Hindawi
Wireless Communications and Mobile Computing
Volume 2021, Article ID 8871150, 11 pages
https://doi.org/10.1155/2021/8871150

Research Article

Transaction Prediction in Blockchain: A Negative Link Prediction Algorithm Based on the Sentiment Analysis and Balance Theory

Ling Yuan[1], **JiaLi Bin**[1], **YinZhen Wei**[2], **Zhihua Hu**[2] **and Ping Sun**[2]

[1]*School of Computer Science, Huazhong University of Science and Technology, 430074, China*
[2]*Huanggang Normal University, 438000, China*

Correspondence should be addressed to YinZhen Wei; wyz_gs@163.com

Received 25 July 2020; Revised 19 December 2020; Accepted 1 February 2021; Published 16 February 2021

Academic Editor: Miguel López-Benítez

User relationship prediction in the transaction of Blockchain is to predict whether a transaction will occur between two users in the future, which can be abstracted into the link prediction problem. The link prediction can be categorized into the positive one and the negative one. However, the existing negative link prediction algorithms mainly consider the number of negative user interactions and lack the full use of emotion characteristics in user interactions. To solve this problem, this paper proposes a negative link prediction algorithm based on the sentiment analysis and balance theory. Firstly, the user interaction matrix is constructed based on calculating the intensity of emotion polarity for social network texts, and a reliability weight matrix (noted as RW-matrix) is constructed based on the user interaction matrix to measure the reliability of negative links. Secondly, with the RW-matrix, a negative link prediction algorithm is proposed based on the structural balance theory by constructing negative link sample sets and extracting sample features. To evaluate the performance of the negative link prediction algorithm proposed, the variable management method is used to analyze the influence of negative sample control error and other parameters on the accuracy of it. Compared with the existing prediction benchmark algorithms, the experimental results demonstrate that the proposed negative link prediction algorithm can improve the accuracy of prediction significantly and deliver good performances.

1. Introduction

Everything in our lives has been digitized with the network technologies including the wireless mesh network with the topology preservation [1] and the wireless sensor networks using the Markov random field [2], both of which assist in providing the last mile Internet access for users. Moreover, with the development of the Blockchain, the technology behind the Bitcoin cryptocurrency system [3], many kinds of cryptocurrencies have been used in digital transactions. The link prediction algorithms can be used to predict whether a cryptocurrency transaction relationship will occur between the users. Tanevski et al. [4] use the Bitcoin OTC network to predict whenever a possible transaction will be made between two users in the network. In the transactions of Blockchain, the node represents a user, and the link represents the trust degree or the ratings or the assessments of the users given to each other after they made a transaction; the value of the link can be either numbers or texts. Thus, the link prediction algorithm, which is based on the information of user relations and user attributes in various networks, will provide more information and support for the decision-making of users who make transactions based on Blockchain technology.

Link prediction is mainly divided into the user similarity matrix [5] and the machine learning-based method [6]. In the user similarity matrix, the value represents the similarity between two nodes, and when the value is greater, the possibility of the existence of links between nodes is greater. The machine learning-based method is to create a model with a set of adjustable parameters. The optimal parameter value is found by the optimization strategy, so the obtained model can reproduce the real network structures and relationship characteristics better. In addition, Yuan and Pradeep found that adding emotional features can improve the accuracy of link prediction [7]. According to the user's emotional

features extracted from different topics, the more likely the two users have common emotional tendencies, the more likely they are to be friends.

However, most platforms prefer to exhibit the positive sentiments and conceal the negative ones. For example, the users can express their positive sentiments with the thumbs-up icon or other functions directly. In contrast, if users want to express negative sentiments, they can only leave messages in the comments section. Hence, most social platforms ignore the negative data; the importance of negative link prediction is underestimated and underutilized [8]. Fortunately, researchers found that the negative link is a good complement to the positive link [9]. The negative link prediction can use the positive link and the interactions of users in the network to predict the possible negative relationships among users, where the interaction among users includes the number of interactions, interaction tendency, and interaction intensity [10]. The existing negative link prediction algorithms mainly focus on the number of negative interactions of users and lack the full utilization of the emotional features in the interaction of users which can improve the accuracy of link prediction [11].

Therefore, this paper proposes a negative link prediction algorithm based on the sentiment analysis. Firstly, with the combination of the sentiment analysis method and the social network, this paper proposes a method to calculate the intensity of the emotional polarity for social network texts. On this basis, we propose a method to construct the user interaction relationship matrix. Based on the user interaction relationship matrix, we construct a reliability weight matrix (noted as RW-matrix) for measuring the reliability of negative links. Secondly, we construct a negative link sample set based on the interactions of users and extract the sample features. Then, the paper proposes a negative link prediction algorithm with the structural balance theory (noted as SABT-NLP).

The remainder of this paper is organized as follows. Section 2 discusses the related work. Section 3 introduces the preliminaries of negative link prediction, and Section 4 describes the details of the proposed methodology. Section 5 presents the experimental evaluation, and finally, Section 6 gives the conclusion.

2. Related Work

Link prediction can use the existing network topology and other information contained in the network to predict the possibility of future connections in the network which have not yet been connected in the network. Symbol networks are one of the most representative methods used in link prediction [12, 13]. Symbol networks are networks with positive and negative signs. For example, YouTube allows users to utilize some of the features to express their opinions about whether they like a video or not. Epinions allows users to rate other users' contents. Such functions in social networks contribute to the development of the symbol network.

Liben-Nowell and Kleinberg [14] proposed a link prediction model and pointed out that the link prediction mainly depends on the similarity between nodes. The more similar the two nodes are, the more likely that a link exists, which can be determined based on the common neighbor method and common path method. However, such methodology is not so suitable for large-scale social networks because of the high costs caused by computing feature values. Song et al. [15] proposed a matrix decomposition method to solve the problem of node similarity in online social networks, which can be applied to large-scale social networks. However, it raises a trick problem; i.e., online social networks do not completely correspond to offline social relationships. Then, Fire et al. [16] proposed a prediction algorithm for the cases lacking some offline friends. Such proposed algorithm can adapt to large-scale social networks as well. Moreover, this methodology can help online social network users to explicitly find people who either know each other or have similar interests with them.

In addition to the analysis of the node similarity to help the link prediction, the attributes like vertices and edges can also be extracted in different scenarios to improve the prediction performance. Benchettara et al. [17] proposed a link prediction algorithm for bipartite social networks based on the extracted attributes of vertices and edges.

Link prediction includes not only the positive one but also the negative one. Leskovec et al. [18] pointed out that the information contained in the negative link can effectively improve the prediction accuracy. Kunegis et al. [19] also confirmed that the negative link prediction can have added value to the social network analysis. Nevertheless, the existing negative link prediction algorithms mainly consider the number of negative interactions among users [18, 19], lacking the full utilization of the sentiment features in the interaction between users. Yuan and Pradeep [7] pointed out that adding emotional features can improve the accuracy of link prediction. Therefore, our paper mainly focuses on how to combine the user interaction information and sentiment analysis to solve the negative prediction problem.

3. Preliminaries

To better understand how to utilize the user interaction information and the positive link to predict the negative link, the basic definition is as follows.

First, a relation network g_p is given that contains the positive link, A is the user content relation matrix, O is the user opinion relation matrix, and then a predictor f is generated by g_p, A, and O, which can predict the negative relation network. Before illustrating our proposed method about the negative link prediction, the meanings of symbols which will be used are shown in Table 1.

3.1. Sentiment Polarity Intensity Quantification. In social networks, texts have the following characteristics: huge in volume, short in length, disorganized in words, and freely expressed in grammar. To perform sentiment analysis and quantification of network texts, we design a method for the short text of the social networks by using polarity intensity quantification. The intensity of the text is quantified based on the polarity intensity of each sentiment word.

Table 1: The meanings of symbols in the paper.

Symbol	Meaning
$\mathcal{G}_p$	A set of positive links known in the network
A	$R^{m\times M}$, user content matrix: the relationship matrix of users and contents
O	$R^{m\times M}$, user opinion matrix: the relationship matrix of users and opinions
Q (quantity)	$R^{m\times M}$, user opinion number matrix: the relationship matrix of users and the number of opinions
S	$R^{m\times M}$, user opinion intensity matrix: the relationship matrix of users and the sentiment intensity of opinions
C (count)	$R^{m\times m}$, user interaction number matrix: the relationship matrix of the number of user interactions
E (emotion)	$R^{m\times m}$, user interaction intensity matrix: the relationship matrix of the sentiment intensity of user interactions
NC (negative count)	$R^{m\times m}$, user negative interaction number matrix: the relationship matrix of the number of negative user interactions
NE (negative emotion)	$R^{m\times m}$, user negative interaction intensity matrix: the relationship matrix of the sentiment intensity of negative user interactions
N (nexus)	$R^{m\times m}$, user interaction matrix: the relationship matrix of user interactions
NN (negative nexus)	$R^{m\times m}$, user negative interaction matrix: the relationship matrix of negative user interactions
Senti	The polarity intensity quantification method

The sentiment words can be divided into basic sentiment words and compound sentiment words. The polarity of the basic sentiment words is based on the SentiWordNet annotation of the sentiment dictionary. The polarity intensity calculation of the compound sentiment words is complicated, which can be determined by the following semantic rules:

(1) Degree Modifiers+Basic Sentiment Words. Degree modifiers give different weights such as $(0.5, 0.7, 0.9, 1.1, 1.3, 1.5)$ depending on the intensity of action. For example, the weight 1.5 represents the sentiment intensity of "super," the weight 1.3 represents the sentiment intensity of "very," the weight 1.1 represents the sentiment intensity of "a little," and the weight 0.5 represents the sentiment intensity of "little." The polarity intensity of this type of compound words is the product of the intensity of the degree modifier and the intensity of the basic sentiment word. If the product exceeds the interval $[-1, 1]$, its boundary is used as the polarity of the compound word.

(2) Repeated Degree Modifiers. For example, the sentiment intensity of "really really like" is stronger than that of "really like." The weight of two "really" needs to be multiplied on the basis of the "like" weight. If the product exceeds the interval $[-1, 1]$, its boundary is used as the polarity of the compound words.

(3) Negatives+Basic Sentiment Words. Such compound words only need to reverse the polarity of the original emotional words. For example, the sentiment intensity of "not good" is the reverse of the intensity of "good."

(4) Negatives, Degree Modifiers, and Basic Sentiment Words Occur Continuously. The combination of such compound words is complicated. Moreover, the different order of appearance of the negatives and degree modifiers will produce the opposite sentiment tendency. If the degree modifiers appear in the middle between the negative and the basic sentiment word, the polarity is the same as that of the basic sentiment word, and the intensity is equal to the negation of the degree modifier. However, if the degree modifier appears before the negative and the basic sentiment word, the polarity of the compound word should be inverted on the polarity of the basic sentiment word, and the intensity should be multiplied by the weight of the degree modifier based on the basic sentiment word. If the product exceeds the interval $[-1, 1]$, the boundary is used as the polarity intensity of the compound word.

(5) Emoticons. Emoticons are one of the popular paradigms for users to express their emotional tendency with graphic animations. In order to express richer emotions, we add emoticons to the sentiment dictionary.

Based on the above semantic rules and the sentiment dictionary, we can compute the polarity and intensity of each sentiment word, which is shown as Equation (1) below:

$$\text{Senti}(t) = \begin{cases} \text{sign}\,(a_1) * \max\,(|a_1|, |a_n|), & \text{if } a_1 \times a_n > 0, \\ a_1 + a_n, & \text{if } a_1 \times a_n < 0, \end{cases} \tag{1}$$

where $\text{Senti}(t)$ represents the sentiment polarity and intensity of the text, t is the text to be quantified, and $a_1, a_2, a_3, \cdots, a_n$ is a sequence of all emotional words in the text t after sorted from large to small with the intensity. If $\text{Senti}(t)$ is positive, the larger the $\text{Senti}(t)$ is, and the stronger the positive sentiment of the text t is. However, if $\text{Senti}(t)$ is negative, the smaller the $\text{Senti}(t)$ is, and the stronger the negative sentiment of the text t is.

3.2. User Interaction Matrix Construction. The user interaction matrix is a comprehensive description to fully express the relationship between users. The interaction between users mainly includes point of praise, point stepping, forwarding,

and comment. The user interaction matrix is constructed as follows:

(1) Initializing the user opinion number matrix Q, user opinion intensity matrix S, user interaction number matrix C, user interaction intensity matrix E, user negative interaction number matrix NC, and user negative interaction intensity matrix NE, all of the elements in this matrix are initialized to 0

(2) In the user opinion relation matrix O, O_{ij} represents the comments that the user u_i gives the opinion P_j. With the polarity intensity quantification method in Section 3.1, we can calculate $\text{Senti}(O_{ij})$. If $\text{Senti}(O_{ij}) < 0$, we can set $O_{ij} = -1$, $S_{ij} = Q_{ij}$, which means that the opinion has negative emotion. On the contrary, If $\text{Senti}(O_{ij}) > 0$, we can set $O_{ij} = 1$, $S_{ij} = Q_{ij}$, which means that the opinion has positive emotion

(3) We set $C = A \times Q^T$, which matches the number of user interactions with the users to represent the number of the user interaction. We calculate $\text{NC} = A \times ((Q - |Q|)/2)^T$ to represent the number of negative user interaction

(4) We set $E = A \times S^T$, which matches the intensity of user opinions with the users to represent the emotional intensity of user interaction. We calculate $\text{NE} = A \times ((S - |S|)/2)^T$ to represent the negative emotional intensity of user interaction

(5) For each element C_{ij} in the user interaction number matrix C, if $C_{ij} = 0$, then $N_{ij} = 0$. Otherwise, $N_{ij} = (C_{ij} \times E_{ij})/\text{MAX}\{E_{ik}\}_{k=1,\cdots,m}$, where N_{ij} represents the interaction intensity between the users u_i and u_j. We can see that the more interactive times between users, the more intense the user interaction. The denominator of this formula is the max value of user emotional interaction to all other users, which is used to normalize the formula

(6) For each element NC_{ij} in the user negative interaction number matrix NC, if $\text{NC}_{ij} = 0$, then $\text{NN}_{ij} = 0$. Otherwise, $\text{NN}_{ij} = (\text{NC}_{ij} \times \text{NE}_{ij})/\text{MAX}\{\text{NE}_{ik}\}_{k=1,\cdots,m}$, where NN_{ij} represents the negative interaction intensity between users u_i and u_j

3.3. Reliability Weight Matrix Construction. The $U = \{u_1, u_2, \cdots, u_m\}$ represents the set of users in the network, where m represents the number of users. A symbol network can be divided into a positive network subgraph $g_{\text{p}}(U, E_{\text{p}})$ and a negative one $g_{\text{n}}(U, E_{\text{n}})$, where E_{p} and E_{n} represent the pair of users with a positive link and a negative one, respectively. E_{o} indicates the pair of users without the links, and the negative link prediction needs to construct the negative link sample from the unlabeled $E_{\text{n}} \cup E_{\text{o}}$. $P = \{p_1, p_2, \cdots, p_M\}$ represents a collection of content published by users, and M represents the number of content.

Based on the related user interaction matrix constructed in Section 3.2, the reliability weight matrix W is defined as Equation (2) below:

$$W_{ij} = \begin{cases} f\left(\dfrac{\text{NC}_{ij} \times \text{NE}_{ij}}{\text{MAX}\{\text{NE}_{ik}\}_{k=1,\cdots,m}}\right), & \text{NC}_{ij} \neq 0, \\ r, & \text{NC}_{ij} = 0. \end{cases} \tag{2}$$

3.4. Structural Balance Theory Equation. The basis of the symbol network is the structural balance theory. The balance theory examines the relationship of a triple, which considers that only "the friend's friend is my friend, the enemy's enemy is my friend" is a balanced relationship, and the other is unbalanced. Only the balanced relationship is stable, and the unbalanced relationship has the tendency to transform into a balanced relationship. The triple of the balance theory model is represented as a triangle with three edges, where the plus sign (+) is used to represent the positive relation on an edge, and the minus sign (-) is used to represent the negative one on an edge. The balance of the triangle structure can be determined by the product of three edges. If the product is positive, the structure is balanced. If the product is negative, the structure is unbalanced.

For example, suppose that s_{ij} indicates the relationship between the users u_i and u_j, which can be considered an edge in a triangle. If $s_{ij} = 1$, it indicates that there is a positive relationship between u_i and u_j. In contrast, if $s_{ij} = -1$, it indicates that there is a negative relationship between u_i and u_j. For example, the triple $\langle u_i, u_j, u_k \rangle$ could be balanced when: $s_{ij} = 1$, $s_{jk} = 1$, and $s_{ik} = 1$ or $s_{ij} = -1$, $s_{jk} = -1$, and $s_{ik} = 1$.

For the negative candidate users, we utilize a triple, $\langle u_i, u_j, u_k \rangle$, to determine whether they are the real, active, and existing users. The process is performed as follows: suppose that there are two users, saying u_i and u_k. If there is a third user u_j that is located in the middle of the users u_i and u_k, the triple is constructed. If the production of three edges in such triple does not meet the requirements of the structural balance theory, then we consider that the negative link is unstable and excluded from the negative link candidate set.

To make generalization better, we specify a matrix B. The x_h and x_l represent links $\langle u_i, u_k \rangle$ and $\langle u_j, u_k \rangle$, respectively. If the $\langle u_i, u_j \rangle$ is a positive link, both x_h and x_l are available, then $B_{hl} = 1$. Otherwise, if both x_h and x_l are unavailable, then $B_{hl} = 0$. According to the structural balance theory, if $B_{hl} = 1$, then the x_h and x_l must be the same type of link. Therefore, the balance theory equation is computed as Equation (3) below:

$$\min \frac{1}{2} \sum_{h,l} B_{hl} \left(w^T x_h - w^T x_l\right)^2 = w^T X \ell X^T w, \tag{3}$$

where ℓ is the Laplacian matrix on B. Equation (3) will be inferred in the negative link prediction algorithm of Section 4.3.

4. Negative Link Prediction Algorithm

The link prediction algorithm can be regarded as a classification problem, and the existing links are used as labels to extract features. Unlike the traditional positive link prediction problems, because many online social networks only open positive links to the public, it is necessary to build negative link sample sets firstly, and the accuracy of the sample sets would directly affect the accuracy of negative link prediction. This section firstly describes the construction algorithm of the negative link sample set and then introduces the feature extraction of the negative link. Finally, a negative link prediction algorithm is proposed.

4.1. Negative Link Sample Set Construction Algorithm. In this section, a negative link prediction sample set construction algorithm is proposed based on the methods presented in Section 3 of the sentiment polarity intensity quantification, the user interaction matrix construction, the RW-matrix construction, and the structural balance theory equation.

The basic idea of constructing the negative link sample set is to select negative interaction user pairs as negative link candidate sets from the negative interaction matrix and then use the structural balance theory and RW-matrix to further filter the candidate set to obtain highly reliable negative link samples.

The process of this proposed algorithm is described as follows:

(1) Initialization of the negative link sample set NS

(2) For each element of a negative user interaction matrix NN_{ij}, if the number of user negative interactions is not 0, that is $\mathrm{NN}_{ij} \neq 0$, add such user pair to the negative link sample candidate setNS

(3) The positive link subgraph g_{p} in the network and the negative link in the candidate set NS form a symbol network g; then, the structural balance theory is used to filter the edges in g

(4) For each user pair $\langle u_i, u_j \rangle$ in the candidate set NSand any useru_k which can form a triple set $\langle u_i, u_j, u_k \rangle$, if this triple set cannot satisfy the structural balance theory, $\langle u_i, u_j \rangle$ will be removed from the candidate set NS

(5) For each user pair $\langle u_i, u_j \rangle$ in the candidate set NSand any useru_k which can form a triple set $\langle u_i, u_j, u_k \rangle$, if this triple set can satisfy the structural balance theory, $\langle u_i, u_j \rangle$ will be preserved in the candidate set NS

(6) For each user pair $\langle u_i, u_j \rangle$ in the candidate set NS, if $(\mathrm{NE}_{ij}/\mathrm{MAX}\{\mathrm{NE}_{ik}\}_{k=1,\cdots,m}) < 0.5$ which indicates that the negative sentiment intensity of the user u_i to u_j is less than half of the maximum negative emotional intensity of the user u_i, to all other users, $\langle u_i, u_j \rangle$ will be removed from the candidate set NS

So far, we can obtain highly reliable negative link samples.

4.2. Feature Extraction of the Negative Link. From the negative link samples, we can extract features. The features of the negative links can be divided into the following categories: user features, user-user pair features, and symbol features. We explain such features in detail.

(1) User Features. It is extracted from each user node. The features of the user u_i include the following information: the in-degree or out-degree of the positive link, the number of the triples which contain the user u_i, the amount of content published by the user u_i, the positive or negative opinions to the content published by the user u_i, and the opinions that the user$u_i$have to other users.

(2) User-User Pair Features. It is used for extracting features from each pair of users $\langle u_i, u_j \rangle$. The extracted features include the following: the number of positive or negative interactions between u_i and u_j and between u_j and u_i, the Jaccard coefficient of the in-degree or out-degree between u_i and u_j, the shortest path between u_i and u_j, and the average value in respect to the sentiment intensity of positive or negative interactions between u_i and u_j.

(3) Symbol Features. The symbol network g is composed of the positive link subgraph g_{p} and the negative link sample set NS, where the weight of the positive link is 1. The weight of the negative link is obtained from the reliability weight matrix. The symbol features for each pair of users include the following: the weighted in-degree or weighted out-degree of the negative links of u_i and u_j, the Jaccard coefficient of the in-degree or out-degree of the negative links of u_i and u_j, and the features of 16 weighted triples proposed by Leskovec et al. [18].

These three types of features are represented by F1, F2, and F3, respectively. In order to obtain the influence of different features on the accuracy of prediction, the importance of each feature can be determined by gradually increasing the feature and by observing the change of the accuracy rate after the new features are added to the original feature set. The detailed experiment data is analyzed in Section 5.3.

4.3. Negative Link Prediction Algorithm. Previously, we obtain a highly reliable negative link sample set by selecting users with negative interactions from the user interaction matrix and utilizing structural balance theory and reliability matrix to filter out useless candidate sets. Meanwhile, feature extraction identifies the features required for the classification. Next, we should select the appropriate classifier to carry out the negative link prediction. Since some noises would be introduced to the construction of a negative link sample, the classifier should have the ability to tolerate noise. Here, the soft interval support vector machine is chosen as the classifier, which is proved to have better noise tolerance. Since the soft interval support vector machine (SVM) has the ability to tolerate noise, we introduce the SVM as the classifier.

Let $\chi = \{x_1, x_2, \cdots, x_N\}$ be a collection of pairs of users in $E_{\mathrm{n}} \in E_{\mathrm{o}}$, where X_i represents the feature vector of the user x_i,

E_n represents the pair of users with the negative link, and E_o indicates the pair of users without the link. The SVM in its standard form in the negative link prediction is given as Equation (4) below:

$$\begin{aligned} &\min_{w,b,\varepsilon} \frac{1}{2} PwP^2 + C \sum_{x_i \in \mathrm{PS} \cup \mathrm{NS}}^{N} \varepsilon_i \\ &\text{s.t.} \quad y_i\left(w^T x_i + b\right) \geq 1 - \varepsilon_i, \quad x_i \in \mathrm{PS} \cup \mathrm{NS} \\ &\varepsilon_i \geq 0, \end{aligned} \tag{4}$$

where ε_i, a slack variable, represents the noise tolerance of the training samples, and C is the penalty parameter ($C > 0$). Notice that the larger the penalty parameter C, the more the errors obtained in the classification penalty, and vice versa. In the negative link prediction algorithm, the noise levels of the positive and negative link samples are different because the positive link sample is trusted in the network and the negative is not, which is inferred from the prediction. Therefore, the slack variables C_p and C_n are introduced in the positive and negative links, respectively, to control the noise. Since the reliability of the negative link is measured by its reliability weight matrix, we introduce the slack variable c_j and set the negative link of $\langle u_i, u_j \rangle$ as x_i. When the c_j is used to control the noise of x_i, Equation (4) is updated to Equation (5) as follows:

$$\begin{aligned} &\min_{w,b,\varepsilon} \frac{1}{2} PwP^2 + C_p \sum_{x_i \in \mathrm{PS}}^{N} \varepsilon_i + C_n \sum_{x_i \in \mathrm{NS}}^{N} c_j \varepsilon_j \\ &\text{s.t.} \quad y_i\left(w^T x_i + b\right) \geq 1 - \varepsilon_i, \quad x_i \in \mathrm{PS} \\ &\qquad y_j\left(w^T x_j + b\right) \geq 1 - \varepsilon_j, \quad x_j \in \mathrm{NS} \\ &\varepsilon_i \geq 0, \quad \varepsilon_j \geq 0. \end{aligned} \tag{5}$$

The balance theory Equation (3) is introduced into the negative link prediction, and the slack variable C_b is given to the balance theory equation to control the error. Then, we can get the updated Equation (6) as follows:

$$\begin{aligned} &\min_{w,b,\varepsilon} \frac{1}{2} PwP^2 + C_p \sum_{x_i \in \mathrm{PS}}^{N} \varepsilon_i + C_n \sum_{x_i \in \mathrm{NS}}^{N} c_j \varepsilon_j + \frac{C_b}{2} w^T X 1 X^T w \\ &\text{s.t.} \quad y_i\left(w^T x_i + b\right) \geq 1 - \varepsilon_i, \quad x_i \in \mathrm{PS} \\ &\qquad y_j\left(w^T x_j + b\right) \geq 1 - \varepsilon_j, \quad x_j \in \mathrm{NS} \\ &\varepsilon_i \geq 0, \quad \varepsilon_j \geq 0. \end{aligned} \tag{6}$$

Equation (6) is an optimization problem with respect to the inequality constraints. We can transform such optimization problem into a dual form. Since w can be represented as $w^* = \sum a_i K(x_i, x)$ by the inner production of eigenvectors,

Table 2: The statistical result of the Epinions dataset.

Category	Number
Number of users	14765
Number of positive links	272513
Number of negative links	52704
Total number of texts	612321
Number of positive comments on the text	6937986
Number of negative comments on the text	163502

Equation (6) can be updated to Equation (7) as follows:

$$\begin{aligned} &\min_{a,b,\varepsilon} \frac{1}{2} a^T K a + C_p \sum_{x_i \in \mathrm{PS}}^{N} \varepsilon_i + C_n \sum_{x_i \in \mathrm{NS}}^{N} c_j \varepsilon_j + \frac{C_b}{2} a^T K \ell K^T a \\ &\text{s.t.} \quad y_i(a_k K(x_k x_i) + b) \geq 1 - \varepsilon_i, \quad u_i \in \mathrm{PS} \\ &\qquad y_j\left(a_k K\left(x_k, x_j\right) + b\right) \geq 1 - \varepsilon_j, \quad u_j \in \mathrm{NS} \\ &\varepsilon_i \geq 0, \quad \varepsilon_j \geq 0. \end{aligned} \tag{7}$$

In Equation (7), K is the Gram matrix of the samples. If we set S_i as Equation (8), we have

$$S_i = \begin{cases} C_p, & x_i \in \mathrm{PS}, \\ C_n c_i, & x_i \in \mathrm{NS}. \end{cases} \tag{8}$$

Then, when the Lagrange multipliers β and γ are factored into consideration, the Lagrange function of Equation (7) is expressed as Equation (9) below:

$$\begin{aligned} L(w, b, \varepsilon, a, \gamma) = {} & \frac{1}{2} a^T \left(K + C_b K \ell K^T\right) a + \sum_{i=1}^{l} s_i \varepsilon_i - \sum_{i=1}^{l} \beta_i \\ & \cdot \left[y_i \left(\sum_k a_k K(x_k, x_i) + b \right) - 1 + \varepsilon_i \right] - \sum_{i=1}^{l} r_i \varepsilon_i. \end{aligned} \tag{9}$$

Take the derivative of b and ε, respectively, to get Equation (10) as follows:

$$L(a, \beta) = \frac{1}{2} a^T \left(K + C_b K \ell K^T\right) a - a^T K J^T Y \beta + \sum_{i=1}^{l} \beta_i. \tag{10}$$

where $J = [I, 0]$, Y is a diagonal matrix of $l \times l$, and l is composed of positive and negative link samples.

When we compute the derivative of a and substitute Equation (10), the optimization problem is transformed into a

Table 3: The meaning and testing value setting of key parameters in the experiment.

	Meaning	Value
C_n	Error control for the negative sample	0, 0.001, 0.01, 0.05, *0.10*, 0.50, 1.0
c_j	Error control for the negative sample X_j	$f(x) = \begin{cases} 0, 0.01, 0.05, 0.10, 0.25 \\ 0.5, 0.75, 1, 1 - 1/\log(1+x) \end{cases}$
C_b	Regularization error control for the balance theory	0, 0.001, 0.01, 0.05, *0.10*, 0.50, 1.0

Lagrange function as Equation (11) below:

$$\max_{\beta} \sum_{i=1}^{l} \beta_i - \frac{1}{2}\beta^T Q \beta$$
$$\text{s.t.} \quad \sum_{i=1}^{l} \beta_i y_i = 0 \tag{11}$$
$$0 \le \beta_i \le s_i,$$

where $Q = YJK(I + C_b \ell K^T)^{-1} J^T Y$.

The process of the negative link prediction is illustrated as follows:

(1) Choose the SVM as the classifier, with Equation (4)

(2) Considering the noise control of positive and negative samples, different error control variables, C_p and C_n, are assigned to the positive and negative samples, respectively

(3) For the negative link sample x_i, a variable c_j is introduced to further control the error. $C_j = W_{ik}$ and W is the RW-matrix, with Equation (5)

(4) The structural balance theory is introduced into the negative link prediction, where the slack variable C_b is given to the balance theory equation to control the error, with Equation (6)

(5) With a series of deduction, an optimization problem is transformed into a Lagrange function following Equations (7)–(10), finally obtaining the negative link prediction model as Equation (11)

5. Results and Analysis

We present an in-depth discussion of our proposed negative link prediction algorithm. Section 5.1 describes the dataset used in the experiments, Section 5.2 explains the experimental platform, and Section 5.3 analyzes the experimental results.

5.1. Experimental Dataset. The Epinions is used as a dataset for experimental evaluation. It is an open commodity review website, which allows users to evaluate the commodity, make comments on statements from other users, and rate the trust or distrust of users. In addition, it also contains the following relationships: positive-negative relationship, users-content attributions relationship, and users-user rates relationship.

Table 4: Classification results of different feature sets under different classifiers.

Feature set	Classifier	Accuracy	Recall ratio	F1 score
F1	SVM	0.20	0.19	0.0195
	Naive Bayes	0.18	0.17	0.175
F1+F2	SVM	0.22	0.22	0.22
	Naive Bayes	0.20	0.20	0.20
F1+F2+F3	SVM	0.25	0.24	0.245
	Naive Bayes	0.22	0.21	0.215

The statistical results of the Epinions dataset are shown in Table 2.

For the data in Table 2, they need to be processed and filtered, and the users who have no positive or negative interactions with others need to be removed. By the way, the data quality indicators [20], such as accuracy, timeliness, completeness, and consistency, are a good choice for evaluating the quality of data for experimental data. Meanwhile, it should be noted that the negative link in Table 2 is only used as an experimental result analysis and for comparison, not for the training and classification of the negative link prediction model.

5.2. Experimental Setting. We mainly evaluate our proposed negative link prediction algorithm with three key parameters involved in the negative link prediction algorithm: C_n, c_j, and C_b. When testing one parameter, the remaining parameters are kept as default values. The detailed information with respect to each parameter is presented in Table 3, where the italicized data is the default value.

In order to evaluate the significance of our proposed algorithm, we choose four groups of baseline algorithms for comparisons, which are described as follows:

(1) Random Algorithm. This is the baseline algorithm in the general link prediction. The links in the network are randomly marked as negative, which indicates that the sampling set of the negative link is generated at random.

(2) The Shortest Path Algorithm. The shorter the shortest path between nodes in the network, the more likely there is a link. In addition, it might have a connection for the nodes with a distance of less than threshold 2. The algorithm considers the nodes as candidates when these nodes do not have a link or the shortest distance threshold is not greater than 2. Nodes in the candidate set are marked as negative links.

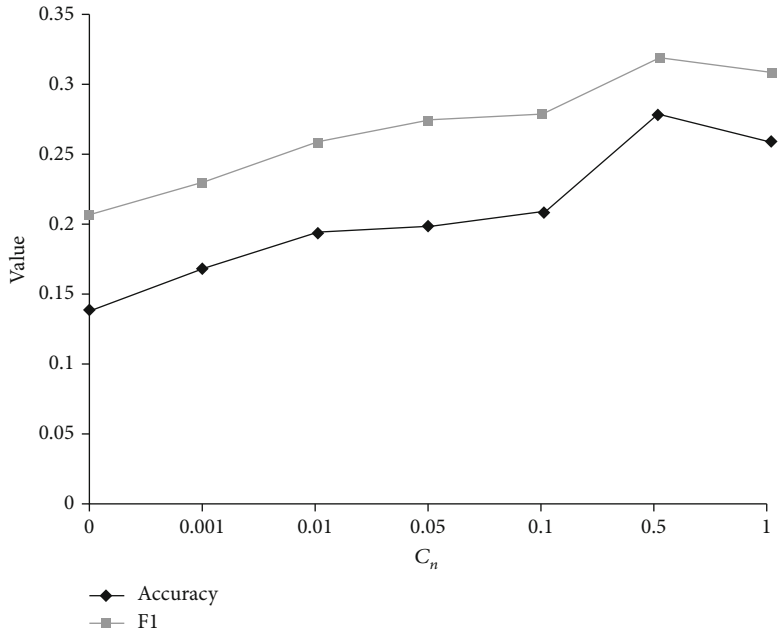

Figure 1: Influence of C_n on the predictive performance.

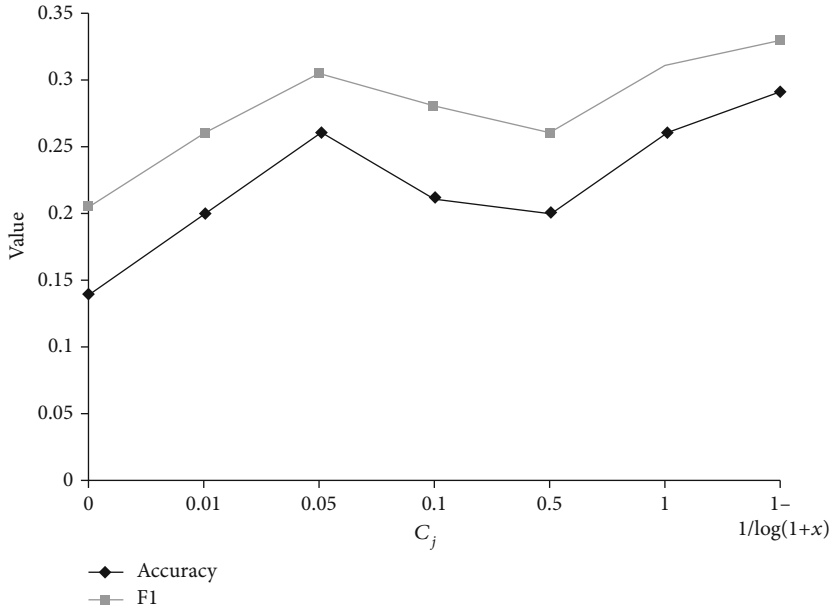

Figure 2: Influence of c_j on the predictive performance.

(3) Negative Interaction Determination Algorithm. Since there is a strong positive relationship between negative user interactions, the nodes are joined into the negative link candidate set when they have negative interactions. Then, nodes in the candidate set are marked as negative links when they are in pairs.

(4) Balanced Negative Interaction Determination Algorithm. Based on the negative interaction determination algorithm, the balance theory is introduced to filter out useless users. In other words, if the nodes do not meet the requirements of the balance theory, they will be dropped. Otherwise, they are added to the negative link candidate set. Then, nodes in pairs are marked as negative links.

5.3. Experimental Analysis

5.3.1. Experiment of Negative Link Feature Set Classification. Three types of features, viz., user features, user-user pair features, and symbol features, are represented as F1, F2, and F3, respectively. The meanings of these features are illustrated in Section 4.2.

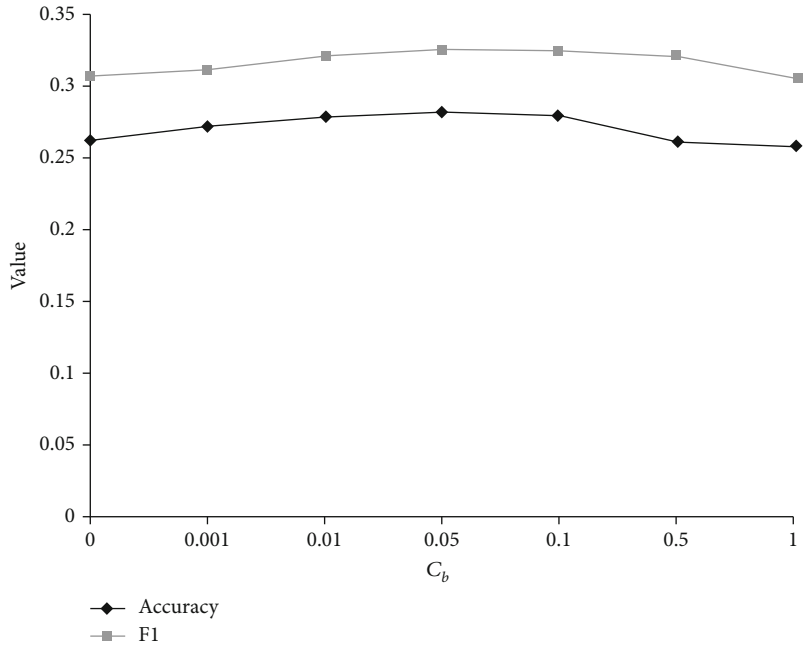

Figure 3: Influence of C_b on the predictive performance.

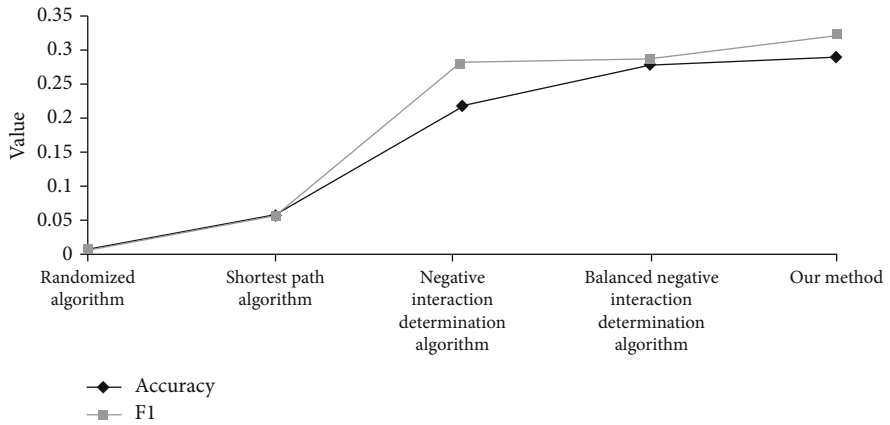

Figure 4: Comparison of the negative link prediction algorithm and the prediction reference methods.

In order to obtain the influence of different features on the prediction accuracy, w adopts a stepwise increasing feature method. For example, the first feature set (F1) contains only user features. The second feature set (F1+F2) contains user features and user-user pair features. And the third feature set (F1+F2+F3) contains all the features. By adding the new features to the original feature set, we can observe the changes in the classification accuracy to determine the importance of each feature.

We use SVM and Naive Bayes as the classifiers. The classification results on the real Epinions dataset are shown in Table 4.

From Table 4, we can find that SVM can achieve higher accuracy, which indicates that the SVM classifier is more suitable for negative link prediction. Furthermore, on each classifier, the F1 value increases consecutively as the feature increases, which indicates that these three types of features are helpful in the classification. More specifically, the F1 value has the fastest growth when adding the symbol features. It means that the symbol features play a crucial role in the classification results in the negative link prediction. The experimental analysis demonstrates that the RW-matrix proposed in our algorithm is reasonable and feasible.

5.3.2. Experiment of Key Parameters in the Negative Link Prediction Algorithm. We use C_n, c_j, and C_b as the key parameters to evaluate the algorithm. The experiment is performed by using the control variable method. When one parameter is tested, the other parameters are kept as the default values.

(1) Negative Sample Error Control Parameter Cn. The experimental results are shown in Figure 1. With the change of C_n, the accuracy rate and the F1 value are with a process that first

rises, gradually stabilizes, and then decreases. The peak value is reached when $C_n = 0.5$. When $C_n = 1$, the positive sample error control parameter should be $C_p = 1$; then, the positive samples and negative samples are the same as error control coefficients. The decreases in the accuracy rate and F1 value indicate that the negative sample should be given a different error control coefficient.

(2) Error Control Parameter cj for the Negative Sample xj. The negative link $\langle u_i, u_j \rangle$ is recorded as a negative sample x_j, $c_j = W_{ij}$ is the error control parameter of x_j, and W is the reliability weight matrix (RW-matrix). The experimental results are shown in Figure 2, and $c_i = f(x) = 1 - (1/\log(1 + x))$ is a better function identified by the previous researchers, which has also achieved higher accuracy in this experiment. Due to a direct relationship between the number of negative interactions and the number of negative links, when $c_j = 0$, the negative link prediction without considering the number of negative interactions is much less accurate. When x_j takes other constants, the accuracy rate decreases to a certain degree compared with $c_i = f(x) = 1 - (1/\log(1 + x))$. This indicates that the RW-matrix can really reflect the reliability of the negative link.

(3) Regularization Error Control Parameter Cb of the Structural Balance Theory. The experimental results are shown in Figure 3. With the change of C_b, the accuracy rate and the F1 value both rise firstly and gradually decrease afterwards. $C_b = 0$ to $C_b = 0.001$ is a significant improvement in accuracy, indicating that the regularization equation of the structural balance theory can improve the performance of negative link prediction. The middle segment remains relatively stable, and the accuracy gradually decreases with the increase of C_b. Such results indicate that the weights should be selected appropriately; otherwise, too large weights would reduce the accuracy rate.

5.3.3. Experiment of the RWSBT-NLP Algorithm and Baseline Algorithm Comparison. The negative link prediction algorithm is compared with four prediction reference methods, illustrated in Section 5.2. The experimental results are shown in Figure 4. The performance of the random algorithm is the worst since the negative link accounts for a small proportion of the overall network. The shortest path algorithm has a larger improvement than the random algorithm, which indicates that the negative link is more likely to exist in the network at a very close distance. The accuracy of the negative interaction determination algorithm has increased dramatically, which indicates that there is a strong link between negative interactions and negative links. The balanced negative interaction determination algorithm improves the accuracy rate compared to the negative interaction determination algorithm, which means that the balance theory does improve the accuracy rate by removing some points that do not meet the balance theory. The accuracy of our proposed negative link prediction algorithm is slightly higher than that of the balanced negative interaction determination algorithm. It indicates that when the negative link prediction algorithms take the sentiment characteristics into account, they can improve the accuracy of the prediction and have a good performance.

6. Conclusions

This paper focuses on the problem of negative link prediction in symbol networks. We propose a negative link prediction algorithm by using the sentiment analysis and structural balance theory. The sentiment analysis is mainly embodied in the construction of the user interaction matrix based on the calculation of the sentiment intensity of social network texts. Based on the user interaction matrix, we construct the reliability weight matrix (RW-matrix). Then, based on the structural balance theory and constructed RW-matrix, we propose the negative link prediction algorithm by building the negative link sample set and extracting the features. With the experiments and conductions with the real dataset, the influence of each parameter on the accuracy of the prediction results has been analyzed based on the control variable method. Compared with the existing predictive algorithms, the proposed negative link prediction algorithm can improve the accuracy of prediction dramatically with good performances.

Data Availability

The Epinions dataset used to support the findings of this study can be available from http://www.trustlet.org/epinions.html.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Acknowledgments

Thanks are due to QuanFeng YAO, Xiang HU, and JiWei HU for their help. This work was supported by the Social Science Fund Planning Project of the Ministry of Education of the People's Republic of China "Research on Data Service and Guarantee for the Fourth Paradigm of Social Science" (20YJA870017).

References

[1] H. Cheng, N. Xiong, A. V. Vasilakos, L. Tianruo Yang, G. Chen, and X. Zhuang, "Nodes organization for channel assignment with topology preservation in multi- radio wireless mesh networks," *Ad Hoc Networks*, vol. 10, no. 5, pp. 760–773, 2012.

[2] H. Cheng, Z. Su, N. Xiong, and Y. Xiao, "Energy-efficient node scheduling algorithms for wireless sensor networks using Markov random field model," *Information Sciences*, vol. 329, pp. 461–477, 2016.

[3] M. H. Miraz and M. Ali, "Applications of blockchain technology beyond cryptocurrency," 2018, https://arxiv.org/abs/1801.03528.

[4] O. Tanevski, I. Mishkovski, and M. Mirchev, "Link prediction on Bitcoin OTC network," 2020.

[5] B. Jeong, J. Lee, and H. Cho, "Improving memory-based collaborative filtering via similarity updating and prediction modulation," *Information Sciences*, vol. 180, no. 5, pp. 602–612, 2010.

[6] J. Tang, H. Gao, X. Hu, and H. Liu, "Exploiting homophily effect for trust prediction," in *Proceedings of the Sixth ACM International Conference on Web Search and Data Mining*, pp. 53–62, New York, 2013.

[7] G. Yuan and K. Pradeep, "Exploiting sentiment homophily for link prediction," in *Proceedings of the 8th ACM Conference on Recommender Systems*, pp. 17–24, New York, 2014.

[8] J. Tang, X. Hu, Y. Chang, and H. Liu, "Predictability of distrust with inter-action data," in *Proceedings of the 23rd ACM International conference on Conference on Information and Knowledge Management*, pp. 181–190, New York, 2014.

[9] H. Ma, M. Lyu, and I. King, "Learning to recommend with trust and distrust relationships," in *Proceedings of the Third ACM Conference on Recommender Systems*, pp. 189–196, New York, 2009.

[10] J. Cho, "The mechanism of trust and distrust formation and their relational outcomes," *Journal of Retailing*, vol. 82, no. 1, pp. 25–35, 2006.

[11] P. Sharma, U. K. Singh, T. V. Sharma, and D. Das, "Algorithm for prediction of links using sentiment analysis in social networks," in *Proceedings of the 7th International Conference on Computing Communication and Networking Technologies*, pp. 110–116, New York, 2016.

[12] K. Jérôme, A. Lommatzsch, and C. Bauckhage, "The Slashdot Zoo: mining a social network with negative edges," in *Proceedings of the 18th International Conference on World Wide Web*, pp. 741–750, New York, 2009.

[13] K. Y. Chiang, C. J. Hsieh, N. Natarajan, I. S. Dhillon, and A. Tewari, "Prediction and cluster-ing in signed networks: a local to global perspective," *Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1177–1213, 2013.

[14] D. Liben-Nowell and J. Kleinberg, "The link-prediction problem for social networks," in *Proceedings of the Twelfth International Conference on Information and Knowledge Management*, pp. 556–559, New York, 2009.

[15] H. H. Song, T. W. Cho, V. Dave, Y. Zhang, and L. Qiu, "Scalable proximity estimation and link prediction in online social networks," in *Proceedings of the 9th ACM SIGCOMM Conference on Internet Measurement*, pp. 322–335, New York, 2009.

[16] M. Fire, L. Tenenboim, O. Lesser, R. Puzis, L. Rokach, and Y. Elovici, "Link prediction in social networks using computationally efficient topological features," in *2011 IEEE Third International Conference on Social Computing*, pp. 73–80, New York, 2011.

[17] N. Benchettara, R. Kanawati, and C. Rouveirol, "Supervised machine learning applied to link prediction in bipartite social networks," in *Proceedings of the 2010 International Conference on Advances in Social Networks Analysis and Mining*, pp. 326–330, New York, 2010.

[18] J. Leskovec, D. Huttenlocher, and J. Kleinberg, "Predicting positive and negative links in online social networks," in *Proceedings of the 19th International Conference on World Wide Web*, pp. 641–650, New York, 2010.

[19] J. Kunegis, J. Preusse, and F. Schwagereit, "What is the added value of negative links in online social networks?," in *Proceedings of the 22nd International Conference on World Wide Web*, pp. 727–736, New York, 2013.

[20] H. Cheng, D. Feng, X. Shi, and C. Chen, "Data quality analysis and cleaning strategy for wireless sensor networks," *EURASIP Journal on Wireless Communications and Networking*, vol. 2018, no. 1, 11 pages, 2018.