IJWR

A Robust Opinion Spam Detection Method Against Malicious Attackers in Social Media

Amir Jalaly Bidgoly*[a], Zoleikha Rahmanian [a], Abbas Dehghani [b]

[a] Department of Information Technology and Computer Engineering, University of Qom, Qom, Iran; jalaly@qom.ac.ir, z.rahmanian@stu.qom.ac.ir

[b] Department of Computer Engineering, Faculty of Engineering, Yasouj University, Yasouj, Iran; Dehghani@yu.ac.ir.

ABSTRACT

Online reviews are crucial in influencing consumer decisions and business practices. However, some individuals exploit this system by posting fake reviews, known as spam opinions, to manipulate perceptions. Spam detection systems face significant challenges in robustness due to their primary focus on identifying spam attacks without accounting for adversaries that target the detection mechanisms. This oversight enables spammers to exploit vulnerabilities in traditional algorithms with complex deceptive strategies, ultimately undermining their effectiveness. This paper proposes a novel multi-layer graph-based method that represents reviews, reviewers, and products as interconnected nodes. This approach captures the complex relationships among them and addresses adversarial attempts to manipulate the detection process. Our approach utilizes three key nodes—opinion, reviewer, and product—to assess the honesty, trust, and reliability of reviews, reviewers, and products in the context of potential deception. Furthermore, we develop a simulation tool capable of generating diverse attack scenarios, including those targeting the detection system itself, enabling a comprehensive evaluation of robustness. We compared the performance of our method with other graph-based techniques through simulations and case studies, demonstrating that our method is a competitive solution among existing alternatives.

Keywords*— *Spam Attack, Spam Detection, Spam Opinion, Deception, Robustness.

1. Introduction

Given the considerable impact that user reviews have on the dynamics of social networks as well as on various commercial websites, it has become increasingly evident and impossible to overlook the fact that a considerable number of malicious individuals are making concerted efforts to fulfill their dishonest goals by spreading false opinions that are devoid of any truth or factual basis. Among these malicious individuals, there exists a spectrum of motivation, with some being driven by a desire to artificially enhance the perceived quality and status of inferior products, while others may be pushed by a competitive need to undermine and damage the hard-earned reputations of their rivals within the marketplace. In light of this widespread and disturbing phenomenon of spam reviews, which poses a significant threat to the authenticity and integrity of user-generated content, researchers who specialize in this field have undertaken the important task of developing a range of advanced detection methodologies that are specifically aimed at identifying and mitigating such fake activities and behaviors [1][2][3].

Within the specific framework of spam detection systems, the term "deception" is utilized to describe the calculated actions of spam attackers who strategically exploit their knowledge of the detection methodologies that are in place, thereby obfuscating the system's ability to differentiate between authentic reviews and those that are fabricated, which in turn facilitates the unchecked proliferation of opinion spam. For instance, when examining repeat-based methodologies (as illustrated in [4]), it becomes clear that attackers have the capability to easily manipulate the detection system by making unique and original reviews that avoid any form of repetition, thereby successfully evading the mechanisms that are designed to flag such fake content. In a related manner, text-based detection systems [5][6][7] empower malicious actors to circumvent the various detection protocols by skillfully steering clear of the

http://dx.doi.org/10.22133/ijwr.2025.495515.1256

Citation A.J. Bidgoly, Z. Rahmanian and A.Dehghani, "A Robust Opinion Spam Detection Method Against Malicious Attackers in Social Media", *International Journal of Web Research*, vol.8, no2.,pp.1-10, 2025, doi: http://dx.doi.org/10.22133/ijwr.2025.495515.1256

*Coressponding Author

Article History: Received: 26 December 2024; Revised: 3 February 2025; Accepted: 29 February 2025.

specific terminology that is known to trigger spam identification, assuming they possess a thorough and comprehensive understanding of the benchmarks that are employed by these systems. To truly be effective, a robust spam detection system must be exactly designed with the necessary resilience to tolerate such devious forms of deception, ensuring that it maintains its effectiveness and operational efficiency even in circumstances where attackers are fully aware of its underlying mechanisms and the operational protocols that govern its function [3][8][9].

However, several challenges pose significant obstacles to developing such a robust spam detection system. Simply reading the review text often provides insufficient clues to distinguish spam from legitimate reviews. Spammers' behaviors can be difficult to identify. To effectively mislead customers, they can mimic the writing styles and review patterns of genuine reviewers. Hence, the challenge of enhancing the robustness of spam detection systems against these attacks remains an open issue in the field. To address this, we propose an unsupervised multi-layer graph-based method that utilizes nodes representing reviews, reviewers, and products, as shown in Figure 1.

Our approach detects spam attackers across various scenarios by calculating trust scores for reviewers, honesty scores for reviews, and reliability scores for products, all updated through an iterative algorithm. The reliability score aims to reflect the true reputation of products, disregarding spam influences. Reviews and attackers are identified based on low honesty and trust scores. Additionally, we have developed a simulation tool to generate a substantial number of reviews according to desired behavior patterns, as comprehensive evaluation of spam detection methods necessitates generating spam reviews. We compared our method with other graph-based techniques through simulation, revealing that our approach significantly enhances system robustness. The results indicate the competitive potential of our robust opinion spam detection method relative to existing alternatives. The main contributions of this study can be summarized as follows:

- A multi-layer graph-based approach that models reviews, reviewers, and products as interconnected nodes, capturing complex relationships and adversarial behaviors.
- Dynamic scoring mechanisms (trust, honesty, and reliability scores) updated iteratively to adapt to evolving spam tactics and resist manipulation.
- A simulation tool for generating diverse attack scenarios, including those targeting the detection system, enabling robust evaluation.

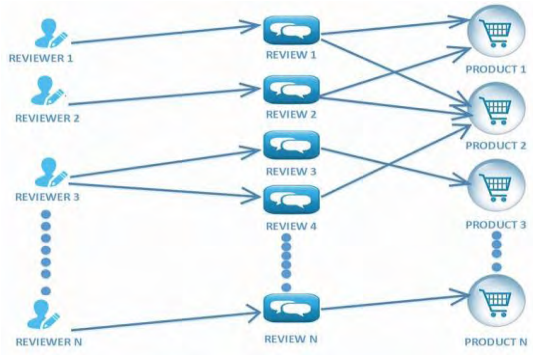

Figure. 1. The interact of different graph nodes

- Enhanced system resilience against sophisticated adversarial strategies, ensuring both spam detection and protection of the detection mechanism.

The paper continues as follows. In the next section, related works are reviewed. In section 3 the motivation and the foundations of the research are presented. The simulator is described in section 4. Further, the proposed method and algorithm are presented in section 5 respectively. Section 6 describes the method using some case studies, and finally, the paper ends in section 7 with the conclusion and future works.

2. Related Work

Due to the rapid growth of social media, many researchers have investigated the security and protection of user data on social media platforms [2][3][10]. An overview of social spam, the spamming process, and social spam taxonomy is given in [2][3][11][12]. Approaches for detecting spam comments are categorized in several ways. Some divide them into two general types: supervised [5][6][7][13][14][15][16] and unsupervised methods[9][17][18]. However, in some other research, they are categorized into three categories: spam reviews, spam attackers and group spam attackers[19].

In the set of spam reviews, the contents of reviews are studied and evaluated. One of the most important approaches in this category is repeated-based detection. These methods try to identify spam reviews by going through repetitive patterns of reviews from the same or different reviewers about similar or different products [14][20]. In addition to this, concept repetition can also be introduced as a measurement criterion for spam comments; the method provided by Alger et al. to identify spam comments [20]. Doing multiple counterfeit reviews is time-consuming and costly. Spam attackers often do not produce a large number of exclusive counterfeit reviews. They tend to copy the existing text. Therefore, identifying similar opinions is a central part of detecting spam comments. Some literature

uses a linguistic character in the text of review [21][22].

In the field of identifying spammers, some methods [9][17][19] use inter relationships between the review, the reviewer, and product graph, as shown in Figure 1, to identify the spammer and also compute the trustworthiness of the reviewer, the honesty of review, and the reliability of the store. Some researchers believe that spammers use a specific period to generate spam comments. The numbers of reviews rise dramatically in that interval; thus, they use burst patterns to identify spam attacker [23][24]. Some of them, for example, use time series to identify spammers [25][26]. Few pieces of research have been done on spammer groups' identification [27][28].

In [29], the authors emphasized the importance of trust-based decision-making in hostile environments, investigating various decision-making methods using formal verification in different attack scenarios and proposing a new approach that significantly improved the robustness of trust and reputation models. Building on this, in [30], the authors introduced a deep reinforcement learning approach to evaluate the robustness of trust and reputation systems (TRSs), enabling the identification and execution of optimal attack plans without prior knowledge, effectively addressing the state space explosion problem inherent in traditional verification methods. Collectively, these studies provide a robust framework for evaluating and enhancing the resilience of reputation systems against malicious attackers. These evidences motivated us to propose a new method for handling the spam attacks.

3. Motivation and Research Foundations

Despite the many types of research that have been taken to identify spam comments, so far, no one has been addressing the issue of spam attacks. Today, smart spammers, with the knowledge of spam detection methods, can easily deceive the spam detection system and continue their malicious activity. For example, in text-based systems, spam attackers deceive the method by modifying the text of reviews. Deception can be performed in two manners: 1) deceptive behavior over time (in length deception) and 2) deceptive behavior over the product (in width deception). In deceptive behavior over time, as can be observed in Figure 2 (the quality of all products is 5), spam attackers exhibit honest behavior for a while, and after gaining enough trust, disclose their deceitful behavior. It means that a smart spammer has a conflicting behavior over time: honest behavior in the first period to increase his/her credentials and dishonest behavior in the next period to achieve his/her malicious goals using the gained social trust. In contrast to in-length deception, in deceptive behavior over the product (in width deception), the spam attacker exhibits conflicting behavior over different products. As can be seen in an instance of this attack in Figure 3, he sends fake reviews for the product that he wants to slander (here's product 2), while writing honest reviews on other products (here's products 2 and 3) to keep its social trust value. As mentioned earlier, detecting these types of attacks needs analyzing the complete knowledge of nodes behaviors, which are used by graph-based spam detection methods; however, the current graph-based techniques can almost be deceived via mentioned attacks.

3.1. Research Methodology

. The proposed methodology for detecting spam reviews consists of the following key steps:

Data Collection:

- Gather reviews, reviewer information, product details, and ratings from the simulation environment.

Trustworthiness Calculation:

- Compute trustworthiness scores for each entity (reviewers, products, and ratings) based on predefined criteria. For example:
  - Reviewer Trustworthiness: Derived from the reviewer's history, such as the number of reviews, consistency in ratings, and account age.
  - Product Trustworthiness: Based on the product's overall ratings, number of reviews, and consistency across reviews.
  - Rating Trustworthiness: Determined by the deviation of a rating from the average rating of the product.

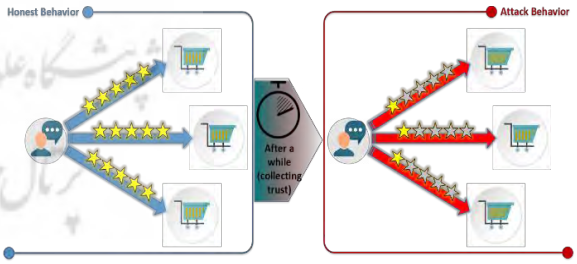

Figure. 2. User deceptive behavior over time

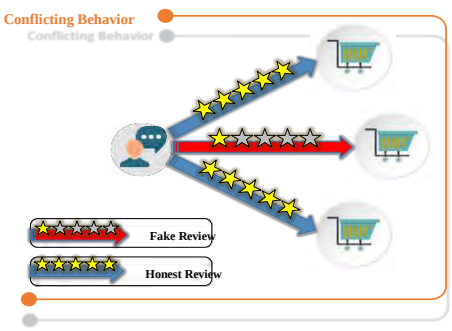

Figure. 3. User deceptive behavior in width

Spam Detection:

- Use the computed trustworthiness scores to identify potential spam reviews. For example:
  - Reviews from low-trustworthiness reviewers are flagged as suspicious.
  - Products with inconsistent ratings or a high number of low-trustworthiness reviews are analyzed further.
  - Ratings that significantly deviate from the product's average rating are marked as potential spam.

Validation:

- Create a simulation environment to validate the proposed algorithm.
- Simulate famous attack plans and validate the detected spam reviews

4. Simulator Software

Since none of the current spam data sets include the spam attacks, and the generation of enough human opinions and performing mentioned attacks are hard in practice, a simulation tool has been developed on the basis of the PDETool platform [31][32] to simulate the spam attack scenarios and evaluate the proposed method. The tool simulates the reviewing process in an e-commerce website and can generate enough samples for any given scenario. Moreover, it is capable of simulating any other desired scenario that may be required in evaluating spam detection methods. The tool defines the reviewing environment as a graph, which includes two types of node: 1) product nodes, and 2) reviewer one. Each product has a defined quality. A reviewer should be connected to a product using connectors to produce a review scenario.

The reviewer nodes have two subtypes: honest and spammer reviewers, which are represented by blue and red users in the tool's graphical user interface. The honest reviewers honestly score the products. To be more specific, their score has a normal distribution with the mean of product quality and a given variance. The variance default value is 0.5; however, it can be changed by the user. In contrast, the spammer behavior is defined using a provided script, which enables the modeler to define any complicated scenario, including spam attacks. Moreover, the software is also capable of defining individual spam behavior for each product. An instance of the defined mode in this tool is given in Figure 4. As shown in the figure, there are 3 reviewers and 3 products in this model. The model includes a spam reviewer (the 3rd reviewer) who falsely scores the last product (i.e., product no. 3) across the red connection.

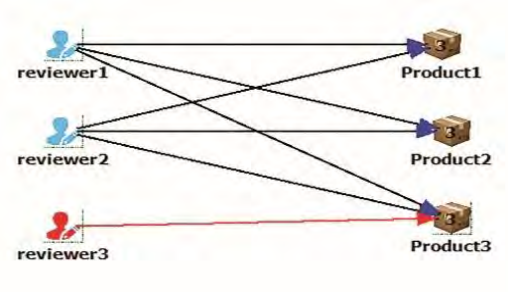

Figure. 4. A view of the simulator environment

5. Proposed Model for Detecting Spam Attacker

In this section, a method for spam and spam attack detection is proposed, which is robust against the mentioned deception scenarios in previous sections. The method is a graph-based model that is defined by three types of nodes: review, reviewer, and product. It estimates the value of trustworthiness, honesty, and reliability for these nodes respectively, which are demarcated in the following in detail. Symbols used in the equations are given in Table 1.

Reviewer Trustworthiness: The reviewer's trust score (denoted by *T(r)*) is the normal honest behavior performed by the user, as shown in Equation (1). It is estimated using the honesty mean of his published reviews and its sequence. The sequence helps the model to give more weight to recent reviews, which is essential for length spam attack detection. The trust of a reviewer (*T(r)*) is calculated through the Equation (1).

$$T(r) = \frac{\sum_{\forall review\ \epsilon\ r \cdot review} review \cdot number * H(r \cdot review)}{\sum_{\forall review\ \epsilon\ r \cdot review} review \cdot number} \quad (1)$$

The trust score *T(r)* quantifies the trustworthiness of a reviewer based on the honesty of their reviews. It rewards reviewers who consistently publish honest reviews and penalizes those who publish spam or dishonest reviews. The score is normalized by the total number of reviews, ensuring fairness and adaptability to evolving reviewer behavior.

Review Honesty: The review's honesty (denoted by H(v)) indicates the accuracy of the opinion, as show in Equation (2). The honesty value is estimated based on its maximum distance from the estimation of the true quality of the product (i.e., product reliability). *H(v)* is defined by the Equation (2).

$$H(v) = \frac{1 - |normalized(v.score) - R(v.product)|}{w}$$
$$R(v.product) > 0.5 \rightarrow W = R \quad (2)$$
$$R(v.product) < 0.5 \rightarrow W = 1 - R$$

The honesty score is a value between zero and one. The higher value indicates a more honest review.

The value of one means that the review is perfectly honest since it fully matches the product's reliability. It is not able that the review scores (i.e., v.score) should be normalized in the range [0, 1] before being used in the above equation (in the case of the systems that use 1-5 scores).

Product Reliability: The reliability score of the product (denoted as *R(p)*) is the estimation of the true quality of the product, as shown in Equation (3). The product reliability score is calculated by Equation (3). It depends on both the trust score of the reviewers and the review's honesty.

$$R(p) = \frac{\sum_{r \in p.reviewers} \sum_{v \in p.r.reviewers} T(r) * H(v) * v.score}{\sum_{r \in p.reviewers} \sum_{v \in p.r.reviewers} T(r) * H(v)} \quad (3)$$

The score is a value in the range of [0, 1]. As all trustworthiness, honesty, and reliability values are interdependent for estimating them, the mentioned equations should be computed in a loop until the result converges to a value. The algorithm output is independent of the initial values of the nodes (trustworthiness, honesty, and reliability). The proposed algorithm for this method can be seen in Figure 5.

The honesty score *H(v)* measures how closely a review's rating aligns with the true quality of the product, as estimated by the product's reliability score *R(p)*. It penalizes reviews that deviate significantly from the product's true quality, ensuring that dishonest or spam reviews are identified.

The weight *W* adjusts the score based on the product's reliability, making the metric contextually meaningful and robust. The reliability score R(p) estimates the true quality of a product by aggregating the contributions of trustworthy reviewers and honest reviews. It gives more weight to reviews from highly trustworthy reviewers, ensuring that the score reflects genuine opinions rather than spam. The iterative updating of *R(p)* ensures that the score remains accurate and adaptive to changes in reviewer behavior and new reviews.

It should be noticed that the only fixed value in the algorithm is 0.5 in Equation (2). A value above 0.5 is more likely to be positive, while a value below 0.5 is more likely to be zero. This serves as a simple threshold for distinguishing between positive and negative classes.

6. Deception Scenarios and Algorithms Implementation

In this section, the efficiency of the proposed method (ROSD) is presented using some spam attack scenarios. Moreover, the results are compared with other well-known graph-based approaches, including Wang's [19] and Fayazbakhsh's models [9]. In all scenarios, 1000 reviews are generated using the simulation tool, and the results of all three methods are presented and compared. Since the result values are [0, 1] for ROSD, [-1, +1] for WNG, and [0, 1] for FYZ, the benchmarking is done through the following defined measure: 1) the ability to detect spams and spammers, and 2) the number of deviations that the spam attacker can create in the actual value of the product's reliability. Whatever spam attacker cannot deflect, the reliability score of a product from its actual score indicates a better system performance. It is noteworthy to consider Fayazbakhsh's model that calculates only a suspicious score of reviewers and products while having no solution for calculating the suspicious score of reviews.

Table 1. Symbols used in proposed model

Definition	Notation
Review	V
Reviewer	R
Product	P
Score of review v	v.score
Product of review v	v.product
Review of reviewer r	r.review
Reviewer of product p	p.reviewer
Reviews of product p by reviewer r	p.r.reviews
The maximum difference between the score of reviewer and the majority of the community	W
Number of review	review.number

Input: *Set of reviewer (r), product (p), set of review (v).*
Output: *trust of reviewer T, honesty of review H, reliability of product R.*
Initialization step: *initialize score T, H, R to 1.*
Repeat until the result converge:
  for *r ∈ reviewer do*
    Compute *T(r) using (1)*
  end for
  for *v ∈ review do*
    Compute *H(v) using (2)*
  end for
  for *p ∈ product*
    Compute *R(p) using (3)*
  end for
end while

Figure. 5. The algorithm of the proposed model

Our work considers five specific deception scenarios, each designed to test the robustness of the proposed spam detection method against different types of spam attacks. These scenarios are incorporated into the graph modeling and calculations as follows. The first scenario involves simple spamming, where a spammer disparages a product without employing deceptive tactics, identifiable through low honesty and trust scores. In the second scenario, an over product attack occurs when a spammer undermines one product while maintaining a façade of honesty for others, detectable by assessing honesty scores across multiple products. The overtime attack scenario illustrates a gradual build-up of trust before a targeted slander, monitored through dynamic updates to honesty and trust scores. Additionally, selective product slandering utilizes real-world data to target specific products, with detection reliant on the analysis of honesty scores and subsequent trust score adjustments. Lastly, selective product promotion similarly leverages real-world data, with inconsistencies in honesty and trust scores serving as indicators of deceptive practices.

6.1. Simple Spamming Against a Product

In the first scenario, the spammer tries to slander a product without any deceptive behavior. As can be seen in Figure 6, there are 10 reviewers and three products. The last reviewer is a spammer who gives zero to product 3. In this scenario, the spam attacker wants to slander the product and does not use a deception scenario. It is important to note that the true quality of all products is considered to be 3 out of 5.

The results are shown in Table 2. As it is presented, ROSD and WNG can find a spammer, while FYZ is unable to detect the spammer as FYZ only tries to find spammers that send high scores (4 or 5 scores). Also in finding the spam reviews, both of the models have acceptable results. As can be seen in the last row of Table 2, in all three models, the spammer has not been able to change the reliability score of the target product significantly from the actual quality. Also, as shown in Table 3, if a spammer simply publishes a positive and fake opinion to promote a product in the same conditions and the spammer constantly gives the score of 5 out of 5 to the corresponding product, then the same results will be achieved. Note that the true quality of the corresponding product (product 3) is considered to be 1 out of 5.

6.2. An over Product Attack

In the second scenario, an overproduction attack is simulated. The simulation model, which is similar to previous ones, is represented in Figure 7 and Figure 8. However, the spammer is connected to all products. He gives the correct score to all products except the last one. As it is presented, only the proposed model can find spam attackers, and this suggests that other models are deceived in this way since they are unable to find the spam attacker. However, in finding the spam review, both models have acceptable results. As can be seen in the last row

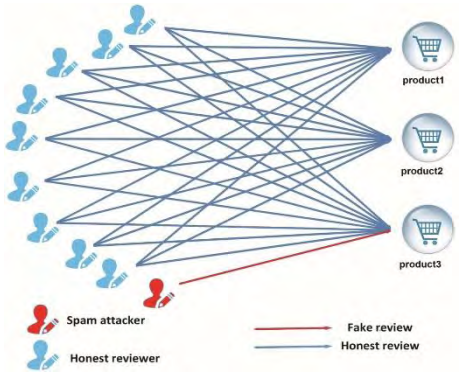

Figure. 6. Scenario 1: Simple spamming against a product

Table 2. Results table for the scenario1- slandering a product

Item	*ROSD*	*WNG*	*FYZ*
The average trustworthiness rating of honest reviewer	0.8667	1	0.9975
The average trustworthiness rating of spam attacker	0	-1	0.9916
The average honesty score of non-spam reviews	0.8662	0.9238	-
The average honesty score of spam reviews	0	-1	-
The average reliability of target products before spammer	0.6	1	0.9942
The average reliability of target products after spammer	0.6067	1	0.9953
Deviation value in product reliability	0.006	0	0.001

Table 3. Results table for the scenario1- promoting a product

Item	*ROSD*	*WNG*	*FYZ*
The average trustworthiness rating of honest reviewer	0.8789	1	0.9963
The average trustworthiness rating of spam attacker	0	-1	0.9967
The average honesty score of non-spam reviews	0.8759	0.8917	-
The average honesty score of spam reviews	0	-1	-
The average reliability of target products before spammer	0.2	-1	0.9941
The average reliability of target products after spammer	0.1915	-1	0.9954
Deviation value in product reliability	0.0085	0	0.001

of Table 4, in all three models, the spam attacker has not been able to change the reliability score of the target product critically from the actual quality. Also, in this scenario, if a spam attacker tries to promote a product and uses false positive scores instead of slandering, the same results have been achieved (here the true score of the selected product is assumed to be 1 out of 5), as indicated in Table 5.

6.3. An Overtime Attack

In this scenario, a slandering attack over time is simulated. As can be seen in Figure 8, there are 3 reviewers and 3 products. The last reviewer is a spam attacker who gives a true score of 3 to product 3 in the intervals of time (20 first reviews) and then gives the score 1 in the intervals of time (20-second reviews); this process continues until the end of the review generation. It should be noted that the true quality of all products is considered to be 3 out of 5.

As illustrated in Table 6, in this scenario results are similar to the earlier scenario; the proposed model may only detect spam attacks. The results for promoting attacks over time are the same, as shown in Table 7.

6.4. Selective Product Slandering with Real Data

In this scenario, an overproduction attack on real data is implemented. 20 spam data records have been artificially added to an existing spam data set to reach this goal [33]. This data set is the opinions collected from the movie Lenz website and includes 16 reviewers and 670 products. To emulate the deception scenario in this data, a spam attacker is added to the data that gives 0.5 (the lowest score in real data is 0.5) to some goal products and sends the correct score (similar to honest reviews) to other productsr The average score of the attacker's target products is about 3.75, so his reviews should definitely be identified as spam. The results are shown in Table.8. As it is presented, only the proposed model can find spam attackers, and in the case of finding the spam review, both models have acceptable results. However, as can be seen in the last row of Table 8, in the proposed model, the FYZ spam attacker has not been able to change greatly the reliability score of the target product from the actual quality.

6.5. Selective Product Promoting with Real Data

In this scenario, all the conditions are the same as in the previous scenario; however, to emulate the deception scenario in this data, a spam attacker is added to the data that gives 5 (the highest score in real data is 5) to some goal products and sends the correct score (similar to honest reviews) to other products. The average score of the attacker's target products is about 1.5, so his reviews should definitely be identified as spam. As before, only the proposed model performs well (Table 9).

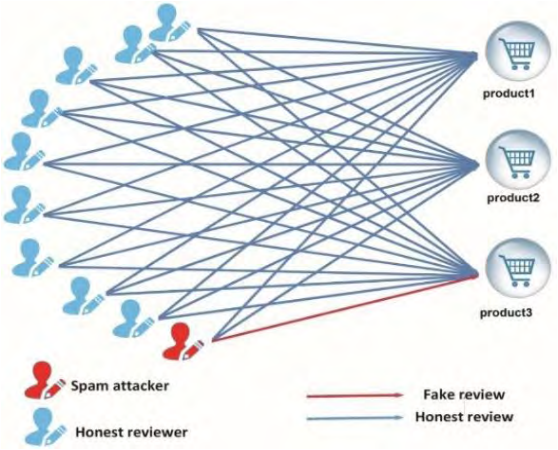

Figure. 7. Scenario 2: An over product attack

Table 4. Results table for the scenario 2- Selective product slandering

Item	*ROSD*	*WNG*	*FYZ*
The average trustworthiness rating of honest reviewer	0.8719	1	0.9960
The average trustworthiness rating of spam attacker	0.5730	1	0.9966
The average honesty score of non spam reviews	0.8684	0.9171	-
The average honesty score of spam reviews	0	-1	-
The average reliability of target products before spammer	0.6	1	0.9965
The average reliability of target products after spammer	0.6060	1	0.9952
Deviation value in product reliability	0.006	0	0.0013

Table 5. Results table for the scenario 2- Selective product promoting

Item	*ROSD*	*WNG*	*FYZ*
The average trustworthiness rating of honest reviewer	0.8704	1	0.9961
The average trustworthiness rating of spam attacker	0.5865	1	0.9962
The average honesty score of non-spam reviews	0.8726	0.8985	-
The average honesty score of spam reviews	0	-1	-
The average reliability of target products before spammer	0.2	-1	0.9935
The average reliability of target products after spammer	0.2016	-1	0.9952
Deviation value in product reliability	0.0016	0	0.002

Table 6. Results table for the scenario3- slandering attack over time

Item	*ROSD*	*WNG*	*FYZ*
The average trustworthiness rating of honest reviewers	0.8651	1	0.9814
The average trustworthiness rating of spam attacker	0.5285	1	0.9890
The average honesty score of non-spam reviews	0.9081	0.8694	-
The average honesty score of spam reviews	0.3486	-1	-
The average reliability of target products before spammer	0.6	1	1
The average reliability of target products after spammer	0.5736	1	0.9816
Deviation value in product reliability	0.0264	0	0.0184

Table 7. Results table for the scenario3- promoting attack over time

Item	*ROSD*	*WNG*	*FYZ*
The average trustworthiness rating of honest reviewers	0.8678	1	0.9900
The average trustworthiness rating of spam attacker	0.5570	1	0.9926
The average honesty score of non-spam reviews	0.9081	0.8694	-
The average honesty score of spam reviews	0.3486	-1	-
The average reliability of target products before spammer	0.6	1	1
The average reliability of target products after spammer	0.6181	1	0.9868
Deviation value in product reliability	0.0181	0	0.0132

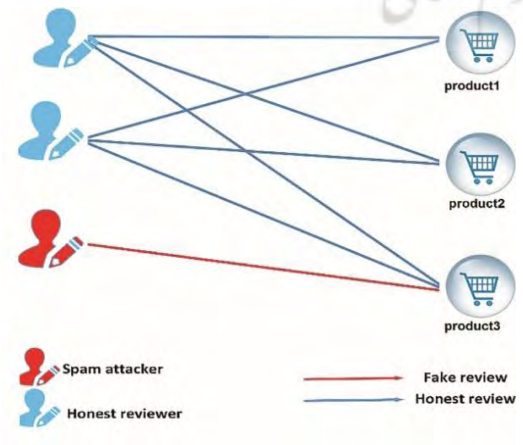

Figure. 8. Scenario 3: An overtime attack

Table 8. Results table for the scenario 4

Item	*ROSD*	*WNG*	*FYZ*
The average trustworthiness rating of honest reviewer	0.9103	0.9307	1
The average trustworthiness rating of spam attacker	0.5596	0.9682	1
The average honesty score of non-spam reviews	0.9140	0.3823	-
The average honesty score of spam reviews	0.1167	-0.1245	-
The average reliability of target products before spammer	0.9106	0.8230	1
The average reliability of target products after spammer	0.8604	0.2506	1
Deviation value in product reliability	0.0502	0.5724	0

Table 9. Results table for the scenario 5

Item	*ROSD*	*WNG*	*FYZ*
The average trustworthiness rating of honest reviewer	0.9119	0.9307	1
The average trustworthiness rating of spam attacker	0.5015	0.9886	1
The average honesty score of non-spam reviews	0.9154	0.3844	-
The average honesty score of spam reviews	0	-0.024	-
The average reliability of target products before spammer	0.2053	-0.6484	1
The average reliability of target products after spammer	0.2053	0.3760	1
Deviation value in product reliability	0	1.022	0

7. Conclusion

This study addresses the ongoing challenge of spam reviews in social networks and commercial platforms, highlighting the need for advanced detection methods capable of combating sophisticated manipulation strategies. We propose a multi-layer graph-based approach that enhances the detection of opinion spam through dynamic scoring mechanisms, improving the identification of unreliable reviews and preserving the integrity of user-generated content. The proposed method's efficiency is evaluated across various attack scenarios and compared with two established models. The results indicate that the proposed model effectively identifies spam attackers in all deception scenarios and shows significant improvement over the other models. It can detect spammers and reduce their trust, while also preventing attackers from undermining product reputation for malicious purposes. Despite these contributions, several avenues for future work

remain to be explored to further bolster the efficacy of spam detection systems. As future work, it could be useful to implement more deceptive scenarios on review spam detection models and resist the currently proposed model against other deceptive scenarios. Future work should explore the integration of machine learning algorithms to refine scoring based on real-time data, expand simulation tools to include diverse attack scenarios, and facilitate cross-platform analysis for a comprehensive spam prevention ecosystem. Overall, while our findings significantly enhance spam detection, continuous innovation and collaboration are essential to address the adaptive strategies of malicious actors.

Declarations

Funding

This research did not receive any grant from funding agencies in the public, commercial, or non-profit sectors.

Authors' contributions

Jalaly Bidgoly, A.: Conceptualization, Investigation, Software, Supervised the research.
Rahmanian, Z.: Writing – Original Draft Preparation, Designed the experiment, Methodology.
Dehghani, A.: Contributed to writing the paper and assisted in the implementation of the experiment.

Conflict of interest

The authors declare that no conflicts of interest exist.

References

[1] C. Esposito, V. Moscato, and G. Sperlì, "Detecting malicious reviews and users affecting social reviewing systems: A survey," *Comput. Secur.*, vol. 133, p. 103407, 2023.

[2] A. Mewada and R. K. Dewang, "A comprehensive survey of various methods in opinion spam detection," *Multimed. Tools Appl.*, vol. 82, no. 9, pp. 13199–13239, 2023, doi: 10.1007/s11042-022-13702-5.

[3] R. K. Dewang and A. K. Singh, "State-of-art approaches for review spammer detection: a survey," *J. Intell. Inf. Syst.*, vol. 50, no. 2, pp. 231–264, 2018, doi: 10.1007/s10844-017-0454-7.

[4] N. Jindal and B. Liu, "Opinion spam and analysis," in *Proceedings of the 2008 international conference on web search and data mining*, 2008, pp. 219–230.

[5] R. Narayan, J. K. Rout, and SI Ke Jena, "Review spam detection using opinion mining," in *Progress in Intelligent Computing Techniques: Theory, Practice, and Applications: Proceedings of ICACNI 2016, Volume 2*, 2018, pp. 273–279.

[6] A. Pak and P. Paroubek, "Twitter as a corpus for sentiment analysis and opinion mining.," in *LREc*, 2010, vol. 10, no. 2010, pp. 1320–1326.

[7] M. Diqi, "TwitterGAN: robust spam detection in twitter using novel generative adversarial networks," *Int. J. Inf. Technol.*, vol. 15, no. 6, pp. 3103–3111, 2023, doi: 10.1007/s41870-023-01352-1.

[8] G. Wang, S. Xie, B. Liu, and S. Y. Philip, "Review graph based online store review spammer detection," in *2011 IEEE 11th international conference on data mining*, 2011, pp. 1242–1247.

[9] S. Fayazbakhsh and J. Sinha, "Review spam detection: a network-based approach," *Final Proj. Rep. CSE*, vol. 590, 2012.

[10] D. Zhang, J. Xu, V. Zadorozhny, and J. Grant, "Fake news detection based on statement conflict," *J. Intell. Inf. Syst.*, vol. 59, no. 1, pp. 173–192, 2022, doi: 10.1007/s10844-021-00678-1.

[11] A. Heydari, M. ali Tavakoli, N. Salim, and Z. Heydari, "Detection of review spam: A survey," *Expert Syst. Appl.*, vol. 42, no. 7, pp. 3634–3642, 2015.

[12] M. S. Lakshmi, A. S. Rani, T. S. Divya, and J. Shravani, "Dynamic Spam Detection in Social Networks: Leveraging Convex Nonnegative Matrix Factorization for Enhanced Accuracy and Scalability," *Int. J. Comput. Eng. Res. Trends*, vol. 11, no. 4, pp. 1–11, 2024.

[13] C. G. Harris, "Detecting deceptive opinion spam using human computation," 2012.

[14] F. H. Li, M. Huang, Y. Yang, and X. Zhu, "Learning to identify review spam," 2011.

[15] E.-P. Lim, V.-A. Nguyen, N. Jindal, B. Liu, and H. W. Lauw, "Detecting product review spammers using rating behaviors," in *Proceedings of the 19th ACM international conference on Information and knowledge management*, 2010, pp. 939–948.

[16] D. Radovanović and B. Krstajić, "Review spam detection using machine learning," in *2018 23rd International Scientific-Professional Conference on Information Technology (IT)*, 2018, pp. 1–4.

[17] N. Jindal and B. Liu, "Analyzing and detecting review spam," in *Seventh IEEE international conference on data mining (ICDM 2007)*, 2007, pp. 547–552.

[18] N. Jindal, B. Liu, and E.-P. Lim, "Finding unusual review patterns using unexpected rules," in *Proceedings of the 19th ACM international conference on Information and knowledge management*, 2010, pp. 1549–1552.

[19] G. Wang, S. Xie, B. Liu, and P. S. Yu, "Identify online store review spammers via social review graph," *ACM Trans. Intell. Syst. Technol.*, vol. 3, no. 4, pp. 1–21, 2012.

[20] S. P. Algur, A. P. Patil, P. S. Hiremath, and S. Shivashankar, "Conceptual level similarity measure based review spam detection," in *2010 International conference on signal and image processing*, 2010, pp. 416–423.

[21] S. Banerjee and A. Y. K. Chua, "Applauses in hotel reviews: Genuine or deceptive?," in *2014 Science and Information Conference*, 2014, pp. 938–942.

[22] M. Ott, C. Cardie, and J. Hancock, "Estimating the prevalence of deception in online review communities," in *Proceedings of the 21st international conference on World Wide Web*, 2012, pp. 201–210.

[23] G. Fei, A. Mukherjee, B. Liu, M. Hsu, M. Castellanos, and R. Ghosh, "Exploiting burstiness in reviews for review spammer detection," in *Proceedings of the international AAAI conference on web and social media*, 2013, vol. 7, no. 1, pp. 175–184.

[24] S. Xie, G. Wang, S. Lin, and P. S. Yu, "Review spam detection via temporal pattern discovery," in *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2012, pp. 823–831.

[25] A. Heydari, M. Tavakoli, and N. Salim, "Detection of fake opinions using time series," *Expert Syst. Appl.*, vol. 58, pp. 83–92, 2016.

[26] S. Xie, G. Wang, S. Lin, and P. S. Yu, "Review spam detection via time series pattern discovery," in *Proceedings of the 21st International Conference on World Wide Web*, 2012, pp. 635–636.

[27] A. Mukherjee, B. Liu, J. Wang, N. Glance, and N. Jindal, "Detecting group review spam," in *Proceedings of the 20th international conference companion on World wide web*, 2011, pp. 93–94.

[28] Z. Wang, T. Hou, D. Song, Z. Li, and T. Kong, "Detecting review spammer groups via bipartite graph projection," *Comput. J.*, vol. 59, no. 6, pp. 861–874, 2016.

[29] A. J. Bidgoly, "Probabilistic analysis of trust based decision making in hostile environments," *Knowledge-Based Syst.*, vol. 211, p. 106521, 2021.

[30] A. J. Bidgoly and F. Arabi, "Robustness evaluation of trust and reputation systems using a deep reinforcement learning approach," *Comput. Oper. Res.*, vol. 156, p. 106250, 2023.

[31] A. Khalili, A. Jalaly Bidgoly, and M. Abdollahi Azgomi, "PDETool: A multi-formalism modeling tool for discrete-event systems based on SDES description," in *Applications and Theory of Petri Nets: 30th International Conference, PETRI NETS 2009, Paris, France, June 22-26, 2009. Proceedings 30*, 2009, pp. 343–352.

[32] A. Khalili, M. Abdollahi Azgomi, and A. Jalaly Bidgoly, "SimGine: A simulation engine for stochastic discrete-event systems based on SDES description," *Simulation*, vol. 89, no. 4, pp. 539–555, 2013.

[33] F. M. Harper and J. A. Konstan, "The movielens datasets: History and context," *Acm Trans. Interact. Intell. Syst.*, vol. 5, no. 4, pp. 1–19, 2015.

Amir Jalaly Bidgoly received the Ph.D. degree in computer engineering from Isfahan University, Isfahan, Iran, in 2015. He is currently an Associate Professor with the Department of Computer Engineering, University of Qom. His-research interests include deep learning and trustworthy machine learning.

Zoleikha Rahmanian received her M.S. degree in Information Technology Engineering from the University of Qom. Her research interest includes spam detection.

Abbas Dehghani is an Assistant Professor at Faculty of Electrical and Computer Engineering, Yasouj, Iran. He received his PhD in Computer Engineering from Iran University of Isfahan, Isfahan, Iran in 2015. He received his B.S. in Computer Engineering from Shiraz University, Iran, in 2002, and M.S. in Computer Architecture from Isfahan University, Iran, in 2007. His research interests include computer architecture, information security, and On-chip wireless communications.