A Privacy-Preserving Friend Recommendation Mechanism for Online Social Networks

Fukang Liu
School of Software &Microelectronics
Peking University, China
1801210584@pku.edu.cn

Guorui Wu
School of Software &Microelectronics
Peking University, China
wugr@ss.pku.edu.cn

Yang Liu
College of Computer Science and Technology
Harbin Institute of Technology, Shenzhen, China
Cyberspace Security Research Center
Peng Cheng Laboratory
liu.yang@hit.edu.cn

ABSTRACT

Friend recommendation systems is widely applied in the context of online social networks (OSNs). Such systems aim to expand users' social network to increase users' engagement with related OSNs. However, during the cold-start stage, in which situation no sufficient information could be used to provide recommendation to new users, social relationships among existing users might be disclosed. As existing users' social relationship might be used to provide recommendation for new users. In this paper, we solve such privacy problem by applying a privacy-preserving friend recommendation mechanism. The novelty of this mechanism lies in its combination of deep learning and differential privacy method**.** The balance of privacy preservation and social recommendation is achieved by introducing node2vec to generate users' latent features, then performing the information fusion with Heterogeneous Information Networks (HIN), and finally using deep neural network (DNN) which accords with differential privacy to make privacy-preserving recommendations. The mechanism is experimented on a real dataset Higgs Twitter Dataset. The result shows that our method can achieve certain balance to obtain good effect of recommendation without disclosing users' privacy.

CCS Concepts

• **Security and privacy → Social network security and privacy**
• **Human-centered computing → Social recommendation**

Keywords

Social recommendation; Differential privacy; Deep learning, Heterogeneous information network.

1. INTRODUCTION

A social network that is not large enough is not attractive to new users[1] as users would soon find out that their friend-lists are empty. Hence, the friend recommendation system, the aim of which is to expand users' social connections and increase their engagement, becomes an essential part of social networks.

Meanwhile, privacy concerns associated with recommendation systems are also increased [1-2]. Ming Sang *et al.*[22] investigates cybercrime on Instagram and propose solutions to the case.

Generally, there are two genres of practical privacy threats in social network:

(1) Utilizing auxiliary graphs to de-anonymize anonymized social graphs, for example, the attack on the anonymized Netflix Prize Dataset [3]. Some researches[1][4] are working to cope with such attacks now.

(2) Utilizing the faultiness of friend recommendation systems in the cold-start stage to crack users' social networks[5]. Typically, an attacker would create lots of fake accounts and establish fake connections between them. Then the accounts will be used to interact with a target user in order to dig out the user's friend lists. As the attackers' accounts are newly registered and the recommendation system is in the cold-start stage, the target user's friends might be recommended to the attacker. Thus, user's social graph would be disclosed [6-7]. This kind of threats is the main problem that we will focus on in this paper.

Existing approaches that try to preserve privacy in recommendation system can be classified into three categories: (1) cryptographic based techniques[4][8], (2) differential privacy based approaches[1][9], and (3) perturbation based techniques[2][10].

Specifically, typical cryptographic based techniques only encrypt users' data in order to protect data transmission. Hence, privacy leaks caused by the output of recommendation systems cannot be prevented by such techniques. While differential privacy-based approaches and perturbation-based techniques are rarely deployed in friend recommendation systems due to the relatively significant impact on the accuracy of recommendation[11].

In this paper, we propose a privacy-preserving friend recommendation mechanism based on deep-learning and differential privacy. The mechanism is novel in its combination of deep learning, differential privacy, and graph neural network. In particular:

(1) The proposed friend recommendation mechanism is based on user social network structure and user interaction data. Differential privacy is applied to prevent practical brute-force attacks.

ICCSP 2020, January 10–12, 2020, Nanjing, China.
ACM ISBN 978-1-4503-7744-7/20/01…$15.00.
DOI: https://doi.org/10.1145/3377644.3377648

[1] See https://signal.org/blog/private-contact-discovery/

(2) An auxiliary social graph generated from the user's original social graph is applied to reduce the effect of Laplace noise, which provides guarantee to the accuracy of recommendation.

(3) We formalize the social network into the form of Heterogeneous Information Networks (HIN), a graph model in which objects and edges are annotated with attributes. Users' features generated from "retweet network'', "mention network'', and "reply network'' are used in the training of recommendation model after integration, which helps to improve the quality and accuracy of recommendation.

We experimented on Higgs Twitter Dataset to evaluate the performance of our approach. The experiment results show that the original accuracy is 72%, and the accuracy after adding Laplace noise is 54%, which is acceptable.

2. PREVIOUS WORK

The main idea of the privacy-preserving techniques based on differential privacy [1][9] is to adopt Laplace mechanism or Exponential mechanism to achieve differential privacy. If the data conforms to differential privacy, then the privacy of users is preserved.

Vaidya *et al.*[12] utilizes the original user data to train models in order to provide recommendation of friend lists. And then they adopted the Laplace mechanism in the models to generate noise to protect users' privacy in real recommendation. Such research is a good start to apply differential privacy in the context of friend recommendation system.

However, some previous researches [13-14] show that although differential privacy has great influence on practice in data-mining applications, none of the proposed theoretical approaches for friend recommendations have been deployed in real recommendation system due to their unacceptable trade-offs between privacy and the accuracy of the recommendation.

In order to improve accuracy of the recommendation which adopted differential privacy, William *et al.*[15] utilize auxiliary graphs to select users to be recommended. Arthi *et al.*[20] and Sarpong *et al.*[21] proposed using privacy-preserving attributes matching protocol to make friend recommendation.

William *et al.*[15] utilize address books to generate candidate friend graph. The idea of such mechanism is to limit the scope of users' candidate friends via data interaction. Such mechanism is relatively effective due to the difficulty of getting the target user's response, as the interaction between target user and attackers is one-way.

Node2vec[16] is a good method to generate users' embedding from users' interaction data, and we generate candidate friends graph based on interaction data of users as well. Node2vec preserves higher order proximity between nodes by maximizing the probability of occurrence of subsequent nodes in fixed length random walks, which means that it can reflect the closeness between users.

Recently, deep neural networks (DNNs) have been successfully applied to graph-structured data. YouTube researchers used DNN in video recommendation [17]. We choose DNN to make friend recommendation as well.

3. METHODOLOGY

Attackers create fake accounts and interact with target users. The reflection in HIN shows that the edges between attackers and users are usually one-way and the weight of "attacker→user'' is greater than "user → attacker''. We first formalize the social network into HIN. Then users' behavior feature generated from "retweet network'', "mention network'', and "reply network'' are fused. In addition, Laplace noise are integrated into user's latent features to achieve differential privacy. Finally, Deep Neural Network (DNN) are applied to generate recommendation model with the fused latent feature.

3.1 Mechanism Overview

In this section, we will introduce our differential privacy-based recommendation mechanism. We formalize "retweet network", "mention network'', and "reply network'' into the form of HIN. After that, we utilize node2vec to generate user's embedding and concatenate users' embedding as training examples. And finally, we add Laplace noise in Uf's (see Figure 3) embedding for protection against practical brute-force attacks. The mechanism overview is as follow (see Figure 1).

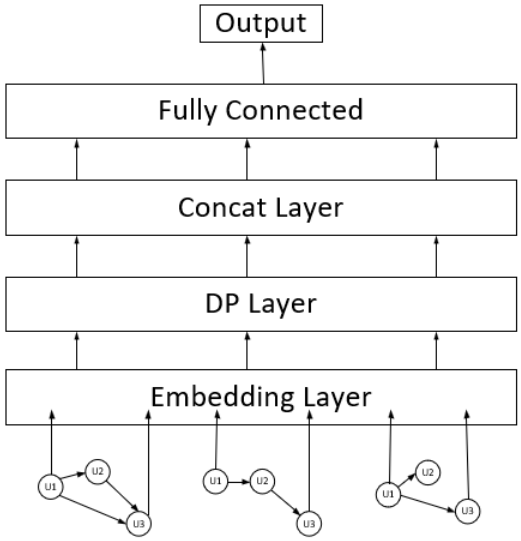


Figure 1. Mechanism overview.

3.2 Generating Users Embeddings

Node2vec[16] is a flexible biased random walk procedure that can explore neighborhoods in a BFS(Breadth-first Search) as well as DFS(Depth-first Search) fashion[18]. It is a 2^{nd} order random walk with two parameters p and q which guide the walk.

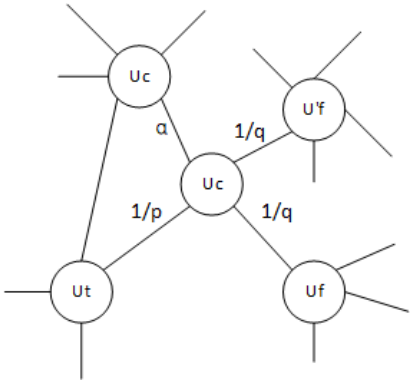


Figure 2. Node2vec parameters

Figure 2 illustrates how parameters p and q control the random walk traverses. Parameter p controls the likelihood of immediately revisiting nodes in the walk. Parameter q allows the search to differentiate between "inward" and "outward" nodes. If q>1, the

random walk is biased towards BFS. In contrast, the random walk is more inclined to DFS.

When generating training samples, we take users who are both one-hop friends and two-hop friends of Ut as positive samples, the triangle(Ut-Uc-Uc) in social network (see Figure 2). We take users who are just two-hop friends of Ut as negative samples. In experiment, we set parameter q with a high value, hence the random walk is biased towards nodes close to original node. The difference between positive and negative samples is more obvious. At the same time, since we limited the users to be recommend to the two-hop friends in the previous screening, so we also limited the distance of random walks.

3.3 Obfuscating Users' Embeddings

The definitions of HIN and HIN schema (a schema graph of entity types and their relations) have been introduced in [6]. Here we just focus on the concepts related to our paper. First, we need to introduce the meta-path in HIN for recommendation.

DEFINITION 3.1: Meta path, A meta-path P is a path defined on the graph of network schema $T_G = (A, R)$, and is denoted in the form of $A_1 \xrightarrow{R_1} A_2 \xrightarrow{R_2} \cdots \xrightarrow{R_l} A_{l+1}$. For simplicity, the meta-path can be denoted by the type names if there exist no multiple relations between the same pair of types: $P = (A_1 A_2 \cdots A_{l+1})$. A path $p = (a_1 a_2 \cdots a_{l+1})$ between a_1 and a_{l+1} is said to follow the meta-path P, if $\forall i, a_i \in A_i$ and each link $e_i = \langle a_i a_{i+1} \rangle$ belongs to each relation R_i in P. These paths are called path instances of P, denoted as $p \in P$.

As shown in Figure 3, we design similar paths according to the definition of meta-paths. Ut represents target user, Uc represents Ut's contacts friends, and Uf represents Uc's contacts friends. Retweet, Reply and Mention are interaction types. The weight of edges represents the number of interactions.

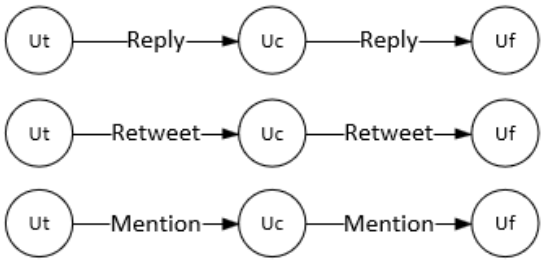


Figure 3. Paths between users

After generating embeddings of the users, we concatenate Ut and Uf's embedding according to the meta-path above as training examples for DNN. In the actual recommendations, Laplace noise is added to Uf's Embedding to satisfy the need of differential privacy, hence we can ensure Uc's social network from being leaked.

Definition 3.2 (ϵ-differential privacy) A randomized algorithm A is differentially private if and only if any two databases D and D' contain at most one different record. And for any possible anonymized output O（O∈Range（A））：$P_r[A(D) = O] \leq e^{\epsilon} \times P_r[A(D') = O]$, where the probability is taken over the randomness of algorithm A.

Theorem 1：Define $M(x) = f(x) + Laplace(0, \sigma)$, when $\sigma \geq \max_{A \approx B} \|f(A) - f(B)\| / \epsilon$, M satisfies ϵ-differential privacy.

The noise we add to Uf's embedding is generated by the function$P(n_j) \propto e^{-\frac{\epsilon |n_j|}{\Delta}}$, where Δ represents sensitivity. It equals Maximum minus minimum in embedding vector. We control the trade-off between privacy and utility by controlling parameter ϵ.

3.4 Recommendation Model

After generating the user's embedding through node2vec and passing through the concat layer, we got samples for training. Given the initialized training examples E_t, we update E_t^k to E_t^{k+1} by the convolution operation:

$$E_t^{k+1} = f_G(W E_t^k + b) \quad (1)$$

E_t^k represents the embedding output of the concat-layer. f_G is the non-linear activation function such as ReLU. W is the weight matrices to be learned, and b represents bias. Adam was used when training the model. After training, we can get original model with high precision. When we use the model to predict, we still use noise-added embedding. Hence users' privacy can be protected, and the recommendation accuracy can maintain acceptable by controlling noise parameters.

4. EVALUATION

In this chapter we will introduce the dataset first, and then introduce the evaluation index, experiment details and experiment results.

4.1 Dataset

Our paper proposes to formalize the social networks into the form of HIN, hence the dataset must be heterogeneous. And the dataset must include social network data and user interaction data. Finally, we choose Higgs Twitter Dataset[19] as the experiment dataset. Four directional networks made available here have been extracted from user activities in Twitter as friends-network, retweet-network, reply-network, and mention-network. The details of the dataset are shown in Table 1.

Table 1. Higgs Twitter Dataset

Network Information	Retweet network	Reply network	Mention network
Nodes	256491	38918	116408
Edges	328132	32523	150818
Number of triangles	21172	244	23068

4.2 Result

The loss function used in this paper is Binary cross entropy. Binary cross entropy is defined as follow.

$$\text{loss} = -\sum_{i=1}^{n} y_i \log \hat{y}_i + (1 - y_i) \log(1 - \hat{y}_i) \quad (2)$$

y_i in formula (2) represents original labels, and $\hat{y}_i$ represents labels predicted by the model.

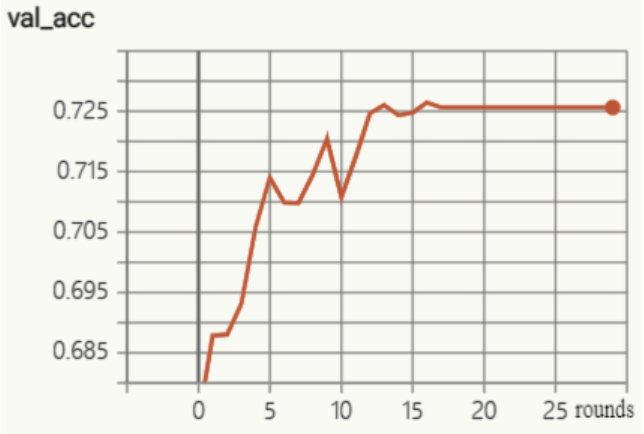
Users' embeddings are generated from retweet-network, reply-network, and mention-network. And then Embeddings are being concatenated to get training examples. Positive and negative samples are defined in section 2.2. The label of positive sample is 1 and the label of negative sample is 0. After data processing, we found that the proportion of samples was inconsistent. The ratio of

positive samples to negative samples was close to 1:39. This problem has been solved by using under sampling.

When we use node2vec to generate samples, we set the random walk distance to [3,5], set p to (1,4) and set parameter q to [10,100].

Differential privacy involves two parameters, sensitivity Δ and privacy parameter ϵ. Sensitivity Δ is an indicator of a function, and L1 sensitivity is defined as $\Delta = \max_{D_1,D_2} \|f(D_1) - f(D_2)\|_1$. D_1 and D_2 are datasets and f is a query function. We set the sensitivity Δ to the maximum of training examples minus the minimum. For the parameter ϵ, according to the definition of differential privacy, when it is close to 0, e^{ϵ} is close to 1, which means that the output probability of the two data sets is close to each other. At this point, user's privacy is well protected, but utility drops sharply. The bigger the ϵ, the worse the privacy protection, but the more accurate the query can be. In order to find suitable privacy parameters, we set ϵ to [0.5,1]. The results of our experiments on Higgs Twitter Dataset are as follows.

Figures below shows the recommendation accuracy in validation datasets, where the vertical axis represents accuracy and the horizontal axis represents the training rounds. The solid line is the smoothed accuracy curve and the dotted line is the original accuracy curve. In Figure 4, although accuracy and loss fluctuated in varying degrees, they eventually achieved about 72%.



(a)accuracy curve in validation dataset

Figure 4. Accuracy curve and loss curve of validation dataset

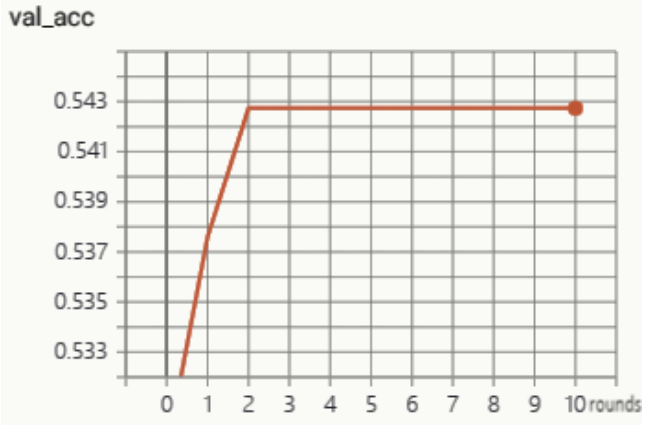


Figure 5. Accuracy curve in validation dataset with ϵ = 1 Laplace noise

Next, we try to add Laplace noise to the validation data and experiment again. By controlling the size of ϵ, we perturb the embedding of Uf to protect Uc's social graph. The experiment results are shown in Figure.5 when $\epsilon = 1$.

We found that accuracy decreased by 0.2, and remained at 0.54 after adding Laplace noise with $\epsilon = 1$. We continue to test the model when $\epsilon = 0.5$. The results are shown in Figure 6.

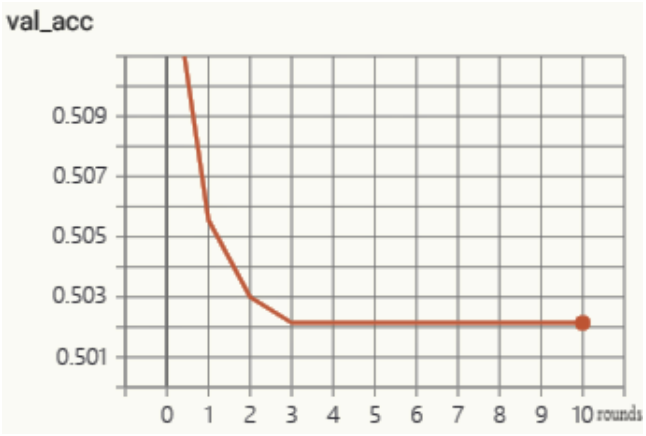


Figure 6. Accuracy curve in validation dataset when $\epsilon = 0.5$.

As the figure shows above, we can conclude that the reduction of ϵ in Laplace noise has a significant effect on the accuracy. The accuracy of the model prediction decreased from 0.7 to 0.5. Through comparative experiments, we can also conclude that the continuous reduction of ϵ has significant impact on the accuracy of the model. The accuracy of the model remains at the level of 0.5, which might be caused by a too small ϵ. We compared our program with baseline and found that the effect is slightly better than baseline.

Table 2. Methods comparison

Method	Our method	Linear regression	Logistic regression
Accuracy	0.543	0.512	0.511

We compare the accuracy degradation of other recommendation systems that adopted differential privacy, and find that the accuracy degradation in our approach is acceptable (see Table 2). For example, Zhu *et al.*[9], performed similar experiments. Although the decrease of ϵ in their experiments is not significant, the prediction accuracy decreases obviously with the decrease of ϵ, which means ours is acceptable.

5. CONCLUSIONS

In this paper, we propose a privacy-preserving mechanism for friend recommendation systems. The mechanism is novel in its combination of deep learning and differential privacy method. The balance of privacy and utility is achieved by introducing node2vec to generate user's latent features, performing the information fusion with HIN, and applying DNN with differential privacy to make privacy-preserving recommendations. Our mechanism guarantees that in cold start stage, attackers can no longer obtain accurate social graph information of target user by unilateral interaction. The experiment based on a real word dataset shows that by adjusting the parameter in Laplace noise, an acceptable recommendation model could be achieved with users' privacy guaranteed by differential privacy.

The privacy-preserving mechanism provides us with a good start to study the balance between privacy and utility in the context of recommendation systems. Our immediate focus of our future work is to refine the approach and expand the application to a wider domain, such as book or music recommendation systems.

6. ACKNOWLEDGMENTS

This work is partly supported by the National Key Research and Development Program of China (2017YFB0802204), Key Research and Development Program for Guangdong Province, China (2019B010136001), and Basic Research Project of Shenzhen, China (JCYJ20180507183624136).

7. REFERENCES

[1] Zhu, X. and Sun, Y. 2016. Differential Privacy for Collaborative Filtering Recommender Algorithm. *In Proceedings of the 2016 ACM on International Workshop on Security And Privacy Analytics* (IWSPA '16). ACM, New York, NY, USA, 9-16. DOI= https://doi.org/10.1145/2875475.2875483.

[2] Xin, Y. and Jaakkola, T. 2014. Controlling privacy in recommender systems. *In Advances in neural information processing systems*, 2618-2626.

[3] Narayanan, A. and Shmatikov, V. 2008. Robust de-anonymization of large datasets (how to break anonymity of the netflix prize dataset). *Computer Science*. The University of Texas at Austin.

[4] Tang, Q. and Wang, J. 2016. Privacy-preserving friendship-based recommender systems. *IEEE Transactions on Dependable and Secure Computing*, 15(5), 784-796.

[5] Mislove, A., Viswanath, B., Gummadi, K. P., and Druschel, P. 2010. You are who you know: inferring user profiles in online social networks. *In Proceedings of the third ACM international conference on Web search and data mining* (WSDM '10). ACM, New York, NY, USA, 251-260. DOI= http://dx.doi.org/10.1145/1718487.1718519.

[6] Sun, Y., Han, J., Yan, X., Yu, P. S., and Wu, T. 2011. Pathsim: Meta path-based top-k similarity search in heterogeneous information networks. *Proceedings of the VLDB Endowment*, 4(11), 992-1003.

[7] Backstrom, L., Dwork, C., and Kleinberg, J. 2007. Wherefore art thou r3579x?: anonymized social networks, hidden patterns, and structural steganography. *In Proceedings of the 16th international conference on World Wide Web* (WWW '07). ACM, New York, NY, USA, 181-190. DOI= https://doi.org/10.1145/1242572.1242598.

[8] Hoens, T. R., Blanton, M., and Chawla, N. V. 2010. A private and reliable recommendation system for social networks. *In 2010 IEEE Second International Conference on Social Computing*, 816-825. IEEE.

[9] Zhu, T., Li, G., Ren, Y., Zhou, W., and Xiong, P. 2013. Differential privacy for neighborhood-based collaborative filtering. *In Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining* (ASONAM '13). ACM, New York, NY, USA, 752-759. DOI= https://doi.org/10.1145/2492517.2492519.

[10] Rebollo-Monedero, D., Parra-Arnau, J., and Forné, J. 2011. An information-theoretic privacy criterion for query forgery in information retrieval. *In International Conference on Security Technology*, 146-154. Springer, Berlin, Heidelberg.

[11] Beigi, G. and Liu, H. 2018. Privacy in social media: Identification, mitigation and applications. arXiv preprint arXiv:1808.02191. DOI= https://arxiv.org/abs/1808.02191.

[12] Vaidya, J., Shafiq, B., Basu, A., and Hong, Y. 2013. Differentially Private Naive Bayes Classification. *In Proceedings of the 2013 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies* (IAT) - Volume 01 (WI-IAT '13), Vol. 1. IEEE Computer Society, Washington, DC, USA, 571-576. DOI= http://dx.doi.org/10.1109/WI-IAT.2013.80.

[13] Dwork, C. 2011. A firm foundation for private data analysis. *Communications of the ACM*, New York, NY, USA, 54(1), 86-95. DOI: https://doi.org/10.1145/1866739.1866758.

[14] Machanavajjhala, A., Korolova, A., and Sarma, A. D. 2011. Personalized social recommendations: accurate or private. *Proceedings of the VLDB Endowment*, 4(7), 440-450.

[15] Brendel, W., Han, F., Marujo, L., Jie, L., and Korolova, A. 2018. Practical Privacy-Preserving Friend Recommendations on Social Networks. *In Companion Proceedings of the The Web Conference 2018* (111-112). International World Wide Web Conferences Steering Committee.

[16] Grover, A. and Leskovec, J. 2016. node2vec: Scalable Feature Learning for Networks. *In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (KDD '16). ACM, New York, NY, USA, 855-864. DOI= https://doi.org/10.1145/2939672.2939754.

[17] Covington, P., Adams, J., and Sargin, E. 2016. Deep Neural Networks for YouTube Recommendations. *In Proceedings of the 10th ACM Conference on Recommender Systems* (RecSys '16). ACM, New York, NY, USA, 191-198. DOI= https://doi.org/10.1145/2959100.2959190.

[18] Perozzi, B., Al-Rfou, R., and Skiena, S. 2014. DeepWalk: online learning of social representations. *In Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining* (KDD '14). ACM, New York, NY, USA, 701-710. DOI= https://doi.org/10.1145/2623330.2623732.

[19] De Domenico, M., Lima, A., Mougel, P., and Musolesi, M. 2013. The anatomy of a scientific rumor. *Scientific reports*, 3, 2980. DOI= https://doi.org/10.1038/srep02980.

[20] Arthi, K. and Reddy, M.C., 2017. "A Secure and Efficient Privacy-Preserving Attribute Matchmaking Protocol for Mobile Social Networks," *International Journal of Network Security*, Vol. 19, No. 3, 2017, 421-429.

[21] Solomon, S., Xu, C., and Zhang, X. 2016. "PPAM: Privacy-preserving Attributes Matchmaking Protocol for Mobile Social Networks Secure against Malicious Users," *International Journal of Network Security*, Vol. 18, No. 4, 2016, 625-632.

[22] Chang, M. and Yen, C. 2019. "Forensic Analysis of Social Networks Based on Instagram," *International Journal of Network Security*, Vol. 21, No. 5, 2019, 850-860.