

Received February 1, 2020, accepted February 15, 2020, date of publication February 20, 2020, date of current version March 3, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2975244

Context-Specific Heterogeneous Graph Convolutional Network for Implicit Sentiment Analysis

ENGUANG ZUO¹, HUI ZHAO, BO CHEN, AND QIUCHANG CHEN

College of Information Science and Engineering, Xinjiang University, Ürümqi 830046, China

Corresponding author: Hui Zhao (zhmerry@126.com)

This work was supported by the National Natural Science Foundation of China (NNSFC) under Grant 61561047.

ABSTRACT Sentiment analysis has attracted considerable attention in recent years. In particular, implicit sentiment analysis is a more challenging problem due to the lack of sentiment words. It requires us to combine contextual information and precisely understand the emotion changing process. Graph convolutional network (GCN) techniques have been widely applied for sentiment analysis since they are capable of learning from complex structures and preserving global information. However, these models either only focus on extracting features from a single sentence and ignore the context semantic background or only consider the textual information and overlook the phrase dependency when constructing the graph. To address these problems, we propose a new context-specific heterogeneous graph convolutional network (CsHGCN) framework that can combine all context representations. It has a complete context that reflects the information on documents more comprehensively. It has a dependency structure that obtains token-token semantic acquisition more accurately. The experimental results on a Chinese implicit sentiment dataset show that our proposed model can effectively identify the target sentiment of sentences, and visualization of the attention layers further demonstrates that the model selects qualitatively informative tokens and sentences.

INDEX TERMS Implicit sentiment, sentiment analysis, heterogeneous graph, graph convolutional network (GCN).

I. INTRODUCTION

Sentiment analysis is an important problem in natural language processing (NLP). According to the expression of subjective and objective emotion in texts, Liu [1] divided texts into implicit emotional and explicit emotional texts. The implicit emotional text is defined as “A language fragment (sentence, clause or phrase) subjective sentiment but contains no explicit sentiment word”. Here are two examples that briefly illustrate the difference between explicit and implicit emotions.

Example 1: 月色真美 (English translation: “The moonlight is so beautiful!”)

Example 2: 一个月有30天,你在家待了30天。(English translation: “There are 30 days in a month, and you have stayed at home for 30 days.”)

The associate editor coordinating the review of this manuscript and approving it for publication was Pengcheng Liu¹.

Example 1 shows an explicit emotional tendency through the word “美 (beautiful)”. However, some texts, such as Example 2, show implicit emotional tendencies with no explicit emotional words. Unstructured texts produced by users in social media contain side writing of real social life, which reflects the behavior of people in real life. These data are widely used in applications of user feedback in the catering industry, online public opinion analysis, and personalized product recommendations [2].

Representation learning is an essential intermediate step in text sentiment classification tasks. In previous studies, convolutional neural networks (CNNs) [3] and recurrent neural networks (RNNs) [4] have been widely applied to text sentiment analysis. To represent the hierarchical relationship between phrases in a sentence, syntax trees or dependency trees are introduced to replace the original sequential structure in CNNs and RNNs [5], [6]. In the past two years, graph neural networks (GNNs) such as graph convolutional network (GCNs) [7], [8] and graph recurrent network (GRNs) [9]

have attracted widespread attention. GNNs are simple and effective and can effectively capture deep-level domain features. They are widely used in the field of sentiment analysis, including the text level [10], [11], sentence level [12], and aspect level [13] etc.

Currently, most of the research on sentiment analysis focuses on explicit sentiment classification tasks. These tasks are already accurate, and the improvement is limited. However, implicit sentiment analysis is a more challenging problem due to the lack of sentiment words. Statistics show that Chinese implicit sentiment sentences account for 15%-20% of the total sentiment sentence.¹ In Chinese texts, the implicit sentiment classification task is not solved better than the explicit one because of the following challenges:

- From the linguistic level of emotional expression, the Chinese sentence can make different emotions by different contextual semantic backgrounds, and the acquisition of semantic features is more complicated.
- From the perspective of words, the implicit text does not contain emotional terms, and the words are relatively objective and neutral, which leads to text representation methods based on the bag of words model not effectively representing the semantics of a sentence.
- Implicit emotional sentences are euphemistic expressions of subjective emotional tendencies, and they are closely related to the individual's cognitive background. There is no formal standard definition for it.

It can be observed that the implicit sentiment classification task has higher requirements for learning text representations, and it needs more semantic information to infer emotional tendency in text. We find that the current research on text sentiment classification still has the following limitations. **Previous studies either focused on extracting features from a single sentence and ignored contextual semantics; or only considered discourse information when constructing the graph, ignoring the dependencies between the tokens.**

In Example 3, the first sentence Example 3-1 is labeled, and the second sentence Example 3-2 is unlabeled. Example 3-2 is the context of the target sentence Example 3-1.

Example 3-1: 照原路走回宿舍, 再次感受三点一线的感觉吧! (label = positive) (English translation: "Let's walk back the way to dormitory and relive the time in high school!" label = positive)

Example 3-2: 高中木有住过校, 但是仍是三点一线.....好怀念的说 (English translation: "I never lived on campus when I was in high school, but I still miss that focused and hard-working time.")

If only the information in Example 3-1 is considered, the classification result could have no effective tendency (label = neutral). Furthermore, with the consideration of Example 3-2, it can be determined that Example 3-1 should have a positive emotional bias (label = positive).

Given the above limitations, we propose a new context-specific heterogeneous graph convolutional network (CsHGCN) framework. First, we separate the emotional target sentence from its context in a document and then represent all the remaining context text as a heterogeneous graph. In the heterogeneous graph, the nodes of the graph are composed of tokens or sentences. A syntactic dependency tree constructs the edge between the Chinese token node, and the boundary between each token and sentence node is built by term frequency-inverse document frequency (TF-IDF). The border between each sentence node is constructed by sentence sequence order. Finally, we propose a framework for building convolutional graph networks (GCNs) [7] over heterogeneous graphs, extracting semantic background information and word dependency. In summary, our main contributions are as follows:

- We propose a new context-specific heterogeneous graph convolutional network (CsHGCN) framework for implicit sentiment analysis. The whole context at the document level is considered a heterogeneous graph. The dependency structure is maintained, and the obtained context features are more accurate.
- We apply a novel GCN model to implicit sentiment classification tasks. GCN is considered as an adaptation of the convolutional CNN for encoding local information of unstructured data, and it can effectively capture deep-level domain features.
- Extensive experimental results show that in implicit sentiment classification, the scores of all models can be improved by the context semantic background and that our proposed CsHGCN has better performance and interpretability.

II. RELATED WORK

A. SENTIMENT ANALYSIS

According to different methods of text processing, sentiment analysis is mainly divided into two categories: the process based on sentiment lexicon and the plan based on deep learning. The advantage of the method based on sentiment lexicon is that it is simple in structure. Nevertheless, the creation of a sentiment lexicon requires feature selection, as well as a large number of labeled data, which costs considerable time and human resources. Moreover, to obtain high classification accuracy, large size and high-quality emotional dictionaries are required. For the above reasons, there are relatively few studies on sentiment analysis based on dictionaries [14], [15].

Methods based on the deep neural networks have stronger knowledge representation ability [16]. CNN [3] and RNN [4] are widely applied to sentiment classification tasks. Zhang et al. [17] proposed that character-level CNN can effectively extract local information of the same convolution window. The model based on RNN can encode sequence information. Ma et al. [18] proposed sensing LSTM to extract common-sense information in the sequence model. Yang et al. [19] proposed a hierarchical attention model

¹http://conference.cipsc.org.cn/smp2019/smp_ecisa_SMP.html

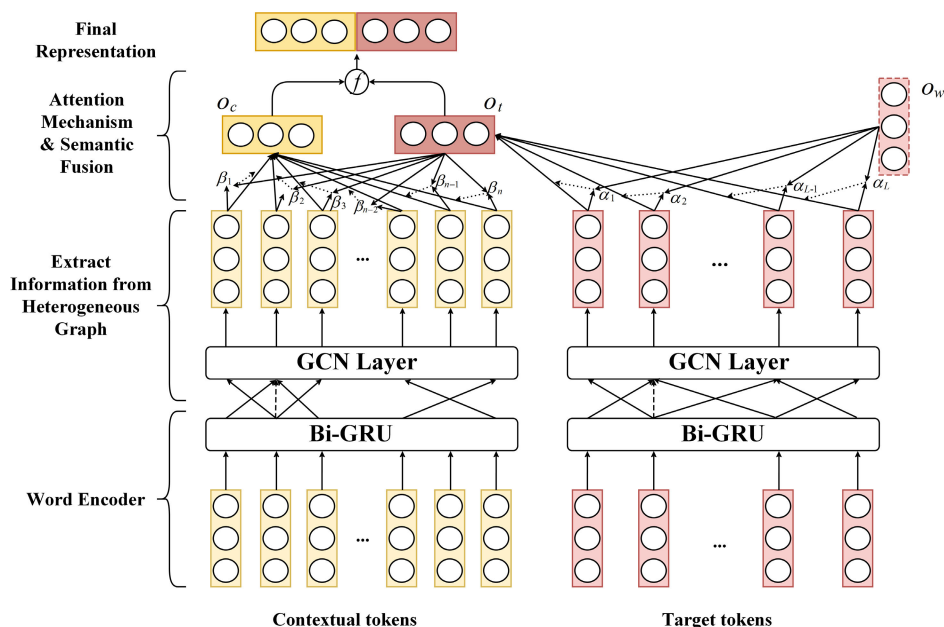


FIGURE 1. Overview framework of context-specific heterogeneous graph convolutional network.

HAN, which achieved excellent results. Tai *et al.* [20] and Teng and Zhang [21] used Tree-LSTM to encode tree-structured sentences to extract hierarchical information for sentiment analysis. Although Tree-LSTM can acquire more accurate semantic information, it has disadvantages such as slow training and difficulty parallelizing. Later, Mou *et al.* [22] combined the advantages of CNN and RNN and proposed a tree-based convolutional neural network (TB-CNN). However, word order features were not considered in TB-CNN. Subsequently, Liao *et al.* [23] proposed a multilayer convolutional neural network model, the semantic dependency tree-based CNN (SDT-CNN), based on the syntactic dependency tree, which can learn implicit dependency representation and a context-explicit semantic background representation of emotional information in the text. However, the context of each sentence in the SDT-CNN model is irrelevant, which may cause some semantic deficiencies. In contrast, the heterogeneous graph we proposed maintains not only the dependency structure but also has edges connected between contexts, so the information obtained is more complete.

B. GRAPH CONVOLUTIONAL NETWORK

In recent years, graph convolutional neural networks have received increasing attention. Scarselli *et al.* [24] proposed a graph neural network (GNN) for encoding an arbitrary graph structure. Bruna *et al.* [25] and others first proposed a CNN structure based on spectral graph theory. The author generalized the CNN model to a regular grid structure. Deferrard *et al.* [26] and others proposed a CNN algorithm based on spectral graph theory, which defined graph convolution. On this basis, Kipf and Welling [7] proposed a graph

convolutional neural network model to obtain the best results on a benchmark data set. In recent research on related NLP tasks, Yao and Mao [10] and others proposed constructing a heterogeneous graph on the entire corpus for text-level classification. Wu *et al.* [27] and others repeatedly eliminated the nonlinearity between GCN layers and folded the function into a linear transformation to reduce the extra complexity of GCNs. Zhang *et al.* [28] and Chen *et al.* [13] introduced the syntactic dependency tree into GCN to encode the syntactic structure and used the new GCN structure in sentiment analysis at the sentence level and aspect level. In previous studies, when building document-level graphs, the edges between token nodes were either based on chronological order or cooccurrence. For comparison, we consider the hierarchical relationships within sentences when constructing heterogeneous graphs, and the structure of the dependency tree is used to obtain more accurate semantic information.

III. CONTEXT-SPECIFIC HETEROGENEOUS GRAPH CONVOLUTIONAL NETWORK

Fig.1 shows the overall framework of our model context-specific heterogeneous graph convolutional network (CsHGCN), which consists of four parts: word coding layer, information extraction layer, attention layer, and final vector representation layer.

In the following sections, the components of CsHGCN are described in detail.

A. WORD REPRESENTATION AND BIDIRECTIONAL GRU CODING

In this component, each sentence is represented as $S = \{w_1, w_2 \dots, w_i \dots, w_n\}$, where w_i is the token. Each

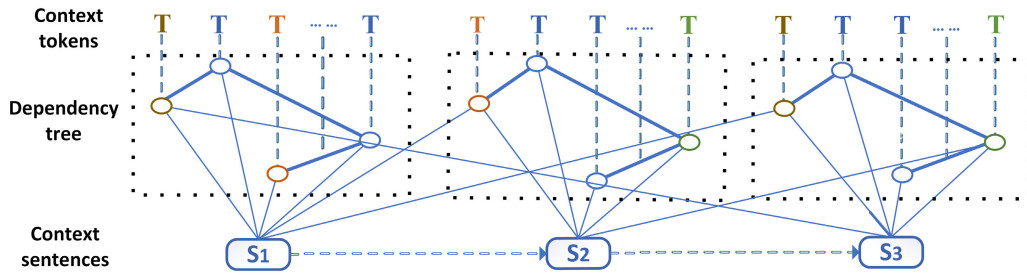


FIGURE 2. Context-specific token sentence graph. Nodes begin with “S” are sentence nodes, others are token nodes. The blue bold edges are token-token edges and the blue thin edges are document-word edges. Except blue, the same color tokens mean the tokens equally. For example, two green “T” represent the same token on different dependency trees.

token in the sentence is mapped to a low-latitude vector space to obtain a word embedding matrix $E \in R^{n \times d_e}$, where n is the size of the vocabulary and d_e is the dimension of the word vector. Sending E to the bidirectional gated recurrent unit (Bi-GRU) [29] network to obtain the hidden layer state vector $H = \{h_1, h_2 \dots, h_i \dots, h_n\}$, where $h_i \in R^m$ represents the hidden layer state vector of the Bi-GRU network at time t , and m represents the vector dimension.

B. CONTEXTUAL-SPECIFIC GRAPH

We construct a text graph of word segmentation nodes and sentence nodes for the context so that GCN can obtain sufficient node information. In text graph $G = (V, E)$, where $|V| = n$ is the number of nodes, which is the summary of nodes with sentences and tokens in a document, and E is the edge set of graph G . It consists of token dependency, term frequency-inverse document frequency (TF-IDF) of the token in the sentence, and the sentence order. The TF-IDF of the token determines the weight of the edge sentence-token, where term frequency (TF) is the number of times the token appears in the sentence, and inverse document frequency (IDF) is the logarithmically scaled inverse fraction of the number of sentences that contain the token in the document. To relate the dependency tree of each sentence, we incorporate sentence order as a feature to represent the relationship between sentence nodes. Matrix $X \in R^{n \times m}$ is composed of the eigenvector $x_v \in R^m$ of n nodes, where m is the eigenvector dimension, and $A \in R^{n \times n}$ is the adjacency matrix of graph G , which is composed of the relationship between each node. Formally, the weight of the edge between node i and node j is defined as for the formula:

$$A_{ij} = \begin{cases} DT(i, j) & i, j \text{ are tokens} \\ TF - IDF_{ij} & i \text{ is sentence, } j \text{ is token} \\ O(i, j) & i, j \text{ are sentences} \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

$DT(i, j)$ is the dependency relationship between the token nodes i and j in the dependency tree. The construction of the text graph is shown in Fig.2. When the input is the

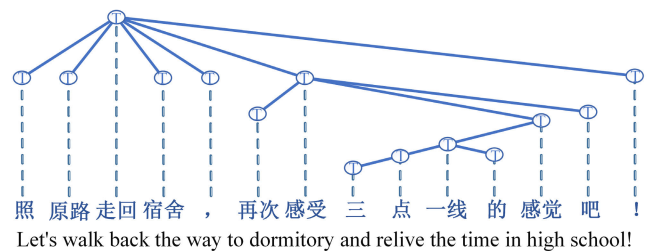


FIGURE 3. A tony demo of target dependency tree.

target sentence, the heterogeneous graph is degraded into a dependency tree, as shown in Fig.3.

C. CONVOLUTIONAL OVER THE HETEROGENEOUS GRAPH

X and A in the constructed text heterogeneous graph G are used as the inputs of the GCN, and the propagation mode between layers is shown in the below equations.

$$H^{j+1} = \rho(\tilde{A}H^{(j)}W_j + b^{(j)}) \quad (2)$$

$$D_{ii} = \sum_j A_{ij} \quad (3)$$

$$\hat{A} = A + I \quad (4)$$

$\tilde{A} = D^{-\frac{1}{2}}\hat{A}D^{-\frac{1}{2}}$ is a symmetrical and normalized adjacency matrix, D is a degree matrix of nodes, I is an identity matrix, W_j is a weight matrix, ρ is an activation function, j is the j -th layer, and b is a bias, $H^{(0)} = X$.

A single-layer GCN can only rely on one layer of convolutions to obtain the information of neighbor nodes, and by deepening the layers of the GCN, it can integrate the knowledge of a broader neighborhood. Therefore, we send the text graph into a simple two-layer GCN, and each node in the graph represents the updated formula below:

$$H^G = ReLU(\tilde{A} ReLU(\tilde{A}XW_0 + b^0)W_1 + b^1)) \quad (5)$$

A two-layered GCN can allow information passing between nodes that are within two steps away. Even if there are no directly connected edges between two nodes in the graph, GCN still enables information exchange between

nodes. The number of sentences in our data set document is relatively small. Further experiments have found that the performance of the two-layer GCN is better than that of the single-layer GCN. It has the same result as Kipf and Welling [7].

D. CONTEXT-SPECIFIC ATTENTION

Not all context tokens contribute equally to judging target sentence sentiment. Hence, we match the essential features related to the target sentence from the matrix $H_{context}^G$ with the overall context information, and a correlation attention weight is set for each context token. First, we make the word-level attention with target sentence vector H_{target}^G that is output from the GCN layer at the word level. The attention is calculated as follows:

$$O_{target\ i}^G = \tanh(W_t h_{target\ i}^G + b_t) \quad (6)$$

$$\alpha_i = \frac{\exp(O_{target\ i}^G O_w^T)}{\exp(\sum_i^L O_{target\ i}^G O_w^T)} \quad (7)$$

$$O_t = \sum_i^L \alpha_i O_{target\ i}^G \quad (8)$$

As shown in (9), we first feed the word representation H_{target}^G obtained by GCN to a layer of the MLP network to obtain O_{target}^G and then calculate the similarity between vector O_w and each of its tokens to obtain α_i , where o_w is the same as HAN [19], and it is the context vector that is randomly initialized and learned during the training process. Finally, the weighted vectors in O_{target}^G are summed to obtain the final representation o_t of the target sentence.

$$O_{context\ i}^G = \tanh(W_t h_{context\ i}^G + b_c) \quad (9)$$

$$\beta_i = \frac{\exp(O_{context\ i}^G O_w^T)}{\exp(\sum_i^L O_{context\ i}^G O_w^T)} \quad (10)$$

$$O_c = \sum_i^L \beta_i O_{context\ i}^G \quad (11)$$

$$r = f(O_t, O_c) \quad (12)$$

Next, we need to match in all context nodes with o_t to obtain a related message. Similarly, we feed the matrix $H_{context}^G$ into the single-layer MLP to obtain $O_{context}^G$ and use the target sentence representation obtained in (8) as the query for each token in $O_{context}^G$. We perform the attention calculation to obtain β_i . Similarly, (11) indicates that the vector in $O_{context}^G$ is weighted and summed according to the allocation weight to obtain the final representation O_c of the context.

The final vector representation is such as (12), where $f(\cdot)$ represents the splicing operation, and r integrates the information interaction of the target sentence and relevant context information.

E. SENTIMENT CLASSIFICATION AND TRAIN

The obtained vector r is fed to the fully connected layer, and the input result is classified into three categories using

TABLE 1. The statistic of SMP-ECISA 2019 dataset.

Dataset	Implicit sentiment			Total	
	neutral	positive	negative	target	doc
Train	7746	3768	4008	15222	13664
Dev	923	469	511	1903	1706
Test	926	463	514	1903	1705

softmax. The formula is as follows:

$$P = \text{softmax}(W_p r + b_p) \quad (13)$$

$P \in R^{(d_p)}$ represents the possibility of classification, d_p represents the number of classifications, and $W_p \in R^{d_p \times m}$ and $b \in R^{d_p}$ represent training weights and biases. During training, a standard gradient descent (SGD) algorithm is used to update the parameters. L^2 - norm regularization is used to prevent overfitting. The loss function is cross-entropy. The formula is as follows:

$$\text{Loss} = - \sum_{i \in D} \sum_{j=1}^{d_p} Y_{ij} \log P_{ij} + \lambda \|\theta\|_2 \quad (14)$$

where D represents the index of the labeled document, d_p represents the output feature dimension the same as (13), Y represents the real label matrix, θ represents all the training parameters, and λ represents the penalty coefficient.

IV. EXPERIMENTS

A. PROBLEM DEFINITION

Our work is to study contextual-based implicit sentiment mining and analysis. The formal task description is as follows:

$$\text{Label} = \mathcal{F}(\text{Doc}(S_c, S_t)) \quad (15)$$

where $\text{Doc}(\cdot)$ represents a text with multiple sentences (number of sentences ≥ 2), and it has at least one target sentence S_t (number of targets ≥ 1). The rest is context S_c , $\mathcal{F}(\cdot)$ represents the model frame, and $\text{Label} \in [\text{neutral}, \text{positive}, \text{negative}]$ represents the set of predictive labels.

B. DATASETS AND EXPERIMENT SETTINGS

We use public data from The Evaluation of Chinese Implicit Sentiment Analysis (SMP-ECISA 2019) for the experiments.² The original data are from Weibo, travel websites, product forums, etc. The initial corpus is noisy, and the expression is informal. Therefore, we perform some preprocessing on the dataset before the text analysis. Our model requires a complete sentence structure and contextual semantics. Therefore, we filtered out the following:

- 1) Sentences without subject-predicate structure (keeping the dependency syntactic structure intact).
- 2) Target sentences without context (contrast test of the influence of context on target sentence judgment, excluding irrelevant variables).

Finally, we randomly extracted 80% of the dataset as the training set, 10% as the validation set, and the remaining 10% as the test set. The detailed statistics are shown in TABLE 1.

²<http://biendata.com/competition/smpecisa2019/>

TABLE 2. The sentiment classification result of different models(%). The best result on each label is bolded; model with * means that the text was removed all contextual sentences; we run all models 5 times and report the mean results.

System	Precision			Recall			F1			Acc
	neutral	positive	negative	neutral	positive	negative	neutral	positive	negative	
TreeLSTM*	80.30	59.67	74.26	86.72	55.29	68.48	83.39	57.40	71.26	74.14
HAN*	76.89	71.65	78.69	92.01	49.68	72.57	83.78	58.67	75.51	76.46
TreeGCN*	80.81	61.68	70.45	92.33	44.39	70.04	86.19	51.69	70.24	74.67
CsHGCN*	80.54	71.84	74.52	89.85	54.00	75.68	84.94	61.65	75.10	77.30
TreeLSTM	78.73	65.74	69.1	93.95	35.64	73.54	85.67	46.22	71.25	74.25
HAN	81.14	69.38	79.48	93.41	55.29	72.37	86.85	61.54	75.76	78.46
TreeGCN	78.47	72.87	75.75	92.12	49.89	73.54	84.75	59.23	74.63	76.83
CsHGCN	83.00	76.52	78.86	94.38	55.29	78.40	88.33	63.84	78.64	80.55

C. MODELS FOR COMPARISON

To fully validate and understand our model, we selected the following baseline models for comparison:

- The HAN [19] model uses a hierarchical attention mechanism at the word level and the sentence level, so the model can give different “attention” to the abilities of sentences and words of different importance in the text. We divide HAN into two parallel models of context, and target sentences are spliced into the final representation after two layers of attention mechanism.
- The Tree-LSTM [20] model is a kind of LSTM network based on a tree structure, which solves the problem of sentiment classification of nonlinear structures such as dependency trees. We also divide the Tree-LSTM into two submodels, context and target sentences, and use their last hidden state as the overall context and the expression of the target sentence.
- The Tree-GCN [28] model uses Bi-LSTM to encode the input word vector to obtain the hidden state with context information and then uses a GCN convolution to obtain the neighboring node information, which enhances the robustness of the GCN.

D. IMPLEMENTATION DETAILS

We set the word embedding dimension to 200 dimensions, the GRU hidden layer size to 64 dimensions, the batch size to 64, the GCN hidden layer size to 128 dimensions, and the initial learning rate to 0.002. Adam [30] was used to train the dataset up to 20 epochs. One hundred batches output a loss value. If the verification loss did not decrease 10 consecutive times, the training stopped.

E. RESULTS

The above TABLE 2 shows the experimental comparison results between our model and other baseline models. It can be seen from the table that our models CsHGCN and CsHGCN* obtained the best classification accuracy with context information or not, respectively.

Specifically, the accuracy of all models increased significantly after adding context information, and this result is consistent with common sense: the expression of implicit sentiments depends on its context and semantic environment. Additionally, compared with other baselines, CsHGCN

performs better on the evaluation indexes of precision, recall, and macro-F1 after adding context.

We also note that Tree-GCN performs better than Tree-LSTM in the tree-based models. Nevertheless, CsHGCN shows the best performance, indicating that GCN’s ability to extract features is stronger than LSTM, and adding heterogeneous graphs can enrich node information.

The HAN model has higher precision for negative labels with and without contextual sentences. One possible reason is that in the absence of context, HAN, which adopts a linear-based information exchange and attention mechanism, is somewhat competitive in the classification tasks of short texts, with the highest precision and F1 values among. The dependency tree in CsHGCN is a nonlinear structure. Without context, the number of edges in the graph is small, and the structure of the graph is simple, which leads to incomplete knowledge learned by our model, making the boundary between positive and negative learned by the model unclear. After adding context, sentence nodes are added to the heterogeneous graph. As the number of edges in the graph increases, the information exchange of heterogeneous graph nodes increases, which makes the precision and F1 value increase by 4.34% and 3.54%, respectively. HAN’s precision and F1 value only increased by 0.79% and 0.25%, respectively, when it has context. This shows that CsHGCN can better exploit contextual information.

F. EFFECT OF HETEROGENEOUS GRAPH

To study the contribution of each component in the context-dependent heterogeneous graph, we ran an ablation study on the test set (TABLE 3).

TABLE 3. An ablation study of the best CsHGCN model. We run all models for 5 times and give mean results(%).

	Model	Accuracy
Bset	CsHGCN	80.55
-	Dependency tree structure	78.67
-	Sentence nodes	74.51

First, we removed the dependency tree structure to observe the impact of the dependency structure of each token on the model. Then, we removed the sentence nodes to keep the effect of the heterogeneous structure on the model. We found the following. (1) When we remove the dependency tree structure, the score drops by 1.88% Acc. (2) Acc drops s by 6.04% when we remove the sentence nodes. (3) Correctly,

TABLE 4. Case study. Visualization of attention scores from CsHGNC on testing examples. Blue denotes the sentence weight, and red denotes the word weight.

Sentence	Attention visualization	Prediction	Label
Context1	我滴 最爱 ， 喝啤酒 ， 吃 小龙虾 Drinking beer and eating crayfish are my favorites .		
Target	哈哈哈 ， 两斤 多 昨天 和 啤酒 雪碧 一起 再见 咯 ！ Hahaha , I had more than two pounds crayfish with beers yesterday !	positive	positive
Context1	根据 媒体报道 ， 乐视 老板 贾跃亭 夫妇 和 乐视系 12.37 亿元 的 资产 遭 上海 市 高级法院 冻结 ， 原因 是 银行贷款 逾期 未 还 。 According to media reports , over \$1.237 billion assets , which belong to the LeTV Ltd. , were frozen by the Shanghai High Court because of the overdue bank loan .		
Context2	除此之外 ， 法院 还 冻结 了 乐视 公司 在 大圣 科技股份 有限公司 的 全部 股权 及 红利 。 In addition , the High Court also froze all the equity and dividends of LeTV Ltd. in Dasheng Tech Ltd .		
Target	据调查 ， 实际上 不仅 是 上海 的 法院 ， 此前 北京 、 天津 、 济南 等 地 的 法院 早 已 有 冻结 乐视系 资产 的 情况 。 ？ According to the investigation , not only the courts in Shanghai but also the courts in Beijing , Tianjin , Jinan , and other places have frozen the assets of LeTV Ltd. since long before .	negative	negative
Context1	本周 日 （ 29 ） 下午 2 点 西单 图书大厦 《 大国 威仪 ： 三军 仪仗队 60 年 纪实 》 八一 前 首发 。 The Chinese Great Powers : The 60th Anniversary of the Honor Guard of the Chinese Army will be released this Sunday (29th) at 2 pm in the Xidan Book Building before August .		
Context2	香港 回归 政权 交接仪式 上 中国 指挥官 与 英国 指挥官 交锋 怎样 出奇 制胜 ？ At the handover ceremony for HongKong's return to China , how did the Chinese commander defeat its British counterpart , surprisingly ?		
Context3	奥运会 会旗 怎样 完美 升起 ？ How perfectly will the Olympic flag be risen ?		
Context4	三次 国庆 大 阅兵 不 为人 知 的 训练 等 ， 记录 一个个 可爱 士兵 的 成长 过程 The unreleased training of three National Day military parades recorded the growth of every lovely soldier .		
Target	独家 图片 与 文字 ： 周恩来 为什么 修改 总统 报告 词 ？ Exclusive pictures and text : Why did Premier ZhouEnlai modify the president's report ?	neutral	neutral

the performance after removing the sentence nodes is more significant than that of the dependency structure model. One possible reason is that the sentence nodes have more edges in the heterogeneous graph, which can obtain richer and more integrated information.

G. CASE STUDY

To better understand how CsHGNC works, we used a few sentences of test set data as an example to show the visualization effect, as shown in TABLE 4. The results of

the following visualizations are not the few that work best, but in most cases, the output that the model visualization can explain.

$$color_t = \frac{\beta_t}{\sqrt{\beta'_s}} \quad (16)$$

$$\beta'_s = \frac{\beta_s}{\sum_i \beta_i} \quad (17)$$

To show the relatively important words in unimportant sentences, the color depth $color_w$ of the words is shown in (16).

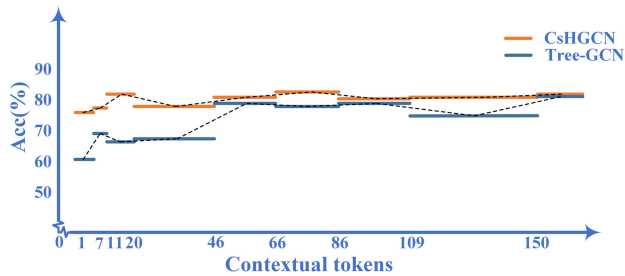


FIGURE 4. The performance over the samples with different number of contextual tokens. The averaged results over 3 runs are reported.

Because in our heterogeneous graph, the second part of the attention of sentence node β_s and context word segmentation node β_t were simultaneous, we defined the color depth β'_s of the sentence as (17), where β_t represents the context sentence node.

In the first example, we chose a document containing one context sentence, and the sentiment label is the implicit positive tendency. It can be seen that the model accurately focused on the “哈哈(Hahaha)” of the target sentence and the “我滴(My)” and “最爱(favorites)” of the context. We chose a document containing two contextual sentences, and the sentiment label was the implicit negative tendency. After training, the model distributed high attention weights in the target tokens “冻结(frozen),” “乐视(LSTV),” “资产(assets),” and the context tokens “有限公司(Ltd.),” “股权(equity),” “红利(dividends),” etc. were desirable. Moreover, the model focused on the context of the second sentence, which shows that the second sentence was more closely related to the target sentence. In the third example, we chose a document with multiple context sentences and without emotional tendencies. It can be seen that the weight of attention for each token in the target sentence is similar. One possible reason is that implicit sentiment sentences have no explicit sentiment words to focus on, and there is no emotional tendency. The model considers that all participles have the same contribution to sentiment classification. It also reflects that the implicit sentiment classification task is much more complicated than the explicit one.

V. DISCUSSION

We performed some statistical experiments to determine the effect of context length on model performance.

According to the statistics, the minimum number of context word segmentation in the test dataset is one token, and the maximum is 833 tokens. In particular, to reduce the contingency of the results, we divide the test data into nine blocks by length, each with approximately 200 documents. The final result is shown in Fig. 4.

We use Tree-GCN as the baseline. It can be seen from the figure that when the context is short, the information is scarce, and it is difficult to infer the sentiment of the target sentence; therefore, the accuracy of each model is not high. At this time, the useful information attention of the target sentence is

essential. As contextual information increases, the accuracy of all models is rising, indicating that contextual information is positively correlated with the accuracy of the model.

VI. CONCLUSION AND FUTURE WORK

We analyze and compare the advantages and disadvantages of existing models and propose a new implicit sentiment classification method, a context-dependent heterogeneous graph convolutional neural network. We build a token-sentence graph for each chapter, and the experimental results show that GCN can obtain global collaborative information very well. The experimental results also show that GCN uses the dependency structure between a token in the heterogeneous graph and the long-distance dependence of sentence nodes to improve the accuracy of the model.

This research may be further improved in the following aspects. First, the attribute information of edges is not considered in our dependency tree, i.e., the labels of each edge. We plan to design a specific GNN to add edge information. Second, we consider incorporating domain knowledge. Finally, due to the high cost of annotation, the lack of implicit sentiment labeled data becomes a major obstacle, and we consider using transfer learning to extract knowledge from a document-level corpus and apply the knowledge to this task.

ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for their valuable remarks and comments.

REFERENCES

- [1] B. Liu, *Sentiment Analysis and Opinion Mining* (Synthesis Lectures on Human Language Technologies), vol. 5. San Rafael, CA, USA: Morgan & Claypool, 2012, pp. 1–184.
- [2] G. Zhou, X. Zhu, C. Song, Y. Fan, H. Zhu, X. Ma, Y. Yan, J. Jin, H. Li, and K. Gai, “Deep interest network for click-through rate prediction,” in *Proc. 24th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2018, pp. 1059–1068.
- [3] Y. Kim, “Convolutional neural networks for sentence classification,” in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2014, pp. 1746–1751.
- [4] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [5] K. S. Tai, R. Socher, and C. D. Manning, “Improved semantic representations from tree-structured long short-term memory networks,” in *Proc. the 53rd Annu. Meeting Assoc. Comput. Linguistics, 7th Int. Joint Conf. Natural Lang. Process.*, vol. 1, 2015, pp. 1556–1566.
- [6] J. Li, T. Luong, D. Jurafsky, and E. Hovy, “When are tree structures necessary for deep learning of representations?” in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2015, pp. 2304–2314.
- [7] T. N. Kipf and M. Welling, “Semi-supervised classification with graph convolutional networks,” in *Proc. 5th Int. Conf. Learn. Represent. (ICLR)*, 2019. [Online]. Available: <https://arxiv.org/abs/1609.02907>
- [8] D. Marcheggiani and I. Titov, “Encoding sentences with graph convolutional networks for semantic role labeling,” in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2017, pp. 1506–1515.
- [9] D. Beck, G. Haffari, and T. Cohn, “Graph-to-Sequence learning using gated graph neural networks,” in *Proc. 56th Annu. Meeting Assoc. Comput. Linguistics*, vol. 1, 2018, pp. 273–283.
- [10] L. Yao, C. Mao, and Y. Luo, “Graph convolutional networks for text classification,” in *Proc. Annu. Conf. Int. Speech Commun. Assoc. (INTER-SPEECH)*, 2013, pp. 3057–3061.
- [11] L. Huang, D. Ma, S. Li, X. Zhang, and H. Wang, “Text level graph neural network for text classification,” in *Proc. Conf. Empirical Methods Natural Lang. Process., 9th Int. Joint Conf. Natural Lang. Process. (EMNLP-IJCNLP)*, 2019, pp. 3435–3441.

- [12] Y. Zhang and Y. Zhang, "Tree communication models for sentiment analysis," in *Proc. 57th Annu. Meeting Assoc. Comput. Linguistics*, 2019, pp. 3518–3527.
- [13] C. Zhang, Q. Li, and D. Song, "Aspect-based sentiment classification with aspect-specific graph convolutional networks," in *Proc. Conf. Empirical Methods Natural Lang. Process., 9th Int. Joint Conf. Natural Lang. Process. (EMNLP-IJCNLP)*, 2019, pp. 4560–4570.
- [14] C. Zhao, S. Wang, and D. Li, "Exploiting social and local contexts propagation for inducing Chinese microblog-specific sentiment lexicons," *Comput. Speech Lang.*, vol. 55, pp. 57–81, May 2019.
- [15] A. Balahur, J. M. Hermida, and A. Montoyo, "Detecting implicit expressions of emotion in text: A comparative analysis," *Decis. Support Syst.*, vol. 53, no. 4, pp. 742–753, Nov. 2012.
- [16] B. Kama, M. Ozturk, P. Karagoz, I. H. Toroslu, and M. Kalender, "Analyzing implicit aspects and aspect dependent sentiment polarity for aspect-based sentiment analysis on informal turkish texts," in *Proc. 9th Int. Conf. Manage. Digit. EcoSyst. (MEDES)*, 2017, pp. 134–141.
- [17] X. Zhang, J. Zhao, and Y. Lecun, "Character-level convolutional networks for text classification," in *Proc. Adv. Neural Inf. Process. Syst.*, Jan. 2015, pp. 649–657.
- [18] Y. Ma, H. Peng, T. Khan, E. Cambria, and A. Hussain, "Sentic LSTM: A hybrid network for targeted aspect-based sentiment analysis," *Cognit. Comput.*, vol. 10, no. 4, pp. 639–650, Mar. 2018.
- [19] Z. Yang, D. Yang, C. Dyer, X. He, A. Smola, and E. Hovy, "Hierarchical attention networks for document classification," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol.*, 2016, pp. 1480–1489.
- [20] K. Sheng Tai, R. Socher, and C. D. Manning, "Improved semantic representations from tree-structured long short-term memory networks," 2015, *arXiv:1503.00075*. [Online]. Available: <http://arxiv.org/abs/1503.00075>
- [21] Z. Teng and Y. Zhang, "Head-lexicalized bidirectional tree LSTMs," *Trans. Assoc. Comput. Linguistics*, vol. 5, pp. 163–177, Dec. 2017.
- [22] L. Mou, G. Li, L. Zhang, T. Wang, and Z. Jin, "Convolutional neural networks over tree structures for programming language processing," in *Proc. 30th AAAI Conf. Artif. Intell. (AAAI)*, 2016, pp. 1287–1293.
- [23] J. Liao, S. Wang, and D. Li, "Identification of fact-implied implicit sentiment based on multi-level semantic fused representation," *Knowl.-Based Syst.*, vol. 165, pp. 197–207, Feb. 2019.
- [24] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardini, "The graph neural network model," *IEEE Trans. Neural Netw.*, vol. 20, no. 1, pp. 61–80, Jan. 2009.
- [25] J. Bruna, W. Zaremba, A. Szlam, and Y. LeCun, "Spectral networks and locally connected networks on graphs," 2013, *arXiv:1312.6203*. [Online]. Available: <http://arxiv.org/abs/1312.6203>
- [26] M. Defferrard, X. Bresson, and P. Vandergheynst, "Convolutional neural networks on graphs with fast localized spectral filtering," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 3844–3852.
- [27] F. Wu, T. Zhang, A. H. de Souza, C. Fifty, T. Yu, and K. Q. Weinberger, "Simplifying graph convolutional networks," 2019, *arXiv:1902.07153*. <https://arxiv.org/abs/1902.07153>
- [28] Y. Zhang, P. Qi, and C. D. Manning, "Graph convolution over pruned dependency trees improves relation extraction," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2018, pp. 2205–2215.
- [29] D. Bahdanau, K. H. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *CoRR*, vol. abs/1409.0473, 2015.
- [30] K. Diederik and J. L. Ba, "ADAM: A method for stochastic optimization," *AIP Conf.*, vol. 1631, pp. 58–62, Dec. 2014.



ENGUANG ZUO was born in Xinjiang, China, in 1993. He received the bachelor's degree from Central South University, in 2016. He is currently pursuing the master's degree in computer application technology with Xinjiang University. His research interests include sentiment analysis and event detection.



HUI ZHAO was born in Xinjiang, China, in 1972. She received the Ph.D. degree from the Dalian University of Technology.

She has been a Professor with the College of Information Science and Engineering, Xinjiang University, since December 2011. Her research interests include artificial intelligence, natural language processing, and image processing.



BO CHEN was born in Henan, China, in 1993. He received the bachelor's degree from Northeast Electric Power University, in 2016. He is currently pursuing the master's degree in computer technology with Xinjiang University. His research interests include data mining and machine translation.



QIUCHANG CHEN was born in Guangdong, China, in 1995. She received the bachelor's degree from the Nanyang Institute of Technology, in 2017. She is currently pursuing the master's degree in computer technology with Xinjiang University. Her research interests include data mining and machine translation.

...