



DSP: Deep Sign Prediction in Signed Social Networks

Wei Yang, Yitong Wang^(✉), and Xinshu Li

School of Computer Science, Fudan University, Shanghai, China
{17210240021, yitongw, xslil8}@fudan.edu.cn

Abstract. In a signed social network, users can express emotional tendencies such as: like/dislike, friend/foe, support/oppose, trust/distrust to others. Sign prediction, which aims to predict the sign of an edge, is an important task of signed social networks. In this paper, we attempt to tackle the problem of sign prediction by proposing a Deep Sign Prediction (DSP) method, which uses deep learning technology to capture the structure information of the signed social networks. DSP considers the “triangle” structures each edge involves comprehensively, and takes both the “balance” theory and the “status” theory into account. We conduct experiments and evaluations on five real signed social networks and compare the proposed DSP method with multiple state-of-the-art methods. The experimental results show that the proposed DSP method is very effective and outperforms other methods in terms of four metrics (AUC, binary-F1, micro-F1, macro-F1).

Keywords: Sign prediction · Balance theory · Status theory · Triangle structure

1 Introduction

With the emergence of online social networks, individuals show more interests in participating in social intercourse on the Internet. Numerous studies focus on unsigned social networks while only a few of them have studied the signed ones. Due to the existence of negative edges, many effective unsigned social network analysis methods cannot be applied to signed social networks directly [4–6]. Signed social networks are usually based on balance theory [1, 2] and status theory [3], both of which are proposed by observing social phenomena. At the same time, with the rapid development of deep learning technology, many scholars have begun to adopt the idea of network embedding to solve the problem of sign prediction. Although these methods have achieved good sign prediction performance, they have some drawbacks. Firstly, most methods basing on balance theory [9, 11] obtain the node embedding by optimizing single balance triangle each edge involves during each round of training process. These methods can have some limitations: 1. The interactions between multiple triangles each edge involves are ignored. 2. Most methods only consider triangles that satisfy the balance theory, and ignore those violating the theory, which will lead to the key information loss. 3. There is no common neighbor between the two endpoints of an edge, which is called a “bridge” edge in the paper [9]. Thus, how to properly model this

type of edges is also an essential aspect. Secondly, for some methods based on skip-gram models [12, 13], they use shallow models to train node embedding and cannot capture the non-linear structure information of the network well. Thirdly, a sign prediction method should fuse balance theory and status theory reasonably to obtain optimal sign prediction performance [9], but there are few methods considering both theories together. Last but not least, as stated in the paper [7], it is necessary to specifically design a solution framework for sign prediction problem.

The main contributions of this paper are:

- This paper proposes an end-to-end framework: Deep Sign Prediction (DSP) method, which uses the deep learning technology and optimizes a loss function specifically designed for sign prediction.
- This paper extends the balance theory by considering all possible “triangle” structures each edge involves and solves the drawbacks of former methods which only model single balanced triangle each edge involves.
- In this paper, a two-layer neural network architecture is designed to combine the balance theory with status theory reasonably.
- We conduct experiments and evaluations on five real signed social networks and compare the proposed DSP method with multiple state-of-the-art methods. Experimental results show that our method outperforms other methods in terms of four metrics (AUC, binary-F1, micro-F1, macro-F1).

2 Related Work

Signed Network Embedding: Signed network embedding methods can be roughly divided into following four categories. The first category does not consider any sociological theory. In [10], Yuan et al. propose a SNE method, which uses a log-bilinear model to train the target embedding of nodes along a given path. The second category makes use of balance theory. For example, SiNE [11] method is based on the extended structural balance theory. In SIGNet [12], the authors propose a scalable signed network embedding method. In [13], Kim et al. propose the SIDE method, which fully encodes the direction and sign of the edges in the learned embedding space. The third category is based on status theory. In SSNE [14], the authors design an energy-based ranking loss function based on status theory. The last type considers both two theories. For example, the BESIDE method in [9].

Sign Prediction: There are many methods solving sign prediction from different perspectives, apart from signed network embedding methods mentioned above. For example, in [8], Leskovec et al. manually extract degree features of the nodes and triad features of edges to train a logistic regression classifier for sign prediction. In [15], Javari et al. design a probability prediction model based on the local and global structure of networks in order to deal with the sparsity problem of signed social networks.

3 Preliminary

Signed Social Network: A signed social network can be modeled as a directed signed graph $G = (V, E, E^+, E^-, S)$, where V, E, E^+, E^- represent the sets of all nodes, signed edges, positive edges, and negative edges in the network; S is a signed adjacency matrix, and each entry S_{ij} of S represents the relationship from node i to j (Specifically, $S_{ij} = 1, -1, 0$ indicates positive, negative, no relationship in the current network).

Sign Prediction: Given a signed social network G , sign prediction aims to predict the sign of edges that are not observed in current network.

4 The Proposed Method

4.1 Model “Triangle” Structures Each Edge Involves

In a signed social network, there are multiple situations for the interaction between any directed edge e_{ij} and any node k . Nodes i, j , and k can form a triangle, which may not necessarily conform to balance theory, or they cannot even form a triangle. And, e_{ij} may be involved in multiple “triangles” at the same time. As shown in Fig. 1, without considering the edge sign, edge e_{ij} and node k can form four possible “triangle” structures (dashed line indicates that edge may not exist). Each “triangle” type corresponds to interactions between the neighbor structure of node i and that of the node j in directed edge e_{ij} . For example, the first type of “triangle” corresponds to the interaction between the out-neighbor structure of node i and the in-neighbor structure of node j .

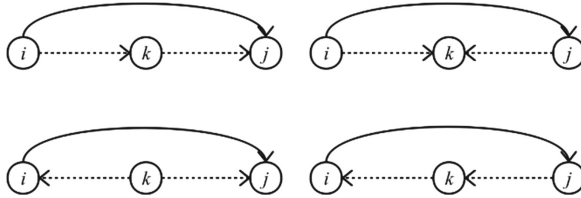


Fig. 1. Four possible “triangle” structures each edge involves.

According to Fig. 1, we generate a “balance” neighbor structure vector for two endpoints of the directed edge e_{ij} :

$$b_{st_i} = [S_i^{out}; S_i^{out}; S_i^{in}; S_i^{in}], b_{en_j} = [S_j^{in}; S_j^{out}; S_j^{in}; S_j^{out}] \quad (1)$$

in formula (1): $b_{st_i}, b_{en_j} \in R^{1 \times 4|V|}$; $S_i^{out} \in R^{1 \times |V|}$ is the i th row of the matrix S , and $S_j^{in} \in R^{1 \times |V|}$ is the j th column of the matrix S .

By constructing b_{st_i} and b_{en_j} for each edge e_{ij} , and training b_{st_i} and b_{en_j} simultaneously, our method can consider the four “triangle” types in Fig. 1. This solution extends the balance theory and considers all possible “triangle” structures each edge involves comprehensively. Next we want to find a function f , which takes b_{st_i} or b_{en_j} as input, and output the “balance” embedding b_{em_i} or b_{em_j} . Namely,

$$b_{em_i} = f(b_{st_i}), b_{em_j} = f(b_{en_j}) \quad (2)$$

in formula (2): $b_{em_i} \in R^{1 \times d}$ and d is the dimension of “balance” embedding.

4.2 Modeling Directed Edges by Status Theory

For a user in signed social networks, her status is determined by two parts: her “subjective” status/self-evaluation, and her “objective” status, which is evaluated by others. The “subjective”/“objective” status can be reflected by the user’s out-neighbor/in-neighbor structure. Then for a node i , we want to find a function $g(\cdot, \cdot)$ which inputs the S_i^{out} and S_i^{in} and outputs a “status” neighbor structure vector: s_{ne_i} .

$$s_{ne_i} = g(S_i^{out}, S_i^{in}) \quad (3)$$

we use the vector addition to define the function g . In formula (3): $s_{ne_i} \in R^{1 \times |V|}$.

After obtaining the “status” neighbor vectors of node i, j , we will learn two functions: st_h and en_h to obtain the “status” embedding: s_{em_i}, s_{em_j} respectively.

$$s_{em_i} = st_h(s_{ne_i}), s_{em_j} = en_h(s_{ne_j}) \quad (4)$$

in formula (4): $s_{em_i}, s_{em_j} \in R^{1 \times d}$, d is the dimension of the “status” embedding.

Based on status theory, we define a status loss function L_{st} , as:

$$L_{st_{ij}} = \max(0, \delta - (Status_i - Status_j) \times (-S_{ij})) \quad (5)$$

in formula (5): $Status_i \in (-1, 1)$ is the status value of node i , which is generated by the non-linear mapping function sta_h ;

$$Status_i = sta_h(s_{em_i}) \quad (6)$$

S_{ij} is the sign of e_{ij} ; δ is the threshold of the difference between the two status values, and we set it to 0.5 according to the previous experimental research.

4.3 Deep Sign Prediction Model

Our network architecture is divided into two parts: the first part is used to extend balance theory, and the second part is used to consider the status theory. The input of DSP is an edge. The detailed network architecture of the DSP method is shown in Fig. 2.

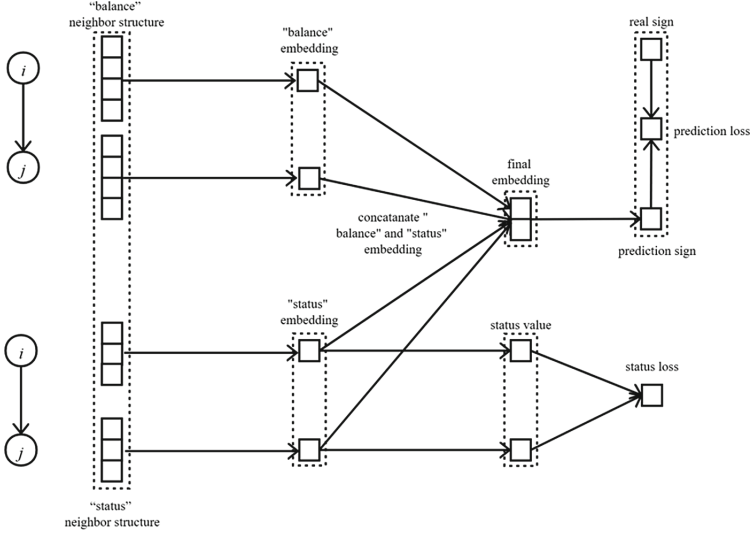


Fig. 2. Network architecture of the DSP method

First, we use a two-layer fully connected neural network to define function f .

$$b_em_i = \tanh(\tanh(b_st_i W^0 + b^0) W^1 + b^1) \quad (7)$$

in formula (7): \tanh is a non-linear activation function; $W^0 \in R^{4|V| \times 2d}$ and $W^1 \in R^{2d \times d}$ are the weight parameters; $b^0 \in R^{1 \times 2d}$ and $b^1 \in R^{1 \times d}$ are the bias parameters.

For the functions st_h and en_h , they are defined by using a layer of fully connected neural network, respectively.

$$s_em_i = \tanh(s_ne_i W^2 + b^2), \quad s_em_j = \tanh(s_ne_j W^3 + b^3) \quad (8)$$

in formula (8): $W^2, W^3 \in R^{|V| \times d}$; $b^2, b^3 \in R^{1 \times d}$.

We also use a layer of fully connected neural network to define the function sta_h .

$$Status_i = \tanh(s_em_i W^4 + b^4) \quad (9)$$

in formula (9): $W^4 \in R^{d \times 1}$; $b^4 \in R^{1 \times 1}$.

Then, we concatenate “balance” embedding and “status” embedding of two endpoints of each edge as the final feature representation.

$$final_{ij} = [b_em_i; b_em_j; s_em_i; s_em_j] \quad (10)$$

in formula (10): $final_{ij} \in R^{1 \times 4d}$ is the final embedding vector.

For the prediction layer, we use a three-layer fully connected neural network to generate the edge’s prediction value.

$$p_{ij} = \text{softmax}(\text{ReLU}(\text{ReLU}(final_{ij}W^5 + b^5)W^6 + b^6)W^7 + b^7) \quad (11)$$

in the formula (11): ReLU and softmax are two non-linear activation functions; $W^5 \in R^{4d \times d}$, $W^6 \in R^{d \times \frac{d}{2}}$, and $W^7 \in R^{\frac{d}{2} \times 2}$; $b^5 \in R^{1 \times d}$, $b^6 \in R^{1 \times \frac{d}{2}}$, and $b^7 \in R^{1 \times 2}$. We use the cross-entropy loss to define the loss of sign prediction.

$$L_{sign_{ij}} = - \sum_m y_{ijm} \log p_{ijm} \quad (12)$$

in the formula (12): $m \in \{0, 1\}$ is the subscript. y denotes the one-hot encoding vector of the edge sign (negative and positive); p defines the predicted probability for each type of sign (negative and positive). For the DSP method, the overall loss function is:

$$L = \frac{1}{|E|} \sum_{e_{ij} \in E} (L_{sign_{ij}} + L_{st_{ij}}) \quad (13)$$

The input scale of the DSP method is $O(|E|)$. We use the Adam [17] algorithm to optimize the DSP model. The learning rate is 0.0001, and the batch size is 128.

5 Experiments

5.1 Datasets

We conduct experiments and evaluations on five real online signed social networks. The specific statistical information of five datasets is shown in Table 1.

Table 1. The statistical information of five datasets

Dataset	Node	Edge	Positive (%)	Negative (%)
Alpha	3783	24186	93.65%	6.35%
OTC	5881	35592	89.99%	10.01%
RfA	7118	107080	78.41%	21.59%
Slashdot	82140	549202	77.40%	22.60%
Epinions	131828	641372	85.30%	14.70%

5.2 Baseline Methods

We compare the proposed DSP method with several state-of-the-art sign prediction methods: two feature engineering methods (All23 [8] and FxG [16]), two unsigned network embedding methods (DW [4] and N2V [6]), three signed network embedding methods (SIGNet [12], SIDE [13], and BESIDE [9]), and the part of extends balance theory in DSP model (DSP_B). For the above methods, we use the same parameters setting recommended by the original papers. For the unsigned network embedding methods, we ignore edge sign during training process. For the node embedding

methods, we concatenate the node embedding to obtain the edge embedding, and then train a logistic regression model for sign prediction. Our method uses the DSP framework for sign prediction and sets the dimension of the embedding vector $d = 40$.

5.3 Sign Prediction

As in previous studies [9], we use AUC, binary-F1, micro-F1, and macro-F1 to evaluate the performance of sign prediction. We randomly divide the datasets into a test set and a training set with a ratio of 2–8. The experimental results are shown in Table 2.

From Table 2, we can see that the DSP method has obtained the best experimental results in most cases. It is better than All23 and FxG, which shows the powerful learning ability of deep model. The performance of two unsigned network embedding methods is relatively poor, which means the unsigned network embedding methods cannot adapt to sign prediction. By comparing DSP with SIGNet and SIDE, we find that using deep models and considering both balance theory and status theory together can achieve better sign prediction performance. DSP is superior to BESIDE, indicating that comprehensive consideration of all possible “triangle” structures each edge involves can capture the latent features related to sign prediction well. The results also indicate that solution framework specifically designed for sign prediction problem is crucial to achieve better sign prediction performance by comparing DSP with other methods of signed network embedding. In most cases, the DSP method is better than DSP_B, which shows that combining balance theory with status theory reasonably can achieve the best sign prediction performance.

Table 2. The result of sign prediction

Dataset	Metric	All23	FxG	DW	N2V	SIGNet	SIDE	BESIDE	DSP_B	DSP
Alpha	AUC	0.8878	0.8793	0.8460	0.8451	0.8678	0.8787	0.8833	0.9143	0.9200
	binary-F1	0.9718	0.9452	0.9681	0.9689	0.9651	0.9682	0.9695	0.9741	0.9742
	micro-F1	0.9464	0.8988	0.9388	0.9403	0.9342	0.9397	0.9422	0.9511	0.9513
	macro-F1	0.7151	0.6411	0.6051	0.6256	0.6963	0.6945	0.7186	0.7808	0.7837
OTC	AUC	0.9109	0.8919	0.8649	0.8664	0.8793	0.8854	0.9069	0.9320	0.9350
	binary-F1	0.9644	0.9308	0.9591	0.9588	0.9550	0.9569	0.9628	0.9714	0.9714
	micro-F1	0.9344	0.8769	0.9240	0.9235	0.9180	0.9210	0.9322	0.9481	0.9481
	macro-F1	0.7734	0.6877	0.7117	0.7112	0.7461	0.7444	0.7912	0.8447	0.8450
RfA	AUC	0.8718	0.8925	0.8080	0.8091	0.9038	0.8369	0.9072	0.9198	0.9210
	binary-F1	0.9047	0.9082	0.8935	0.8894	0.9162	0.8955	0.9193	0.9238	0.9242
	micro-F1	0.8442	0.8495	0.8205	0.8133	0.8657	0.8257	0.8701	0.8778	0.8787
	macro-F1	0.7388	0.7450	0.6614	0.6452	0.7889	0.6851	0.7940	0.8111	0.8125
Slashdot	AUC	0.8873	0.8141	0.8049	0.8045	0.8852	0.8466	0.8903	0.9258	0.9271
	binary-F1	0.9063	0.8617	0.8780	0.8776	0.9030	0.8919	0.9131	0.9266	0.9262
	micro-F1	0.8462	0.7803	0.7965	0.7959	0.8472	0.8265	0.8617	0.8847	0.8843
	macro-F1	0.7394	0.6641	0.6325	0.6320	0.7718	0.7264	0.7878	0.8289	0.8289
Epinion	AUC	0.9433	0.9240	0.8673	0.8682	0.9214	0.9197	0.9391	0.9643	0.9664
	binary-F1	0.9555	0.9441	0.9400	0.9399	0.9556	0.9579	0.9660	0.9712	0.9712
	micro-F1	0.9213	0.9032	0.8927	0.8926	0.9231	0.9262	0.9411	0.9501	0.9501
	macro-F1	0.8060	0.7906	0.7177	0.7183	0.8337	0.8300	0.8734	0.8925	0.8924

We compare the prediction performance of each algorithm on a test set that contains only “bridge” edges. The experimental results on “bridge” edges are shown in Table 3. From Table 3, we can see that BESIDE achieves the best performance on all baseline methods, which means that BESIDE method with status theory is quite useful for modeling “bridge” edges. The proposed DSP and DSP_B methods are both superior to BESIDE, showing that considering the neighbor structures of two endpoints of each edge comprehensively, the “bridge” edges can be trained and predicted well.

Table 3. The performance of sign prediction on the “bridge” edges.

Dataset	Metric	All23	FxG	DW	N2V	SIGNet	SIDE	BESIDE	DSP_B	DSP
Alpha	AUC	<u>0.8321</u>	0.8093	0.7769	0.7840	0.8021	0.8313	0.8208	0.8696	0.8742
	binary-F1	0.9804	0.9014	0.9796	<u>0.9809</u>	0.9642	0.9716	0.9798	0.9817	0.9810
	micro-F1	0.9618	0.8237	0.9602	<u>0.9628</u>	0.9323	0.9454	0.9608	0.9648	0.9634
	macro-F1	0.6313	0.5356	0.5915	0.5942	0.6583	0.6370	<u>0.7160</u>	0.7535	0.7435
OTC	AUC	0.8442	0.8065	0.7948	0.8003	0.8121	0.8120	<u>0.8584</u>	0.8755	0.8814
	binary-F1	0.9645	0.8820	0.9667	0.9681	0.9540	0.9580	<u>0.9683</u>	0.9708	0.9708
	micro-F1	0.9324	0.7956	0.9363	0.9390	0.9147	0.9213	<u>0.9407</u>	0.9457	0.9455
	macro-F1	0.6283	0.5585	0.6208	0.6299	0.6802	0.6635	<u>0.7651</u>	0.7977	0.7958
RfA	AUC	0.8064	0.8867	0.7899	0.7696	0.8527	0.8027	0.8590	0.8853	0.8860
	binary-F1	0.8184	0.8536	0.8174	0.8035	0.8426	0.8136	<u>0.8595</u>	0.8749	0.8750
	micro-F1	0.7238	<u>0.8057</u>	0.7390	0.7129	0.7885	0.7401	<u>0.8057</u>	0.8277	0.8279
	macro-F1	0.6210	<u>0.7823</u>	0.6801	0.6355	0.7600	0.6920	<u>0.7722</u>	0.7994	0.8001
Slashdot	AUC	0.8615	0.7951	0.7844	0.7846	0.8674	0.8277	<u>0.8724</u>	0.9085	0.9094
	binary-F1	0.8930	0.8492	0.8783	0.8734	0.8939	0.8837	<u>0.9086</u>	0.9178	0.9172
	micro-F1	0.8218	0.7654	0.7890	0.7881	0.8337	0.8146	<u>0.8547</u>	0.8709	0.8705
	macro-F1	0.6805	0.6610	0.6153	0.6122	0.7550	0.7138	<u>0.7778</u>	0.8098	0.8100
Epinion	AUC	0.8575	0.8573	0.7918	0.7948	0.8409	<u>0.8709</u>	0.8609	0.9039	0.9066
	binary-F1	0.8962	0.8694	0.8832	0.8837	0.9036	0.9142	<u>0.9306</u>	0.9395	0.9389
	micro-F1	0.8211	0.8064	0.8090	0.8090	0.8463	0.8639	<u>0.8873</u>	0.9025	0.9017
	macro-F1	0.6244	0.7476	0.6800	0.6752	0.7622	0.7931	<u>0.8159</u>	0.8443	0.8440

6 Conclusion and Future Work

This paper presents a DSP method specifically for solving sign prediction task. DSP makes use of the powerful learning ability of deep learning to capture the complex structure of signed social networks. At the same time, the DSP method extends the balance theory, comprehensively considers all possible “triangle” structures each edge involves. Finally, the DSP method reasonably combines balance theory with status theory. We perform two types of comparative experiments on the five real signed social network datasets. The experimental results on four commonly used evaluation metrics show the superiority of our proposed methods.

Although DSP achieves excellent sign prediction performance, there are still some directions that can be further explored. For example, in the future, we will explore more ways of combining balance theory with status theory. Moreover, we will explore the attribute information of nodes in the next research work.

References

1. Heider, F.: Attitudes and cognitive organization. *J. Psychol.* **21**(1), 107–112 (1946)
2. Cartwright, D., Harary, F.: Structural balance: a generalization of Heider's theory. *Psychol. Rev.* **63**(5), 277–293 (1956)
3. Leskovec, J., Huttenlocher, D., Kleinberg, J.: Signed networks in social media. In: *Proceedings of SIGCHI*, pp. 1361–1370. ACM (2010)
4. Perozzi, B., Al-Rfou, R., Skiena, S.: Deepwalk: online learning of social representations. In: *Proceedings of SIGKDD*, pp. 701–710 (2014)
5. Tang, J., Qu, M., Wang, M., Zhang, M., Yan, J., Mei, Q.: Line: large-scale information network embedding. In: *Proceedings of WWW*, pp. 1067–1077 (2015)
6. Grover, A., Leskovec J.: node2vec: scalable feature learning for networks. In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 855–864 (2016)
7. Cui, P., Wang, X., Pei, J.: A survey on network embedding. In: *Proceedings of IEEE Transactions on Knowledge and Data Engineering* (2018)
8. Leskovec, J., Huttenlocher, D., Kleinberg, J.: Predicting positive and negative links in online social networks. In: *Proceedings of WWW*, pp. 641–650. ACM (2010)
9. Chen, Y., Qian, T., Liu, H., Sun, K.: “Bridge” enhanced signed directed network embedding. In: *The 27th ACM International Conference on Information and Knowledge Management*, pp. 773–782. ACM (2018)
10. Yuan, S., Wu, X., Xiang, Y.: SNE: signed network embedding. In: Kim, J., Shim, K., Cao, L., Lee, J.-G., Lin, X., Moon, Y.-S. (eds.) *PAKDD 2017. LNCS (LNAI)*, vol. 10235, pp. 183–195. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-57529-2_15
11. Wang, S., Tang, J., Aggarwal, C., Chang, Y., Liu, H.: Signed network embedding in social media. In: *Proceedings of SDM*, pp. 327–335. SIAM (2017)
12. Islam, M.R., Aditya Prakash, B., Ramakrishnan, N.: SIGNet: scalable embeddings for signed networks. In: Phung, D., et al. (eds.) *PAKDD 2018. LNCS (LNAI)*, vol. 10938, pp. 157–169. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-93037-4_13
13. Kim, J., Park, H., Lee, J., Kang U.: SIDE: Representation learning in signed directed networks. In: *Proceedings of WWW*, pp. 509–518 (2018)
14. Lu, C., Jiao, P., Liu, H., Wang, Y., Xu, H., Wang, W.: SSNE: status signed network embedding. In: Yang, Q., Zhou, Z.-H., Gong, Z., Zhang, M.-L., Huang, S.-J. (eds.) *PAKDD 2019. LNCS (LNAI)*, vol. 11441, pp. 81–93. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-16142-2_7
15. Javari, A., Qiu, H., Barzegaran, E., Jalili, M., Chang, K.: Statistical link label modeling for sign prediction: smoothing sparsity by joining local and global information. In: *Proceedings of ICDM*, pp. 1039–1044 (2017)
16. Kumar, S., Spezzano, F., Subrahmanian, V., Faloutsos, C.: Edge weight prediction in weighted signed networks. In: *Proceedings of ICDM*, pp. 221–230 (2016)
17. Kingma, D., Ba, J.: Adam: a method for stochastic optimization. In: *Proceedings of ICLR* (2014)