Real-Time Identity-Deception Detection Techniques for Social Media: Optimizations and Challenges

Michail Tsikerdekis
Western Washington University

Identity-deception detection methods have been proposed for social-media platforms with high effectiveness, but their efficiency can vary. Previous literature has not examined the potential of these methods to work as real-time monitoring systems. Such implementations further highlight the challenges of applying computationally intensive methods in online environments that involve large datasets and high speeds of data. This paper attempts to classify detection methods based on the approaches and identifies factors that, in real-time systems, will impact the effectiveness and efficiency of these methods. Optimizations are proposed that can limit the computational overhead. Further challenges involving real-time identity-deception detection are discussed.

Identity deception in social media has received attention in recent years.[1,2] The rapid growth of the user population for social-media platforms coupled with the ease of creating new accounts has increased identity-deception attacks.[3] These attacks can create disruptions in the normal operations of online communities and impact their users. Research attention to novel identity-deception detection methods has addressed the issue of identity deception using computational solutions. These are more effective than detection performed by humans. However, the computational overhead is rarely taken into account within the size and scope of social-media platforms. These computational solutions can only realistically be expected to have an impact on reducing identity-deception cases if they can monitor online communities on a real-time basis. The longer a deceptive account is allowed to operate unchecked, the larger its negative impact on an online

Published by the IEEE Computer Society
1089-7801/18/$33.00 ©2018 IEEE

community. The ideal goal is to continuously monitor a new account and render judgments based on its actions until a confidence threshold is surpassed that would allow for classifying it as legitimate or malicious. However, literature has not examined the potential of these detection methods to monitor these online communities in real time.

This paper provides a survey of existing identity-deception detection methods and identifies their potential application as real-time monitoring systems in large-scale social-media platforms. Their performance is discussed based on the estimated time complexity and their ability to cope as the data velocity increases. Optimizations for these and future identity-deception detection techniques are proposed for large-scale social-media platforms that can help mitigate the computational overload when monitoring systems in real time. Given the scope and size limitations of this article, the approach taken is theoretical and aims to open a discussion on novel approaches where real-time detection systems can be made possible for social-media platforms.

IDENTITY-DECEPTION ATTACKS

The social-media identity-deception attacks that this paper considers are identity theft and identify forgery. These involve generating accounts and employing identity management,[4] as well as behavioral strategies (such as navigating to a particular page) aimed at deceiving others. Identity-management strategies are context-specific and begin with the selection of login credentials and subsequent editing of various attributes associated with the account (such as age and employment). Table 1 shows categories and examples of such attacks. Identity creation describes scenarios in which a legitimate identity in real life does not exist in an online platform (such as Facebook), yet the attacker creates the identity based on the data available from other sources.[1] Profile cloning involves cases in which a legitimate identity is present and cloned.[3] Disruption attacks are direct in their nature (such as spamming on a page until banned), and attackers often expect to be caught. Sockpuppetry, on the other hand, involves attacks that are more sophisticated; often, a puppeteer's account is also present in the platform (albeit probably banned).[5] Finally, Sybil attacks are a special case of sockpuppetry with a network-centric focus.[6] Multiple identities are generated towards achieving an often long-term goal.

Table 1. Identity-deception attacks.

Category	Subcategory	Examples
Identity theft	Identity creation	Trojan horse (such as seeking to gain access to the identity's network)
	Profile cloning	Phishing and scamming
Identity forgery	Disruptive attacks	Trolling (such as misinformation and annoying messages)
	Sockpuppetry	Spreading propaganda, phishing, scamming, and circumventing bans
	Sybil attacks	Increasing reputation, skewing votes, and reviewing scores

IDENTITY-DECEPTION DETECTION METHODS

Several methods have been proposed to detect identity-deception attacks. Some have explicitly aimed at targeting a particular identity-deception category, while others need to be generalized.

This paper classifies methods based on the major approach or theory that guides the detection method.

Immediate-Comparison Methods

Methods that fall under the immediate-comparison category aim to primarily identify cases of profile cloning, sockpuppetry, and Sybil attacks. The general approach compares data directly from a user's account to an existing set of users. It aims to identify cases in which duplicate accounts exist in the system or in which mismatched information exists about a user. The approach can further be divided based on the types of data utilized (verbal and/or non-verbal).

Profile-based methods

Profile-based methods aim to cross-validate account profiles with an existing set of users. The features can be verbal (text) or non-verbal (such as duration between actions). A tested approach using profile-based features identified duplicate accounts in a criminal database.[7] Duplicate records in the data were suspected to be the result of individuals who may have reported false names under separate arrests. The technique traversed through the list, and each record's profile was compared to the rest of the dataset to identify cases based on a profile score and a particular threshold. The technique was highly effective even in the presence of missing data. However, the original method's time complexity was high (asymptomatic time complexity of $O(N)$ for checking a new user against the total user set). The overhead of such a technique could be larger if profile scores for users have to be updated when an account change occurs (such as employment information updated).

A similar approach was used in the social-networking site LinkedIn.[3] The technique utilized profile-based features to identify cases of manual or automatic profile cloning through bots. The method was proposed as a comparison tool meant to be utilized by a user. It analyzed a user's profile and, using LinkedIn's search, attempted to find profiles that may have cloned part of the original. If it were to be applied for all accounts based on real-time changes on accounts, the method would offer high precision at the expense of computational overheads. The estimated time complexity of the method requires an account to be tested against the rest of the dataset ($O(N)$).

Profile-based features can also be used to cross-reference the validity of an account on a social network, A, using information found in a set of anonymized social networks, AG_i.[8] The trustworthiness of features for an account can be evaluated based on anonymized social networks that also contain the account under investigation. The method demonstrated a high accuracy for various network structures; however, the computational overhead of a part of the proposed algorithm had a worst case of $O(|N|^2)$ in respect to the nodes in A.

Verbal-only methods

An alternative approach compares a particular account to the rest of the dataset based on verbal communication. This approach allows for identification of not only cases regarding profile cloning but also cases of sockpuppetry and Sybil attacks.

The approach has been used on Wikipedia to identify sockpuppet accounts.[5] The authors built a natural-language-processing algorithm that analyzed lexical features for a set of Wikipedia accounts to detect likely sockpuppets. The accuracy of the method is considerably high; however, the computational overhead is also high. Comparing a user to all other accounts will result in a time complexity of $O(N)$. But, unlike profile-based approaches, an algorithm will have to traverse for each user, N_i, through all account content, R_i (revisions on article pages in the case of Wikipedia). Effectively, the operation translates to a comparison of all revisions made by users. If no index is used, an algorithm will result in a complexity of $O(N * R)$, while if a binary search tree is used for the revision dataset, the complexity can be reduced to $O(N \log R)$. Platforms such as Twitter have considerably higher proportions of user-content pairs than Wikipedia, rendering the real-time application of this method more impractical.

Baseline Methods

Methods under this category relate to the expectancy-violations theory. The theory looks at how individuals react to unexpected behavior (such as evidence of deception unintentionally leaked by a deceiver). This is translated to a computer deception detector—often using machine-learning algorithms—that is looking for violations of expected behavior, which can often reveal deception. In particular contexts, one can identify a "universal" expected legitimate user behavior, which serves as a baseline. Deviations from this baseline are considered deceptive cues and indicative of identity deception. Contrary to immediate-comparison approaches, baseline approaches require a previously established supervised learning model that is often the result of a particular machine-learning algorithm. The quality of the model and that of the training data will influence the effectiveness of the baseline and the prediction accuracy of a model. These models are more effective at resolving most of the identity-deception cases such as scamming and sockpuppetry. Approaches under this category are further divided into verbal, non-verbal, and hybrid.

Verbal methods

Verbal approaches utilize the content delivered by an individual and look for deceptive cues that are known to be associated with deceptive behavior. Often, these relate to lexical, prosodic, or even sentiment features of speech, whether written or audible.[9] These methods have been extensively applied to spam detection and website scamming attacks.[10] In social media, the approach restricts the attention of the detection algorithm to one account and focuses on verbal data produced from the account. These methods can be extremely effective, as well as computationally efficient ($O(C)$ where C is the content produced by a particular user) since they examine only content produced by one individual. However, their effectiveness is also dependent on the nature of the social-media platform. For example, sentiment analysis in short text is less accurate,[11] so the effectiveness of the method may be limited on Twitter.

The method has been found to be effective in the field of forensics in distinguishing adults posing as children on social networks.[12] Training data establish a baseline for age and gender by utilizing a combination of natural language processing and stylistic language fingerprinting. The approach is highly effective and efficient.

Non-verbal methods

Non-verbal approaches work in a similar manner. They include metadata, summarized actions of an individual, and even metrics that represent user actions or behavior within a particular platform's cyberspace. They have been found to be effective and computationally efficient.[13] In the event that some of the summarized data have already been pre-calculated, the methods can be substantially more efficient than verbal natural-language-processing approaches.

Hybrid methods

Recently, a hybrid approach in which verbal and non-verbal data are combined under one supervised learning model has been proposed, although results have not yet been reported.[14] It is likely that the method could potentially be more effective at detecting identity deception at the expense of efficiency (due to the need for more data processing).

Hybrid Approaches

Recent techniques have attempted to combine the two broader categories to detect identity deception. This is often achieved on the basis of generating a baseline model; however, it often also involves immediate comparisons to existing accounts or other data on the platform. An example of this can be seen through the use of social-network analysis. When a particular user u_i needs to be tested for identity deception, a graph G is built for all users V and connections E between them. Connections can be based on friendships but can also consist of any connection deemed important for an online community (such as followers on Twitter and common articles edited on

Wikipedia). The user's profile in G is compared using various social-network metrics (such as closeness centrality) in respect to all users in V. Then, using a baseline, a decision is rendered on whether the "profile" of u_i matches that of known identity-deception accounts. The decision can be made using supervised learning algorithms or based on an arbitrary metric and a particular threshold.

A recent study has identified, with high precision, Sybil attacks using social-network data.[6] The approach utilized a graph to determine the probable clusters of Sybil and non-Sybil accounts. The time complexity of the approach was $O(n \log n)$. The computational benefits of this approach are based on the fact that it examined the degree of a vertex as opposed to more computationally intensive graph metrics. A similar study proposed a graph approach using a game-theoretic model to explain deception.[15] The study demonstrated the rich potential of utilizing social-network data for deception detection. Implementing similar techniques in real time holds great potential but also poses substantial computational challenges.

REAL-TIME POTENTIAL DETECTION METHODS

This paper posits that, to utilize these methods in real time, one has to consider three primary factors that influence their efficiency: the number of users on a platform, technology infrastructure, and data velocity. Figure 1 depicts a representation of a hypothetical real-time implementation of an identity-deception detection system on social media.

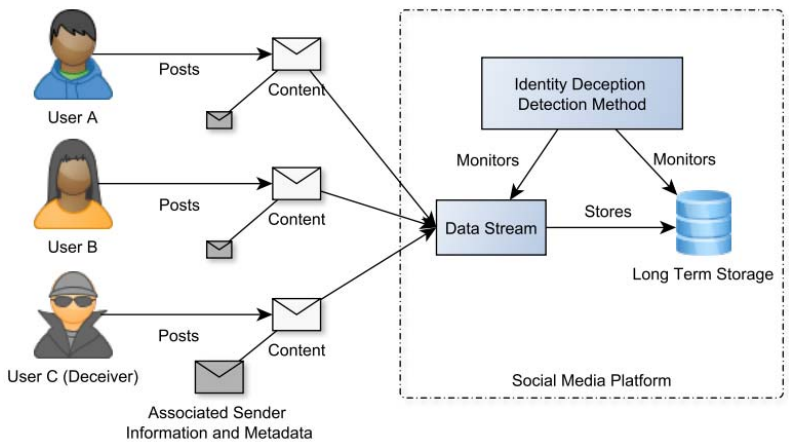

Figure 1. Depiction of an identity-deception detection system on a social-media platform intended to be used for real-time monitoring.

Social-media platforms vary substantially in respect to their user volumes. This will also have an impact on data volume that has to be analyzed by a particular identity-deception detection algorithm. When this needs to occur in real time, the size of the user population may be restrictive for some methods unless optimizations are considered. Additionally, the computational overhead can vary between the methods presented. The overhead may also be influenced by the existing infrastructure. For example, real-time applications have been said to require an active stream that provides an algorithm with the latest changes, but there is also a need for seamless access to older data.[16] If a requirement is not met, some of the methods cannot feasibly be implemented for real-time detection. Finally, when discussing real-time application, data velocity is perhaps the most important factor. For the purposes of this paper, data velocity is defined as the intensity of change in the data. Although different measurements may be used depending on the social-media platform, data velocity describes the flow of data over time. For example, in the context of Twitter, this may be translated as the number of tweets per second. Data velocity will influence the number of times a detection method needs to be executed over a time interval.

Baseline methods are the most efficient. They are independent of the number of users on a platform and require limited support by infrastructure. As such, they can afford to be used in platforms with higher data velocities. Immediate-comparison approaches, on the other hand, can be heavily influenced by the total number of users. As such, when data velocity is high, they will suffer from redundancies that will make real-time application improbable for large social-media platforms. Finally, hybrid approaches are by far the most expensive methods for identity-deception detection. As the total number of users rises, so does the need for substantially more infrastructure to support data processing. For example, social-network analyses are notorious for the memory requirements as graph sizes grow. Each user tested is required to be added to the existing graph, and many metrics will have to be recalculated for all vertices in a graph. Few metrics are not bound by the need for calculating the same metric for all vertices.

OPTIMIZATIONS FOR REAL-TIME OPERATION

This paper identifies several optimizations that can help overcome some of the aforementioned limitations while attempting to preserve most of the effectiveness. The estimated impact on effectiveness is reported based on the literature. The optimizations are meant to be representative, but not exhaustive.

Window Optimization

This optimization aims to apply restrictions on the visibility of data, which effectively reduces the amount of data and the computational overhead while attempting to limit the loss of information—which may cause a reduction in an algorithm's detection accuracy. These methods can impact efficacy positively or negatively depending on the context and the bounds of a system. In the context of real-time detection, it can be further divided based on its application on an active stream of data or complete dataset.

Applying window on stream

While many real-time applications rely on a stream to monitor changes, some attempt to store the most recent set of changes based on a window.[16] This helps reduce the need to query old data directly from a database and provides data available for immediate analysis. The size of this window can be determined based on time (such as changes during the past day) or based on data (such as most recent 1,000 changes). The optimization can benefit baseline approaches, especially in contexts in which recent behavior can be more indicative of a deceptive account than the complete history of an account. However, the optimization is not applicable to methods that rely on account history, as well as immediate-comparison and hybrid approaches.

Applying window on dataset

Methods that rely on accessing past records of data (aside from the active stream) can also be optimized based on windows. Like with the previous optimization focused on the stream, windows can be applied in relation to time or data. The impact of this optimization can radically reduce time complexity for algorithms. For example, a window has been used for detecting duplicate records on a criminal database that reduced the time complexity from $O(N)$ to $O(\log N)$ when testing for a particular user.[7] This came at no cost to efficacy. Similar windows can benefit social-network analysis. For example, localized graph statistics can be calculated based on a restricted view of the overall network based on an n-step reach from the particular user to be tested. This reduces the amount of vertices in a graph and the computational overhead. Finally, the method may yield higher efficacy in baseline approaches, because it could reduce the error caused by the assumption that a community exhibits uniform behavior. As users change behaviors over time, establishing a baseline on recent data may be a more effective strategy.

Adaptive window optimization

Window optimization can be taken a step further. This is often overlooked by literature since the aim is to demonstrate primarily effectiveness of an approach rather than efficiency for real-time implementation. However, social media do not have a static number of users nor a static data velocity. For example, catastrophes can often result in users becoming more active on social-media platforms such as Twitter. Bursts of activity can potentially impact an otherwise functioning real-time deception-detection system. If windows are applied as an optimization technique, having them automatically adjust their size based on data velocity will likely render them more efficient. The tradeoff of accuracy (or inaccuracy) due to limited data can further be remedied by using an adjustable penalty on the probability that an account is deceptive.

Monitoring only Non-Cleared Accounts

This optimization aims to reduce computational overhead without impacting efficiency. It is ideal for baseline methods and can be applied on the stream of data. Actions that come from accounts that have already been cleared as non-deceptive accounts do not have to be monitored. This is likely to occur after an account has been monitored for a said time. For example, a study that used baseline behavior determined that deceptive and legitimate accounts' non-verbal data tend to increasingly deviate from one another with the age of accounts.[13] As such, administrators may decide that an account could be considered cleared after N days of repeatedly "clean" activity. Another study has also reported no cost to accuracy by establishing a trusted set of accounts (or trusted features of accounts).[8] However, a more accurate restriction in most social media would be to clear an account on the basis of actions made by a user, rather than of time. This is because an account can often be old but have limited activity.

Probabilistic Sampling

The aim of sampling is to reduce computational overhead; however, it can come at the expense of effectiveness. It can be applied to immediate-comparison, baseline, and hybrid methods. The implementation can be established on the stream of data and the dataset. The method is similar to those found on airport security random checks or tax-fraud detection. Since real-time deception detection iterates upon the actions made by a user, the likelihood that a deceptive account will be selected over time increases as a user's actions increase. As such, in a uniform random sampling strategy, a balance between the probability threshold and the frequency can balance the probability of eventual detection. Use of probabilistic sampling using game theory can also be a more effective alternative to uniform sampling strategies. Bayesian Stackelberg games have been shown to be effective at preventing an adversary from guessing a random strategy, as well as at providing defenders with the ability to add weights to different types of likely adversaries based on the severity of damage they can inflict (such as vandalism versus sockpuppetry attacks).[17] Furthermore, the efficiency of the optimization can be affected by the size of the user population. It can be adapted to adjust to data velocity or remain independent, although the latter may impact detection accuracy.

Landmark Optimizations

Several approaches that involved social-network analyses[15,18] can also be optimized through improvements to navigational operations in graphs. Often, proximity of an account to other accounts on a graph can be indicative of identity deception. Distance calculation in large graphs found in social-media platforms can be expensive. Instead, if landmark nodes can be selected ahead of time (preprocessing), computational overhead can drastically decrease.[19] The expense of this optimization is the reduced accuracy of a path calculation; however, it can often be negligible depending on the method selected (some also allow for exact estimation). The optimization can be beneficial as the user population grows, since it mitigates the impact of data velocity. It also reduces computational overhead for the infrastructure supporting the detection system.

BALANCING EFFICIENCY AND EFFECTIVENESS

Attaining an ideal balance between efficiency and effectiveness is a considerable challenge that pertains to the application of these optimizations. At a low level, all these methods aim to reduce computational complexity by reducing the data volume that is expected to be parsed by a detection method, which in turn reduces infrastructure requirements. In practical terms, the processing capacity of the detection system needs to be equal or larger to the growth rate of accounts. In social-media applications, the user population often exhibits a growth that follows an exponential trend before transitioning into a long logarithmic tail (for example, Wikipedia's user population growth). Since the underlying system (user population) is finite, the trend over time is often similar to that of probing rates seen by internet worms (for example, Slammer). As such, detection-system optimizations are most useful early on the lifecycle during the linear or exponential growth of a population in which the system will be under the most stress. The loss in detection rates (effectiveness) can be compensated for by either adjusting parameters for these methods or combining them. Some of the tradeoffs and benefits of these optimizations are described below.

Window optimizations reduce computational requirements at the cost of lower effectiveness due to data-volume reduction. However, accuracy may not be affected in the event that data quality is high (such as when there is no missing data) and there is low variance in the data.—in other words, when the difference between attacker and legitimate user is "distinct" in the data. This will also depend largely on the context and how user actions are recorded and compared. For example, we would need larger data windows to determine differences in writing style compared to measuring a user's average time between actions. This is also relevant for cases of weak models (high bias) that do not benefit from the addition of more data.

Clearing accounts after passing a certain trust threshold can also benefit the system from repeatedly evaluating the same users. However, in this approach, false negative accuracy for detection methods becomes particularly important (eliminating doubt on cleared accounts). This reversal in perspective may also require changes in the methods since the aim is not to detect but rather to establish trust. Computationally, the optimization holds promise even if the influx rate of new users surpasses the "clearing" rate. By reducing the threshold for false negatives, the detection system will require fewer resources at the cost of allowing some adversaries to pass undetected.

A similar challenge can also be observed with probabilistic sampling. Random sampling rates have a linear relationship with detection rates in respect to the user population. As such, the computational gains and loss of accuracy are directly comparable. However, the relationship between efficiency and effectiveness becomes more promising when sampling is informed on the basis of indicators (for example, if an account registers in the middle of the night, it is more likely to be a deceiver). As such, biased sampling will likely retain more of the unsampled effectiveness while exhibiting an equal reduction in computational complexity compared to random sampling.

Finally, landmark optimizations in networks hold the potential for reducing processing requirements for graph statistics by building shortcuts for many of the calculations. These often have parameters that define how aggressive algorithms should approximate measurements (such as shortest distance). The relationship between measurement error and landmark parameter size will determine resource requirements and accuracy. In some algorithms, this relationship follows a near-linear trend,[20] and so establishing the loss in effectiveness can be more easily estimated. This is also dependent on how important network statistics are in the original detection method.

CHALLENGES AND OPPORTUNITIES

Identity-deception detection approaches are summarized in Table 2 based on the main purpose and real-time factors that can affect them along with applicable optimizations. Real-time applications for these methods can vary. Most optimizations fit almost all approaches proposed in this paper. The impact on detection accuracy by these optimizations is mainly determined by how aggressive their said parameters will be. Further challenges described in this paper relate to the use of baselines in real-time detection methods and identity management.

Interpersonal deception theory posits that a deceiver's behavior changes over time and adapts based on a victim's responses and environmental factors.[21] These changes are made by a deceiver to ensure deception success. Current deception-detection methods do not look for these temporal adaptations in an account's profile (verbal and non-verbal). Instead, they focus on an account's complete history. However, since accounts are tracked constantly in real-time monitoring, these changes in behavior can be captured and incorporated into algorithmic models. These moves and counter-moves can be used to estimate an attacker's behavior even in active defense scenarios.

Table 2. Identity-deception detection approaches and proposed optimizations.

Identity-Deception Detection Method	Ideal for	Factor(s) with Largest Impact	Optimizations
Immediate-comparison	Profile cloning Sockpuppetry Sybil attacks	User population Data velocity	Window optimization on stream Window optimization on dataset Adaptive windows Probabilistic sampling
Baseline	Identity creation Disruptive attacks Sockpuppetry	Data velocity	Window optimization on stream Window optimization on dataset Adaptive windows Probabilistic sampling Monitoring only non-cleared users
Hybrid	Profile cloning Sybil attacks Sockpuppetry Disruptive attacks	User population Data infrastructure Data velocity	Window optimization on stream Window optimization on dataset Adaptive windows Probabilistic sampling Landmark optimizations

Furthermore, baseline approaches often track behaviors in binary form (deceptive versus non-deceptive account) and assume that all individuals need to behave in accordance to an established baseline. An attacker may be able to trick a detection system by acting legitimately until such time when an attack takes place. Looking for leakage cues may be a more effective and computationally efficient method, but current machine-learning algorithms need large amounts of training data from multiple users and as such are problematic in their application for a single user. Alternative cognitive or game-theoretic models may need to be developed that more closely reflect the internal state of individuals and look for subtle deviations from a predicted legitimate user behavioral path.

Subsequent identity-management approaches that share cross-platform data will need to be developed. Cross-referencing across datasets has been proven to be effective in past studies and can help act as an initial security measure for newly created accounts. This is especially true for cases of identity creation. However, the approach may be limited for cases of identity forgery in

which pseudonymous accounts are allowed by a platform. In such cases, access-control protocols (such as gradually granting privileges to a user) based on real-time trust-management systems may prove to be more effective. An attempt by a user to access a feature beyond the limits set by a system may be an early indicator of the user being an attacker.

Finally, identity-deception prevention can also decrease computational overhead for algorithms. Currently, attempts towards "universal" login credentials through social-media logins and other services such as OpenID aim to accommodate a user demand for a reduction in login credentials. However, these logins could potentially offer summarized identity-deception detection information derived from various websites that they have been used in. Such standardization of reliable identity data (such as times during which or locations from which an individual usually operates) will require extensive studies and collaborations across scientific fields. If successful, a new account created on Facebook for use in other websites will have to demonstrate its legitimacy through its activity pattern on Facebook (or whichever other social login is used).

CONCLUSION

Over the course of a decade, progress has been made in not only understanding how deceivers operate online in social media but also how their behaviors can be tracked and identified. However, successful approaches and algorithm lack plans for implementation in realistic large-scale social-media platforms. This paper describes a classification of existing deception-detection approaches and proposes factors and optimizations that need to be considered for a real-time implementation of these approaches in social media. These recommendations call for a re-examining of detection approaches and their potential use in social media. Given the rapid growth of social media and data velocity, it is likely that technology will not save these detection approaches, but rather will offer innovative new ways to make them practical for real-time applications.

REFERENCES

1. M. Tsikerdekis and S. Zeadally, "Online Deception in Social Media," *Communications of the ACM*, vol. 57, no. 9, 2014, pp. 72–80.
2. M. Tsikerdekis and S. Zeadally, "Detecting and Preventing Online Identity Deception in Social Networking Services," *IEEE Internet Computing*, vol. 19, no. 3, 2015, pp. 41–49.
3. G. Kontaxis et al., "Detecting social network profile cloning," *IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops)*, 2011, pp. 295–300.
4. E. Bertino and K. Takahashi, *Identity Management: Concepts, Technologies, and Systems (Information Security & Privacy)*, Artech House Publishers, 2011.
5. T. Solorio, R. Hasan, and M. Mizan, "A Case Study of Sockpuppet Detection in Wikipedia," *Proceedings of the Workshop on Language Analysis in Social Media*, 2013, pp. 59–68.
6. Q. Cao et al., "Aiding the detection of fake accounts in large scale social online services," *Proceedings of the 9th USENIX conference on Networked Systems Design and Implementation*, 2012, p. 15.
7. G.A. Wang et al., "Automatically detecting criminal identity deception: an adaptive detection algorithm," *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, vol. 36, no. 5, 2006, pp. 988–999.
8. C. Dai et al., "Privacy-Preserving Assessment of Social Network Data Trustworthiness," *International Journal of Cooperative Information Systems*, vol. 23, no. 2, 2014.
9. J. Hirschberg et al., "Distinguishing deceptive from non-deceptive speech," *Interspeech, Proceedings of Eurospeech'05*, 2005, pp. 1833–1836.
10. A. Ntoulas et al., "Detecting Spam Web Pages Through Content Analysis," *Proceedings of the 15th International Conference on World Wide Web*, 2006, pp. 83–92.

11. M. Thelwall et al., "Sentiment strength detection in short informal text," *Journal of the American Society for Information Science and Technology*, vol. 61, no. 12, 2010, pp. 2544–2558.
12. A. Rashid et al., "Who Am I? Analyzing Digital Personas in Cybercrime Investigations," *Computer*, vol. 46, no. 4, 2013, pp. 54–61.
13. M. Tsikerdekis and S. Zeadally, "Multiple account identity deception detection in social media using nonverbal behavior," *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 8, 2014, pp. 1311–1321.
14. A.M. Kuruvilla and S. Varghese, "A detection system to counter identity deception in social media applications," *International Conference on Circuit, Power and Computing Technologies (ICCPCT)*, 2015, pp. 1–5.
15. A.C. Squicciarini and C. Griffin, "An Informed Model of Personal Information Release in Social Networking Sites," *International Conference on Privacy, Security, Risk and Trust (PASSAT)*, 2012, pp. 636–645.
16. M. Stonebraker, U. Cetintemel, and S. Zdonik, "The 8 Requirements of Real-time Stream Processing," *SIGMOD Rec.*, vol. 34, no. 4, December 2005, pp. 42–47.
17. J. Pita et al., "Using Game Theory for Los Angeles Airport Security," *AI MAGAZINE*, vol. 30, no. 1, 2009, pp. 43–57.
18. C. Peng et al., "Predicting Information Diffusion Initiated from Multiple Sources in Online Social Networks," *Sixth International Symposium on Computational Intelligence and Design (ISCID)*, 2013, pp. 96–99.
19. K. Tretyakov et al., "Fast Fully Dynamic Landmark-based Estimation of Shortest Path Distances in Very Large Graphs," *Proceedings of the 20th ACM International Conference on Information and Knowledge Management* (CIKM), 2011, pp. 1785–1794.
20. M. Potamias et al., "Fast Shortest Path Distance Estimation in Large Networks," *Proceedings of the 18th ACM Conference on Information and Knowledge Management*, 2009, pp. 867–876.
21. D.B. Buller and J.K. Burgoon, "Interpersonal Deception Theory," *Communication Theory*, vol. 6, no. 3, August 1996, pp. 203–242.

ABOUT THE AUTHOR

Michail Tsikerdekis is an assistant professor in the Computer Science Department at Western Washington University. His research interests include deception, data mining, cybersecurity, and social computing. Tsikerdekis has a PhD in informatics from Masaryk University. Contact him at Michael.Tsikerdekis@wwu.edu.