**ORIGINAL ARTICLE**

# Multiplex network embedding for implicit sentiment analysis

Xiaofang Zhao[1] · Yuhong Liu[2] · Zhigang Jin[1]

## Abstract

As one of the hot research directions in natural language processing, sentiment analysis has received continuous and extensive attention. Different from explicit sentiment words indicating sentiment polarity, implicit sentiment analysis is a more challenging problem due to the lack of sentiment words, which makes it inadequate to use traditional sentiment analysis method to judge the polarity of implicit sentiment. This paper takes sentiment analysis as a special sign link prediction problem, which is different from traditional text-based method. In particular, by performing the word graph-based text level information embedding and heterogeneous social network information embedding (i.e. user social relationship network embedding, and user-entity sentiment network embedding), the proposed scheme learns the highly nonlinear representations of network nodes, explores early fusion method to combine the strength of these two types of embedding modeling, optimizes all parameters simultaneously and creates enhanced context representations, leading to better capture of implicit sentiment polarity. The proposed method has been examined on real-world dataset, for implicit sentiment link prediction task. The experimental results demonstrate that the proposed method outperforms state-of-the-art schemes, including LINE, node2vec, and SDNE, by 20.2%, 19.8%, and 14.0%, respectively, on accuracy, and achieves at least 14% gains on AUROC. For sentiment analysis accuracy, the proposed method achieves AUROC of 80.6% and accuracy of 78.3%, which is at least 31% better than other models. This work can provide useful guidance on the implicit sentiment analysis.

**Keywords** Sentiment analysis · Network embedding · Word co-occurrence · Heterogeneous social network · Link prediction

## Introduction

The unstructured text generated by users, who share feelings and express attitudes in social media, contains users' behaviors in real life, and forms the sentiment links between these users and entities (e.g. commodities, movies or friends). These sentiment links between users and entities contains a lot of available information, which has been successfully applied in many tasks, such as intention mining [1], personalized recommendation system [2, 3] and so on.

However, in online social network platforms, the contents published by users are usually short sentences, clauses or phrases, and because people express their feelings or opinions in various ways, they can use explicit emotional words or implicit ones. In this case, it is inadequate to use traditional sentiment classification algorithms to judge the polarity of sentiment relationship among uses and entities. Therefore, according to the expression of subjective and objective emotion in texts, Liu [4] divides emotion into sub-objective opinion and fact-implied opinion, the former is the subjective statement of emotional tendency or opinion, while the latter is the implicit expression of emotion through objective statement. Liao et al. [5] defined implicit sentiment sentence as "a language segment that expresses subjective emotions but does not contain explicit affective words".

Most current user sentiment analysis literature focuses on making a sentiment decision according to users' text descriptions. However, on online social media, most people express their attitudes on different entities with very limited words (e.g., Sina Weibo, Twitter), where explicit sentiment

✉ Zhigang Jin
  zgjin@tju.edu.cn

  Yuhong Liu
  yhliu@scu.edu

[1] School of Electrical and Information Engineering, Tianjin University, Tianjin, China

[2] Department of Computer Science and Engineering, Santa Clara University, Santa Clara, USA

polarity words are missing. In such cases, the sentiment polarity is likely to be hidden and not obvious from the short text. Furthermore, different context semantic backgrounds can result in different sentiment polarity. In these situations, traditional sentiment analysis methods often fail to retrieve users' hidden real attitudes, making it critical to be able to capture implicit sentiment, especially sentiment from the limited content published by users. Unfortunately, there is very little prior work focusing on implicit sentiment analysis for online social media. Different from explicit sentiment words indicating sentiment polarity, implicit sentiment analysis task is more difficult mainly because of the following reasons:

1. From the perspective of linguistic and words, different context semantic backgrounds can result in different sentiment polarity; and the implied text does not contain sentiment words. The above two factors make it challenging for the bag-of-words model-based text representation methods to effectively express the semantics of sentences, and the acquisition of semantic features is more complicated. In this case, it is unrealistic to extract the user's sentiment towards the entity by using the traditional sentiment classifiers.

2. Existing methods [12–14] focus on extracting features from discourse information and use the structural relationship of text data to construct graph, such as word co-occurrence, syntactic relationship, and context relationship and so on, and train classifiers to predict sentiment polarity. However, users' social structure and the user-entity-aware information, which exert significant impacts on the formation of social networks' sentiment links, are often ignored.

3. Network embedding methods aim to learn the low dimensional potential representation of nodes in the network [20–26], and these methods have achieved significant performance in the task of link prediction. However, most of these methods only focus on predicting whether there are links between node pairs, while cannot retrieve whether there are negative links or positive links between node pairs. So these methods cannot be directly applied to the field of sentiment analysis.

Text semantic information and social network information are important sources for learning node representations in the task of implicit sentiment analysis. However, there is no relevant research to analyze the hidden sentiment polarity expressed by users to entities with very limited words in online social networks (such as Sina Weibo and twitter). To our best knowledge, this is the first work that models sentiment polarity relationship as sentiment links between users and entities, and propose a novel multiplex network embedding framework based on word

graph-based text level information and the heterogeneous social network information (user social relationship network, user-entity sentiment network) to extract nodes' low dimensional embedding representation and further to predict the sign of implicit sentiment links.

The proposed MEISP model adopts deep autoencoders to map each user structure, sentiment network and text structure information into a low-dimensional vector space, in this process, nodes' highly nonlinear representations from these three networks can be preserved. The learned node representations are fused together by concatenation aggregation and nonlinear fusion for further sentiment link prediction. First of all, because both the user social relationship network embedding and the sentiment network embedding are equally important to the final representation, this paper aggregate these two embeddings into a final social network level embedding through a concatenation function. Then an early fusion method is used to fuse the text-based information and social network-based information. Through the nonlinear fusion of social information and text contents information, a powerful network embedded representation learning framework is formed. Furthermore, the joint optimization of these components can enhance the learning representation performance of network nodes, and solves the problem of implicit sentiment analysis.

As far as we know, this is the first work that studies both social network-based and text-based multiplex network embedding for implicit sentiment mining and analysis. The main contributions of this work are summarized below:

• This paper formulates and elaborates the problem of sentiment analysis in social networks, that is, to find the sentiment tendency hidden in user comments.

• In this paper, the implicit sentiment analysis problem is formally defined as a sentiment sign link prediction problem, which considers the comprehensive context at both the text level and the social network relation level. Based on this, a holistic sentiment polarity analysis scheme is designed to reveal users' real attitude hidden in short texts.

The rest of this paper is organized as follows. Section "Related works" reviews the related works from two aspects of sentiment analysis and network embedding. Section "Problem formulation" introduces the proposed formal definition of the implicit sentiment analysis problem. Section "Multiplex network embedding" proposes MEISP model and shows how multiplex information network embedding is handled to perform implicit sentiment link predictions. Section "Experiment" carries out a variety of experiments and compared with some well-known network embedding methods and sentiment analysis algorithms.

# Related works

Since, this work aims to propose an advanced network embedding framework for implicit sentiment analysis, this section summarizes related works from two aspects: sentiment analysis and network embedding.

## Sentiment analysis

Sentiment analysis is one of the most important tasks in the field of natural language processing. The existing text-based studies of sentiment classification can be mainly grouped into two categories: lexicon-based and corpus-based approaches. Lexicon-based approaches mainly use the sentiment polarity associated with the sentiment dictionary to calculate the sentiment polarity of each sentence or document. Corpus-based approaches [6–8] take sentiment classification as a special case of text categorization problem, which utilize machine learning methods to extract reasonable features from texts and feed them into a classifier to predict sentiments. Nowadays, more people are used to express their attitudes on different entities in online social networks, forming user to entity sentiment links. These sentiment links imply positive or negative semantics. Most of current user sentiment analysis literature focuses on making a positive, neutral, or negative sentiment decision according to users' text descriptions. And more and more attention has been paid to the task of user-entity sentiment analysis. For example, Li et al. [9] integrate both named entity information and sentiment level information together to form label sequences and adopt an approach based on graphical models. Gan et al. [10] propose a self-attention based hierarchical dilated convolutional neural network for multi-entity sentiment analysis. Ding et al. [11] design an entity-level sentiment analysis tool consisting of sentiment classification and entity recognition, which can classify issue comments into < sentiment, entity > tuples. However, these above entity sentiment analysis are mainly rely on text content, and all contain explicit words. Compared with explicit sentiment analysis, the lack of sentiment words in sentences makes the expression of sentiment more euphemistic and these traditional sentiment analysis methods often fail to retrieve users' hidden real attitudes, making the implicit sentiment analysis more challenging.

Existing works of implicit sentiment analysis mainly focuses on the discrimination of implicit sentiment sentences. To capture implicit structures, recent studies show promising results by learning word embedding with neural language models, such as word2vec [41] model. The embedded word vector obtained by these methods has better representation and reasoning ability in the semantic space and can be used as the input of various deep learning models, such as LSTM, and CNN models. Kauter et al. [12] propose a new fine-grained method to identify explicit and implicit sentiment in financial news and reviews. Liao et al. [5] propose a multi-level semantic fusion method based on representation learning, which can learn and fuse three different levels (i.e. words, sentences and documents) of features to recognize factual implicit sentiment sentences, however, this model is limited by grammatical structure. Bi-directionally long short-term memory [13] model can effectively capture the semantics of long dependent sequences, and assign different weights to words in sentences automatically via introduction of attention mechanism [14, 15]. Wei et al. [16] propose an implicit sentiment classification model based on multi-polarization orthogonal attention mechanism, which can simulate the differences between vocabulary and specific sentiment polarity attention effectively, and improve the performance of implicit sentiment classification. Zuo et al. [17] propose a context-based heterogeneous graph convolution network model, which first regards the context of the whole document as a heterogeneous graph to maintain its dependency structure, and then employs graph convolution network to obtain the features of implicit sentiment sentences and context.

However, existing implicit sentiment analysis often faces three major challenges: (1) there is a lack of sentiment words; (2) the words are relatively objective and neutral; and (3) the words' sequences, co-occurrence relationship are neglected. The traditional bag of words-based models, however, can neither accurately capture the meaning and text structure of words in the text, nor effectively represent the semantics of a sentence. In addition, since words are organized into text content by forming phrases, sentences and so on to express users' opinion and the different positions of words in a sentence will have a great impact for a detailed understanding of the text.

Graph-based text representation method is one of the effective methods to solve the above problems. In the graph, nodes represent features and edges represent the relationship between different nodes. Although there are various graph-based text representation models, such as word graphs and ngram-graphs [18], to better capture the inherent characteristics of text in social media, this paper uses word co-occurrence graph to represent the relationship between words in short text content.

Representing the text in a graph structure that considers all the information for sentiment classification has achieved excellent performance and has attracted considerable attention in recent years. For example, Bijari et al. [19] propose to represent sentences via graph scheme and Node2vec

embedding is used as feature learning algorithms to extract the text features. Zhao et al. [20] design a bidirectional attention mechanism with position encoding to capture the aspect-specific representations. Gui et al. [21] propose to build a heterogeneous network by constructing the relationship among words, users and products.

On social networks, users with similar hobbies and concepts often form personalized social communities. For example, Xiao et al. [22] and Zhao et al. [23] both consider user structure and model the social relationship of users. In addition, part of the sentiment prediction work also considers user-product-aware information, for example, user product neural network (UPNN) [24], user product attention (UPA) [25], user product deep memory network (UPDMN) [26] and hierarchical user attention and product attention neural network (HUAPA) [27] have achieved excellent performance in sentiment analysis by considering internal and external connections of users and products.

These above researches demonstrate that both the representation of a text in a graph structure, user social structure and the connections between users and entities are effective for sentiment analysis and opinion mining. These studies inspire this work to take into account both aspects in the proposed framework.

## Network embedding

The purpose of network embedding is to represent high-dimensional and sparse vector space with low dimensional and dense vector space. The learned features can be used in machine learning tasks such as classification, regression, clustering and so on. Network embedding methods mainly include matrix eigenvectors calculation-based methods and neural network-based methods. Locally linear embedding (LLE) [28] and Laplacian Eigenmaps (LE) [29] are conventional matrix eigenvectors calculation-based methods. Nowadays, neural network-based embedding methods, such as DeepWalk [30], which inspired by the neural network language model Word2vec [31] achieves excellent performance and has been widely used. LINE [32] proposes to establish the first- and second-order proximity between nodes, and can be used in large networks. Node2vec [33] designs a biased random walk procedure to learn a mapping of nodes that maximizes the likelihood of preserving network neighborhoods of nodes. SDNE [34] uses auto-encoder to capture network structures and learn user representations.

The above researches are network embedding models and methods. In addition, there are also some literatures focusing on embedding representation algorithms for different entities. For example, Zhang et al. [35] propose a new optimization model to capture the hidden relationship between item content features for cold-start and content-based

recommendations. He et al. [36] propose the bipartite graph neural network (BGNN), which consists of two central operations as interdomain message passing and intra-domain alignment to learn the embedding representations of user and item nodes. Wang et al. [37] propose SHINE model, which utilizes deep neural network structure to learn the low-dimensional representation of nodes to predict sentiment links between users and celebrities. Yuan et al. [38] propose to fuse multi-networks information to learn the nodes' low-dimensional embedding representation for user behavior classification.

The above studies on network embedding mainly focus on the field of recommender systems and user behavior analysis. The proposed scheme is different from these studies in two aspects. First, this work utilizes network embedding to address a different problem. In particular, this paper models the implicit sentiment analysis problem from special link prediction angle and explore user-entity modeling to find the implicit sentiment in social network comments. Second, this different problem raises new challenges that are not addressed in existing studies. In particular, implicit sentiment analysis requires the proposed embedding approach to be able to capture rich information from different networks, such as user social network, user-entity sentiment network, and word graph-based text level comments. These networks are highly heterogeneous. To address these challenges, a novel multiplex network embedding method called MEISP is proposed in this work. Specifically, deep neural network architecture is introduced for embedding these heterogeneous networks, and a multilayer perceptron structure-based early fusion method is adopted to fuse the text information and social network information. Furthermore, the proposed scheme can also extract nodes' highly nonlinear representations while preserving the original networks structure.

## Problem formulation

To reveal the implicit sentiment polarity of social media short text, this paper explores and provides a novel angle to integrate user profiles and latent topics from text and social network structure, which is different from existing text-based methods. Specifically, rather than focusing on word-word relationship, the text embedding module processes text into graphs and represent each text as a graph of vectors. By jointly optimizing the first-order and second-order proximity in the graph, the proposed text embedding module can effectively capture the non-consecutive and long-distance semantics. Furthermore, inspired by the theory in sociology, we consider that social network information also plays an important role in understanding the context and predicting

latent sentiment links. Therefore, beyond text embedding, this paper also integrates embedded information from social network structures. These implicit structures can significantly enrich the input information for sentiment analysis, especially for the dominant short-texts published on online social networks. In the subsections below, the input and output of the sentiment link prediction task are defined.

*Input* When predicting user sentiments in social networks, it is very complex to build a joint graph model by integrating user relationships and users' sentiments on entities. This work addresses this challenge by introducing multiplex networks embedding to learn the low dimensional vectorization representation of nodes in each graph. Through the feature learning algorithms, it can pay more attention to the important nodes and relationships, while ignoring the unimportant nodes and relationships. To extract the node embeddings, this paper constructs three kinds of graph network [39] (a) word graph (W–W) (b) user social relationship network (U–U), and (c) user-entity sentiment relationship polarity network (U-E-P).

## Word graph

The occurrence, ordering, and positioning of words and the relationship between different parts of a text play the most important role in determining the sentiment polarity of a sentence. Therefore, a first fundamental question is how to select an appropriate graph-based text representation models to reveal the innate and essential information of a text graph. As far as we know, word co-occurrence graph is an effective way to represent the relationship between one word and another in social media comments. This paper defines the text graph as $G_T = (W, E)$, where each word in a sentence is considered as a node. In the graph $G_T$, if the corresponding words appear simultaneously in the window of $k$ words, it can be defined that there is a co-occurrence relationship between these words. Generally, 2 to 10 seem to be appropriate according to different experiments. Therefore, this paper uses a word graph with a window size of 5.

## User social relation network

A user's social relation connections can measure the local relationship of this user and has an indirect impact on his/her behaviors. Therefore, based on such connections, users can be categorized into different communities, which is of great significance for analyzing the dissemination of social network information. In this paper, the user structure graph is denoted as $G_R = (U, R)$, where $U = \{u_i\}_{i=1}^N$ is the set of users; $R = \{r_{ij}\}_{i,j \in U}^{|R|}$ is user connections, $r_{ij}$ can take the value of 1 or 0, which denote the user $i$ follow the user $j$ or not.

## Sentiment relationship network

In social networks, a large number of users express their attitudes towards entities, and this topological relationship forms sentiment network. The sentiment network is denoted as $G_P = (U, V, P)$, in particular, $U$ and $V$ represent users set and entities set contained in social media contents, respectively; $P = \{p_{ij}\}_{i \in U, j \in V}$ is sentiment links among users and entities node pairs. Each $p_{ij}$ can take the value of $+1$, $-1$ or 0, which denote sentiment polarity between the user $i$ and entity $j$ are positive, negative or unobserved polarity relationship, respectively.

*Output* Taking $G_R$, $G_P$ and $G_T$ as inputs, and through a well-designed multi network fusion method to learn the node low dimensional representation, and then to predict sentiment relation $\hat{G}_P$ between uses and entities hidden in user comments.

## Multiplex network embedding

In this section, we first show the overall framework of MEISP model, and then introduce how to extract node representations from three networks and learning algorithm in details.

### Framework

The proposed method consists of two main modules: word graph-based text representation module and heterogeneous social graph network embedding module.

To capture the implicit text feature hidden in the user published content, this paper uses network embedding methods to learn the low dimensional potential representation of nodes in the network. For text representation, first, this paper processes text into graphs, and represent each text as a graph of vectors. Then, feature learning technology (representing the learning stage) is adopted to capture non-consecutive and long-distance semantics, and the internal features of the word graph-based text representation are determined.

In addition to text content, the topological relationship of users to entities forms the explicit sentiment network. To capture such information, rather than directly adopt the adjacency matrix of user-entity sentiment network, which is too long and sparse for further processing, this paper divides the original social network into user relationship structure network and user entity sentiment network, and learns their node embedding representations separately through two distinct autoencoders. Concatenation operation is then used to aggregate these above embeddings into final heterogeneous social network representation, since it can preserve more

information out of the two types of heterogeneous information networks.

Once the word graph-based text embedding and heterogeneous social graph network embedding are obtained, multiple non-linear layers are stacked to learn better representations of data. In this way, the word graph-based text modeling can complement the learning of heterogeneous social network modeling, realize the seamless integration of these two modules. In the following subsections, this paper will introduce the proposed MEISP framework in detail. Figure 1 shows the work-flow of the proposed framework.

## Word graph-based text representation

Node embedding representation is the process of vectorizing the nodes in a graph. In the process of learning representation, it can focus more on the important nodes and edges
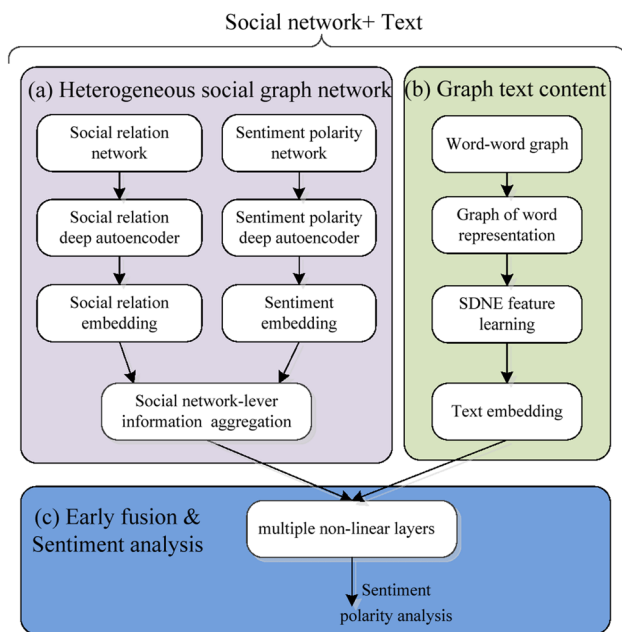


**Fig. 1** Framework of the MEISP model

and ignore the unimportant nodes and edges. In the past decades, shallow networks have been used to learn network embedding representations and some high performance models have been proposed. At the same time, node embedding can be used to reveal the essential information of graph. Inspired by the recent success of deep learning, SDNE [34] a semi-supervised deep model, which has multi-layer nonlinear functions, has been proposed. The composition of multilayer nonlinear functions can map data to highly nonlinear space, and by jointly optimizing first-order and second-order proximity in the semi-supervised depth model, this method can preserve the local and global network structure, and is

robust to sparse networks. Figure 2 shows the deep neural network for text embedding. Next, this paper will discuss the workflow of feature learning for graph-based text contents.

Word co-occurrence is the most effective way to represent text as a word graph. For a given text $T$, this paper defines the text as a graph $G_T = (W, E)$, $W = (w_1, w_2 \cdots w_n)$ denote the node set of vertices, and $w_i$ is the $i - th$ vertex in the text. If two words appear in a context within a window of size $k$, they are considered to have a word-word relation, and $E = \left\{ e_{ij} \middle| i \in W, j \in W \right\}$ denotes the set of edges between vertices pairs $(w_i, w_j)$, and in this study, $k$ is set as 5.

In this model, the auto-encoder is an unsupervised model, which consists of encoder and decoder. The encoder maps the input data to the representation space through multiple nonlinear operations, while the decoder maps the embeddings from the low dimensional potential space to the reconstruction space through multiple nonlinear functions.

For the $i - th$ vertex, the adjacency matrix $s_i$ is used as input, and each $s_i$ contains the neighbor structure information of vertex $i$, the representation of each hidden layer [34] is as follows:

$$y_i^{(1)} = \sigma \left( W^{(1)} x_i + b^{(1)} \right), \tag{1}$$

$$y_i^{(l)} = \sigma \left( W^{(l)} x_i + b^{(l)} \right), l = 2, \ldots, L. \tag{2}$$

After obtaining $y_i^{(l)}$, the output $\hat{x}_i$ can get by reversing the operation process of the encoder.

The goal of the autoencoder is to minimize the reconstruction error of the output and input, so the reconstruction process can make the vertexes with similar structure have similar embedding representation vectors. Therefore, it is necessary to use a weighted loss function with a high penalty coefficient for non-zero elements. In other words, by reconstructing the second-order nearest neighbors between the vertices to preserve the global network structure. And then in order to make the two connected nodes very close in the mapping space, we also define the first-order similarity. The first order and the second order similarity are denoted as:

$$L_{1st} = \sum_{i,j=1}^{n} s_{i,j} \left\| \left( \hat{y}_i^{(K)} - y_j^{(K)} \right) \right\|_2^2 = \sum_{i,j=1}^{n} s_{i,j} \left\| (\hat{y}_i - y_j) \right\|_2^2, \tag{3}$$

$$L_{2nd} = \sum_{i=1}^{n} \left\| (\hat{x}_i - x_i) \odot b_i \right\|_2^2, \tag{4}$$

where $\odot$ is Hadamard product, $\mathbf{b}_i = \left\{ b_{i,j} \right\}_{j=1}^{n}$. If $s_{i,j} = 0$, then $b_{i,j} = 1$, otherwise $b_{i,j} = \beta > 1$. The parameter $\beta$ controls the penalty term coefficient of non-zero elements.

The semi-supervised learning method ensures the first-order similarity and the second-order similarity respectively, the minimized objective function of joint optimization is:
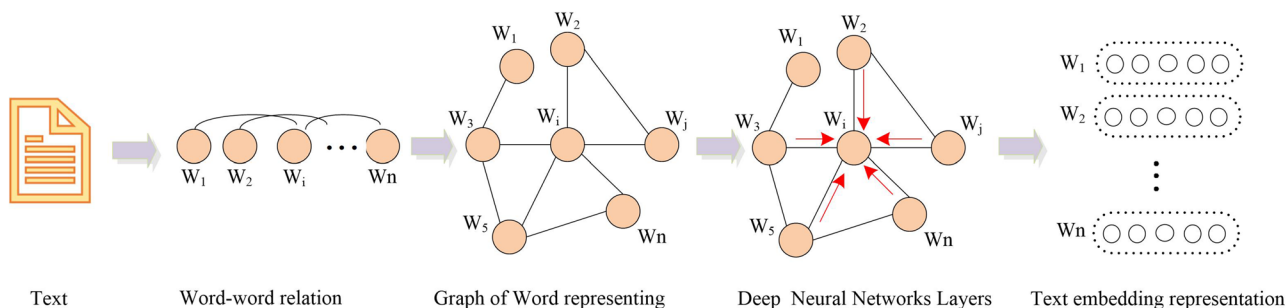
**Fig. 2** Illustration of a deep neural network for text embedding

$$L = L_{2\text{nd}} + \alpha L_{1\text{st}}$$

$$= \sum_{i=1}^{n} \left\| (\hat{x}_i - x_i) \odot b_i \right\|_2^2 + \alpha \sum_{i,j=1}^{n} s_{i,j} \left\| (y_i - y_j) \right\|_2^2, \quad (5)$$

$\alpha$ is the parameter controlling the first-order loss. The representations of word graph network can be denoted as $h_i^T = y_i^{(L)}$. It is note that not all text words contribute equally important to the text sentiment, hence, this paper tries to learn unified embedding by assigning different importance weights to different words. The weight coefficients can be calculated as follows:

$$z_i^T = \tanh\left(W h_i^T + b\right), \quad (6)$$

$$\alpha_i^T = \frac{\exp\left(z_i^T z_w\right)}{\exp\left(\sum_i^W z_i^T z_w\right)}, \quad (7)$$

where the word representation $h_i^T$ obtained is used as input of multi-layer perceptron model, and then get the node representation $z_i^T, z_w$ is the context word matrix. The final representation of the vertex $v_i$ is:

$$\mathbf{z}_i^T = \sum_{r=1}^{W} \left(\alpha_i \cdot h_i^T\right). \quad (8)$$

## Sentiment relationship network

For the given sentiment relationship network $G_P = (U, V, P)$, $U$ and $V$ are user and entity nodes set respectively, sentiment polarity adjacency vector can be denoted as $x_P = \left\{ p_{ij} \middle| i \in U, j \in V \right\}$. The sentiment polarity adjacency vector $x_P$ contains global sentiment information of user $i$ to entity $j$. However, in sentiment prediction task, due to the sparsity of adjacency vector $x_P$, it cannot be used as sentiment relation directly. Recently, network embedding methods, which can learn rich network structure and semantic information, has attracted a lot of attention, and motivated by excellent
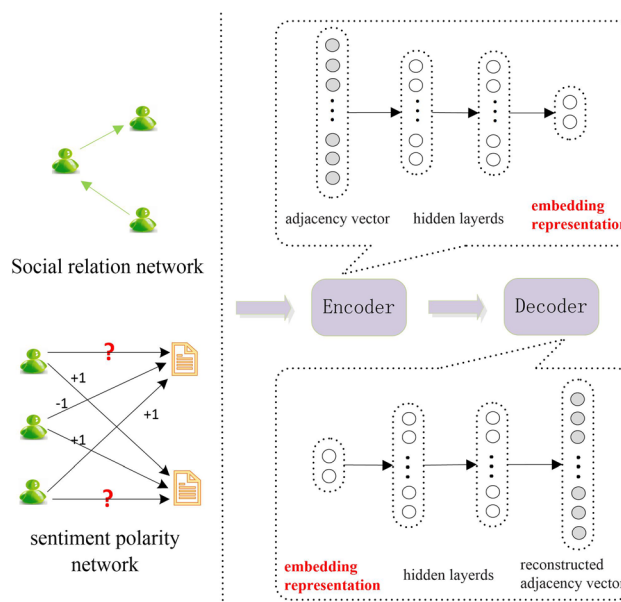


**Fig. 3** The architecture of autoencoder for user social relation and user-entity sentiment polarity network embedding

performance of deep autoencoder in capturing highly nonlinear network structures via deep models, this paper introduces deep autoencoder structure to learn sentiment relationship network embedding. In the sentiment relationship autoencoder, encoder compresses inputs into potential space representation, decoder reconstructs outputs from potential space representation. Both encoder and decoder are neural network models and include multiple nonlinear layers. Figure 3 shows the architecture of autoencoder for user social relation and user-entity sentiment network embedding.

Given the adjacency matrix $x_{Pi}$ as input, the hidden representations of each layer $l$ are:

$$x_{Pi}^l = \sigma\left(W_P^l x_{Pi}^{l-1} + b_P^l\right), l = 1, 2, \cdots L_P, \quad (9)$$

where $W_P$ and $b_P$ are the weight and bias in the sentiment deep architecture, respectively, $\sigma(\cdot)$ is the nonlinear

activation function, $L_P$ is the number of layers of autoencoder. The output of $L_P/2$ layer $\mathbf{z}_i^P = x_{Pi}^{Lp/2}$ denotes the sentiment polarity network embedding representation, and the output of $L_P$ layer $x_{Pi}^{Lp}$ denotes sentiment reconstruction vector $x_{Pi}$. In the process of sentiment polarity network embedded representation and data reconstruction, our basic goal is to minimize the reconstruction loss between input and output data. Therefore, the reconstruction loss term of sentiment polarity deep autoencoder is:

$$L(x_P) = \sum_{i \in U, j \in V} \left\| (x_P - \hat{x}_P) \odot l_P \right\|_2^2, \tag{10}$$

where $(x_P - \hat{x}_P)$ is used to measure the distance between $x_P$ and $\hat{x}_P$, $\odot$ is the Hadamard product and $l_{P_{ij}}$ is used to define reconstruction weight vector of sentiment link. In other words, if there is a sentiment connection between user node $u_i$ and entity node $v_j$, whether it's positive or negative, this paper defines the weight vector $l_{P_{ij}} = \gamma > 1$, else $l_{P_{ij}} = 1$. The meaning of the reconstruction weight vector lies in that we impose more penalty to the reconstruction error of the non-zero elements than that of zero elements in input $x_{P_i}$.

## User social relation network

For the given user social relation network $G_R = (U, R)$, this paper collects the social relation among user pairs $(u_i, u_j)$. The relationship of each user pair can take a value of 1 or 0, indicating whether user $i$ follows user $j$. Similar to sentiment network embedding, this paper also employs deep neural network to embed user social relation. Given the adjacency matrix $x_{Ri}$ as input, the hidden representations of each layer $l$ are:

$$x_{Ri}^l = \sigma\left( W_R^l x_{Ri}^{l-1} + b_R^l \right), l = 1, 2, \cdots L_R, \tag{11}$$

where $W_R$ and $b_R$ are the weight and bias in the social relation deep architecture, respectively. $\sigma(\cdot)$ is the nonlinear activation function. The output of $L_R/2$ layer $\mathbf{z}_i^R = x_{Ri}^{L_R/2}$ denotes the user social relation network embedding representation, and the output of $L_R$ layer $x_{Ri}^{L_R}$ denotes social relation reconstruction vector $x_{Ri}$.

The design of loss function of social relation embedding is the same reconstruction loss term of sentiment polarity network, and then the social relation loss is:

$$L(x_R) = \sum_{i \in U, j \in U} \left\| (x_R - \hat{x}_R) \odot l_R \right\|_2^2, \tag{12}$$

where $(x_R - \hat{x}_R)$ is used to measure the distance between $x_R$ and $\hat{x}_P$, $\odot$ is the Hadamard product and $l_{R_{ij}}$ is used to define reconstruction weight vector of user social relation link. That is if there is a link between $u_i$ and $u_j$, this paper defines the weight vector $l_{R_{ij}} = \gamma > 1$, else $l_{R_{ij}} = 1$. The meaning of

the reconstruction weight vector lies in that we impose more penalty to the reconstruction error of the non-zero elements than that of zero elements in input $x_{R_i}$.

## Optimization and sentiment prediction

Our work is to study heterogeneous social network-based and text-based multiplex network embedding for implicit sentiment mining and analysis. To combine the strength of both word graph-based text representation and heterogeneous social graph network modeling, an intuitive way is to choose concatenation to fuse the above information. However, the embedding representation obtained by concatenation is not effective for link prediction. Instead, early fusion model can optimizing all parameters simultaneously, so that the interaction between the word graph-based text modeling and heterogeneous social network modeling is more closely. Therefore, this paper uses the following methods to fuse the three kinds of network information.

First, for a given use/item node $i$, both the social relation network embedding and the sentiment network embedding are equally important to the final representation. Therefore, the proposed method first aggregates these two embedding into a final node embedding $\mathbf{z}_i^S$ through a concatenation function, and the finally social network-based embedding can be denoted as:

$$\mathbf{z}_i^S = \left\langle \mathbf{z}_i^R, \mathbf{z}_i^P \right\rangle. \tag{13}$$

Then, to fuse the text-based information and social network-based information, this paper integrates these two modules by early fusion model, that is $\mathbf{z}_i^S$ and $\mathbf{z}_i^T$ are used as the input of the multi-layer perceptron model, and then the representations of each hidden layer can be calculated by the following formulas:

$$\mathbf{z}_i^{t(0)} = \begin{bmatrix} \mathbf{z}_i^T \\ \mathbf{z}_i^S \end{bmatrix}, \tag{14}$$

$$\mathbf{z}_i^{(l)} = \delta_k \left( \mathbf{W}^{(l)} \mathbf{z}^{(l-1)} + \mathbf{b}^{(l)} \right), l = 1, 2, \cdots L, \tag{15}$$

$\delta_k$ is the activation function, $L$ is the number of hidden layers. The final embedding representation can be denoted as $\mathbf{z}^L$. Through the above nonlinear fusion, heterogeneous social network modeling and word graph-based text modeling are combined. In particular, for a given user node $i$ and an entity node $j$, inner product operation is used to predict sentiment $p_{ij}$, then, according to the reconstruction loss of three types deep autoencoders and supervised loss between predicted sentiment link polarity and ground truth, the objective function can be considered as:

$$\mathbb{L} = L_P + \lambda_1 L_R + \lambda_2 L_T + \lambda_3 L_S + \lambda_4 L_{reg}$$

$$= \sum_{i \in U, j \in V} \left\| \left( x_i^P - \hat{x}_i^P \right) \odot l_P \right\|_2^2 + \lambda_1 \sum_{i \in U, j \in U} \left\| \left( x_i^R - \hat{x}_i^R \right) \odot l_R \right\|_2^2$$

$$+ \lambda_2 \left( \sum_{i=1}^w \left\| \left( \hat{x}_i^T - x_i^T \right) \odot b_i \right\|_2^2 + \alpha \sum_{i,j=1}^n s_{i,j} \left\| \left( y_i^T - y_j^T \right) \right\|_2^2 \right)$$

$$+ \lambda_3 \sum_{p_{ij}=0,1} \left( f\left( e_i, e_j \right) - p_{ij} \right)^2 + \lambda_4 L_{reg},$$

$$(16)$$

where $\lambda_1$, $\lambda_2$ $\lambda_3$ and $\lambda_4$ are balancing parameters. The first three terms in Eq. (16) are the reconstruction loss terms of sentiment autoencoder, user social relationship autoencoder, and text autoencoder, respectively. The fourth term is the supervised loss term for penalizing the divergence between predicted sentiment and ground truth. The fifth term is the regularization term that prevents over-fitting.

## Experiment

In this section, this paper conducted several experiments to bring out different aspects of the proposed MEISP method in comparison with the baselines and the state-of-the-art sentiment analysis methods. All of our experiments are run on Intel Xeon 2.30 GHz processor, with 256 GB RAM cluster and Nvidia Quadro P620 GPUs. The operating system and software platforms are Ubuntu 16.04, TensorFlow 1.14 and Python 3.5.

## Database

To verify the application performance of the proposed model in social media, this paper collected Weibo data for experiments. To remove noise, text with less than 10 words is firstly filtered out. Secondly, useless symbols such as spaces and emoticons contained in the text are removed, and only keep the words and numbers in the sentence. Then Jieba

is used to segment the text. The stop words in the text are removed. Finally, the obtained text content is transformed into word graph network for feature learning. After the above processing the obtained data contains useful Weibo towards Movies dataset which contains users' sentiment towards entities with 8735 texts, 1983 users, 1357 entities, 32,732 social links. And then, through the entity-level sentiment extraction method, this paper establishes the sentiment dataset with label information, which consists of users' sentiment relation and social relation. The current amount of data is enough to extract valuable information from complex network data and complement text information for online opinion mine and sentiment analysis. Therefore, the experimental results are statistically significant.

## Parameters setting and sensitivity

For word graph-based text encoder, the reconstruction weight of non-zero elements $\beta = 10$ and the weight of first-order term $\alpha = 0.05$. Besides, this paper designs a 4-layer encode architecture for both social relationship autoencoder and sentiment autoencoder, trade-off parameter $\lambda_4 = 0.01$. Based on these parameters settings, this paper evaluates embedding dimensions, reconstruction weight $\gamma$ of sentiment and social autoencoder, the balance parameters $\lambda_1$, $\lambda_2$, $\lambda_3$ and different number of hidden layers of early fusion layer on the performance.

1. Embedding layer dimension, reconstruction weight γ of social network autoencoders

   Specifically, this paper first fixes the balancing parameters $\lambda_1 = \lambda_2 = 1$, $\lambda_3 = 20$, and test the embedding layer dimension and reconstruction weight $\gamma$ ($l_R = l_P = \gamma$).

   Figure 4a shows the influence of the embedded layer dimension and the reconstruction weight of the social autoencoders on the accuracy of the model, and this paper gets the following results. First, with the embedding dimension increased from 10 to 100, the per-
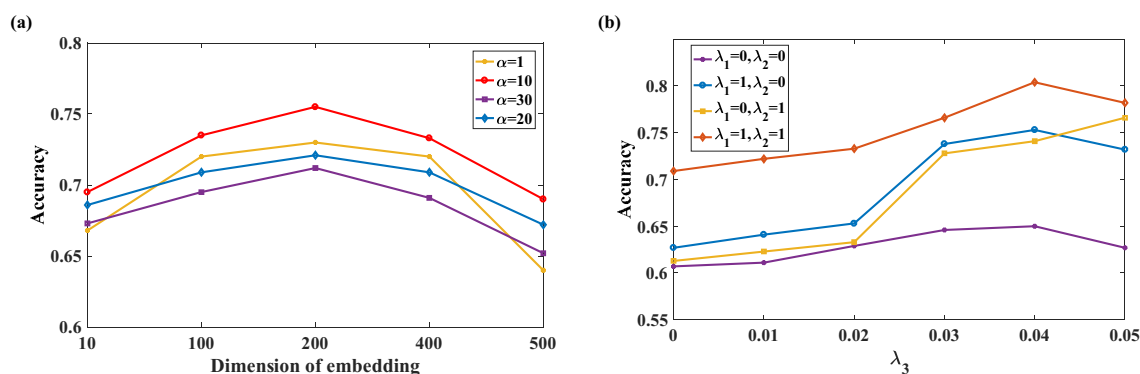


**Fig. 4** **a** Embedding layer dimension and **b** balancing parameters reconstruction weight of social autoencoders

formance is improved because the larger embedding dimension indicates that more useful information can be encoded into the autoencoder. However, with the embedding dimension increased from 100 to 500, it may introduce noises and then the performance starts to decline. Second, this paper sets $\gamma = 1$, $\gamma = 10$, $\gamma = 20$ and $\gamma = 30$, respectively, to test the influence of different reconstruction weights of non-zero elements $\gamma$ on the performance. The result shows that when $\gamma = 20$, the accuracy is the best, because when $\gamma = 1$ and $\gamma = 10$ there is not much discrimination between zero elements and non-zero elements in the MEISP model reconstruction stage; when $\gamma = 30$ the performance will decrease because large $\gamma$ makes MEISP insensitive to the dissimilarity between nodes.

2. Balancing parameters

Second, by fixing the reconstruction weights of social autoencoder and sentiment autoencoder non-zero elements $l_R = l_P = \gamma = 20$, the sentiment and social embedding dimensionality $d = 128$, this paper sets $\lambda_1$ and $\lambda_2$ to 0 or 1, respectively, and vary the value of $\lambda_3$ to study performance. For different values of $\lambda_1$ and $\lambda_2$, it indicates whether the word graph embedding module and the social relation embedding module are considered in the MEISP model. From Fig. 4b we make the following observations. (1) When $\lambda_1 = 1$ or $\lambda_2 = 1$, the model shows relatively better performance because the social relation information or word co-occurrence graph information is added into the sentiment network, respectively. When $\lambda_1 = 1$ and $\lambda_2 = 1$, the performance of MEISP is further improved. This paper will further use ablation experiments to verify the accuracy of sentiment link prediction of different network embedding models in section "Network embedding analysis". (2) The performance improves at the beginning when the $\lambda_3$ increases from 1 to 20. This is because MEISP focuses more on sentiment link prediction errors. However, when $\lambda_3$ continuously increases (e.g., $\lambda_3 = 30$) the performance starts to drop slowly. The reason is that too large $\lambda_3$ breaks the trade-off among loss terms in the objective function.

3. Different number of hidden layers of early fusion

After the text-based information and social network-based information embedding representation, this paper tries to integrate these two kinds of information by early fusion method. The generalization ability of early fusion improves with the increase of the number of hidden layers. However, because of the optimization difficulties, it may also degrade the performance of the model. In this situation, this paper tests the performance of link prediction under different number of early fusion hidden layers.

**Table 1** Comparison of link prediction results with different number of early fusion hidden layers

| Number of hidden layers | AUROC | Micro-F1 |
|---|---|---|
| No hidden layer | 0.8043 | 0.7674 |
| One layer | 0.8151 | 0.7749 |
| Two layers | 0.8263 | 0.7958 |
| Three layers | 0.8288 | 0.7734 |

**Table 2** Different model types

| | Model | Embedding Network | | |
|---|---|---|---|---|
| | | U-E-P | U–U | W–W |
| 1 | MEISP$_N$ | Yes | No | No |
| 2 | MEISP$_W$ | Yes | No | Yes |
| 3 | MEISP$_R$ | Yes | Yes | No |
| 4 | MEISP | Yes | Yes | Yes |

As can be seen from Table 1, first, when there are only input and output layers, the AUROC and Micro-F1 values of sentiment prediction are relatively weak. With the increase of the number of hidden layers, the performance of the model is gradually improved. This indicates that the use of deeper architecture can improve the sentiment analysis performance effectively, and the generalization ability of the model is enhanced. However, as the number of hidden layers increases, the training time of CPU it also increases, and even the tendency of "over fitting". Specifically, if there is only one layer in the early fusion layer, it takes 22.1 s for one epoch, while there are three hidden layers, it will take three times longer.

## Network embedding analysis

The purpose of multiplex network embedding is to learn the low-dimensional vector representation of different nodes, and then to reveal implicit sentiment polarity via similarity measurement function. In MEISP model, it contains word co-occurrence relations graph embedding (W–W), social relationship network embedding (U–U) and user-entity sentiment network embedding (U-E-P). In this section, this paper conducts ablation study to verify the effects of different models on sentiment analysis performance. Table 2 shows different model types.

MEISP$_N$ is set as a baseline model, which only contains user-entity sentiment polarity network for sentiment prediction. On the basic of MEISP$_N$ model, it adds word co-occurrence graph embedding and social relation graph embedding respectively, and record them as MEISP$_W$ and

**Table 3** Results of different network embedding analysis

| | Model | | | |
|---|---|---|---|---|
| | MEISP$_N$ | MEISP$_W$ | MEISP$_R$ | MEISP |
| $P_{precision}$ | 0.721 | 0.776 | 0.796 | 0.832 |
| $R_{recall}$ | 0.748 | 0.769 | 0.805 | 0.827 |
| F1 | 0.737 | 0.772 | 0.803 | 0.823 |

**Table 4** Experimental results on sentiment link prediction task

| Metric | Unsigned network embedding | | | Our model |
|---|---|---|---|---|
| | LINE | Node2vec | SDNE | MEISP |
| Accuracy | 0.627 | 0.631 | 0.689 | 0.829 |
| AUROC | 0.638 | 0.637 | 0.676 | 0.816 |

MEISP$_R$ model. Furthermore, in order to validate the effectiveness of word co-occurrence and user social relation on sentiment prediction, both the above two graph networks are embedded on top of the baseline model as MEISP.

First, we randomly hide 50% edges from the user-entity sentiment network, and construct a balanced test set in which the number of positive links equal to the number of negative links, and then the remaining network used to train the node representations via different models. The Precision, recall and F1value of experiment results are shown in Table 3.

From Table 3, we have the following observations. First, with the addition of U–U link information network and W–W information network to the MEISP$_N$ model, the accuracy of sign link prediction has been gradually improved. Compared with the MEISP$_N$ model, the MEISP$_w$ model has increased the Precision, Recall, and F1 values by 7.63%, 2.81%, and 4.75%, respectively. Compared with the MEISP$_N$ model, the MEISP$_R$ model has increased the precision, Recall, and F1 values by 10.40%, 7.62%, and 8.95%, respectively. It is proved that the MEISP$_w$ and MEISP$_R$ models can improve the performance of sentiment link prediction, optimize the effect of sentiment analysis, and have universal applicability by introducing user relationships, word graph information. Second, for MEISP model, U-E-P sentiment polarity network, U–U link information network, and W–W graph information network are introduced to realize the embedding and fusion of multiple networks for sentiment link prediction. When compared with the MEISP$_N$ model, MEISP has improved the precision, recall, and F1 values by 15.40%, 10.56%, and 11.67%, respectively. Third, comparing the MEIS$_P$ model with the MEISP$_W$ and MEISP$_R$ models, the precision, recall, and F1 values have been improved significantly.

In summary, these comparative experiments fully verify that the MEISP model proposed in this paper has significantly improved the sentiment analysis of Weibo by adopting user relationships and word co-occurrence information.

## Sentiment link prediction

In order to verify the embedding effect of MEISP model, we use link prediction as the downstream task, and compare it with baseline embedding methods LINE [32], Node2Vec [33] and SDNE [34].

For Weibo datasets, the training set and the testing set are given, and we randomly split the training set into actual training and validation data with a ratio of 9:1. The hyperparameters are tuned according to the performance on the validation set. Empirically, we set the learning rate as 0.01 with Adam [40] optimizer and the dropout rate as 0.5. We have done two groups of experiments. First, we randomly take 50% links of sentiment network out as the test set, and the remaining network is used to train network embedding. Then, the accuracy and AUROC are adopted as evaluation metrics. The experimental results are shown in Table 4.

It can be seen from Table 4 that our proposed MEISP model achieves Accuracy of 82.9% and AUROC of 81.6%. In term of accuracy, the MEISP method outperforms LINE, node2vec by 20.2%, 19.8%, respectively, and MEISP is about 14% higher than the network embedding model SDNE which also uses deep autoencoder. And achieves at least 14% gains on AUROC compared with the other three schemes.

Second, we randomly take 50–100% of the sentiment network links out as the test set to further verify the sentiment analysis results in the case of sparse network. The final results are shown in Fig. 5.

As can be seen from Fig. 5, with the increase of removed sentiment links from 50 to 100% in the test set, the AUROC values of all models decreased. For MEISP model, when the removed sentiment links are 50% and 100%, the AUROC values are 81.6% and 75.45%, respectively, which decreases by about 6%. However, for LINE, Ndode2vec and SDNE models, AUROC values decreased by 21%, 15% and 12%,
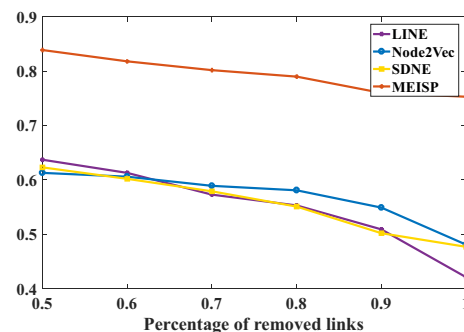


**Fig. 5** Accuracy and AUROC on link prediction tasks

respectively. It demonstrate the robustness of MEISP model is better than other models when the network is sparse.

The above two experiments show that compared with LINE, Node2vec and SDNE embedding methods, the proposed MEISP model can get better embedding representation results. Second, deep autoencoder structure-based multiplex network embedding is effective and more suitable for modeling sentiment analysis problem from link prediction angle.

## Cold start scenario

It is difficult to judge the hidden sentiment relationship between new users and entities, when new user or entity nodes are just added in the network, which is the cold start problem. The most intuitive way is to use the information from the target network to learn new nodes embedding. For example, in the sentiment link prediction problem, the model only uses the information from the user-entity sentiment link network to solve the cold start problem. However, due to the little interaction between the new arrival node and the existing target network, the performance of sentiment link prediction is not satisfactory. The proposed scheme does not only consider the interaction between the new added node and the original network, but also makes full use of the side information between heterogeneous nodes.

In this section, we study the performance of MEISP when new nodes join the existing target network. We construct a test set of newly text, and the user social relations and user entity sentiment relationships contained in the new text are not used in the training process. The results for all and new added sentiment link prediction are shown in Table 5.

**Table 5** The comparison results for all and new added sentiment link prediction

|         |                   | LINE  | Node2vec | SDNE  | MEISP |
|---------|-------------------|-------|----------|-------|-------|
| Accuracy | All sentiment link  | 0.621 | 0.637    | 0.675 | 0.816 |
|         | New sentiment link  | 0.571 | 0.537    | 0.659 | 0.809 |
| Micro-F1 | All sentiment link  | 0.627 | 0.631    | 0.688 | 0.823 |
|         | New sentiment link  | 0.508 | 0.524    | 0.621 | 0.807 |

From the results, for MEISP, the Micro-F1 of new sentiment link prediction decreases by 1.6% when compared with all sentiment link prediction results, but for LINE, Node2Vec and SDNE, new sentiment link prediction results decrease by 11.9%, 10.7% and 6.7%, respectively. It can be seen that MEISP is still perform better in the cold start scenario, because the proposed multiplex network embedding method can mine the information from social network information and text information, which makes the obtained information for sentiment relationship analysis more complete.

## Sentiment analysis accuracy

At present, many sentiment analysis methods are proposed from the feature extraction angle and word embedding representation angle and most of them have achieved excellent performance. In this section, we compare MEISP model with a number of existing methods as listed below.

1. Traditional machine learning methods
   *TextFeature* extracts text features including word and character n-grams, sentiment lexicon features, etc., and then train a SVM classifier.
   *Trigram* trains a support vector machine (SVM) classifier with unigrams, bi-grams and trigrams as features.
2. Deep learning method

**Text-based sentiment analysis models**
   *SVM + word2vec* word embeddings trained by word2vec and SVM is used as classifier.
   *UPNN* user product neural network [24], which incorporates user and product information using CNN.

**Link prediction-based sentiment analysis models**
   *MEISP* our proposed sentiment classification method uses graph-based text representation module and heterogeneous social network embedding modules to reveal implicit sentiment links.

In this part of the experiment, balanced data is selected in order to reduce the bias to a certain category, which ultimately affect the sentiment analysis accuracy. And the optimal parameter setting is adopted for all models. The results are shown in Table 6.

**Table 6** Comparison with existing methods

| Dataset | Metric | Traditional machine learning methods | | Deep learning method | | |
|---------|--------|------------------|----------------|----------------|------|-------|
|         |        |                  |                | Text-based | | Link prediction-based |
|         |        | TextFeature + SVM | Trigram + SVM | word2vec + SVM | UPNN | MEISP |
| Weibo | Accuracy | 0.423 | 0.407 | 0.407 | 0.437 | 0.783 |
|       | AUROC    | 0.418 | 0.421 | 0.387 | 0.406 | 0.806 |

As can be seen from Table 6, first, compared with the traditional sentiment analysis methods of Trigram and Textfeature, Word2vec + SVM and SSWE + SVM are more effective due to the deep learning-based word representation. Second, we notice that our method significantly outperforms UPNN model which is designed for sentiment classification specifically. The main difference between the MEISP and UPNN is the word representations derived. For UPNN model, it incorporate user and product level information into a convolutional neural networks, and only uses word embeddings trained with context information from text content only. In the MEISP model, it generates word representations from constructed multiplex networks embedding model, which inherently consider word emergence, ranking positioning and the relationship between different components of the text, as well as the heterogeneous social network information (user social relation network, user-entity sentiment network) during representation learning.

## Error analysis

For the given user-entity sentiment relationship network $G_P = (U, V, P)$, the sentiment polarity adjacency vector $x_P$ contains global sentiment information of user $i$ to entity $j$. However, in sentiment prediction task, due to the sparsity of adjacency vector $x_P$, it cannot be used as sentiment relation directly. The proposed MEISP model can capture word graph-based text information and heterogeneous social graph network information. We analyzed our predicted sentiment labels in section "Experiment" experiment part. In the confusion matrix, we observed that our model incorrectly linked the sentiment polarity relationship between users and entities. This is because (1) when the context is short, the text information is scarce, or (2) when the heterogeneous social networks are sparse, it is difficult to infer the sentiment of the user to entity. Therefore, the accuracy of each model is not high. As contextual information and heterogeneous social network information increase, the accuracy of all models is rising, indicating that text information and heterogeneous networks information are positively correlated with the accuracy of the model.

## Conclusions

In this paper, sentiment analysis is formulated as a special sign link prediction problem, which is addressed by combining user social link information, user-entity sentiment polarity information and word graph-based text level information through heterogeneous graph network embedding. In particular, a novel multi-network embedding method called MEISP is proposed for more effective and efficient embedding. This proposed work is different from current sentiment classification researches, which mainly determine the sentiment polarity of text from grammatical and semantic rules. In addition, it is different from the current multi-network embedding methods by considering both text information and social network information. Furthermore, it also extracts nodes' highly nonlinear representations and text structure information while preserving the structure of original networks. Extensive experiments are conducted to evaluate the performance of the proposed scheme. Experimental results prove the competitiveness of MEISP against multiple baselines and demonstrate the effectiveness of adopting user link information, user-entity sentiment polarity information and word graph-based text level information for embedding representation nodes in social networks to predict sentiment sign link polarities. The future works can focus on employing other graph-based representation methods to extract hidden characteristics of a network, exploiting the attribute information of edges to enrich the initial features of the network, and employing other innate features and information in the social media to enhance sentiment analysis techniques.

## Declarations

# References

1. Rashid A, Farooq MS, Abid A et al (2021) Social media intention mining for sustainable information systems: categories, taxonomy, datasets and challenges. Syst Complex Intell. https://doi.org/10.1007/s40747-021-00342-9

2. Yi S, Liu X (2020) Machine learning based customer sentiment analysis for recommending shoppers, shops based on customers' review. Complex Intell Syst 6:621–634. https://doi.org/10.1007/s40747-020-00155-2

3. Hu H, Wei Y, Zhou Y (2021) Product-harm crisis intelligent warning system design based on fine-grained sentiment analysis of automobile complaints. Complex Intell Syst. https://doi.org/10.1007/s40747-021-00306-z

4. Liu B (2012) Sentiment analysis and opinion mining[M]. Claypool Publishers, Morgan

5. Liao J, Wang S, Li D (2019) Identification of fact-implied implicit sentiment based on multi-level semantic fused representation. Knowl Based Syst 165(1):197–207. https://doi.org/10.1016/j.knosys.2018.11.023

6. Bryar AH, Tarik AR, Seyedali M (2021) Formal context reduction in deriving concept hierarchies from corpora using adaptive evolutionary clustering algorithm star. Syst Complex Intell. https://doi.org/10.1007/s40747-021-00422-w

7. Hassan BA, Rashid TA, Mirjalili S (2021) Performance evaluation results of evolutionary clustering algorithm star for clustering heterogeneous datasets. Data Brief. https://doi.org/10.1016/j.dib.2021.107044

8. Hassan BA, Rashid TA (2021) A multidisciplinary ensemble algorithm for clustering heterogeneous datasets. Neural Comput Appl. https://doi.org/10.1007/s00521-020-05649-1

9. Li H, Lu W (2017) Learning latent sentiment scopes for entity-level sentiment analysis. In: Proceedings of the thirty-first AAAI conference on artificial intelligence (AAAI-17)

10. Gan CQ, Wang L, Zhang ZF (2020) Multi-entity sentiment analysis using self-attention based hierarchical dilated convolutional neural network. Future Gener Comp Sy 112:116–125. https://doi.org/10.1016/j.future.2020.05.022

11. Ding J, Sun HL, Wang X, Liu XD (2018) Entity-level sentiment analysis of issue comments. In: Proceedings of the 3rd International Workshop on Emotion Awareness in Software Engineering, ACM. https://doi.org/10.1145/3194932.3194935

12. Kauter M, Breesch D, Hoste V (2015) Fine-grained analysis of explicit and implicit sentiment in financial news articles. Expert Syst Appl 42(11):4999–5010. https://doi.org/10.1016/j.eswa.2015.02.007

13. Chen T, Xu R, He Y, Wang X (2017) Improving sentiment analysis via sentence type classification using BiLSTM-CRF and CNN. Expert Syst Appl 72:221–230. https://doi.org/10.1016/j.eswa.2016.10.065

14. Yang Z, Yang D, Dyer C, He X (2016) Hierarchical Attention Networks for Document Classification. In: Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, ACL, San Diego, California, 1480–1489. https://doi.org/10.18653/v1/N16-1174

15. Wang Y, Huang M, Zhu X, Zhao L (2016) Attention-based LSTM for Aspect-level Sentiment Classification. In: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing. ACL, Austin, Texas, 606–615. https://doi.org/10.18653/v1/D16-1058

16. Wei JY, Liao J, Yang Z, Wang S, Zhao Q (2020) BiLSTM with MultiPolarity orthogonal attention for implicit sentiment analysis. Neurocomputing 383:165–173. https://doi.org/10.1016/j.neucom.2019.11.054

17. Zuo E, Zhao H, Chen B, Chen Q (2020) Context-specific heterogeneous graph convolutional network for implicit sentiment analysis. IEEE Access 8:37967–37975. https://doi.org/10.1109/ACCESS.2020.2975244

18. Violos J, Tserpes K, Psomakelis E, Psychas K, Varvarigou T (2016) Sentiment analysis using word-graphs. In: Proceedings of the 6th International Conference on Web Intelligence, Mining and Semantics, WIMS 2016, ACM, Nîmes, France. https://doi.org/10.1145/2912845.2912863

19. Bijari K, Zare H, Kebriaei E, Hadi V (2020) Leveraging deep graph-based text representation for sentiment polarity applications. Expert Syst Appl 144:113090. https://doi.org/10.1016/j.eswa.2019.113090

20. Zhao P, Hou L, Wu O (2020) Modeling sentiment dependencies with graph convolutional networks for aspect-level sentiment classification. Knowl Based Syst 193:105443. https://doi.org/10.1016/j.knosys.2019.105443

21. Gui L, Zhou Y, Xu R, He Y, Lu Q (2017) Learning representations from heterogeneous network for sentiment classification of product reviews. Knowl Based Syst 124:34–45. https://doi.org/10.1016/j.knosys.2017.02.030

22. Xiao Y, Li X, Liu Y, Liu H, Li Q (2018) Correlations multiplexing for link prediction in multidimensional network spaces. Sci China Inform Sci 61:112103. https://doi.org/10.1007/s11432-017-9334-3

23. Zhao K, Zhang Y, Zhang Y, Xing C, Li c (2019) Learning from user social relation for document sentiment classification. In: Proceeding of International Conference on Database Systems for Advanced Applications. Springer, 86–103. https://doi.org/10.1007/978-3-030-18579-4_6

24. Tang D, Qin B, Liu T (2015) Learning semantic representations of users and products for document level sentiment classification. In: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing. ACL, 1014–1023. https://doi.org/10.3115/v1/P15-1098

25. Chen H, Sun M, Tu C, Lin Y (2016) Neural Sentiment Classification with User and Product Attention. In: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, ACL, Austin, Texas, 1650–1659. https://doi.org/10.18653/v1/D16-1171

26. Dou Z (2017) Capturing User and Product Information for Document Level Sentiment Analysis with Deep Memory Network. In: Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, ACL, Copenhagen, Denmark, 521–526. https://doi.org/10.18653/v1/D17-1054

27. Wu Z, Dai XY, Yin C, Huang S, Chen J (2018) Improving review representations with user attention and product attention for sentiment classification. Proceeding of Thirty-second AAAI conference on artificial intelligence. AAAI press, New Orleans, pp 5989–5996

28. Wang J (2012) Geometric structure of high-dimensional data and dimensionality reduction. Springer, Heidelberg, pp 203–220

29. Belkin M, Niyogi P (2002) Laplacian eigenmaps and spectral techniques for embedding and clustering. Adv Neural Inf Process Syst 14(6):585–591

30. Perozzi B, Al-Rfou R, Skiena S (2014) Deep walk: Online learning of social representation. In: Proceedings of the 20th ACM

SIGKDD International Conference on Knowledge Discovery and Data Mining, New York, ACM, 701–710

31. Mikolov T, Chen K, Corrado G, et al., (2013). Efficient estimation of word representations in vector space. In: Proceedings of Workshop at International Conference on Learning Representations, Scottsdale, ICLR

32. Tang J, Qu M, Wang M, et al (2015) Line: Large-scale information network embedding. In: Proceedings of the 24th international conference on world wide web, Florence, ACM, 1067–1077

33. Grover A, Leskovec J (2016) node2vec: Scalable feature learning for networks. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. San Francisco, ACM, 855–864.

34. Wang D, Cui P, Zhu W (2016) Structural deep network embedding. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, ACM, 1225–1234

35. Zhang H, Sun Y, Zhao M et al (2020) Bridging user interest to item content for recommender systems: an optimization model. IEEE T Cybernetics 50(10):4268–4280. https://doi.org/10.1109/TCYB.2019.2900159

36. He C, Xie T, Rong Y, et al (2019) Bipartite Graph Neural Networks for Efficient Node Representation Learning. arxiv. https://arxiv.org/pdf/1906.11994v2.pdf

37. Wang H, Zhang F, Hou M, et al (2018) SHINE: Signed Heterogeneous Information Network Embedding for Sentiment Link Prediction. In: Proceedings of the 11th ACM International Conference on Web Search and Data Mining, Marina Del Rey, ACM, 592–600

38. Yuan WW, He KY, Han GJ et al (2019) User behavior prediction via heterogeneous information preserving network embedding. Future Gener Comp Sy 92:52–58. https://doi.org/10.1016/j.future.2018.09.036

39. Sun Y, Han J (2013) Mining heterogeneous information networks: a structural analysis approach. ACM SIGKDD Explor Newsl 14(2):20–28

40. Diederik K, Ba J (2015) ADAM: A method for stochastic optimization, In: Proceeding of 3rd International Conference on Learning Representations. San Diego, ICLR

41. Mikolov T, Chen K, Corrado G., Dean J. (2013) Efficient estimation of word representations in vector space. In Proceedings of Workshop at International Conference on Learning Representations (ICLR)