Signed Random Walk with Restart: A New Approach To Personalized Ranking in Signed Networks

Erdem Tuna Yildiz

Complex Network Theory and Applications

University of Zurich

As the number of users engaging in online interactions continues to grow, the identification of the relationships between them has become increasingly vital in forming accurate clusters of communities and conduct targeted researches. While most social networks utilize signed networks to indicate trust or distrust among users, determining rankings for users in these networks is a challenging task, particularly when attempting to identify negative (distrust) connections with precision. Traditional methods such as PageRank and Random Walk with Restart prove ineffective in this regard, as they only take into account positive connections. Additionally, the intricacies of real-world networks are far more complex than the simple "the friend of a friend is a friend" or "the enemy of an enemy is a friend." To address these issues, this paper presents a new algorithm, Signed Random Walk With Restart[1], which aims to provide personalized rankings for each node in relation to a seed node. The algorithm initiates a random walker with a positive sign at the seed node, and changes its sign if the edge is negative while maintaining it if the edge is positive. Results indicate that the algorithm is capable of achieving up to 83% accuracy in sign prediction. Overall, the proposed algorithm provides a more comprehensive and accurate approach to understanding the relationships within signed networks.

Introduction

Interactions between users in social networks with signed edge usually forms with users' relative thought on another user's perspective, sale or comment on another user. In most of the online platforms, users are sharing their opinions on other users to indicate their trust or distrust measures. It is crucial to point out these trust-distrust scores in between users because it gives an overall relative trustworthiness of the nodes and play a significant role on how important the node is in the network is. Figuring out the nodes that have high trust or distrust scores becomes handy to maintain high quality of the network. Users with high distrust scores could be shown less to other users while high trust scored nodes can be more centralized making other users feel that the network is more trustworthy with low amount of trolls which could contribute to their overall experience on the network. When a user is selling a product online, buyers look for the users that has high rating thinking that making transactions with high rated users will be much more reliable and tend to have lower risk of theft. However, user can boost his/her rating by registering with multiple different accounts. With SRWR, this problem can be improved, as the algorithm is able to solve complex trust-distrust measures by considering the relative trustworthiness scores of numerous different nodes, instead of just looking at simple friend-of-a-friend edges.

Ranking the nodes based on their respective trust scores is essential for tasks such as link predictions and community detection. However, popular methods such as PageRank or Random Walk With Restart are limited in their ability to interpret negative edges and only consider positive edges. Another issue with these algorithms is their lack of accuracy when the network is complex and composed of unexpected edge link formations, such as a friend of a friend being an enemy. Additionally, these algorithms tend to give a global ranking to the nodes, rather than a personalized ranking that takes into account the relationships between each node. Personalized ranking is much more efficient compared to global ranking, as it can provide personalized recommendations, such as suggesting people to follow on a social network or displaying more relevant posts based on the user's hobbies and interests. Furthermore, personalized ranking can recommend for each individual node, rather than only considering the general trend of the network.

In this research, we examine the effectiveness of using a Random Walk with Restart[1] algorithm on a Bitcoin Alpha network dataset to assess anonymous user reputations and minimize fraudulent and risky transactions. The goal is to determine the accuracy of the method and compare it with other conventional algorithms. From a network theory point of view, the approach involves assigning individual rankings to nodes instead of global rankings. The proposed method, called the Signed Random Walk with Restart (SRWR), is an efficient iterative algorithm that calculates personalized rankings. It works by assigning a sign to a walker, which changes depending on the edge sign between two nodes. This enables the algorithm to take into account complex edge relationships

Significance Statement

The central idea behind using network analysis algorithms to assign trust-distrust scores to nodes is to rank the nodes in a network based on their relevance to a specific topic or in relation to another node. This ranking has numerous advantages, such as discovering individuals with similar interests, targeting advertisements based on interests, and facilitating community analysis. SRWR is an algorithm that provides personalized rankings to nodes in a network relative to a designated seed node. By utilizing this algorithm, it becomes easier to display personalized ads, thus enhancing the user experience by presenting more relevant content.

with respect to the seed node and to interpret negative edges due to sign changes. We evaluate the effectiveness of the proposed method by applying it to a real network data set for link prediction. Our results show that the algorithm achieves an accuracy of 83 percent in link prediction.

Methods

The Signed Random Walk with Restart (SRWR) algorithm is a model that ranks nodes in a signed network based on trust and distrust relationships. It operates by introducing a signed walker with a positive sign at the selected seed node, whose sign changes as it moves through edges. Positive indicates the node is a friend to the seed node, while negative means it is an enemy. SRWR ranks nodes by considering edge relationships during the walk, unlike conventional methods such as PageRank and Random Walk with Restart. The ranking is determined by the frequency of positive and negative signed walkers visiting a node. The walker can either make a signed random walk or restart, with a restart probability of c. The ranking of each node is calculated based on the probabilities of positive and negative walkers being at that node.

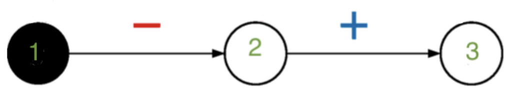

Fig. 1. Friend of an enemy case.

Considering a simulation of a 'friend of an enemy is my enemy' with three nodes as given in Figure 1, the seed node s would have a negative edge with the second node and the second node would have a positive edge with the third node t. Suppose a random walker goes through s towards t. Conventional methods such as PageRank and Random Walk With Restart would not consider the edge signs which means conventional algorithms would fail to consider edge relationships during the walk. Considering the given arbitrary situation above, SRWR would successfully interpret the edge relationships and give accurate rankings of trust-distrust to the nodes. It would start 'positive' on the seed node, and would change its sign to negative since the edge is negative, and would keep its negative sign since the edge has negative sign. So, SRWR would rank node 2 and 3 as 'enemy' which is expected. With this method, the SRWR can successfully rank the nodes in other cases such as "friend's friend", "friend's enemy" and "enemy's enemy" etc.

The trust-distrust relationships of the network with respect to the seed node are determined by having the surfer navigate through it. If the positive walker visits a node more frequently than the negative walker, SRWR will assign a positive ranking to that node. Conversely, if the negative walker passes through a node more frequently, SRWR will assign it a negative ranking. This methodology allows for ranking the entire network in relation to the seed node.

The walker has two options: a signed random walk or a restart. During a signed random walk, the walker will make transition to one of its neighboring nodes and either change or maintain its sign, depending on the sign of the edge, with a probability of 1-c (where c is the restart probability). The second option is a restart, which occurs with a probability of c. This returns the walker to the seed node and changes its sign to positive if it was negative, or keeps it positive if it was already positive, as the walker starts as a friend of the seed node.

The ranking of each node is determined by measuring two probabilities: r_u^+ and r_u^-. r_u^+ represents the trust SRWR score, or the probability that a positive surfer is at node u. r_u^- represents the distrust SRWR score, or the probability that a negative surfer is at node u. A high r_u^+ indicates that node u is trustworthy for the seed node, while a high r_u^- suggests the opposite. The overall score of a node with respect to the seed node is calculated as rd = r_u^+ - r_u^-, and the relative trust score is then determined based on this value.

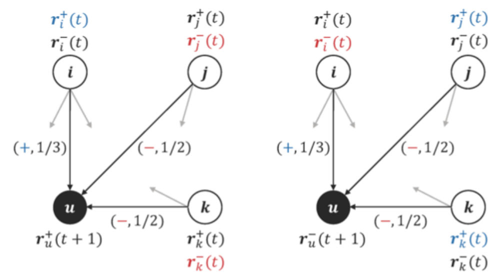

Fig. 2. Example scenario for r_u^+ and r_u^- calculation[1].

The calculation of r_u^+ and r_u^- scores can be explained using the example presented in Figure 2. Firstly, the transition probabilities of each pair are calculated based on the node degrees. For instance, the transition probability of a positive edge from node i to node u is 1/3, as node i has 3 outgoing edges. It is crucial to consider both the edge sign and walker's sign when determining the positive or negative score of a surfer at a given node. To have a positive surfer on node u at time t+1, either a positive surfer on one of u's neighbors at time t must move through u via a positive edge, or a negatively signed walker must traverse u at time t via a negative edge. For a walker to have a negative sign on node u, a positive surfer must traverse a negative edge, or a negative surfer must traverse a positive edge. With the consideration of the reset probability, c, the calculation of r_u^+ for Figure2(a) and r_u^- for Figure2(b) turns out to be:

$$\mathbf{r}_u^+(t+1) = (1-c)\left(\frac{\mathbf{r}_i^+(t)}{3} + \frac{\mathbf{r}_j^-(t)}{2} + \frac{\mathbf{r}_k^-(t)}{2}\right) + c\mathbf{1}(u=s) \qquad [1]$$

where 1(u=s) is 1 if u is the seed node s and 0 otherwise.

$$\mathbf{r}_u^-(t+1) = (1-c)\left(\frac{\mathbf{r}_i^-(t)}{3} + \frac{\mathbf{r}_j^+(t)}{2} + \frac{\mathbf{r}_k^+(t)}{2}\right) \qquad [2]$$

It is worth noting that we do not add the restarting score for r_u^- because walker's sign becomes positive when it goes back to the seed node.

A new summarized equation can be introduced to be able to make the equations [1] and [2] recursive so that calculation of

r_u^+ and r_u^- can be extended through each node of the network.

$$\mathbf{r}_u^+ = (1-c)\left(\sum_{v\in\overleftarrow{\mathbf{N}}_u^+}\frac{\mathbf{r}_v^+}{|\overrightarrow{\mathbf{N}}_v|} + \sum_{v\in\overleftarrow{\mathbf{N}}_u^-}\frac{\mathbf{r}_v^-}{|\overrightarrow{\mathbf{N}}_v|}\right) + c\mathbf{1}(u=s)$$
$$\mathbf{r}_u^- = (1-c)\left(\sum_{v\in\overleftarrow{\mathbf{N}}_u^-}\frac{\mathbf{r}_v^+}{|\overrightarrow{\mathbf{N}}_v|} + \sum_{v\in\overleftarrow{\mathbf{N}}_u^+}\frac{\mathbf{r}_v^-}{|\overrightarrow{\mathbf{N}}_v|}\right) \quad [3]$$

New symbols and variables needs to be introduced to vectorize equation [3].

Symbol	Definition
G	signed input graph
n	number of nodes in G
m	number of edges in G
s	seed node (= query node, source node)
c	restart probability
$\overleftarrow{\mathbf{N}}_u$	set of in-neighbors to nodes u
$\overrightarrow{\mathbf{N}}_u$	set of out-neighbors from nodes u
$\mathbf{A}$	$(n \times n)$ signed adjacency matrix of G
$\lvert\mathbf{A}\rvert$	$(n \times n)$ absolute adjacency matrix of G
$\mathbf{D}$	$(n \times n)$ out-degree matrix of $\lvert\mathbf{A}\rvert$, $\mathbf{D}_{ii} = \sum_j \lvert\mathbf{A}\rvert_{ij}$
$\tilde{\mathbf{A}}$	$(n \times n)$ semi-row normalized matrix of $\mathbf{A}$
$\tilde{\mathbf{A}}_+$	$(n \times n)$ positive semi-row normalized matrix of $\mathbf{A}$
$\tilde{\mathbf{A}}_-$	$(n \times n)$ negative semi-row normalized matrix of $\mathbf{A}$
$\mathbf{q}$	$(n \times 1)$ starting vector (= sth unit vector)
$\mathbf{r}^+$, $\mathbf{r}^-$	$(n \times 1)$ trust and distrust SRWR score vector, resp.
$\mathbf{r}^d$	$(n \times 1)$ relative trustworthy vectors, $\mathbf{r}^d = \mathbf{r}^+ - \mathbf{r}^-$

Fig. 3. Symbols[1]

Signed adjacency matrix $\mathbf{A}$ of graph G is a matrix where the values are 1 or -1 depending on the edge sign and zero otherwise. Semi row matrix $\tilde{A} = D^{-1}$A. The positive semi-row matrix and negative semi-row matrix are introduced to capture the positive and negative values in $\tilde{A}$, where the positive semi-row matrix contains only the positive values and the negative semi-row matrix contains only the negative values.

Given the information of positive and negative semi row matrix and the table, the equation [3] can be written as:

$$\mathbf{r}^+ = (1-c)\left(\tilde{\mathbf{A}}_+^\top\mathbf{r}^+ + \tilde{\mathbf{A}}_-^\top\mathbf{r}^-\right) + c\mathbf{q}$$
$$\mathbf{r}^- = (1-c)\left(\tilde{\mathbf{A}}_-^\top\mathbf{r}^+ + \tilde{\mathbf{A}}_+^\top\mathbf{r}^-\right) \quad [4]$$

$\mathbf{q}$ is the starting vector with seed node index element is 1, otherwise 0.

In both synthetic and real networks, the relationships between nodes are often inconsistent with the general principles of "friend of a friend is a friend," "friend of an enemy is enemy," "enemy of an enemy is a friend," and "enemy of a friend is an enemy." This is due to random edge sign generation in synthetic networks and imbalanced relationships in real networks. To account for the uncertainties in trust-distrust relationships, the model introduces stochastic parameters β and γ as balance attenuation factors. β represents the uncertainty of "enemy of enemy is a friend," and γ represents "friend of an enemy is enemy." From an algorithmic point of view, when a walker moves from seed node s to node B and then to node C, both of which have positive edges, the walker retains its positive sign and ranks node C as trustworthy for seed node s. However, this may not always be the case due to the uncertainty of friendship. To consider this, the walker carries its positive sign with probability β or changes its sign with probability 1-β when it moves through a positive edge. Similarly, when a negative walker moves through a negative edge, it changes its sign with probability γ or keeps its sign with probability 1-γ. Since it is impossible to find β and γ values for the network, the algorithm will be running with different values of β and γ values and the highest accuracy will be recorded. By this way, the uncertainty of the network can also be estimated which gives a lot of information about the network in general. After inserting β and γ parameters, equation [4] becomes:

$$\mathbf{r}^+ = (1-c)\left(\tilde{\mathbf{A}}_+^\top\mathbf{r}^+ + \beta\tilde{\mathbf{A}}_-^\top\mathbf{r}^- + (1-\gamma)\tilde{\mathbf{A}}_+^\top\mathbf{r}^-\right) + c\mathbf{q}$$
$$\mathbf{r}^- = (1-c)\left(\tilde{\mathbf{A}}_-^\top\mathbf{r}^+ + \gamma\tilde{\mathbf{A}}_+^\top\mathbf{r}^- + (1-\beta)\tilde{\mathbf{A}}_-^\top\mathbf{r}^-\right) \quad [5]$$

The algorithm for implementing the Signed Random Walk with Restart is an iterative process that computes trust-distrust measures for each node in a network with respect to a seed node, which is selected at the beginning of the algorithm. The algorithm uses Equation 5 to determine the SRWR score for each node.

The first step of the algorithm is the normalization. The function for normalization takes signed adjacency matrix $\mathbf{A}$ as input and returns positive and negative semi-row normalized matrix. To start, the out-degree diagonal matrix $\mathbf{D}$ is found from the absolute adjacency matrix $|\mathbf{A}|$, which is 1 if there is an edge between the nodes and 0 otherwise. Then, semi-row normalized matrix $\tilde{\mathbf{A}}$ is found where $\tilde{\mathbf{A}} = \mathbf{D}^{-1}$ $\mathbf{A}$. For the final step, $\tilde{\mathbf{A}}$ splits into two parts: the positive semi-row normalized matrix $\mathbf{A}_+$ takes only the positive values from $\tilde{\mathbf{A}}$ and sets the rest to 0, while the negative semi-row normalized matrix $\mathbf{A}_-$ takes only the negative values from $\tilde{\mathbf{A}}$ and sets the rest to 0.

The algorithm is followed by the iteration phase. The iteration function takes positive semi-row normalized matrix $\mathbf{A}_+$, negative semi-row normalized matrix $\mathbf{A}_-$, seed node $\mathbf{s}$, restart probability $\mathbf{c}$, β, γ and the error toleration ϵ as input and returns SRWR vector scores of $\mathbf{r}_+$ and $\mathbf{r}_-$. The iteration phase is shown below:

1. $\mathbf{q}$ is set to be 1 for the seed node index and zero otherwise.
2. Initialize $\mathbf{r}_+ = \mathbf{q}$ and $\mathbf{r}_- = 0$. $\mathbf{r}^{'}$ is vertical concatenation $\mathbf{r}_+$ and $\mathbf{r}_-$.
3. Update $\mathbf{r}_+$ and $\mathbf{r}_-$ using equation [5].
4. $\mathbf{r}$ = vertical concatenation of $\mathbf{r}_+$ and $\mathbf{r}_-$
5. $\delta = |\mathbf{r}$ - $\mathbf{r}^{'}|$
6. $\mathbf{r}^{'} = \mathbf{r}$
7. repeat 3-6 until $\delta < \epsilon$
8. return $\mathbf{r}_+$ and $\mathbf{r}_-$

Results

The algorithm was tested on two different networks: a synthetic network and a real network. The synthetic network was used to examine simple relationships between nodes and confirm the algorithm's accuracy in predicting edge signs, while ignoring β and γ values to evaluate the algorithm without friendship uncertainties. The real network was then used to test the algorithm's performance, taking into account the friendship uncertainties.

The Watts-Strogatz algorithm is a good option for evaluating synthetic network task due to its small-world property that mimics real-world networks with high clustering and short distances. A circular graph with N nodes is first established with connections to c nearest neighbors, then edges are rewired with a probability of p. By manipulating edge signs, the algorithm's performance can be tested without the beta and gamma values of friendship uncertainties, allowing for evaluation without stochastic calculations. In a 8-node Watts-Strogatz network, with each node connected to 5 nearest neighbors and a rewiring probability of 0.05, random edges were removed from a seed node and the algorithm was tested. The results showed a 91% accuracy rate in edge sign prediction.

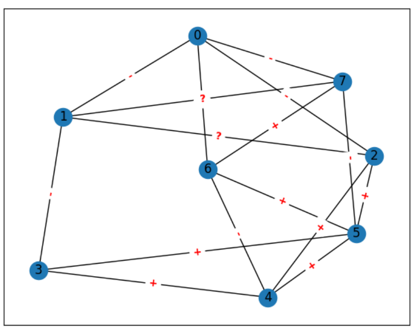

Fig. 4. Watts-Strogatz Graph with missing signs.

Bitcoin Alpha is a platform where anonymous individuals trade Bitcoin with one another. To ensure the platform's operation, trust between users is essential as registration with real identities is not required. To maintain security, users are given the ability to rate other users. Transactions with high-rated users are considered safe, while low-rated users are often seen as risky and potentially fraudulent. The network comprises approximately 4000 nodes and 24000 edges. It has a right-skewed degree distribution.

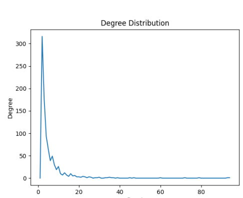

Fig. 5. Degree Distribution

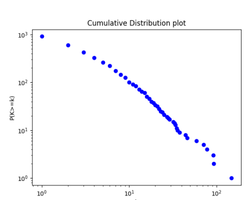

Fig. 6. Complementary Cumulative Distribution

Being able to give personalized ranking to this network is essential since global ranking methods will be lowering the transaction amount by evaluating couple users high all the time. Personalized ranking of each node with respect to a seed node is more efficient for this network's operation compared to a global ranking of nodes. PageRank and Random Walk With Restart may result in a small number of users monopolizing the market and having the most transactions. These algorithms will assign high scores to nodes with the most successful transactions, which may disadvantage users with fewer transactions. The number of transactions is not necessarily the best evaluation of security, as some users may have low transaction numbers but still be secure. A global ranking may also lower transaction traffic through the platform, which is not beneficial.

Additionally, high global ranked users may behave differently towards friends and business partners, being ranked high as long as they have more friends than enemies. The SRWR algorithm allows for the entire network to be ranked with respect to a seed node, allowing for the interpretation of trust-distrust relationships on an individual basis, creating a more diverse transaction environment."

The node with the highest degree centrality was selected as the seed node for the SRWR algorithm. This is because nodes with high degree centrality are often highly connected in the graph, hold a significant amount of information about the network, and typically have access to nodes in other clusters. After picking the seed node in the network, 20% of the edge signs were removed from the seed node. To evaluate the optimal β and γ friendship uncertainties, the algorithm was run on the seed node with different β and γ values ranging from 0.1 to 1, incrementing by 0.1.

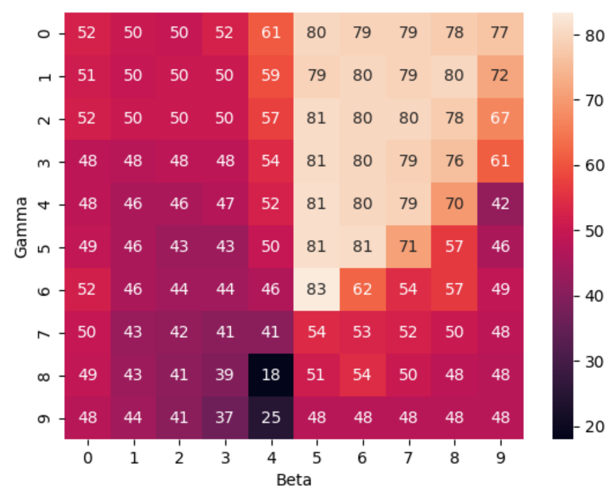

Fig. 7. SRWR accuracy with different β and γ values (0.1x for x and y values.)

The algorithm has the highest accuracy of 83% when β is 0.5 and γ is 0.6.

SRWR accuracy score on BitcoinAlpha for link prediction was compared to the findings from J.Jung and W.Jin's paper[1] where they compared algorithms SRWR, Random Walk with Restart (RWR), Modified Random Walk with Restart (M-RWR), Modified Personalized SALSA (M- PSALSA), Personalized Signed spectral Rank (PSR), Personalized Negative

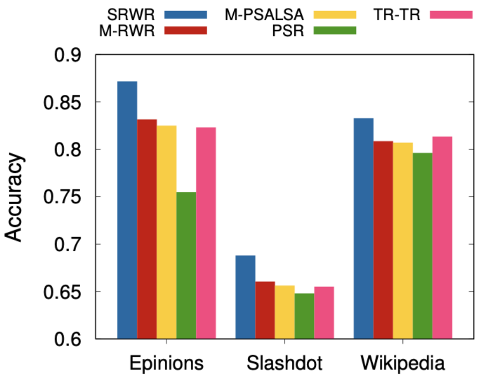

Fig. 8. Accuracies of RWR, M-RWR, M-PSALSA, PSR and PNR on different networks Epinions, Slashdot and Wikipedia[1].

Rank (PNR), and Troll-Trust Model (TR-TR) on different networks. Our SRWR accuracy finding on BitcoinAlpha is close to their SRWR scores on different datasets which proves that SRWR is a valid personalized ranking method in different networks and scenarios.

Discussions

The SRWR algorithm was utilized in the BitcoinAlpha network to predict the signs of edges in relation to the seed node with the highest degree centrality. The objective of the study was to remove 20% of the edges from the seed node and allow the SRWR to predict their signs. SRWR achieved an accuracy of 83% for BitcoinAlpha network. Comparing other algorithms in different networks of Epinions, Slashdot and Wikipedia, it can be concluded that the SRWR algorithm is efficient and outperforms current conventional methods.

Epinions is a free website where users can register and review products. They are allowed to express their opinions on a variety of items. As shown in Figure 8, all of the algorithms performed well compared to other network datasets because users are rewarded if they have written enough useful reviews. The contribution of nodes is also a critical factor for the algorithms to function properly. When the nodes have a consensus on writing the best and most comprehensive reviews, the accuracy of the network connection and algorithm scores improves. Providing incentives for the nodes to reach a consensus is another factor in enhancing the overall quality of the network and accuracy of the algorithms.

On the other hand, all algorithms, including SRWR, have the lowest accuracy on Slashdot. Slashdot is a technology news website where users can write and express their opinions, which are then rated by editors. There are several reasons why the ranking algorithms score much lower on this network. Firstly, users do not have any incentives to write sensible reviews and are free to express any opinion, even if it is not their own (such as trolling). Secondly, instead of having users rank each other, a group of editors rate the opinions. This can lead to bias and a lack of consensus on writing the most optimal review. As a result, the network is of low quality and has more randomly signed edges, causing the algorithms to score low.

Although there are differences in performance among the previously discussed algorithms, the quality of the network remains one of the most crucial factor for link prediction and any analysis of community structure. Imbalanced and randomly connected edges will negatively impact the performance of the algorithm, regardless of its complex mathematics and optimized coding. In more structured networks, SRWR has been deemed the most optimal and demonstrates the highest performance in terms of link prediction accuracy comparatively.

Conclusion

The Signed Random Walk with Restart (SRWR) algorithm was used to evaluate personalized ranking in a network with respect to a seed node, rather than providing a global ranking. The main distinction of this algorithm is that the walker has its own sign, starting as positive at the seed node, and changing its sign based on the edge signs between the nodes. If the edge is positive, the walker retains its sign, and if the edge is negative, the walker changes its sign.

Each node in the network had two probabilities that determined its relative trust-distrust score. The probability that a positive surfer was at node "u" was represented by "r_u^+," while the probability that a negative surfer was at node "u" was represented by "r_u^-." A high value of "r_u^+" indicated that node "u" was trustworthy with respect to the seed node, while a high value of "r_u^-" indicated that node "u" was not trustworthy with respect to the seed node. The trust-distrust score of a node with respect to the seed node was determined by the difference between "r_u^+" and "r_u^-." The SRWR algorithm attempted to find the trust-distrust score for each node by using a recursive process.

The Signed Random Walk with Restart (SRWR) algorithm was optimized for improved performance through vectorization. The signed adjacency matrix $\mathbf{A}$ was introduced, where each element of the matrix represented the sign of the corresponding edge (1 for positive and -1 for negative). The matrix was later transformed into a semi-row matrix ($\tilde{\mathbf{A}}$), which was then separated into two matrices, $\mathbf{A}_+$ and $\mathbf{A}_-$, to hold positive and negative values separately.

The introduction of two stochastic variables, β and γ, was necessary to handle the uncertainties present in real-world networks where connections such as "enemy of an enemy is a friend" or "friend of an enemy is an enemy" exist. These variables help to effectively deal with the complexities arising from these uncertainties.

The walker had two options: to jump to a neighboring node with respect to its position with probability 1-c, or to reset to the seed node and change its sign to positive, regardless of its current sign, with probability c. This walking step was repeated recursively until the error margin became lower than a pre-determined variable, epsilon.

The SRWR algorithm was run on a real network called BitcoinAlpha, where users trade Bitcoin among each other. Given that users register on the platform without providing their real identities, the security of the platform is critical to its continued operation. To test the accuracy of the SRWR, a seed node with the highest degree centrality was selected, and 20% of its edge signs were removed. The SRWR algorithm was then run with different values of β and γ, ranging from 0.1 to 1 and incrementing by 0.1. With 4000 nodes and 24,000 edges in the network, the algorithm recursively provided a personalized ranking for each node and predicted the edge signs. The algorithm showed an accuracy of 83% when beta and gamma values were set to 0.5 and 0.6, respectively.

Although SRWR runs efficiently and outperforms current conventional algorithms, the structure of the network is still crucial for proper operation. In more randomized and biased signed networks, such as Slashdot, all algorithms, including SRWR, scored lower in comparison. In more organized and structured networks, where nodes have incentive to perform their best, all algorithms scored much higher. Despite the network structure, SRWR still scored the highest among other algorithms.

References

1. J. Jung, W. Jin, L. Sael and U. Kang, "Personalized Ranking in Signed Networks Using Signed Random Walk with Restart," 2016 IEEE 16th International Conference on Data Mining (ICDM), Barcelona, Spain, 2016, pp. 973-978, doi: 10.1109/ICDM.2016.0122.

2. Bhatia, Chaitanya. "Random Walk With Restart and Its Applications - Chaitanya Bhatia." Medium, 9 Dec. 2021, medium.com/chaitanyabhatia/random-walk-with-restart-and-its-applications-f53d7c98cb9

3. Papers With Code - Slashdot Dataset. paperswithcode.com/dataset/slashdot.

4. Disney, Andrew. "Social Network Analysis: Understanding Centrality Measures." Cambridge Intelligence, 27 Oct. 2022, cambridge-intelligence.com/keylines-faqs-social-network-analysis.

5. S. Kumar, F. Spezzano, V.S. Subrahmanian, C. Faloutsos. Edge Weight Prediction in Weighted Signed Networks. IEEE International Conference on Data Mining (ICDM), 2016.

6. S. Kumar, B. Hooi, D. Makhija, M. Kumar, V.S. Subrahmanian, C. Faloutsos. REV2: Fraudulent User Prediction in Rating Platforms. 11th ACM International Conference on Web Searchand Data Mining (WSDM), 2018.

7. Stanford Network Analysis Project. "Soc-Sign-Bitcoin-Alpha." snap.stanford.edu, Stanford University, https://snap.stanford.edu/data/soc-sign-bitcoin-alpha.html.