectly. Too many tweets may indicate offensive behavior, but also may be the result of a very active legitimate account such a high profile celebrity or a certified news agency.
- **Number of Followers (Numerical):** A large number of followers is typically an indicator of an account which can be trusted, provided that the account has been in existence sufficiently long to attract a critical mass of followers.
- **Number of Friends (Numerical):** By *friends* is meant accounts which follow each other. This is a stronger indicator than the preceding one, yet fake accounts or bots controlled by the same source may easily create phony networks immitating real ones to take advantage of this property.
- **Account is Verified (Binary):** True if account is verified by Twitter.
- **Account Changed Profile (Binary):** True if the account bio has changed during the past six months. Fake and suspicious accounts tend to frequently transform, often in an Ovidian way, in order to achieve their malicious objectives.
- **Account Favourites (Numerical):** The number of tweets the account has marked as favourite. A very high number may point at a bot indiscriminately following other accounts or topics in order to locate potential targets.

Once tweet scores are obtained, the raw individual trust score of each account is the average of those of its tweets. A critical question arising is whether there are sufficient tweets for each account in order to ensure a fair classification. The distribution of tweets per accounts was assumed to follow a normal distribution as shown in equation (9). The rationale behind this assumption is that since the total number of tweets is a result of the tweeting activity of a large number of accounts, then by the central limit theorem (CLT) the resulting distribution is Gaussian.

$$f\left(x;\mu_0,\sigma_0^2\right) \stackrel{\triangle}{=} \frac{1}{\sigma_0\sqrt{2\pi}}\exp\left(-\frac{\left(x-\mu_0\right)^2}{2\sigma_0^2}\right) \tag{9}$$

This assumption was tested with the Kolmogorov-Smirnoff test for each dataset separately. Table 2 has the test results as well as the estimated μ_0 and σ_0^2. For a description of the datasets see section 4.

For each of the datasets described in the next subsection the parameters μ_0 and σ_0^2 were estimated. The six sigma property of the normal distribution mandates that 99.5% of its mass in concentrated in the interval centered in the mean value μ_0 with a length of three standard deviations $3\sigma_0$ on each side, resulting in a total size of $6\sigma_0$. In view of this property, it follows that if μ_0 is high enough and σ_0 is small enough, then only at most 0.25% of the accounts in each dataset has insufficient tweets for a sound statistical analysis. Given that fake and troll accounts typically tweet a lot as explained in (Im et al., 2020), they are bound to be contained in the accounts where a significant tweet sample exists in the datasets.

Table 2: Statistical properties for datasets.

Dataset	**Gaussian?**	$\hat{\mu}_0$	$\hat{\sigma}_0$
Terror	Yes	189.32	45.72
US Election	Yes	362.67	27.44
Harvey	Yes	537.11	16.96

The Gaussian distribution parameters are estimated through their respective sample estimates as follows. Assume that for each account there are available n_s tweets and the j-th such tweet has received a score of $y[j]$. Equation (10) yields the sample mean:

$$\hat{\mu}_0 \stackrel{\triangle}{=} \frac{1}{n_s}\sum_{j=1}^{n_s} y[j] \tag{10}$$

Since each tweet trust score is essentially an estimator in the statistical signal processing sense, averaging n_s tweet scores leads to the estimator variance divided by $\sqrt{n_s}$. Thus, the more tweets available for an account, the more reliable the raw trust score.

The sample variance is obtained by equation (11):

$$\hat{\sigma}_0 \stackrel{\triangle}{=} \frac{1}{n_s-1}\left(\sum_{j=1}^{n_s}\left(y[j]-\hat{\mu}_0\right)^2\right)^{\frac{1}{2}} \tag{11}$$

The normalizing factor in the above equation ensures an unbiased variance estimator.

3.5 Group Attributes for Twitter

Since estimating similarity between accounts is difficult in the general case, it makes sense to establish this piece of ground truth directly. An online survey was organized asking social media users to anonymously complete Web forms. Additionally, a Web application was developed on purpose and linked to a relational database where answers where collected and sorted for subsequent processing. The evaluation scenario complied with the following directives:

- Evaluation was anonymous since Twitter screen names were mapped to hashed ones.

- Participating netizens were not presented with the results. Otherwise, a bandwagon effect (Howard, 2019) might have influenced their decisions.

Please note that at that time the EU data protection directive 95/46/EC was in effect. The broader and more detailed general directive protection regulation (GDPR)[4] directive was enacted much later.

Each netizen answered questions about its own interests as shown in table 3. Based on these answers each of N the netizens was represented as a binary vector. Since almost every netizen expressed a positive stance towards news, this category was omitted.

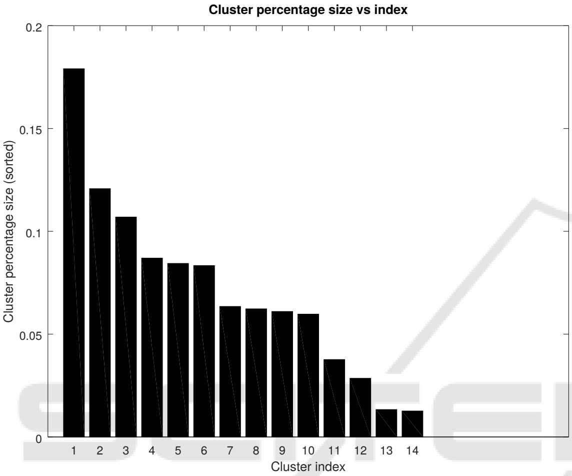

Figure 2: Sorted cluster percentage size.

The k-means with the Hamming distance $h(\cdot,\cdot)$ as the metric between data points was used to derive account clusters with a maximum of 16. Blank clusters were allowed. Since k-means operate in a probabilistic way, the number of clusters and their respective sizes may vary. To address this, $\lceil\sqrt{N}\rceil$ runs were made and the clustering maximizing the total inter-cluster distance $\bar{d}$ was selected. Assume that clusters C_j and $C_{j'}$ with $j \neq j'$ have respectively $|C_j|$ and $|C_{j'}|$ elements each. Equation (12) is the inter-cluster distance $d_{j,j'}$ between them:

$$d_{j,j'} \overset{\triangle}{=} \frac{1}{|C_j|\,|C_{j'}|} \sum_{\mathbf{w}\in C_j} \sum_{\mathbf{w}'\in C_{j'}} h(\mathbf{w}, \mathbf{w}') \tag{12}$$

Adding the inter-cluster distance for every distinct pair of clusters (j, j') yields $\bar{d}$ as shown in equation (13). Maximizing $\bar{d}$ means that clusters are more discernible and therefore clustering is more clear.

$$\bar{d} \overset{\triangle}{=} \sum_{(j,j')} d_{j,j'} \tag{13}$$

Figure 2 shows the clustering achieving the maximum $\bar{d}$. Observe there are only 14 clusters instead of the maximum 16. Moreover, there is one relatively large cluster followed by three groups of sizes two, three, and four respectively. However, only the three largest clusters have more than 10% of the original data points, whereas four clusters have a percentage size less than 5%. The similarity matrix $\mathbf{F}_g$ contains the normalized metric values only for data points of the same cluster. Data points of different clusters are assumed to have zero similarity. Note that every dataset contains the netizens participated to this survey to ensure that there is a core truth in them.

3.6 Non-linear Kernel for Twitter

The final component of the tailored Twitter version of the general trust scheme of equation (3) is the final non-linear kernel $T(\cdot)$. The particular selection is the sigmoid function of equation (14):

$$T(u;\beta_1) \overset{\triangle}{=} \frac{1}{1+\exp(-\beta_1 u)} \tag{14}$$

The particular selection has a number of attractive properties including the following:

- It has a domain which is similar to U, the standard interval used in this work. Still translation and scaling are in order as will be explained below.
- Its derivative is also smooth and thus the mapping to the final trust score has no discontinuities.
- It has close ties to the signal estimation field. Specifically it has a natural interpretation as a bit estimator in the presence of white noise.

In the preceding equation the argument u is the linear combination of the individual and group trust scores as explained earlier:

$$u \overset{\triangle}{=} w_i \mathbf{f_i}[j] + w_b \mathbf{F_g}[\neg j; j]^T \mathbf{f_i}[\neg j] \tag{15}$$

The first derivative of the sigmoid function has the following recursive form which allows higher order derivatives to be expressed as polynomials of the original function as shown in equation (16):

$$\frac{\partial T(u)}{\partial u} = \beta_1 T(u)(1 - T(u)) \tag{16}$$

The dependence of the derivative on both the current value $T(u)$ and $1 - T(u)$ is what keeps the kernel values bounded as they are opposing factors.

Since $|T(u)| \leq 1$ it immediately follows that the derivative is close to zero when u is close to $\pm 5/\beta_1$ whereas for any intermediate values it holds that:

$$\left|\frac{\partial T(u)}{\partial u}\right| \leq \beta_1 \tag{17}$$

[4] https://gdpr.eu

Table 3: Sentiment per topic.

	News	Sports	Cinema	Music	Technology
Positive	94.32%	72.53%	68.03%	58.74%	57.21%
Negative	5.68%	7.47%	31.97%	41.26%	42.79%

This implies that the maximum increase rate of $T(u)$ is bounded and controlled. Therefore, it can be adjusted to any particular scenario as appropriate.

4 EXPERIMENTS

4.1 Results

For the construction of the datasets we relied on the Twitter API together with Twitter4j[5], a Java based library. The tweets were progressively collected from 01/06/2019 to 01/06/2020. Table 4 contains information about them. In the experiments the following classification methods, which were readily available in the MLlib library, were used:

- Naive Bayes is perhaps the simplest classifier (Jiang et al., 2019; Chen et al., 2020).
- Logistic regression is a very common binary classification scheme (Sur and Candès, 2019; Wu et al., 2019; Denoeux, 2019).
- Support vector machine (SVM) is a fairly sophisticated classifier where the minimum distance between classes is guaranteed to be maximized (Wang and Chen, 2020; Shen et al., 2019; Chen et al., 2019).

These datasets consisting of English tweets were collected from trending topics using topic sampling:

- **Terror Dataset:** The first one responds to a discussion topic about a social situation with duration in time and very different activity levels from time to time.
- **US Election Dataset:** This dataset reflects a discussion topic regarding the elections as well as the two candidates with quite linear activity in time.
- **Harvey Dataset:** The third topic deals with an emerging tragic event. The related discussion has a bursty activity for the first few days but then fades to very low activity levels.

In order to evaluate the model of equation (3) performance each classifier-dataset combination was executed ten times. Although the respective sizes of the training and testing segments were constant, each time their contents were random in order to ensure a fair assessment. The values of accuracy, precision, recall, and F1 score was the arithmetic mean of the respective results. These figures were obtained directly by the MLlib for its classifiers.

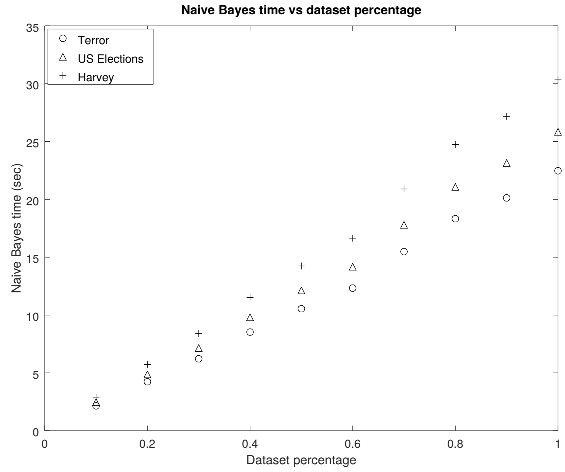

Figure 3: Time for the naive Bayes.

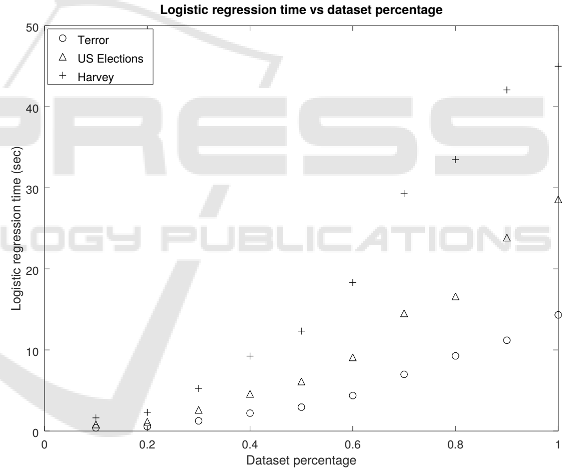

Figure 4: Time for the logistic regression.

Figures 3, 4, and 5 contain the total execution time in seconds for the model of equation (3). Each point is derived by the arithmetic mean of ten executions. To see the scaling patterns, a fraction of each dataset was given to the model. From these figures it can be clearly seen that the SVM variant is the slower of the three but achieves higher accuracy. The logistic regression is quicker but only at the expense of lower accuracy, whereas the naive Bayes is very fast but the accuracy is even lower.

[5] http://twitter4j.org

Table 4: Dataset Features.

Topic	Tweets	Keywords
Terror	285.000	terrorist, terror, attack
US Elections	410.000	donald, trump, joe, biden
Harvey	920.000	Category 4, harvey, hurricane

Table 5: Scores for the terror dataset.

Classifier	Accuracy	Precision	Recall	F1 score
Naive Bayes	97.7%	36.1%	50.9%	42.3%
Logistic Regression	98.2%	40.6%	11.8%	18.3%
Support Vector Machine	98.3%	50.3%	17.9%	26.4%

Table 6: Scores for the US election dataset.

Classifier	Accuracy	Precision	Recall	F1 score
Naive Bayes	97.9%	36.9%	49.5%	42.3%
Logistic Regression	98.2%	31.4%	11.2%	16.5%
Support Vector Machine	98.3%	55.6%	19.1%	28.3%

Table 7: Scores for the Harvey dataset.

Classifier	Accuracy	Precision	Recall	F1 score
Naive Bayes	99.2%	15.9%	53.9%	28.1%
Logistic Regression	99.7%	22.2%	7.2%	10.9%
Support Vector Machine	99.8%	52.6%	10.2%	17.1%

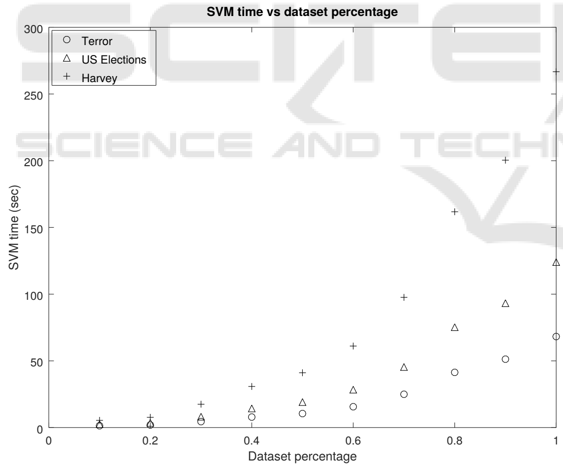

Figure 5: Time for the SVM.

4.2 Recommendations

In view of the results presented earlier in this section as well as of the recommendations given in the scientific literature, it is generally recommended that the following best practices be put in place when estimating Twitter trust levels:

- Since Web trust is a human trait, efforts should be made in order to involve humans in the general trust evaluation process. Still this has to be as non-invasive as possible. Moreover, the implications of the “Who watches the watchmen?” (*Quis custodies ipsos custodes?*) question should be very carefully considered.
- Alternative ML models should be trained with the same data in order to gain insight into the activity patterns of trusted accounts. This is especially true since trolls and fake news farms continue to adapt to the digital countermeasures against them.
- When in doubt and in high risk cases, insert a human in the decision loop. This may be tedious or slow but eventually it is more accurate if the human operator has the proper data available. To this end, although outside the scope of this conference paper, data summarization and visualization techniques can help.

5 FUTURE WORK DIRECTIONS

This conference paper focuses on the computation over Spark of the trustworthiness of a massive number of Twitter accounts. A novel general Web trust model is proposed based on a combination of online individual activity attributes and that of the activity of similar accounts. Three large English tweet datasets obtained from topic sampling using trending topics served as benchmarks. The proposed approach achieved high accuracy values. Moreover, out of the three alternative ML models available and based on the aforementioned performance metrics the SVM outperforms both the logistic regression and the naive Bayes, while out of the latter two the logistic regressor fares much better.

Regarding future research work, the scalability issues emerging from applying the proposed approach to much larger datasets should be investigated. Research may well cover some of the numerous applications of Web trust, which include fake news discovery, viral marketing, branch loyalty, or massive online fact checking for political campaigns.

ACKNOWLEDGEMENTS

Andreas Kanavos, Aristidis Ilias and Christos Makris have been co-financed by the European Union and Greek national funds through the Regional Operational Program “Western Greece 2014-2020”, under the Call “Regional Research and Innovation Strategies for Smart Specialisation - RIS3 in Information and Communication Technologies” (project: 5038701 entitled “Reviews Manager: Hotel Reviews Intelligent Impact Assessment Platform”).

REFERENCES

Adali, S. et al. (2010). Measuring behavioral trust in social networks. In *ISI*, pages 150–152.

Adib, P., Alirezazadeh, S., and Nezarat, A. (2017). Enhancing trust accuracy among online social network users utilizing data text mining techniques in Apache Spark. In *ICCKE*, pages 283–288.

Ajao, O., Bhowmik, D., and Shahrzad, Z. (2018). Fake news identification on Twitter with hybrid CNN and RNN models. In *SMSociety*, pages 226–230.

Albanie, S., Nagrani, A., Vedaldi, A., and Zisserman, A. (2018). Emotion recognition in speech using cross-modal transfer in the wild. In *ACM Conference on Multimedia Conference*, pages 292–301.

Alexopoulos, A., Drakopoulos, G., Kanavos, A., Sioutas, S., and Vonitsanos, G. (2020). Parametric evaluation of collaborative filtering over Apache Spark. In *SEEDA-CECNSM*.

Attia, M., Nasr, M., and Kassem, A. (2016). e-mail systems in cloud computing environment privacy, trust and security challenges. *IJERA*, 6:63–68.

Baldwin, S. (2017). “how to pronounce meme”. Three YouTube channels. *Humanities*, 6(1):10.

Boselie, F. and Wouterlood, D. (1989). The minimum principle and visual pattern completion. *Psychological research*, 51(3):93–101.

Buntain, C. and Golbeck, J. (2017). Automatically identifying fake news in popular Twitter threads. In *SmartCloud*, pages 208–215.

Buskens, V. (2002). *Social Networks and Trust*. Springer Science & Business Media.

Castillo, C., Mendoza, M., and Poblete, B. (2011). Information credibility on Twitter. In *WWW*, pages 675–684.

Chen, S., Webb, G. I., Liu, L., and Ma, X. (2020). A novel selective naïve Bayes algorithm. *Knowledge Based Systems*, 192.

Chen, Y., Xiong, J., Xu, W., and Zuo, J. (2019). A novel online incremental and decremental learning algorithm based on variable support vector machine. *Cluster Computing*, 22(Supplement).

Chiu, M., Shu, J., and Hui, P. (2018). Emotion recognition through gait on mobile devices. In *PerCom Workshops*, pages 800–805.

Ciordas-Hertel, G., Schneider, J., Ternier, S., and Drachsler, H. (2019). Adopting trust in learning analytics infrastructure: A structured literature review. *The Journal of Universal Computer Science*, 25(13):1668–1686.

Davis, S. K., Morningstar, M., Dirks, M. A., and Qualter, P. (2020). Ability emotional intelligence: What about recognition of emotion in voices? *Personality and Individual Differences*, 160.

Denoeux, T. (2019). Logistic regression, neural networks and Dempster-Shafer theory: A new perspective. *Knowledge Based Systems*, 176:54–67.

Drakopoulos, G., Kafeza, E., and Al Katheeri, H. (2019a). Proof systems in blockchains: A survey. In *SEEDA-CECNSM*, pages 1–6.

Drakopoulos, G., Kafeza, E., Mylonas, P., and Al Katheeri, H. (2020). Building trusted startup teams from LinkedIn attributes: A higher order probabilistic analysis. In *ICTAI*.

Drakopoulos, G., Kanavos, A., Mylonas, P., and Sioutas, S. (2017). Defining and evaluating Twitter influence metrics: A higher order approach in Neo4j. *SNAM*, 7(1):52:1–52:14.

Drakopoulos, G., Pikramenos, G., Spyrou, E. D., and Perantonis, S. J. (2019b). Emotion recognition from speech: A survey. In *WEBIST*, pages 432–439.

Egger, M., Ley, M., and Hanke, S. (2019). Emotion recognition from physiological signal analysis: A review. *Electronic Notes in Theoretical Computer Science*, 343:35–55.

Furnell, S. and Spafford, E. H. (2019). The Morris worm at 30. *ITNOW*, 61(1):32–33.

Garg, K., Saraswat, P., Bisht, S., Aggarwal, S. K., Kothuri, S. K., and Gupta, S. (2019). A comparitive analysis on e-voting system using blockchain. In *IoT-SIU*, pages 1–4. IEEE.

Gray, E., Seigneur, J., Chen, Y., and Jensen, C. D. (2003). Trust propagation in small worlds. In *iTrust*, pages 239–254.

Haimson, O. L. and Hoffmann, A. L. (2016). Constructing and enforcing “authentic” identity online: Facebook, real names, and non-normative identities. *First Monday*.

Hallikainen, H. and Laukkanen, T. (2018). National culture and consumer trust in e-commerce. *International Journal of Information Management*, 38(1):97–106.

Howard, J. (2019). Bandwagon effect and authority bias. In *Cognitive Errors and Diagnostic Mistakes*, pages 21–56. Springer.

Im, J., Chandrasekharan, E., Sargent, J., Lighthammer, P., Denby, T., Bhargava, A., Hemphill, L., Jurgens, D., and Gilbert, E. (2020). Still out there: Modeling and identifying russian troll accounts on Twitter. In *ACM Conference on Web science*, pages 1–10.

Jain, N., Kumar, S., Kumar, A., Shamsolmoali, P., and Zareapoor, M. (2018). Hybrid deep neural networks for face emotion recognition. *Pattern Recognition Letters*, 115:101–106.

Jiang, L., Zhang, L., Yu, L., and Wang, D. (2019). Class-specific attribute weighted naive Bayes. *Pattern Recognition*, 88:321–330.

Kamvar, S. D., Schlosser, M. T., and Garcia-Molina, H. (2003). The EigenTrust algorithm for reputation management in P2P networks. In *WWW*, pages 640–651.

Kim, Y. and Peterson, R. A. (2017). A meta-analysis of online trust relationships in e-commerce. *Journal of Interactive Marketing*, 38:44–54.

Kyriazidou, I., Drakopoulos, G., Kanavos, A., Makris, C., and Mylonas, P. (2019). Towards predicting mentions to verified Twitter accounts: Building prediction models over MongoDB with keras. In *WEBIST*, pages 25–33.

Lewis, M., Zettersten, M., and Lupyan, G. (2019). Distributional semantics as a source of visual knowledge. *PNAS*, 116(39):19237–19238.

Li, X., Jiang, P., Chen, T., Luo, X., and Wen, Q. (2020). A survey on the security of blockchain systems. *Future Generation Computer Systems*, 107:841–853.

Liang, G., Weller, S. R., Luo, F., Zhao, J., and Dong, Z. Y. (2018). Distributed blockchain-based data protection framework for modern power systems against cyber attacks. *IEEE Transactions on smart grid*, 10(3):3162–3173.

Lumbreras, A. and Gavaldà, R. (2012). Applying trust metrics based on user interactions to recommendation in social networks. In *ASONAM*, pages 1159–1164.

Massa, P. and Bhattacharjee, B. (2004). Using trust in recommender systems: An experimental analysis. In *International conference on trust management*, pages 221–235. Springer.

Obimbo, C., Speller, A., Myers, K., Burke, A., and Blatz, M. (2018). Internet worms and the weakest link: Human error. In *CSCI*, pages 120–123. IEEE.

O'Donovan, J. and Smyth, B. (2005). Trust in recommender systems. In *IUI*, pages 167–174.

Oliveira, T., Alhinho, M., Rita, P., and Dhillon, G. (2017). Modelling and testing consumer trust dimensions in e-commerce. *Computers in Human Behavior*, 71:153–164.

Papaoikonomou, T., Kardara, M., Tserpes, K., and Varvarigou, T. A. (2013). The strength of negative opinions. In *EANN*, volume 384, pages 90–99.

Patil, S., Deshpande, S., and Potgantwar, A. D. (2017). Product recommendation using multiple filtering mechanisms on Apache Spark. *International Journal of Scientific Research in Network Security and Communication*, 5(3):76–83.

Ranganathan, H., Chakraborty, S., and Panchanathan, S. (2016). Multimodal emotion recognition using deep learning architectures. In *WACV*, pages 1–9. IEEE.

Richardson, M., Agrawal, R., and Domingos, P. M. (2003). Trust management for the Semantic Web. In *ISWC*, pages 351–368.

Risnanto, S., Rahim, Y. B. A., and Herman, N. S. (2019). Preparatory component for adoption e-voting. In *TSSA*, pages 31–34. IEEE.

Salahdine, F. and Kaabouch, N. (2019). Social engineering attacks: A survey. *Future Internet*, 11(4):89.

Schirmer, A. and Adolphs, R. (2017). Emotion perception from face, voice, and touch: Comparisons and convergence. *Trends in cognitive sciences*, 21(3):216–228.

Sengupta, J., Ruj, S., and Bit, S. D. (2020). A comprehensive survey on attacks, security issues and blockchain solutions for IoT and IIoT. *Journal of Network and Computer Applications*, 149.

Sharma, A. and Sharma, T. (2017). HR analytics and performance appraisal system. *Management Research Review*.

Shen, M., Tang, X., Zhu, L., Du, X., and Guizani, M. (2019). Privacy-preserving support vector machine training over blockchain-based encrypted IoT data in smart cities. *IEEE Internet of Things Journal*, 6(5):7702–7712.

Sur, P. and Candès, E. J. (2019). A modern maximum-likelihood theory for high-dimensional logistic regression. *PNAS*, 116(29):14516–14525.

Tavakolifard, M., Almeroth, K. C., and Gulla, J. A. (2013). Does social contact matter? modelling the hidden Web of trust underlying Twitter. In *WWW*, pages 981–988.

Terzi, D. S., Terzi, R., and Sagiroglu, S. (2017). Big data analytics for network anomaly detection from netflow data. In *UBMK*, pages 592–597. IEEE.

Ventocilla, E. (2019). Big data programming with Apache Spark. In *Data science in practice*, pages 171–194. Springer.

Wang, M. and Chen, H. (2020). Chaotic multi-swarm whale optimizer boosted support vector machine for medical diagnosis. *Applied Soft Computing*, 88.

Wang, S.-H., Phillips, P., Dong, Z.-C., and Zhang, Y.-D. (2018). Intelligent facial emotion recognition based on stationary wavelet entropy and Jaya algorithm. *Neurocomputing*, 272:668–676.

Wu, S., Sanghavi, S., and Dimakis, A. G. (2019). Sparse logistic regression learns all discrete pairwise graphical models. In *NIPS*, pages 8071–8081.

Xue, P., Li, B., Wang, N., and Zhu, T. (2019). Emotion recognition from human gait features based on DCT transform. In *International Conference on Human Centered Computing*, pages 511–517. Springer.