MG-SIN: Multigraph Sparse Interaction Network for Multitask Stance Detection

Heyan Chai, Jinhao Cui, Siyu Tang, Ye Ding, *Member, IEEE*, Xinwang Liu, *Senior Member, IEEE*, Binxing Fang, *Member, IEEE*, and Qing Liao, *Member, IEEE*

***Abstract*— Stance detection on social media aims to identify if an individual is in support of or against a specific target. Most existing stance detection approaches primarily rely on modeling the contextual semantic information in sentences and neglect to explore the pragmatics dependency information of words, thus degrading performance. Although several single-task learning methods have been proposed to capture richer semantic representation information, they still suffer from semantic sparsity problems caused by short texts on social media. This article proposes a novel multigraph sparse interaction network (MG-SIN) by using multitask learning (MTL) to identify the stances and classify the sentiment polarities of tweets simultaneously. Our basic idea is to explore the pragmatics dependency relationship between tasks at the word level by constructing two types of heterogeneous graphs, including task-specific and task-related graphs (tr-graphs), to boost the learning of task-specific representations. A graph-aware module is proposed to adaptively facilitate information sharing between tasks via a novel sparse interaction mechanism among heterogeneous graphs. Through experiments on two real-world datasets, compared with the state-of-the-art baselines, the extensive results exhibit that MG-SIN achieves competitive improvements of up to 2.1% and 2.42% for the stance detection task, and 5.26% and 3.93% for the sentiment analysis task, respectively.**

***Index Terms*— Graph neural network (GNN), multitask learning (MTL), sentiment analysis, social media, stance detection.**

NOMENCLATURE

$\mathbf{Y}^{\text{st}} = \{y_i^{\text{st}}\}$	Set of labels of stance detection task.
$\mathbf{Y}^{\text{se}} = \{y_i^{\text{se}}\}$	Set of labels of sentiment analysis task.
$\mathbf{e} = \{e_i\}$	Set of target texts.
$\mathbf{S} = \{s_i\}$	Set of tweets.
st-graph	Stance-specific graph.
se-graph	Sentiment-specific graph.
tr-graph	Task-related graph.
$t \in \{\text{st, se}\}$	Stance detection task or sentiment analysis task.
gt-graph	Types of heterogeneous graphs and gt $\in$ {st, se, tr}.
$\mathcal{A}^{\text{gt}}$	Adjacency matrix of gt-graph and gt $\in$ {st, se, tr}.

Manuscript received 21 January 2023; revised 28 September 2023; accepted 18 October 2023. Date of publication 13 November 2023; date of current version 6 February 2025. This work was supported in part by the National Natural Science Foundation of China under Grant 62076079, Grant 61976051, and Grant U19A2067; in part by the Guangdong Major Project of Basic and Applied Basic Research under Grant 2019B030302002; in part by the Major Key Project of Peng Cheng Laboratory (PCL) under Grant PCL2022A03; and in part by the Shenzhen Science and Technology Program under Grant ZDSYS20210623091809029. *(Corresponding author: Qing Liao.)*

Heyan Chai, Jinhao Cui, and Siyu Tang are with the Department of Computer Science and Technology, Harbin Institute of Technology, Shenzhen, Guangdong 518055, China (e-mail: chaiheyan@stu.hit.edu.cn; cuijinhao@stu.hit.edu.cn; tangsiyu999@stu.hit.edu.cn).

Ye Ding is with the School of Cyberspace Security, Dongguan University of Technology, Dongguan 523808, China (e-mail: dingye@dgut.edu.cn).

Xinwang Liu is with the College of Computer, National University of Defense Technology, Changsha 410073, China (e-mail: xinwangliu@nudt.edu.cn).

Binxing Fang and Qing Liao are with the Department of Computer Science and Technology, Harbin Institute of Technology, Shenzhen 518055, China, and also with the Peng Cheng Laboratory, Shenzhen 518066, China (e-mail: fangbx@cae.cn; liaoqing@hit.edu.cn).

Digital Object Identifier 10.1109/TNNLS.2023.3328659

I. INTRODUCTION

STANCE detection is an important task in natural language understanding (NLU), aiming at identifying one's attitude or standpoint toward a specific target or event, which can provide powerful information for widespread applications including fake news detection [1], [2], sentiment analysis [3], [4], and societal issues analysis [5]. As a consequence, there is a pressing need for exploring stance detection on social media with a relatively limited number of annotated datasets. Taking the COVID-19 pandemic as an example, since people have fewer opportunities for face-to-face communication, they turn to social media platforms such as Twitter, Facebook, and other microblogs to share their views regarding various epidemic prevention policies, yielding large amounts of stance detection data on social media sites [6]. To help health officials better estimate the expected impact of epidemic prevention policies, it is essential for governments to understand the public's opinion or stance about a series of epidemic prevention policies, such as stay at home mandates and wearing face masks in public. As shown in Fig. 1(a), the tweet 1 "I have an immune system that works fine, masks harm our immune system." expresses the user's attitude ("Against") about the target "wearing a face mask (WFM)" and sentiment polarity ("Positive") for the whole sentence. And, the tweet 2 expresses the user's attitude ("Favor") about the target "Stay at home orders" and sentiment polarity ("Positive") for the whole sentence.

Some early studies have been proposed to address stance detection on social media. These methods either leverage manually engineered features to tackle the stance detection problem by way of analyzing online activity of users [7] or utilize deep learning technology, including recurrent neural networks (RNNs), convolutional neural networks (CNNs) [8], [9], and attention mechanisms [10] to detect users' stances.

Tweet 1: I have an immune system that works fine, masks harm our immune system.
— Target: Wearing a face mask
— Stance Label: *Against*
— Sentiment Label: *Positive*
Tweet 2 : It's essential that staying at home is taken to stop the spread.
— Target: Stay at home orders
— Stance Label: *Favor*
— Sentiment Label: *Postive*

(a)

Tweet 1: Also what the fuck gives a man the right to tell a woman what to do with her body.
— Target: Legalization of Abortion
— Stance Label: *Favor*
— Sentiment Label: *Negative*
Tweet 2 : I can tell a woman what to do with her body.
— Target: Legalization of Abortion
— Stance Label: *Against*
— Sentiment Label: *Negative*

(b)

Fig. 1. Examples paired with their targets, stance expressions, and sentiment labels to show the roles that words play for different tasks. (a) Example 1. (b) Example 2.

Besides, some graph neural networks (GNNs) combined with attention mechanisms [11] or similarity matrix [12] were proposed to automatically learn the importance of neighbors of nodes, aiming at improving performance of stance detection. They primarily focused on extracting statistical and contextual features in annotated datasets. Extracting features requires expensive labor costs and the guidance of expert knowledge, which is not always feasible in practice. Moreover, due to noise information in tweet texts, the contextual feature extracted from sentences cannot provide enough useful information to identify users' stances. Some recent works have shown that multitask learning (MTL) methods can help improve the performance of stance detection by taking sentiment analysis tasks as auxiliary tasks [13]. For example, Sun et al. [14] leveraged a hierarchical attention network to determine stance and sentiment simultaneously. Different from them, this article focuses on solving the stance detection and sentiment analysis tasks simultaneously. Existing MTL methods largely focused on employing attention mechanisms to fuse different representations from distinct tasks to obtain richer stance representation at sentence level, ignoring the fine-grained task interaction (relationship between words and tasks). It can be argued that words may play different roles with respect to different tasks when jointly training stance detection tasks and sentiment analysis tasks. As such, it is desirable to model the roles of different words with respect to the different tasks in order to capture the relationship between tasks at multiple granularity levels. Noting that this article refers to the roles of different words with respect to the different tasks as pragmatics information of words. This can facilitate the learning of shared information between tasks and reduce the interference of noisy information, so as to achieve more accurate identifying results.

Fig. 1(b) shows a motivation example, where the tweet texts are paired with their corresponding targets, stance expressions, and sentiment labels. As shown in Fig. 1(b), noting that there are some identical words in "Tweet 1" and "Tweet 2," specifically, "a woman" and "what to do with her body." But, stance expressions relating to these words for the identical target "Legalization of Abortion" are opposite, and sentiment labels also present opposite polarities. That is, directly employing the stance/sentiment information associated with statistical and contextual information of words to identify the stance and classify sentiment polarity of the novel tweet may produce wrong results. The main reason is that the same words or expressions may have distinct importance in different sentences. Specifically, the pragmatics information of words is constantly changing, which leads to words yielding different contributions toward tasks. Therefore, it is important to capture the pragmatics information of words in tweet texts and adapt it for different tasks, which could facilitate the learning of stance expressions and sentiment representations in MTL framework. It can be argued that the main challenges in MTL stance detection problem are to adapt the roles that words play for different tasks, and model the relationships between tasks at word level to facilitate the learning of richer task-specific representations.

To better tackle the stance detection and sentiment analysis problem on social media, this article proposes a novel multigraph sparse interaction network (MG-SIN) to leverage fundamental word-level pragmatics dependencies of task representations. Specifically, MG-SIN first constructs the task-specific graphs, including st-graph and se-graph, to capture the pragmatics importance of words toward different tasks based on task-specific pragmatics weights at the word level. Then, MG-SIN considers the fine-grained relationship between words and categories of tasks to construct the tr-graph via deriving the word-category pragmatics dependency information for each word. At last, a novel graph-aware module of MG-SIN is proposed to effectively leverage task-specific and task-related information by a designed sparse interaction mechanism between graphs to fuse useful information for each task and filter noisy information from other tasks. The contributions of this work are summarized as follows.

1) A novel MG-SIN is proposed to facilitate the tasks to adaptively fuse useful information from the information-sharing network, so as to improve the performance of stance detection and sentiment analysis tasks simultaneously.
2) This article is the first to study the relationship between tasks in MTL by leveraging pragmatics information of words and word-category pragmatics dependency relationship at the word level. The pragmatics information of words can be utilized to capture more fine-grained representations of tasks.
3) A novel graph-aware module with a sparse interaction mechanism between graphs is proposed to facilitate the information sharing between tasks, which allows each task to selectively fuse the useful information shared by other tasks and filter the noisy information from the information-sharing network.

4) Experimental results on two real-world datasets demonstrate that our proposed method outperforms the state-of-the-art models both in stance detection and sentiment analysis tasks by considerable margins.

II. Problem Description

This section elucidates problem statement and notions in Section II-A and introduces the detailed definition of the task in Section II-B.

A. Problem Statement and Notations

For controversial topics or hot events on Twitter or other social media platforms, people may express their comments or thoughts about them by posting tweets. Under this scenario, this article defines the stance detection task as identifying people's standpoint or attitude expressed in each tweet toward specific target topics by using a classifier. Meanwhile, this article also defines the sentiment analysis task as classifying the sentiment polarity of each tweet using a classifier. More concretely, given a tweet t_i and the corresponding target e_i as input, the classifier should predict the stance from the set $\hat{y}^{\mathrm{st}} = \{\mathrm{Favor}, \mathrm{Against}, \mathrm{None}\}$ and the sentiment polarity of tweet t_i from the set $\hat{y}^{\mathrm{se}} = \{\mathrm{Positive}, \mathrm{Negative}, \mathrm{Neither}\}$ simultaneously. Note that both stance detection and sentiment analysis tasks are supervised classification problems, where tweets in the training and test datasets are mutually exclusive. For presentation simplicity, the frequently used notions are summarized in Nomenclature.

B. Definition of the Task

Definition 1 (Target Text): Each target, denoted by $e_i \in \mathbf{e}$ represents a hot topic, a people, et al. on the social media.

Definition 2 (Tweet Text): Given a set of tweets $\mathbf{S} = \{s_1, s_2, \ldots, s_{|\mathbf{S}|}\}$, each tweet s_i consists of one or more sentences.

Definition 3 (Sentiment Analysis Task): Given a set of tweets $\mathbf{S} = \{s_1, s_2, \ldots, s_{|\mathbf{S}|}\}$ and corresponding targets $\mathbf{e} = \{e_1, e_2, \ldots, e_{|\mathbf{e}|}\}$, predict sentiment label $\hat{y}_i^{\mathrm{se}}$ for each piece of unlabeled tweet s_i such that $\hat{y}_i^{\mathrm{se}} \approx y_i^{\mathrm{se}}$.

Definition 4 (Stance Detection Task): Given a set of tweets $\mathbf{S} = \{s_1, s_2, \ldots, s_{|\mathbf{S}|}\}$ and corresponding targets $\mathbf{e} = \{e_1, e_2, \ldots, e_{|\mathbf{e}|}\}$, predict stance label $\hat{y}_i^{\mathrm{st}}$ for each piece of unlabeled tweet s_i such that $\hat{y}_i^{\mathrm{st}} \approx y_i^{\mathrm{st}}$.

Definition 5 (Multitask Stance Detection): Given a set of tweets $\mathbf{S} = \{s_1, s_2, \ldots, s_{|\mathbf{S}|}\}$ and corresponding targets $\mathbf{e} = \{e_1, e_2, \ldots, e_{|\mathbf{e}|}\}$, find a model $\mathcal{F}$ that predicts the sentiment label $\hat{y}_i^{\mathrm{se}}$ and the stance label $\hat{y}_i^{\mathrm{st}}$ of the tweet s_i simultaneously, such that $\mathcal{F}(s_i) = \{\hat{y}_i^{\mathrm{se}}, \hat{y}_i^{\mathrm{st}}\} \approx \{y_i^{\mathrm{se}}, y_i^{\mathrm{st}}\}$.

III. Methodology

A. Overview of MG-SIN

This section describes our proposed MG-SIN model in detail. The architecture of the MG-SIN is demonstrated in Fig. 2, which mainly consists of four major components: 1) encoding block, which derives the word representations of the input text with text encoder (described in Section III-B); 2) multigraph construction block, which constructs heterogeneous graphs (st-graph, se-graph, and tr-graph) and learns fundamental word-level pragmatics dependencies of tasks representations; 3) multigraph sparse interaction block, which is designed to capture the relationship between tasks and achieve more efficient information sharing among tasks at multiple granularity levels; and 4) task attention block, which captures the richer stance and sentiment representations and outputs the final predictions.

B. Encoding Block

For a tweet text consists of n words $s = \{w_i\}_{i=1}^{n}$ and the corresponding target e consisted of m words $e = \{w_i\}_{i=1}^{m}$, n and m are the length of the text s and target e, respectively. MG-SIN uses the pretrained BERT-base[1] model as the Text Encoder to encode both the tweet text s and the target e. First, the tweet s and its corresponding target e are processed into the input pair format of BERT as

$$[\mathrm{CLS}]\ s\ [\mathrm{SEP}]\ e\ [\mathrm{SEP}].$$

Then, the tweet-target pair are feed into BERT to obtain a d_m-dimensional hidden representation $\boldsymbol{h} \in \mathbb{R}^{(n+m)\times d_m}$ for each input pair

$$\boldsymbol{h} = \mathrm{BERT}([\mathrm{CLS}]s[\mathrm{SEP}]e[\mathrm{SEP}]) \tag{1}$$

where $\boldsymbol{h} = \{h_1, h_2, \ldots, h_{(m+n)}\}$ denotes the representation of the input pair and $h_i \in \mathbb{R}^{d_m}$ denotes the vector representation of ith word.

C. Multigraph Construction

To capture the relationship between nodes of graph, some works [15], [16] use similarity matrix to fuse different types of information. Differently, to understand and adapt the role of words that play in different tasks, this article computes the pragmatics weight for each word toward different tasks and then constructs heterogeneous graphs to leverage fundamental word-level pragmatics dependencies of task representations.

MG-SIN builds the syntactical dependency tree [17] for each input sequence (target text e and tweet text s) to capture the word dependencies of input sentence. MG-SIN first constructs a dependency tree $\mathcal{T}^s \in \mathbb{R}^{n\times n}$ of tweet text s according to syntactical analysis tool[2] and obtain a set of root words w^r of dependency tree $\mathcal{T}^s$. Since the tweet s and its corresponding target e are the individual text, a novel approach is designed to build the relationship between s and e. Specifically, MG-SIN connects all words of e to the root words w^r of the $\mathcal{T}^s$, so as to derive the dependency tree of input sequence $\mathcal{T}$. Formally

$$\mathcal{T}(w_i, w_j) = \begin{cases} 1, & \text{if } w_i \in e \text{ and } w_j \in w^r \\ \mathcal{T}^s(w_i, w_j), & \text{otherwise} \end{cases} \tag{2}$$

where w^r denotes the root words [17] of $\mathcal{T}^s$, and $\mathcal{T} \in \mathbb{R}^{(m+n)\times(m+n)}$ denotes dependency tree of input sequence.

[1] https://github.com/google-research/bert
[2] We use spaCy toolkit: https://spacy.io/.

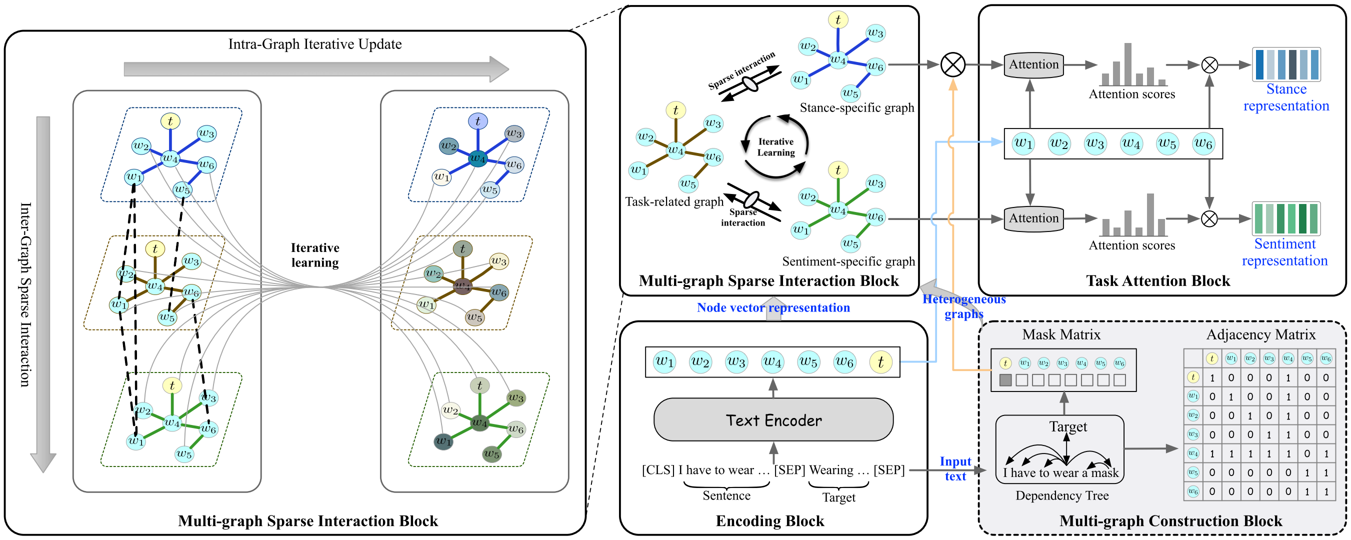


Fig. 2. Overall architecture of MG-SIN, which includes four components: 1) encoding block derives the word representation of input text; 2) multigraph construction block constructs heterogeneous graphs by the dependency tree and word-level pragmatics dependency information; 3) multigraph sparse interaction block captures the relationship between tasks with a sparse interaction mechanism; and 4) task Attention block is to learn the task representations.

1) Task-Specific Graph Construction: The task-specific graph aims to emphasize the crucial word relations for specific tasks and ignore the unconsidered ones. Specifically, if words have high optimistic pragmatics weights, their weights of graph edge will be large. In contrast, the words with low pragmatics weights show that the weights of their edges would be much diminished. MG-SIN proposes a novel approach to automatically capture the task-specific pragmatics weights of the words by integrating pragmatics information focused on specific tasks in the corpus.

First, MG-SIN computes word frequency $\omega(w_i)$ by calculating the times of word w_i appearing in the corpus, which can be defined as

$$p(w_i) = \frac{N(w_i)}{N}, \quad \omega(w_i) = \frac{p(w_i) - \mu(p)}{\delta(p)}, \quad w_i \in s \tag{3}$$

where $N(w_i)$ is the number of times w_i appeared in the corpus, N is the total number of words in the corpus, $\mu(\cdot)$ and $\delta(\cdot)$ are the mean value and standard deviation function, respectively.

Second, as sentences from these two types of labels (label_+ and label_-) contain more clearer pragmatics information, we only utilize Favor (label_+) and Against (label_-) category to compute the stance-specific pragmatics weight $\phi^t(w_i)|_{t=\text{st}}$, and utilize Positive ($\text{label}_+$) and Negative ($\text{label}_-$) category to compute the sentiment-specific pragmatics weight $\phi^t(w_i)|_{t=\text{se}}$. For tweet text s, this article computes the task-specific pragmatics weight $\phi^t(w_i)$ by

$$\rho^t(w_i) = \left| \frac{N^t(w_i, \text{label}_+)}{N^t(\text{label}_+)} - \frac{N^t(w_i, \text{label}_-)}{N^t(\text{label}_-)} \right|, \quad t \in \{\text{st}, \text{se}\} \tag{4}$$

$$\phi^t(w_i) = \frac{\rho^t(w_i) - \mu(\rho^t)}{\delta(\rho^t)}, \quad w_i \in s \tag{5}$$

where $N^t(w_i, \text{label}_\pm)$ and $N^t(\text{label}_\pm)$ are the number of occurrences of w_i and the total number of words in different stance or sentiment task, respectively. Since the target text e cannot form a sentence, this article defines pragmatics weight of target words by

$$\phi^t(w_j) = 1, \quad w_i \in e. \tag{6}$$

Based on the syntactical dependency tree $\mathcal{T}$ of input sequence, MG-SIN finally leverages the task-specific pragmatics weight of words $\phi^t(w_i)$ and word frequency $p(w_i)$ to derive the adjacency matrix $\mathcal{A}^t \in \mathbb{R}^{(m+n)\times(m+n)}$ of stance-specific and se-graph by

$$\mathcal{A}^t_{i,j} = \begin{cases} \sum\limits_{k\in\{i,j\}} \phi^t(w_k)p(w_k), & \text{if } \mathcal{T}(w_i, w_j) = 1 \\ 0, & \text{otherwise} \end{cases} \tag{7}$$

where $t \in \{\text{st}, \text{se}\}$ denotes the stance and sentiment task.

2) Task-Related Graph Construction: Since the relationship between tasks in different categories is different, this article designs a tr-graph to capture the fine-grained relationship between tasks to facilitate the learning of task representation. In tr-graph, a novel weighting scheme of words is proposed to automatically capture the relationship between words and categories of a specific task (also called task-related pragmatics weight of words), so as to explore the word-level task interaction.

MG-SIN first computes the category-aware word importance $\varphi(w_i, c^t_i)$ by incorporating the importance of the word w_i within a specific category c^t_i and the relationships between words and category. That is, if words exclusively occurred and with high frequency in a category, the words have a high weight to words. Formally

$$\varphi^t(w_i, c^t_i) = \frac{|w_i^{c^t_i}|}{|N_{c^t_i}|} \times \frac{|w_i^{c^t_i}|}{|w_i|} \tag{8}$$

where c^t_i is the ith category of task t, $|w_i^{c^t_i}|$ is the number of occurrences of w_i in c^t_i, $N_{c^t_i}$ is the total number of words in the category c^t_i, and $|w_i|$ is the times of w_i occurs in the training dataset.

Based on the category-aware word importance of words, MG-SIN concatenates the importance weights of words to different categories of a task to form the distribution of word-categories in the specific task, defined as $\eta^t(w_i)$. Formally

$$\eta^t(w_i) = \left[\gamma^t(w_i, c_1^t), \ldots, \gamma^t(w_i, c_i^t)\right] \tag{9}$$

where $t \in \{\text{st}, \text{se}\}$, c_i^t denoted the ith category of task t, and $\gamma^t(w_i, c_i^t)$ is defined as

$$\gamma^t(w_i, c_i^t) = \frac{\varphi^t(w_i, c_i^t) - \mu(\varphi^t(w_i, c_i^t))}{\delta(\varphi^t(w_i, c_i^t))} \tag{10}$$

where $\mu(\cdot)$ and $\delta(\cdot)$ are the mean and standard function.

MG-SIN can compute the task-related pragmatics weight of words $\xi(w_i)$ based on category-aware word importance of a word in stance detection task and sentiment analysis task, defined as $\eta^{\text{st}}(w_i)$ and $\eta^{\text{se}}(w_i)$, respectively. Formally

$$\xi(w_i) = \frac{\eta^{\text{st}}(w_i) \times \eta^{\text{se}}(w_i)}{\|\eta^{\text{st}}(w_i)\|_2^2 + \|\eta^{\text{se}}(w_i)\|_2^2 - \eta^{\text{st}}(w_i) \times \eta^{\text{st}}(w_i)}. \tag{11}$$

Based on the syntactical dependency tree $\mathcal{T}$ of input sequence, MG-SIN finally leverages the task-related pragmatics weight of words $\xi(w_i)$ to derive the adjacency matrix $\mathcal{A}^{\text{tr}} \in \mathbb{R}^{(m+n)\times(m+n)}$ of tr-graph by

$$\mathcal{A}_{i,j}^{\text{tr}} = \begin{cases} \sum\limits_{k\in\{i,j\}} \xi(w_k), & \text{if } \mathcal{T}(w_i, w_j) = 1 \\ 0, & \text{otherwise.} \end{cases} \tag{12}$$

D. Multigraph Sparse Interaction Block

This section discusses how to leverage fundamental word-level pragmatics dependencies of task representations based on heterogeneous graphs described in Section III-C. Since different graphs preserve different properties and structures, it is necessary to design a novel mechanism to harmonize the different information from heterogeneous graphs, so as to facilitate the information fusion of heterogeneous graphs during the learning process. A graph-aware module with a sparse interaction mechanism between graphs is proposed to facilitate information sharing between tasks. For each layer of heterogeneous graphs, MG-SIN performs two kinds of iterative learning: Intragraph Iterative Update and intergraph iterative sparse interaction (see Fig. 2).

The aim of Intragraph Iterative Update is to integrate the pragmatics information of a single graph into the context representation. Each node in the lth layer is updated by aggregating neighbor information of each node within a graph according to the adjacency matrices, which can be defined as

$$\boldsymbol{g}_{\text{intra}}^{\text{gt},l} = \sigma\left(\widehat{E}_{\text{gt}}^{\frac{1}{2}} \widehat{\mathcal{A}}^{\text{gt}} \widehat{E}_{\text{st}}^{-\frac{1}{2}} \boldsymbol{g}^{\text{gt},l} \boldsymbol{W}^{\text{gt},l}\right), \quad gt \in \{\text{st}, \text{se}, \text{tr}\} \tag{13}$$

where $\boldsymbol{W}^{\text{gt},l}$ is the weight matrix of gt-graph (gt $\in$ {st, se, tr}) for the lth layer, and $\sigma(\cdot)$ is a nonlinear activation function and $\boldsymbol{g}^{\text{gt},l}$ is the node feature matrix of gt-graph within the lth graph layer. Adjacent matrix $\widehat{\mathcal{A}}^{\text{gt}} = \mathcal{A}^{\text{gt}} + \boldsymbol{I}$, where $\boldsymbol{I}$ is the identity matrix and $\widehat{E}_{\text{gt}} = \sum_j \widehat{\mathcal{A}}_j^{\text{gt}}$. Noting that the initial inputs of all graphs are $\boldsymbol{h}$ derived from the encoding block in Section III-B, defined as, $\boldsymbol{g}^{\text{gt},0} = \boldsymbol{h} = \{h_1, h_2, \ldots, h_{m+n}\}$.

Intergraph Iterative Sparse Interaction aims to automatically fuse the pragmatics information of different graphs to help task obtain useful information while filtering the noisy information that hinders the learning of all tasks. More concretely, MG-SIN seeks an adaptive heterogeneous graph information interaction mechanism that decides what node information of nodes of graph should be shared across which tasks and what node information should be task-specific, which can improve the performance of all tasks.

MG-SIN seeks a binary random variable $\boldsymbol{u}_l^t \in \mathbb{R}^{m+n}$ ($t \in \{\text{st}, \text{se}\}$) for each layer l and each task t that determines whether each node of tr-graph in the lth layer is selected to propagate/exchange information between different graphs or skipped, yielding the best overall performance on all learning tasks. The process of intergraph iterative sparse interaction is defined as

$$\boldsymbol{g}_{\text{inter}}^{t,l+1} = \boldsymbol{g}_{\text{intra}}^{t,l} + \boldsymbol{W}^{t,l} \odot \boldsymbol{u}_l^t \times \boldsymbol{g}_{\text{intra}}^{\text{tr},l}, \quad t \in \{\text{st}, \text{se}\} \tag{14}$$

$$\boldsymbol{g}_{\text{inter}}^{\text{tr},l+1} = \alpha\ \boldsymbol{g}_{\text{intra}}^{\text{tr},l} + \frac{1-\alpha}{2} \boldsymbol{g}_{\text{inter}}^{\text{st},l} + \frac{1-\alpha}{2} \boldsymbol{g}_{\text{inter}}^{\text{se},l} \tag{15}$$

where $\odot$ indicates the hadamard product, α is the trade-off weight, and $\boldsymbol{W}^{t,l}$ is the trainable parameter. Here, we use $\boldsymbol{g}_{\text{inter}}^{t,l+1}$ to indicate the output of stance-specific and se-graphs after multigraph sparse interaction block. Note that, for sake of simplicity, this article defines $\boldsymbol{g}^{\text{st}} = \boldsymbol{g}_{\text{inter}}^{\text{st},l+1}$ and $\boldsymbol{g}^{\text{se}} = \boldsymbol{g}_{\text{inter}}^{\text{se},l+1}$.

However, the binary random variable $\boldsymbol{u}_l^t$ is discrete and nondifferentiable, and it is impossible to directly optimize $\boldsymbol{u}_l^t$. MG-SIN adopts Gumbel-Softmax Sampling trick [18] to address this issue, so as to use backpropagation. Instead of directly sampling discrete variable $\boldsymbol{u}_l^t$, this article generates it by

$$\boldsymbol{u}_l^t = \underset{j\in\{0,1\}}{\arg\max}\left(\log \pi_l^t(j) + G_l^t(j)\right) \tag{16}$$

where $\pi_l^t = [1 - \boldsymbol{\beta}_l^t, \boldsymbol{\beta}_l^t]$ is the distribution vector of binary random variable $\boldsymbol{u}_l^t$ and $\boldsymbol{\beta}_l^t$ denotes the probability matrix that $\boldsymbol{u}_l^t = \mathbf{1}$. G_l^ts are i.i.d samples drawn from Gumbel(0,1).[3] Due to the nondifferentiable operation argmax in (16), MG-SIN uses the reparameterization trick [18] to relax one-hot ($\boldsymbol{u}_l^t \in \{0, 1\}$) to $\boldsymbol{v}_l^t \in \mathbb{R}^2$

$$\boldsymbol{v}_l^t(j) = \frac{\exp\left(\left(\log \pi_l^t(j) + G_l^t(j)\right)/\tau\right)}{\sum_{i\in\{0,1\}} \exp\left(\left(\log \pi_l^t(i) + G_l^t(i)\right)/\tau\right)} \tag{17}$$

where $j \in \{0, 1\}$ and τ is the temperature of softmax operation. Note that the related variable $\boldsymbol{v}_l$ are given by (17) in both forward and backward passes.

E. Task Representations

This section captures significant task representations based on the final graph representations and word representations derived from BERT. Due to target-based stance detection, this article calculates the target-oriented stance representations $\boldsymbol{g}_{\text{mask}}^{\text{st}}$. MG-SIN first masks out vectors of nontarget words (words in tweet text) by mask matrix mask and keeps the

[3]The Gumbel(0,1) distribution is sampled by drawing $\boldsymbol{U} \sim \text{Uniform}(0,1)$ and $G = -\log(-\log \boldsymbol{U})$.

target word unchanged as follows:

$$\text{mask}_i = \begin{cases} 0, & 0 \le i < n \\ 1, & n \le i < n+m. \end{cases} \tag{18}$$

Therefore, the target-oriented graph representation are obtained by $\boldsymbol{g}^{\text{st}}_{\text{mask}} = \text{mask} \odot \boldsymbol{g}^{\text{st}}$, where $\boldsymbol{g}^{\text{st}}$ is the final output of st-graph. A retrieval-based attention mechanism [19] is employed to retrieve significant features relevant to the target words. The target-aware attention weights are defined as

$$\psi_j = \sum_{i=1}^{m+n} h_j^\top \boldsymbol{g}^{\text{st},i}_{\text{mask}}, \quad \varepsilon_j = \frac{\exp(\psi_{\text{j}})}{\sum_{i=1}^{n} \exp(\psi_i)} \tag{19}$$

where h_j is the vector representation of jth word derived from BERT in Section III-B. The final stance representation $\boldsymbol{r}^{\text{st}}$ is computed by

$$\boldsymbol{r}^{\text{st}} = \sum_{i=1}^{m+n} \varepsilon_i h_i. \tag{20}$$

Since sentiment analysis task is sentence-based, not target-based, it is unnecessary to calculate the target-aware attention weights for sentiment analysis. Therefore, based on attention mechanism, this article computes the sentiment representation $\boldsymbol{r}^{\text{se}}$ by

$$\psi'_j = \sum_{i=1}^{m+n} h_j^\top \boldsymbol{g}^{\text{se}}_i, \quad \varepsilon'_j = \frac{\exp(\psi'_{\text{j}})}{\sum_{i=1}^{n} \exp(\psi'_i)} \tag{21}$$

where $\boldsymbol{g}^{\text{se}}$ is the final output of se-graph. The final sentiment representation $\boldsymbol{r}^{\text{se}}$ is computed by

$$\boldsymbol{r}^{\text{se}} = \sum_{i=1}^{m+n} \varepsilon'_i h_i \tag{22}$$

where h_i is the vector representation of ith word derived from BERT in Section III-B.

F. Model Training

To facilitate each task to selectively fuse the useful information and filter the noisy information, it is desirable to form a compact graph-aware interaction model in our proposed method. Therefore, this article proposes a regularization $\mathcal{L}_{\text{sp}}$ to guarantee the sparsity of interactions between heterogeneous graphs by

$$\mathcal{L}_{\text{sp}} = \frac{\sum_{l \le L} \log \boldsymbol{u}^{\text{st}}_l + \sum_{l \le L} \log \boldsymbol{u}^{\text{se}}_l}{2}. \tag{23}$$

Moreover, to avoid excessive sparse interactions between heterogeneous graphs caused by $\mathcal{L}_{\text{sp}}$, that is, the graph-aware sparse interaction model degrades into three separate graph networks (st-graph, se-graph, and tr-graph), MG-SIN proposes a regularization $\mathcal{L}_{\text{sh}}$ to encourage information sharing across graphs. Since bottom layers of heterogeneous graph layers contain low-level knowledge, MG-SIN minimizes the weighted sum of L_1 distances between different tasks to encourage the sharing at bottom layers, which is defined as

$$\mathcal{L}_{\text{sh}} = \frac{1}{2} \sum_{l \le L-1} \frac{L-l}{L} \left(\left| \boldsymbol{u}^{\text{st}}_{l+1} - \boldsymbol{u}^{\text{st}}_l \right| + \left| \boldsymbol{u}^{\text{se}}_{l+1} - \boldsymbol{u}^{\text{se}}_l \right| \right) \tag{24}$$

where L is the max layer number of iterative learning.

TABLE I

STATISTICS OF SEMEVAL16 AND COVID19 DATASET

Dataset	Target	# Train	# Test
SemEval16	Atheism (AT)	513	220
	Climate Change is Concern (CC)	395	169
	Feminist Movement (FM)	664	285
	Hillary Clinton (HC)	689	295
	Legalization of Abortion (LA)	653	280
	Total	2914	1249
COVID19	Wearing a Face Mask (WFM)	843	124
	Anthony S. Fauci, M.D. (AFM)	1018	123
	Keeping Schools Closed (KSC)	705	134
	Stay at Home Orders (SHO)	842	144
	Total	3408	525

The final objective function of stance detection and sentiment analysis task is defined by cross-entropy loss and L_2-regularization

$$\mathcal{L}(\Theta) = \lambda_1 \sum_{i=1}^{|\mathbf{S}|} y^{\text{st}}_i \log \hat{y}^{\text{st}}_i + \lambda_2 \sum_{i=1}^{|\mathbf{S}|} y^{\text{se}}_i \log \hat{y}^{\text{se}}_i + \lambda_3 \mathcal{L}_{\text{sp}} + \lambda_4 \mathcal{L}_{\text{sh}} \tag{25}$$

where λ_1, λ_2, λ_3, and λ_4 are the weights of stance detection, sentiment analysis task, sparsity, and sharing regularization term. $\hat{y}^{\text{task}}_i$ is the probability distribution of task representation of ith tweet by fully connected layer, $\hat{y}^{\text{task}}_i = \text{softmax}(\boldsymbol{W}^{\text{task}} \boldsymbol{r}^{\text{task}} + \boldsymbol{b}^{\text{task}})$ and y^{task}_i is the ground-truth label of ith tweet. Θ is the total parameters.

IV. EXPERIMENTAL SETUP

A. Dataset Description

The proposed model MG-SIN is evaluated on two benchmark datasets on social media in both stance detection and sentiment analysis scenarios.

1) SemEval16 [20] is from subtask A of SemEval-2016 task 6, which collects users' tweets toward a specific target, and each tweet is annotated with a stance and sentiment label (Positive, Neither, Negative). SemEval16 contains five targets with 2914 labeled training data instances and 1249 test instances. The statistics of SemEval16 are shown in Table I.
2) COVID19 [6] contains four controversial targets that arose as the virus continued its spread in the United States (USA), and each tweet is annotated with a stance and sentiment label (Positive, Neither, Negative). Due to Twitter's information privacy policy, we only crawl Twitter by using the Twitter Streaming API[4] according to the tweet ID lists given by [6]. The detailed statistics of COVID19 are listed in Table I.
3) CLiCS[5] includes 9700 tweets toward the target, Climate Change, and each tweet is annotated with a stance and sentiment label (Positive, Neither, Negative). Due to Twitter's information privacy policy, we only crawl Twitter by using the Twitter Streaming API[6] according to the tweet ID lists given by [21].

[4]https://developer.twitter.com/en/portal/petition/in-review

[5]https://github.com/apoorva-upadhyaya/Elsevier_StanceDetection_ClimateChange/tree/main/data

[6]https://developer.twitter.com/en/portal/petition/in-review

TABLE II
COMPARISON OF THE STANCE DETECTION RESULTS OVER FIVE TARGETS ON THE SEMEVAL16 DATASET (%)

Categories	Models	F_{avg}	F_{favor}	$F_{against}$	AT	CC	HC	FM	LA	$MacF_{avg}$
					F_{avg}	F_{avg}	F_{avg}	F_{avg}	F_{avg}	
STL	CT-BERT	69.01	63.88	74.14	67.83	50.73	63.95	61.51	64.63	61.73
	T-DAN	50.04	37.67	62.42	60.71	49.23	51.32	58.35	56.05	55.13
	SCN	57.26	45.99	68.53	61.98	46.30	55.46	52.12	55.62	54.30
	KEMLM	57.27	46.18	68.36	56.68	42.80	60.96	59.88	59.31	55.93
	HAN	69.79	67.75	71.82	70.53	49.56	61.23	57.50	66.16	61.00
	MELT	73.48	71.34	75.62	66	**71**	67	63	66	66.6
	ASDA	70.47	68.41	72.51	74.93	59.31	67.01	56.43	61.66	63.87
Graph	TextGCN	66.27	58.68	73.87	66.52	52.76	55.85	57.29	64.64	59.41
	TextING	67.43	61.54	73.32	67.27	52.96	58.76	58.91	63.27	60.23
	TensorGCN	66.52	59.54	73.50	68.19	49.92	54.79	57.69	63.07	58.73
MTL	AT-JSS	72.33	70.22	74.43	69.22	59.18	68.33	61.49	68.41	65.33
	Tchebycheff	58.11	51.40	64.82	54.21	48.95	54.85	51.63	46.05	51.14
	BanditMTL	57.83	51.91	63.75	53.27	51.52	52.21	53.85	50.63	52.29
	MTIN	70.28	68.23	72.32	65.86	65.17	65.77	62.89	64.56	64.85
	SP-MT	73.99	71.84	76.15	69.51	63.52	63.22	67.51	**70.54**	66.84
Ours	MG-SIN *w/o* SA	68.35	62.42	74.28	68.13	52.06	56.86	58.47	61.55	59.41
	MG-SIN (Ours)	**76.06**	**73.84**	**78.27**	**75.27**	64.02	**69.23**	**68.50**	67.66	**68.94**

B. Evaluation Metrics and Implementation Details

The official evaluation metrics of SemEval16 are used to evaluate the performance of proposed model both on SemEval16 and COVID19 dataset, which is the F_{avg} for the Favor and Against categories and macroaverage of the F1-score (MacF_{avg}), with None category disregarded [20], [22], [23]. The F1 score of label Favor and Against is computed as follows:

$$F_{\text{favor}} = \frac{2P_{\text{favor}}\ R_{\text{favor}}}{P_{\text{favor}}\ + R_{\text{favor}}}, \quad F_{\text{against}} = \frac{2P_{\text{against}}\ R_{\text{against}}}{P_{\text{against}}\ + R_{\text{against}}} \tag{26}$$

where P and R are precision and recall, respectively. The F1 average is computed as

$$F_{\text{avg}} = \frac{F_{\text{favor}}\ + F_{\text{against}}}{2}. \tag{27}$$

This article calculates the F_{avg} for each target. This article averages the F_{avg} on each target to get MacF_{avg}. Note that the overall F_{avg} is not the average of F_{avg} of each target, but for all of the testing data. For sentiment analysis task, this article uses the Accuracy (Acc) and F1 score (F1-score) across all targets to evaluate the performance. This article also averages the Acc and F1-score on each target to get Acc_{avg} and F1-$\text{score}_{\text{avg}}$ to evaluate the whole performance of model on all targets.

This article follows [6], [20] and use the official train/test splits in the SemEval16 and COVID19 dataset for all comparison models. In our MG-SIN, the dimension of embedding is set as 768. The number of GCN blocks is set to 3. Adam is used as the optimizer with learning rate of $1e{-}5$ to train the model, and the batch size is set to 16, and $(\lambda_1, \lambda_2, \lambda_3, \lambda_4) = (0.8, 0.7, 5e{-}4, 5e{-}4)$. The optimal α is set to 0.4. The version of python is v3.8.3, and that of PyTorch is v1.7.0. For reducing the experimental error, this article performed three independent runs for each model to account for variability, and report average results over the three runs.

C. Comparison Methods

For the fairness of comparative experiments, this article uses various comparison models to compare with our MS-GIN both on stance detection and sentiment analysis task, which can be categorized into three categories.

1) *STL Methods:* This article adopts seven models to compare with MG-SIN model on stance detection task. Specifically, CT-BERT [24] is a transformer-based model, pretrained on a large corpus of Twitter text on the topic of COVID-19. T-DAN [25] used an attention calculation method to locate the crucial words for the targets. SCN [26] proposed the stance-wise convolution module to absorb the correlation between stances. KEMLM [27] is a BERT-based method that uses weighted log-odds-ratio to capture the distribution of stance. HAN [28] used hierarchical attention approach to well leverage various linguistic information. MELT [29] proposed a hierarchical message-encoder pretrained transformer of stance detection. ASDA [30] used data augmentation method to improve the stance detection. For sentiment analysis task, this article adopts several models to compare with MG-SIN. Specifically, MCNN-MA [31] utilized the multichannel CNN to learn the sentiment representation. ABCDM [32] captured the temporal information by employing two Bi-LSTM layers and attention mechanism to improve the sentiment analysis. Co_LSTM [33] proposed a hybrid network of CNN and LSTM to classify the sentiment.
2) *Graph-Based Methods:* TextGCN [12] built a text graph for a sentence based on the relationship between words. TextING [34] fulfilled the inductive learning of new words and then induced the sentiment distribution of each sentence. TensorGCN [35] proposed a new framework TensorGCN for text classification.
3) *MTL Methods:* AT-JSS [22] utilized attention mechanism to incorporate the target and sentiment information

to identify the stances. Tchebycheff [36] proposed a novel Tchebycheff procedure for multitask text classification problems. BanditMTL [37] utilized adversarial multiarmed bandit to improve the multitask text classification. SP-MT [38] and MTIN [13] captured interaction relationship between tasks to improve the stance detection and sentiment analysis task simultaneously. Note that this article adopts an additional method for sentiment analysis task, that is, MTL-SFU [39] that took negation as an auxiliary task to help improve the performance of sentiment analysis.

V. Experimental Results and Analysis

This section discusses and analyzes the experimental results of our MG-SIN. Sections V-A and V-B have demonstrated the comparison results on stance detection and sentiment analysis tasks, respectively. Subsequently, this article analyzes the impact of pragmatics weight of words (in Section V-C) and sparse interaction block (in Section V-D). Section V-E shows the ablation study of our MG-SIN. Finally, this article explores the impact of coefficients of different tasks (in Section V-G), tradeoff parameter (in Section V-F), and the number of heterogeneous graphs iterations (in Section V-H).

A. Performance of Stance Detection Task

1) Performance on SemEval16 Dataset: Table II shows the performances of all comparable approaches and our model in terms of the F_{avg} and MacF_{avg} on the SemEval16 dataset. It can be observed that our MG-SIN achieves the best performance on overall evaluation metrics (F_{avg}, F_{against}, F_{favor}, and MacF_{avg}) and shows good performance on various topics. Specifically, MTL methods have comparative performance (73.99% in F_{avg}), perhaps suggesting that semantically sparse tweets and noisy information, such as grammar errors, abbreviations of terms, and spelling errors, cannot be tackled well, and these MTL models are insufficient to learn the richer stance representation. This demonstrates that MG-SIN with employing spare interaction mechanism, outstandingly improves the performance of stance detection by leveraging the useful information from the other learning task. Graph-based methods have the worst performance (67.43% in F_{avg}), which is remarkably lower than our MG-SIN. This finding suggests that our proposed task-related pragmatics graphs indeed improve the performance of stance detection task. Single task learning (STL) methods have second place performance (73.48% in F_{avg}), which illustrates that leveraging pragmatics information of words can potentially help obtain richer stance representation. The performance of MG-SIN w/o SA (removing the Sentiment Analysis task from MG-SIN) is significantly lower than MG-SIN, which illustrates the benefits of our proposed task spare interaction mechanism. Overall, the empirical results show that our model could significantly improve stance detection on SemEval16 dataset.

2) Performance on COVID19 Dataset: To illustrate the effectiveness of MG-SIN in larger dataset, this article conducts the comparison experiments on the COVID19 dataset. Table III shows the detailed stance detection results in terms of F_{avg} and MacF_{avg} on the COVID19 dataset. It is clear that MG-SIN achieves the best performance on overall evaluation and shows good performance on various topics other than WFM. Specifically, graph-based methods that only consider the semantic information of words have the worst performance (60.34% in F_{avg}) because of ignoring the pragmatics information of words. MTL methods have comparative performance (78.11% in F_{avg}), remarkably lower than our MG-SIN, which shows the effectiveness of our MG-SIN. STL methods have median performance with MG-SIN, such as MELT and CT-BERT model. However, there is still a gap between MG-SIN because MG-SIN can utilize the information from other tasks to facilitate the learning of itself. Note that MG-SIN w/o SA model is constructed by removing the sentiment analysis task, which obtains poor performance. This demonstrates that modeling the relationship between tasks at word level can outstandingly improve the performance of stance detection. Therefore, the experimental results show the effectiveness of our MG-SIN.

3) Performance on CLiCS Dataset: Table IV shows the performance of all comparison methods and our proposed MG-SIN method in terms of the F_{favor}, F_{against}, and MacF_{avg} on CLiCS dataset. We can observe that our MG-SIN achieves the best performance on all evaluation metrics, which demonstrates the effectiveness and superiority of our proposed MG-SIN. Concretely, our MG-SIN method obtains 3.9% improvement in MacF_{avg} compared to the existing SOTA baselines. Our proposed MG-SIN method is superior to existing MTL methods, such as Tchebycheff, BanditMTL, MTIN, and SP-MT. This illustrates that our proposed spare interaction mechanism can fully capture the interaction feature between stance detection and sentiment analysis task, so as to facilitate the learning of target-oriented stance representation.

B. Performance of Sentiment Analysis Task

1) Performance on SemEval16 Dataset: To verify that our MG-SIN can improve the performance of stance detection and sentiment analysis tasks simultaneously, this article also reports the comparison results of sentiment analysis task in terms of the Acc and F1-score (Acc_{avg} and F1-$\text{score}_{\text{avg}}$ are the average of Acc and F1-score on all targets) on SemEval16 dataset. As shown in Table V, it is observed that our MG-SIN achieves the best performance on overall evaluation metrics and shows best performance on various topics. Compared with existing best STL models, such as CT-BERT and MG-SIN has 3.96% improvement and 9.53% improvement in Acc_{avg} and F1-$\text{score}_{\text{avg}}$, respectively, which verifies the effectiveness of our MG-SIN in sentiment analysis task. Owing to the failure of modeling the fine-grained relationship between tasks, AT-JSS, MTL-SFU, MTIN, and SP-MT overall perform worst since they neither leverage the task-specific pragmatics information nor filter the noisy information shared by the other task. Analogously, graph-based methods can capture rich semantic information, but there is still a gap between MG-SIN and graph-based methods due to the negligence of word-category pragmatics dependency relationship at word level. Comparatively, the model that considers the pragmatics information of words and semantic information of sentence (MG-SIN w/o

TABLE III
COMPARISON OF THE STANCE DETECTION RESULTS OVER FOUR TARGETS ON THE COVID19 DATASET (%)

Categories	Models	F_{avg}	F_{favor}	$F_{against}$	WFM	AFM	KSC	SHO	$MacF_{avg}$
					F_{avg}	F_{avg}	F_{avg}	F_{avg}	
STL	CT-BERT	76.67	80.74	72.59	**80.84**	66.06	78.64	68.99	74.86
	T-DAN	68.31	72.30	64.31	67.04	63.57	56.52	56.82	60.99
	SCN	56.65	64.10	49.19	68.44	49.85	46.79	44.16	52.31
	KEMLM	56.09	65.04	47.15	70.69	63.02	54.33	55.62	60.92
	HAN	73.25	78.02	70.30	73.56	67.36	75.47	72.14	72.18
	MELT	77.58	80.70	74.46	77.45	72.71	82.24	74.51	76.55
	ASDA	74.40	77.39	71.41	74.28	69.73	78.87	71.46	73.42
Graph	TextGCN	58.96	69.50	48.42	49.84	52.46	60.75	56.62	54.92
	TextING	60.34	68.99	51.68	55.05	55.88	62.61	59.77	58.33
	TensorGCN	59.42	67.91	50.92	51.84	58.68	61.83	61.37	58.43
MTL	AT-JSS	69.77	74.45	65.09	54.24	64.73	56.49	60.20	58.91
	Tchebycheff	66.27	71.42	61.11	62.69	51.34	56.67	56.25	56.73
	BanditMTL	68.05	69.44	66.67	63.81	56.95	60.00	65.15	61.48
	MTIN	67.97	70.70	65.24	60.53	64.92	65.67	70.21	65.33
	SP-MT	78.11	80.39	75.84	77.73	72.97	82.54	74.78	77.00
Ours	MG-SIN *w/o* SA	59.72	65.74	53.69	59.69	58.87	60.46	57.77	59.20
	MG-SIN (Ours)	**80.30**	**83.53**	**77.07**	80.17	**75.26**	**85.13**	**77.13**	**79.42**

TABLE IV
COMPARISON OF THE STANCE DETECTION RESULTS ON CLICS DATASET (%)

Models	F_{favor}	$F_{against}$	$MacF_{avg}$
CT-BERT	92.44	69.93	81.19
T-DAN	91.26	65.92	78.59
KEMLM	92.31	71.14	81.72
TextGCN	90.35	60.67	75.51
Tchebycheff	88.23	69.67	78.94
BanditMTL	84.84	62.52	73.68
MTIN	87.76	51.31	69.53
SP-MT	92.25	57.57	74.91
MG-SIN (Ours)	**93.56**	**76.45**	**85.01**

TABLE V
COMPARISON OF THE SENTIMENT ANALYSIS RESULTS OVER FIVE TARGETS ON THE SEMEVAL16 DATASET (%)

Models	AT		CC		HC		FM		LA		Acc_{avg}	$F1\text{-}score_{avg}$
	Acc	*F1-score*	*Acc*	*F1-score*	*Acc*	*F1-score*	*Acc*	*F1-score*	*Acc*	*F1-score*		
CT-BERT	80.61	83.12	72.58	75.80	85.54	85.17	82.00	74.95	78.81	79.06	79.90	79.62
MCNN-MA	64.99	61.82	54.04	50.29	70.40	66.70	78.70	61.96	69.52	61.79	67.53	60.51
Co_LSTM	64.70	60.24	49.31	53.88	71.41	65.62	74.90	58.87	64.17	57.63	64.89	59.25
ABCDM	71.06	71.52	53.85	56.26	74.24	73.02	72.50	59.47	69.40	62.49	68.21	64.55
TextGCN	78.52	80.30	70.48	75.03	84.21	83.21	80.70	72.23	76.62	75.47	78.10	77.25
TextING	79.76	80.68	69.19	75.80	85.34	84.83	80.60	70.81	77.43	75.31	78.46	77.49
TensorGCN	78.97	80.31	67.72	74.41	84.68	84.06	81.00	73.29	79.63	77.99	78.40	78.02
AT-JSS	76.21	77.07	62.92	66.30	69.60	67.35	76.10	53.70	69.76	57.64	70.93	64.41
Tchebycheff	68.65	62.62	58.22	27.64	77.78	80.47	73.30	46.17	71.67	65.73	69.93	56.53
BanditMTL	66.99	61.22	55.62	46.42	74.36	75.78	74.60	57.94	55.56	55.81	65.43	59.43
MTL-SFU	67.73	68.27	47.14	51.97	70.28	66.27	73.20	62.11	70.95	63.56	65.86	62.44
MTIN	79.66	82.32	73.63	76.68	85.28	84.94	85.86	72.86	78.85	81.64	80.66	79.69
SP-MT	78.94	80.01	65.18	68.83	72.10	69.92	78.83	63.02	74.24	73.34	73.86	71.02
MG-SIN *w/o* ST	79.76	80.59	72.10	76.13	84.68	80.74	81.01	71.47	80.56	79.00	79.62	77.59
MG-SIN (Ours)	**84.52**	**87.42**	**75.82**	**81.34**	**88.39**	**88.09**	**88.30**	**84.91**	**82.29**	**82.99**	**83.86**	**84.95**

ST) performs slightly better but still lower than our MG-SIN. This demonstrates that our proposed sparse interaction mechanism of tasks can significantly improve the performance of sentiment analysis task.

2) Performance on COVID19 Dataset: To illustrate the generalization of MG-SIN, this article evaluates the performance of MG-SIN on a larger dataset (COVID19 dataset) shown in Table VI. It is clear that MG-SIN consistently outperforms all comparison models in terms of the Acc and F1-score. Among them, compared with previous promising graph-based methods, such as TensorGCN, our MG-SIN improves 5.35% on $\mathrm{Acc}_{\mathrm{avg}}$ and 11.69% on $\mathrm{F1\text{-}score}_{\mathrm{avg}}$, which verifies that

TABLE VI
COMPARISON OF THE SENTIMENT ANALYSIS RESULTS OVER FOUR TARGETS ON THE COVID19 DATASET (%)

Models	WFM		AFM		KSC		SHO		Acc_{avg}	$F1\text{-}score_{avg}$
	Acc	*F1-score*	*Acc*	*F1-score*	*Acc*	*F1-score*	*Acc*	*F1-score*		
CT-BERT	70.10	63.28	74.25	78.02	73.88	73.41	79.40	73.13	74.41	71.96
MCNN-MA	68.38	54.46	55.28	49.09	72.39	66.28	69.00	47.46	66.26	54.32
Co_LSTM	56.85	51.77	57.72	47.29	73.63	54.66	70.10	50.38	64.30	51.02
ABCDM	59.98	51.82	55.83	52.38	67.16	62.94	70.60	57.69	63.39	56.21
TextGCN	70.57	60.38	71.24	67.67	77.78	71.34	78.50	64.02	74.51	65.85
TextING	71.96	59.83	69.15	66.51	78.47	73.75	77.50	70.36	74.28	67.61
TensorGCN	72.22	64.55	72.68	73.32	**79.86**	70.21	79.90	71.11	76.16	69.80
AT-JSS	68.98	40.82	66.12	59.28	69.39	68.66	63.10	56.11	66.91	56.22
Tchebycheff	63.74	51.45	64.77	44.65	50.00	53.57	63.10	50.59	60.39	50.07
BanditMTL	64.99	65.64	65.04	39.47	66.67	73.33	66.70	53.33	65.84	57.95
MTL-SFU	64.16	54.43	56.64	55.68	69.90	66.12	75.00	66.21	66.42	60.61
MTIN	76.32	72.34	76.42	75.33	77.87	80.85	83.64	76.47	78.56	76.24
SP-MT	76.32	72.39	77.04	83.01	75.85	75.91	80.05	78.93	77.32	77.56
MG-SIN *w/o* ST	70.05	65.62	71.24	71.58	76.62	75.57	81.94	77.55	74.96	72.58
MG-SIN (Ours)	**79.43**	**79.45**	**80.18**	**85.33**	78.94	**78.04**	**87.50**	**83.13**	**81.51**	**81.49**

TABLE VII
COMPARISON OF THE SENTIMENT ANALYSIS RESULTS ON CLiCS DATASET (%)

Models	Acc_{avg}	$F1\text{-}score_{avg}$
CT-BERT	60.13	61.81
ABCDM	39.66	35.99
TextGCN	47.35	53.31
Tchebycheff	41.66	44.12
BanditMTL	58.05	61.72
MTIN	56.85	60.03
SP-MT	44.20	37.24
MG-SIN (Ours)	**62.81**	**64.62**

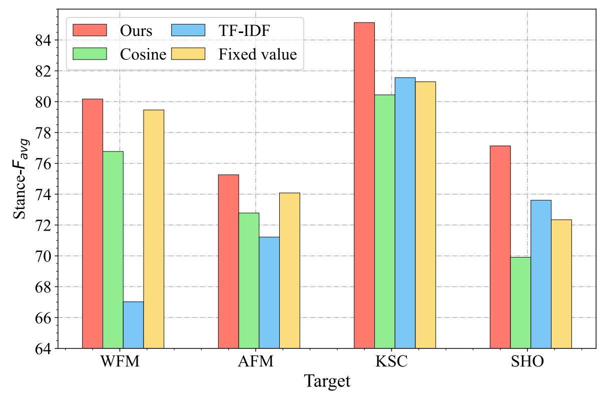


Fig. 3. Impact of different edge weight of heterogeneous graphs.

leveraging the task-specific pragmatics dependency information of words could potentially lead to improved sentiment analysis results. Compared with previous noteworthy MTL methods, such as SP-MT, our MG-SIN improves 4.19% on Acc_{avg} and 3.93% on $\text{F1-score}_{\text{avg}}$, which further demonstrates that the superiority of sparse interaction mechanism between tasks. The STL methods (such as CT-BERT) perform slightly worse than our MG-SIN but comparable performance with MG-SIN w/o ST constructed by removing the stance detection task from MG-SIN, which further verifies the superiority of multitask sparse interaction mechanism. Overall, the experimental results on COVID19 dataset show the effectiveness and superiority of our MG-SIN.

3) Performance on CLiCS Dataset: Table VII shows the performance of sentiment analysis task on CLiCS dataset. It is clear that our proposed MG-SIN method achieves best performance on all evaluation metrics compared to existing SOTA baselines, demonstrating the effectiveness and superiority of our proposed method. Specifically, our proposed MG-SIN obtains 2.9% improvement on sentiment analysis task compared to existing SOTA MTL baselines, such as Tchebycheff, BanditMTL, MTIN, and SP-MT. This demonstrates that our proposed MG-SIN is superior to existing MTL methods and our proposed multigraph sparse interaction mechanism indeed help improve performance of all tasks simultaneously.

C. Impact of Task-Specific Pragmatics Weight

To further verify that the task-specific pragmatics information can enrich the heterogeneous graph representation and improve the performance of task, experiments are conducted on COVID19 dataset by fixing the structure of MG-SIN network and replacing the task-specific pragmatics weight with cosine similarity (MG-SIN with Cosine), or word frequency (MG-SIN with TF-IDF), or fixed value "1" (MG-SIN with Fixed value). As shown in Fig. 3, it is observed that our MG-SIN significantly outperforms all other variants of MG-SIN across all targets, which verifies the effectiveness of task-specific pragmatics information in learning task representations. Specifically, the model MG-SIN with Fixed value has comparable performance with our MG-SIN on WFM and AFM targets, but it has poor performance on KSC and SHO targets which is remarkably lower than our MG-SIN. This finding suggests that adopting task-specific pragmatics information has better generalization than any other approaches across various targets. Overall, this experiment illustrates the effectiveness of task-specific pragmatics information.

D. Effectiveness of Sparse Interaction Block

To verify the significance of our proposed sparse interaction mechanism in improving performance of tasks, this article

TABLE VIII
ABLATION STUDY RESULTS ON COVID19 DATASET (%)

Model	WFM		AFM		KSC		SHO		Average	
	ST	SE	ST	SE	ST	SE	ST	SE	ST	SE
w/o mask matrix	77.41	73.10	72.50	81.26	77.14	76.44	72.84	80.75	74.97(↓4.45)	77.88(↓3.60)
w/o ST attention	73.43	73.25	64.72	81.17	71.51	74.83	72.50	77.02	70.54(↓8.88)	76.57(↓4.92)
w/o SE attention	79.39	72.78	69.99	80.79	75.99	74.10	75.58	75.94	75.24(↓4.18)	75.90(↓5.59)
w/o $\mathcal{L}_{sh}$	79.54	71.28	73.92	76.95	82.14	76.10	76.07	76.58	77.92(↓1.50)	75.23(↓6.26)
w/o $\mathcal{L}_{sp}$	78.66	77.27	71.58	80.74	81.01	**79.87**	75.19	72.42	76.61(↓2.81)	77.58(↓3.91)
w/o $\mathcal{L}_{sh}$ & $\mathcal{L}_{sp}$	78.36	71.65	71.14	77.84	78.15	73.12	75.00	73.46	75.66(↓3.76)	74.02(↓7.47)
MG-SIN (Ours)	**80.17**	**79.45**	**75.26**	**85.33**	**85.13**	78.04	**77.13**	**83.13**	**79.42**	**81.49**

ST and SE denote stance detection and sentiment analysis task respectively, **Average** is the average performance on all targets. This paper reports the F_{avg} and *F1-socre* of each target.

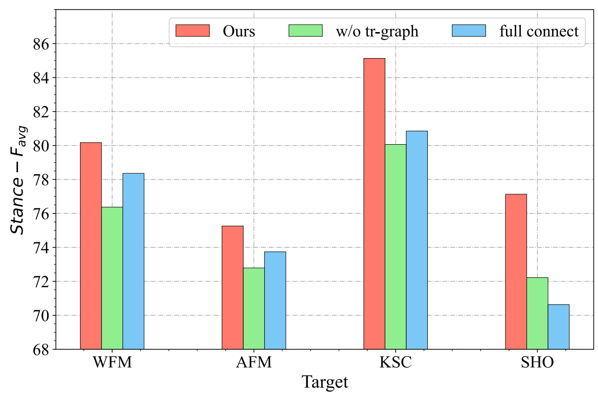


Fig. 4. Impact of sparse interaction block.

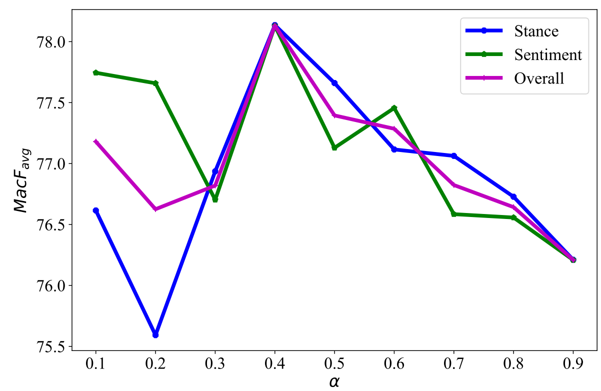


Fig. 5. Impact of the trade-off parameter α. Overall is the average performance of stance and sentiment tasks.

constructs two variants: 1) MG-SIN without tr-graph that is no interaction between st-graph and se-graph by removing tr-graph (Let $\boldsymbol{u}^{\text{st}}$ and $\boldsymbol{u}^{\text{se}}$ be constant value "0" described in Section III-D) and 2) MG-SIN with full connect that is to make tr-graph fully connect with st-graph and se-graph (Let $\boldsymbol{u}^{\text{st}}$ and $\boldsymbol{u}^{\text{se}}$ be constant value "1"). The experimental results are shown in Fig. 4. Noting that model MG-SIN without tr-graph performs overall worst, and thus it demonstrates that the interaction between tasks is significant in improving task performance. Model MG-SIN with full connect has the median performance, and its performance is remarkably lower than MG-SIN across all targets. This finding confirms our intuition presented in Section III-D that sparse interaction between tasks can help task filter the noisy information that hiders the learning of all tasks. Therefore, this experiment verifies the effectiveness and significance of our proposed sparse interaction mechanism.

E. Ablation Study

To analyze the impact of different components of our MG-SIN model, an ablation study is performed over stance detection (ST) and sentiment analysis (SE) task on COVID19 dataset and report the results in Table VIII. It is observed that the removal of mask matrix seriously degrades the performance of all tasks, especially stance detection task, which indicates that mask matrix is effective in improving task performance. Removal of ST attention module leads to evident performance degradation (8.88%) in ST task, which indicates the ST attention module is important in detection stances. Similarly, for sentiment analysis task, the removal of SE attention module also seriously reduces the performance (5.59%) of SE task, which illustrates that SE attention module indeed improves the performance of SE task. In addition, this article analyzes the impact of regularization terms (described in Section III-F) by removing the $\mathcal{L}_{\text{sh}}$ and $\mathcal{L}_{\text{sp}}$, and they have slight performance degradation. This indicates that employing these regularization terms can indeed help our MG-SIN model properly improve all task performance.

F. Impact of Trade-off Weight α

To investigate the impact of trade-off weight α (described in Section III-D), this article varies α from 0.1 to 0.9 and shows the results in Fig. 5. Noting that model with $\alpha = 0.4$ performs overall better than other values, and thus this article sets trade-off weight α to 0.4 in our MG-SIN. Moreover, in the case of the α greater than 0.4, the overall performance of MG-SIN fluctuates with the increasing α value and essentially tends to decline. Note that a larger α value represents less interaction among heterogeneous graphs described in (15), and thus it implies that too little or too many task interactions among task graphs can degrade the overall performance of MG-SIN. Therefore, it makes sense to set an appropriate α value to balance the performance of all tasks.

G. Impact of Task Weight λ_1 and λ_2

Considering the importance of the interaction between stance and sentiment tasks, this article investigates the impact

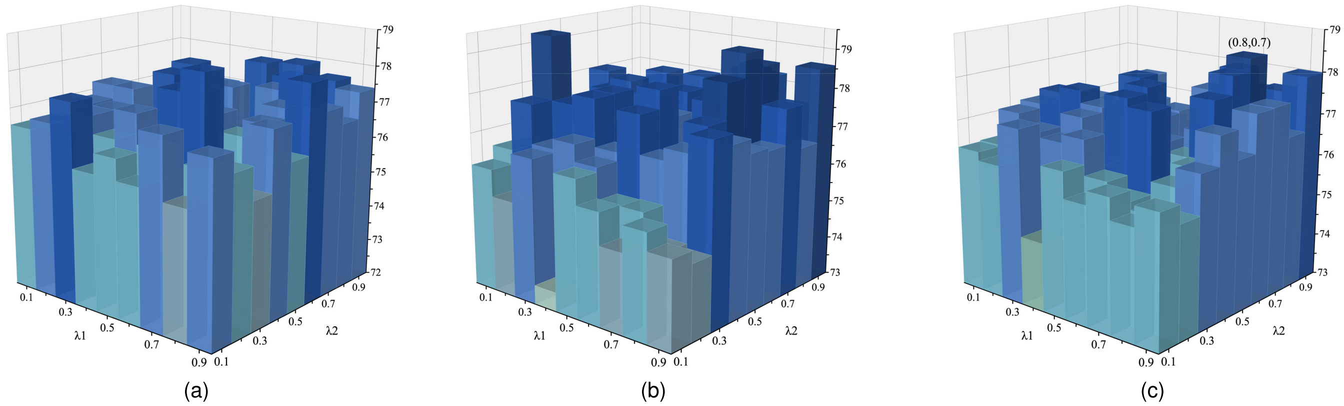


Fig. 6. Impact of different λ. (a) Performance of stance detection task. (b) Performance sentiment analysis task. (c) Average performance.

of the task weight λ_1 and λ_2 on the performance of our MG-SIN model. Specifically, this article varies the λ_1 and λ_2 from 0.1 to 0.9 and shows the results in Fig. 6. It is observed that the performance of MG-SIN varies as the value of λ, which illustrates that task weight is an important factor affecting the performance of the model. Noting that model with a larger value of λ performs overall better than lower value, and this article sets the (λ_1, λ_2) to (0.8,0.7) in our model. As shown in Fig. 6(a), the effect of λ_2 on stance task performance is greater than that of weight sentiment analysis task λ_1 on stance task. Similarly, Fig. 6(b) shows that the performance of sentiment analysis task fluctuates with the increasing value of λ_1. This implies that stance detection and sentiment analysis tasks can help each other improve their performance, and they can also degrade their performance to a certain extent.

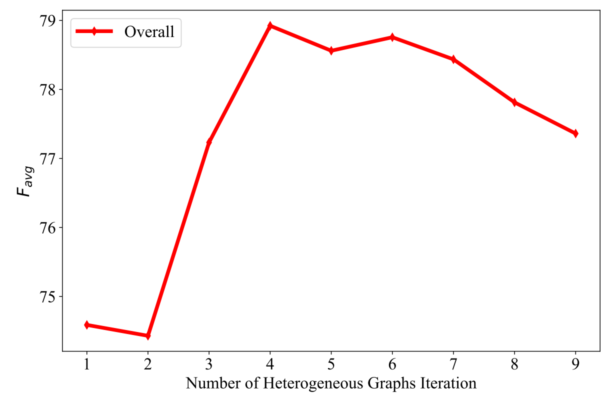


Fig. 7. Performance of setting different number of heterogeneous graph iterations.

H. Impact of Number of Task Graph Iterations

To analyze the impact of the heterogeneous graph iteration number on the performance of our proposed model, this article varies the iteration number from 1 to 9 and reports the overall experimental in terms of F_{avg} results on COVID19 dataset over four targets in Fig. 7. Note that the model with four iterations performs overall better than other values, and thus this article sets the number of heterogeneous graph iterations to four in our MG-SIN. When the number of graph iterations is less than three, the MG-SIN has poor performance since the inadequate network structure is not enough to fully exploit the task-specific pragmatics information for all learning tasks. Moreover, it is clear that the performance of MG-SIN tends to decline as the number of graph iterations increases. This implies that roughly increasing the number of iterations can easily degrade the learning ability of the model due to a sharp increase in model parameters.

I. Case Study

To better understand how our MG-SIN works, this article selects two sentences from SemEval16 and COVID19 datasets to analyze the stance detection task. As shown in Table IX, this article presents a case study and visualizes the attention weights of each word toward given targets, together with their predictions of stance detection task and corresponding ground truth labels. Note that, the background color of a word represents the varying levels of its importance to the stance detection task, with darker colors indicating a higher significance of the word to task representation.

The first example "It should be a hate crime if someone makes a threat against women." has the subjective words "hate crime," which distinctly showcases the user's standpoint on the target, and the subjunctive word "should," which could complicate the identification of underlying semantics. BERT and MTIN fail to capture the importance of words "hate crime" and "should," leading to inaccurate predictions. TextGCN and MG-SIN successfully capture the correlation between those words and the given target, so as to make accurate predictions The second example "Avoiding places where I would be asked to wear a mask." includes the words of negative meaning "Avoiding" in the sentence, which has the potential to confuse models, leading them to incorrect stances. BERT, TextGCN, and MTIN are capable of capturing the negation, yet they are failing to establish the relationship between the negative word and the given target, ultimately resulting in incorrect predictions. Our proposed MG-SIN correctly captures the relationship between words and given target by leveraging the task-related pragmatics dependency information, which implies that our MG-SIN effectively harmonizes the pragmatics dependency information and semantic information. This

TABLE IX

CASE STUDY. VISUALIZATION OF ATTENTION SCORES FROM BERT, TEXTGCN, MTIN, AND MG-SIN ON TESTING EXAMPLES OF SEMEVAL16 AND COVID-19 DATASETS, ALONG WITH THEIR PREDICTIONS AND CORRESPONDING GROUND TRUTH LABELS

Model	Target	Attention Visualization	Prediction	Label
BERT	FM	It should be a hate crime if someone makes a threat against women	Against✗	Favor
	WFM	Avoiding places where I would be asked to wear a mask.	Favor✗	Against
TextGCN	FM	It should be a hate crime if someone makes a threat against women	Favor✓	Favor
	WFM	Avoiding places where I would be asked to wear a mask.	Favor✗	Against
MTIN	FM	It should be a hate crime if someone makes a threat against women	Against✗	Favor
	WFM	Avoiding places where I would be asked to wear a mask.	Against✓	Against
MG-SIN	FM	It should be a hate crime if someone makes a threat against women	Favor✓	Favor
	WFM	Avoiding places where I would be asked to wear a mask.	Against✓	Against

TABLE X

CASE STUDY. VISUALIZATION OF ATTENTION SCORES FROM TEXTGCN, MTIN, AND MG-SIN METHODS ON TESTING EXAMPLES OF COVID-19 DATASET, ALONG WITH THEIR PREDICTIONS AND CORRESPONDING GROUND TRUTH LABELS

Noisy Type	Model	Target	Attention Visualization	Prediction	Label
Abbreviations of Terms	TextGCN	WFM	He won't be happy (un)til everyone has a bag over our heads	Against✗	Favor
	MTIN		He won't be happy (un)til everyone has a bag over our heads	Against✗	
	MG-SIN		He won't be happy (un)til everyone has a bag over our heads	Favor✓	
Spelling Errors	TextGCN	KSC	All parents aren't teained(trained) to be teachers	Favor✗	Against
	MTIN		All parents aren't teained(trained) to be teachers	Favor✗	
	MG-SIN		All parents aren't teained(trained) to be teachers	Against✓	
Grammar Errors	TextGCN	WFM	(I am) Not wearing a mask because I am in my own greenhouse	Against✓	Against
	MTIN		(I am) Not wearing a mask because I am in my own greenhouse	Against✓	
	MG-SIN		(I am) Not wearing a mask because I am in my own greenhouse	Against✓	

further demonstrates that our MG-SIN can capitalize on the pragmatics information of words to enhance the model's performance.

Moreover, to illustrate that our MG-SIN can effectively filter noisy information, this article selects three sentences with some noisy information (including grammar errors, abbreviations of terms, and spelling errors) and uses three models to identify the stance toward a given target. As shown in Table X, this article performs a case study and visualizes the attention scores learned by TextGCN, MTIN, and our MG-SIN models, together with their predictions of stance detection task and corresponding ground truth labels. Note that, the words inside the parentheses are correct forms of errors.

The first sentence in Table X "He won't be happy til everyone has a bag over our heads" has noisy information, abbreviations of terms "won't," which can bring extra difficulty in detecting implicit semantics. TextGCN and MTIN methods are misled by these errors and fail to identify the connection between the words "won't" and given target, leading to wrong predictions. Our MG-SIN can accurately capture implicit contextual semantics and identify the right stance. The second sentence "All parents aren't teained to be teachers" has a spelling error "teained (trained)" and an abbreviation "aren't," which are important for identifying the user's stance. However, TextGCN and MTIN models are misled by these errors and cannot recognize contextual information of these misspelled words, resulting in false prediction. Our MG-SIN can accurately model the relationship between these errors and word "teachers," ignoring the interference from noisy information. Similarly, the third sentence "Not wearing a mask because I am in my own greenhouse" has a grammar error, the lack of a subject predicate "I am." All methods can filter this type of noise information by mining contextual information from sentences, ultimately achieving accurate predictions.

VI. CONCLUSION

This article proposes a novel MG-SIN to improve the performance of stance detection and sentiment analysis tasks simultaneously on social media. This article first constructs heterogeneous pragmatics dependency graphs to capture both contextual semantic information and pragmatics dependency information of words, which can be utilized to capture the more fine-grained task representations. To address the data noise problem of the information sharing network, this article proposes a graph-aware module with a sparse interaction mechanism to facilitate information sharing between tasks, which allows the tasks to selectively fuse the useful information shared by other tasks while filtering the noisy information from the information sharing network. The experimental results show the effectiveness of our proposed model.

Our proposed MG-SIN method constructs three heterogeneous graphs for each given sentence and uses the GCN module to capture contextual information from words. Due to the use of GCN, the proposed MG-SIN method is unsuitable for long text, such as document text or long news. For a given long text, the constructed heterogeneous graphs are very large and sparse, making our model a bit computationally expensive, which hinders the application of our MG-SIN in large-scale, long-text scenarios. Therefore, our MG-SIN method's weakness is the expensive computation cost in the long text.

In future work, we will focus on zero-shot stance detection on social media, which can identify the stance of an unseen target in the training data. In this article, we demonstrate the effectiveness of multigraph sparse interaction mechanism in capturing target-dependent stance expression as well as filtering the target-independent expression. In my personal view, the proposed sparse interaction mechanism can also boost the performance of zero-shot stance detection by capturing target-dependent and -independent stance representation. Moreover, zero-shot stance detection is a challenging and important problem in practical application, which can simulate the way humans reason, to identify new targets that have never been seen before.

REFERENCES

[1] C. Li et al., "Joint stance and rumor detection in hierarchical heterogeneous graph," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 6, pp. 2530–2542, Jun. 2022.

[2] Q. Liao et al., "An integrated multi-task model for fake news detection," *IEEE Trans. Knowl. Data Eng.*, vol. 34, no. 11, pp. 5154–5165, Nov. 2022.

[3] F. Huang, X. Li, C. Yuan, S. Zhang, J. Zhang, and S. Qiao, "Attention-emotion-enhanced convolutional LSTM for sentiment analysis," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 9, pp. 4332–4345, Sep. 2022.

[4] H. Chai, J. Cui, Y. Wang, M. Zhang, B. Fang, and Q. Liao, "Improving gradient trade-offs between tasks in multi-task text classification," in *Proc. 61st Annu. Meeting Assoc. Comput. Linguistics*. Stroudsburg, PA, USA: Association for Computational Linguistics, 2023, pp. 2565–2579.

[5] L. Grimminger and R. Klinger, "Hate towards the political opponent: A Twitter corpus study of the 2020 U.S. elections on the basis of offensive speech and stance detection," in *Proc. 11th Workshop Comput. Approaches Subjectivity, Sentiment Social Media Anal. (WASSA@EACL)*. Stroudsburg, PA, USA: Association for Computational Linguistics, Apr. 2021, pp. 171–180.

[6] K. Glandt, S. Khanal, Y. Li, D. Caragea, and C. Caragea, "Stance detection in COVID-19 tweets," in *Proc. 59th Annu. Meeting Assoc. Comput. Linguistics, 11th Int. Joint Conf. Natural Lang. Process. (ACL/IJCNLP)*. Stroudsburg, PA, USA: Association for Computational Linguistics, Aug. 2021, pp. 1596–1611.

[7] G. Zarrella and A. Marsh, "Mitre at SemEval-2016 task 6: Transfer learning for stance detection," in *Proc. 10th Int. Workshop Semantic Eval. (SemEval-)*. San Diego, CA, USA: The Association for Computer Linguistics, 2016, pp. 458–463.

[8] Q. Sun, Z. Wang, Q. Zhu, and G. Zhou, "Stance detection with hierarchical attention network," in *Proc. 27th Int. Conf. Comput. Linguistics (COLING)*, Santa Fe, NM, USA: Association for Computational Linguistics, Aug. 2018, pp. 2399–2409.

[9] X. Ma, Y. Liu, and C. Ouyang, "Capturing semantic features to improve Chinese event detection," *CAAI Trans. Intell. Technol.*, vol. 7, no. 2, pp. 219–227, Jun. 2022.

[10] U. A. Siddiqua, A. N. Chy, and M. Aono, "Tweet stance detection using an attention based neural ensemble model," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol. (NAACL-HLT)*, Minneapolis, MN, USA, vol. 1. Stroudsburg, PA, USA: Association for Computational Linguistics, Jun. 2019, pp. 1868–1873.

[11] T. Yang, L. Hu, C. Shi, H. Ji, X. Li, and L. Nie, "HGAT: Heterogeneous graph attention networks for semi-supervised short text classification," *ACM Trans. Inf. Syst.*, vol. 39, no. 3, pp. 1–29, Jul. 2021.

[12] L. Yao, C. Mao, and Y. Luo, "Graph convolutional networks for text classification," in *Proc. 33rd AAAI Conf. Artif. Intell., 31st Innov. Appl. Artif. Intell. Conf., 9th AAAI Symp. Educ. Adv. Artif. Intell. (EAAI)*. AAAI Press, 2019, pp. 7370–7377.

[13] H. Chai, S. Tang, J. Cui, Y. Ding, B. Fang, and Q. Liao, "Improving multi-task stance detection with multi-task interaction network," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*. Stroudsburg, PA, USA: Association for Computational Linguistics, 2022, pp. 2990–3000.

[14] Q. Sun, Z. Wang, S. Li, Q. Zhu, and G. Zhou, "Stance detection via sentiment information and neural network model," *Frontiers Comput. Sci.*, vol. 13, no. 1, pp. 127–138, Feb. 2019.

[15] Z. Li, F. Nie, X. Chang, Y. Yang, C. Zhang, and N. Sebe, "Dynamic affinity graph construction for spectral clustering using multiple features," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 12, pp. 6323–6332, Dec. 2018.

[16] R. Zhou, X. Chang, L. Shi, Y.-D. Shen, Y. Yang, and F. Nie, "Person reidentification via multi-feature fusion with adaptive graph learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 5, pp. 1592–1601, May 2020.

[17] D. Chen and C. Manning, "A fast and accurate dependency parser using neural networks," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2014, pp. 740–750.

[18] E. Jang, S. Gu, and B. Poole, "Categorical reparameterization with gumbel-softmax," in *Proc. 5th Int. Conf. Learn. Represent. (ICLR)*, 2017, pp. 1–12.

[19] 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)*. Stroudsburg, PA, USA: Association for Computational Linguistics, 2019, pp. 4567–4577.

[20] S. M. Mohammad, S. Kiritchenko, P. Sobhani, X. Zhu, and C. Cherry, "SemEval-2016 task 6: Detecting stance in tweets," in *Proc. 10th Int. Workshop Semantic Eval. (SemEval@NAACL-HLT)*, San Diego, CA, USA: Association for Computer Linguistics, Jun. 2016, pp. 31–41.

[21] A. Upadhyaya, M. Fisichella, and W. Nejdl, "Towards sentiment and temporal aided stance detection of climate change tweets," *Inf. Process. Manage.*, vol. 60, no. 4, Jul. 2023, Art. no. 103325.

[22] Y. Li and C. Caragea, "Multi-task stance detection with sentiment and stance lexicons," in *Proc. Conf. Empirical Methods Natural Lang. Process., 9th Int. Joint Conf. Natural Lang. Process. (EMNLP-IJCNLP)*, Hong Kong. Stroudsburg, PA, USA: Association for Computational Linguistics, Nov. 2019, pp. 6298–6304.

[23] G. Zhao and P. Yang, "Pretrained embeddings for stance detection with hierarchical capsule network on social media," *ACM Trans. Inf. Syst.*, vol. 39, no. 1, pp. 1:1–1:32, 2020.

[24] M. Müller, M. Salathé, and P. E. Kummervold, "COVID-Twitter-BERT: A natural language processing model to analyse COVID-19 content on Twitter," 2020, *arXiv:2005.07503*.

[25] Y. Yang, B. Wu, K. Zhao, and W. Guo, "Tweet stance detection: A two-stage DC-BILSTM model based on semantic attention," in *Proc. 5th IEEE Int. Conf. Data Sci. Cyberspace (DSC)*, Jul. 2020, pp. 22–29.

[26] D. Yang et al., "Stance detection with stance-wise convolution network," in *Proc. 9th CCF Int. Conf. Natural Lang. Process. Chin. Comput. (NLPCC)* (Lecture Notes in Computer Science), vol. 12430. Cham, Switzerland: Springer, 2020, pp. 555–567.

[27] K. Kawintiranon and L. Singh, "Knowledge enhanced masked language model for stance detection," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol., NAACL-HLT*. Stroudsburg, PA, USA: Association for Computational Linguistics, 2021, pp. 4725–4735.

[28] Z. Wang, Q. Sun, S. Li, Q. Zhu, and G. Zhou, "Neural stance detection with hierarchical linguistic representations," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 28, pp. 635–645, 2020.

[29] M. Matero, N. Soni, N. Balasubramanian, and H. A. Schwartz, "MeLT: Message-level transformer with masked document representations as pre-training for stance detection," in *Proc. Findings Assoc. Comput. Linguistics, EMNLP*. Stroudsburg, PA, USA: Association for Computational Linguistics, 2021, pp. 2959–2966.

[30] Y. Li and C. Caragea, "Target-aware data augmentation for stance detection," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol. (NAACL-HLT)*. Stroudsburg, PA, USA: Association for Computational Linguistics, 2021, pp. 1850–1860.

[31] Y. Feng and Y. Cheng, "Short text sentiment analysis based on multi-channel CNN with multi-head attention mechanism," *IEEE Access*, vol. 9, pp. 19854–19863, 2021.

[32] M. E. Basiri, S. Nemati, M. Abdar, E. Cambria, and U. R. Acharya, "ABCDM: An attention-based bidirectional CNN-RNN deep model for sentiment analysis," *Future Gener. Comput. Syst.*, vol. 115, pp. 279–294, Feb. 2021.

[33] R. K. Behera, M. Jena, S. K. Rath, and S. Misra, "Co-LSTM: Convolutional LSTM model for sentiment analysis in social big data," *Inf. Process. Manage.*, vol. 58, no. 1, Jan. 2021, Art. no. 102435.

[34] Y. Zhang, X. Yu, Z. Cui, S. Wu, Z. Wen, and L. Wang, "Every document owns its structure: Inductive text classification via graph neural networks," in *Proc. 58th Annu. Meeting Assoc. Comput. Linguistics*. Stroudsburg, PA, USA: Association for Computational Linguistics, 2020, pp. 334–339.

[35] X. Liu, X. You, X. Zhang, J. Wu, and P. Lv, "Tensor graph convolutional networks for text classification," in *Proc. 34th AAAI Conf. Artif. Intell., 32nd Innov. Appl. Artif. Intell. Conf. (IAAI), 10th AAAI Symp. Educ. Adv. Artif. Intell. (EAAI)*, 2020, pp. 8409–8416.

[36] Y. Mao, S. Yun, W. Liu, and B. Du, "Tchebycheff procedure for multi-task text classification," in *Proc. 58th Annu. Meeting Assoc. Comput. Linguistics*. Stroudsburg, PA, USA: Association for Computational Linguistics, 2020, pp. 4217–4226.

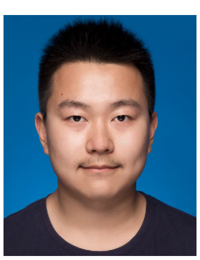
[37] Y. Mao, Z. Wang, W. Liu, X. Lin, and W. Hu, "BanditMTL: Bandit-based multi-task learning for text classification," in *Proc. 59th Annu. Meeting Assoc. Comput. Linguistics, 11th Int. Joint Conf. Natural Lang. Process. (ACL/IJCNLP)*. Stroudsburg, PA, USA: Association for Computational Linguistics, 2021, pp. 5506–5516.

[38] A. Upadhyaya, M. Fisichella, and W. Nejdl, "A multi-task model for sentiment aided stance detection of climate change tweets," in *Proc. 16th Int. AAAI Conf. Web Social Media (ICWSM)*. AAAI Press, 2023.

[39] J. Barnes, E. Velldal, and L. Øvrelid, "Improving sentiment analysis with multi-task learning of negation," *Natural Lang. Eng.*, vol. 27, no. 2, pp. 249–269, Mar. 2021.

Heyan Chai received the M.Eng. degree in computer science and technology from the Department of Computer Science and Technology, Harbin Institute of Technology, Shenzhen, China, in 2020, where he is currently pursuing the Ph.D. degree.

His current research interests include data mining, multitask learning, and social media analysis.



Jinhao Cui received the B.E. degree in information management and information system (economic management) from the Harbin Institute of Technology, Harbin, China, in 2021, where he is currently pursuing the M.S. degree with the Department of Computer Science and Technology.

His research interests include data mining and graph contrastive learning.

Siyu Tang received the B.E. degree in software engineering from Zhengzhou University, Zhengzhou, China, in 2021. She is currently pursuing the M.S. degree with the Department of Computer Science and Technology, Harbin Institute of Technology, Shenzhen, China.

Her research interests include aspect-based sentiment analysis and social media data mining.

Ye Ding (Member, IEEE) received the Ph.D. degree in computer science and engineering from the Department of Computer Science and Engineering, The Hong Kong University of Science and Technology, Hong Kong, in 2014, under the supervision of Prof. Lionel M. Ni.

He is currently an Associate Professor with the Dongguan University of Technology, Dongguan, China. His research interests include spatial-temporal data analytics and machine learning.

Xinwang Liu (Senior Member, IEEE) received the Ph.D. degree from the National University of Defense Technology (NUDT), Changsha, China, in 2013.

He is currently a Professor with the School of Computer, NUDT. He has published more than 90 peer-reviewed papers, including those in highly regarded journals and conferences, such as IEEE Transactions on Pattern Analysis and Machine Intelligence, IEEE Transactions on Knowledge and Data Engineering, IEEE Transactions on Image Processing, IEEE Transactions on Neural Networks and Learning Systems, IEEE Transactions on Multimedia, IEEE Transactions on Information Forensics and Security, International Conference on Machine Learning (ICML), Conference on Neural Information Processing Systems (NeurIPS), Conference on Computer Vision and Pattern Recognition (CVPR), International Conference on Computer Vision (ICCV), Association for the Advancement of Artificial Intelligence (AAAI), and International Joint Conferences on Artificial Intelligence (IJCAI). His current research interests include kernel learning, multiview clustering, and unsupervised feature learning.

Dr. Liu is also an Associate Editor of IEEE Transactions on Neural Networks and Learning Systems and *Information Fusion* journal. More information can be found at https://xinwangliu.github.io/.

Binxing Fang (Member, IEEE) received the M.S. degree in computer science and technology from Tsinghua University, Beijing, China, in 1984, and the Ph.D. degree in computer science and technology from the Harbin Institute of Technology, Harbin, China, in 1989.

He is currently a member of the Chinese Academy of Engineering, Beijing. His current research interests include computer networks, information and network security, and artificial intelligence security.

Qing Liao (Member, IEEE) received the Ph.D. degree from The Hong Kong University of Science and Technology, Hong Kong, in 2016.

She is currently a Professor with the School of Computer Science and Technology, Harbin Institute of Technology, Shenzhen, China. Her research interests include data mining, artificial intelligence, and information security.