Graph-Based Cross-Granularity Message Passing on Knowledge-Intensive Text

Chenwei Yan, Xiangling Fu, Xinxin You, Ji Wu, *Senior Member, IEEE*, and Xien Liu, *Member, IEEE*

***Abstract*—In knowledge-intensive fields such as medicine, the text often contains numerous professional terms, specific text fragments, and multidimensional information. However, most existing text representation methods ignore this specialized knowledge and instead adopt methods similar to those used in the general domain. In this paper, we focus on developing a learning module to enhance the representation ability of knowledge-intensive text by leveraging a graph-based cross-granularity message passing mechanism. To this end, we propose a novel learning framework, the Multi-Granularity Graph Neural Network (MG-GNN), to integrate fine-grained and coarse-grained knowledge at the character, word, and phase levels. The MG-GNN performs learning in two stages: 1) inter-granularity learning and 2) intra-granularity learning. During inter-granularity learning, semantic knowledge is extracted from character, word, and phrase granularity graphs, whereas intra-granularity learning focuses on fusing knowledge across different granularity graphs to achieve comprehensive message integration. To enhance the fusion performance, we propose a context-based gating mechanism to guide cross-graph propagation learning. Furthermore, we apply MG-GNN to address two important medical applications. Experimental results demonstrate that our proposed MG-GNN model significantly enhances the performance in both diagnosis prediction and medical named entity recognition tasks.**

***Index Terms*—Multi-granularity, graph neural network, electronic medical record, medical NER, diagnosis prediction.**

I. Introduction

MEDICAL texts, as knowledge-intensive texts, contain rich medical semantic knowledge and valuable clinical experience. This information can be represented at different levels of granularity: fine-grained knowledge, such as single characters, and coarse-grained knowledge, such as phrases and sentences [1]. As depicted in Fig. 1, medical texts contain both word-level and phrase-level knowledge, illustrating their knowledge-intensive nature. Compared with word-level knowledge, longer terminologies in the medical domain, such as disease names, symptoms, and treatment, often have a coarser granularity. Effective representation learning across these various granularities is vital for understanding the overall semantics of medical texts. This process involves encoding the information into semantic text representations, optimizing and enriching these representations with domain-specific knowledge, and applying them to diverse downstream tasks and clinical practices [2], including medical named entity recognition, diagnosis prediction, and International Classification of Diseases (ICD) coding.

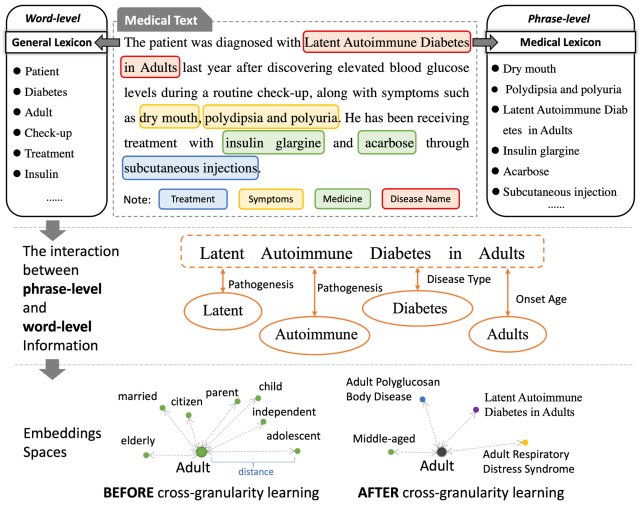

Fig. 1. An example of knowledge across different granularities in a medical text segment. Best viewed in color.

From the perspective of different granularities, fine-grained information provides more detailed knowledge from various aspects for a complete understanding of the semantics. For example, as shown in Fig. 1, integrating the word embedding of "Adults" into the embedding of "Latent Autoimmune Diabetes in Adults" can emphasize the age of onset of the disease. After the interaction of cross-granularity knowledge, the embeddings of "Adults" can be influenced by the given disease. In the semantic space, the word "adult" becomes closer to adult-related diseases, whereas it becomes more distant from words associated with other age groups, such as "adolescent" and "elderly".

Received 30 January 2023; revised 21 May 2024, 24 July 2024, and 3 September 2024; accepted 29 September 2024. Date of publication 2 October 2024; date of current version 11 October 2024. This work was supported in part by the National Key R&D Program of China under Grant 2023ZD0506501 and Grant 2021YFC2500803, in part by the Ningxia Key R&D Program of China under Grant 2023BEG02064, in part by the National Natural Science Foundation of China under Grant 82071171, and in part by by BUPT Excellent Ph.D. Students Foundation under Grant CX2021122. The associate editor coordinating the review of this article and approving it for publication was Prof. Jiajun Zhang. *(Corresponding authors: Xiangling Fu; Xien Liu.)*

Chenwei Yan and Xiangling Fu are with the Beijing University of Posts and Telecommunications and Key Laboratory of Trustworthy Distributed Computing and Service (BUPT), Ministry of Education, Beijing 100876, China (e-mail: chenwei.yan@bupt.edu.cn; fuxiangling@bupt.edu.cn).

Xinxin You and Xien Liu are with the Department of Electronic Engineering, Tsinghua University, Beijing 100190, China (e-mail: yxx23@mails.tsinghua.edu.cn; xeliu@mail.tsinghua.edu.cn).

Ji Wu is with the Department of Electronic Engineering, Tsinghua University, Beijing 100190, China, and also with the College of AI, Tsinghua University, Beijing 100190, China (e-mail: wuji_ee@mail.tsinghua.edu.cn).

Digital Object Identifier 10.1109/TASLP.2024.3473308

In contrast, coarse-grained information provides more abstract and complete information [3], as well as boundary information for the terms [4]. For example, "patch" refers to an adhesive piece applied to the skin in the sentence "The patient used a nicotine skin patch to help quit smoking", whereas it refers to an abnormal area in the macula of the eye in the sentence "The eye specialist found a macular patch during the examination". Thus, the meaning of a word is closely related to its context. The incorporation of the coarse-grained information into the learning of fine-grained embeddings can increase their accuracy. However, in practice, coarse-grained knowledge is more easily discarded, especially for Chinese natural language processing (NLP) tasks, where Chinese word segmentation (CWS) errors often result in irregular coarse-grained terminologies or phrases being incorrectly segmented by word segmentation tools. A large amount of useful coarse-grained information between the word level and the document level, such as professional expressions, may be overlooked in graph-based models [5], [6]. As a result, the coarse-grained information that should be incorporated into word and document representations is not fully utilized during feature extraction.

Several studies [4], [7], [8], [9] have shown that different granularities of knowledge in text are both useful and complementary. Zhang et al. [7] and Ma et al. [10] incorporated word-level information into character-based models to recover coarse-grained knowledge. Xia et al. [8] combined word-level and sentence-level information to address overlapping issues in named entity recognition. Yao et al. [5] employed a graph-based model to connect word-level and document-level information to enhance the final text representation. Ma et al. [3] expanded the text sampling granularity from the word level to the sentence level. Zhu et al. [11] extracted phrases to generate local-level and global-level features.

Similarly, in the medical domain, some methods consider different granularities of knowledge. Lee et al. [12] integrated word-level information into characters for medical named entity recognition, however, they ignored phrase-level information. Yao et al. [13] employed trigger phrases as domain knowledge to guide disease prediction, however, these phrases were spliced together without being connected to the corresponding fine-grained information. In summary, medical texts provide multiple granularities of knowledge owing to their knowledge-intensive nature, but most existing approaches fail to fully exploit the different granularities of knowledge, let alone the semantic relationships between these granularities. Therefore, to obtain more accurate medical text representations, leveraging the knowledge-intensive and multi-granularity nature of medical texts and enhancing the interaction between different granularities of knowledge are essential.

Thus, in this paper, we propose a novel multi-granularity learning framework, named the **M**ulti-**G**ranularity **G**raph **N**eural **N**etwork (**MG-GNN**), along with two message passing modes for the interaction of different granularities of knowledge.

(1) Top-Down mode: The semantics of a single Chinese character are rich and varied, as a character can convey different meanings depending on the context of the sentence. Coarser-grained information can provide more complete and abstract semantic information. In other words, the interpretation of a character's meaning is determined by the surrounding coarser-grained information. Therefore, incorporating the entire coarse information into the understanding of individual characters is essential.

(2) Bottom-Up mode: Phrases are made up of words, and the words are made up of characters. Fine-grained information can provide more detailed knowledge from different perspectives. Since the contributions of each character to the word are not equal, the key to forming a coarse-grained knowledge representation is to identify and capture the vital fine-grained information features.

Specifically, in this work, we construct a character-granularity graph, a word-granularity graph, and a phrase-granularity graph for each medical document. The edges of the character and word graphs are built by their co-occurrence information, and the edges of the phrase graph are built by the similarity between phrases. The graph neural network is exploited in these three graphs to obtain the inter-granularity neighborhood information. After the inter-granularity message passing, we utilize a context-based gating mechanism to establish the message passing between graphs, and the gating mechanism contains top-down or bottom-up modes. In summary, our contributions are as follows:

- We propose a graph neural network-based learning framework to integrate multi-granularity knowledge, which enhances the use of knowledge and allows mutual complementarity between different granularities of knowledge.
- We design two message passing modes in intra-granularity aggregation to enable the interaction between fine-grained knowledge and coarse-grained knowledge. The top-down mode compresses prior knowledge into a representation of fine-grained granularity, whereas the bottom-up mode adds fine-grained semantics to sentence representations.
- We conduct extensive experiments on two medical tasks, and the results demonstrate that our framework can capture better representations of medical texts, and it outperforms the state-of-the-art baselines.

II. Related Work

This section reviews recent studies on text modeling via graph neural networks (GNNs) and multi-granularity text information fusion.

A. Text Modeling Via Graph Neural Networks

In natural language processing (NLP) tasks, transferring natural language symbol information into digital information, such as vectors or matrices, to facilitate computer understanding and processing is key. A good text representation needs to be able to fully express the semantic connotation of the text. Early text modeling methods have evolved from methods based on bag-of-words (BOW) to methods based on word embeddings represented by word2vec [14] and Glove [15]. These methods convert words into continuous and dense distributed representations in low-dimensional space. The emergence of word embeddings

allowed natural languages to establish perfect connections with deep neural network models. Convolutional neural networks (CNNs) [16] and recurrent neural networks (RNNs) [17], [18] are widely used in text modeling for NLP applications. These models capture the semantic information from the text sequence.

Recently, graph neural networks have become popular for text modeling. GNNs [19] provide much richer neighbors with multiple relations rather than only focusing on sequentiality, and enable the long-term and nonconsecutive semantics to be captured [20]. In terms of constructing text into graphs, Vashishth et al. [21] utilize dependency parsing to construct the word graphs. Yao et al. [5] built a graph on the entire corpus to generate the word and document representations. The graph is built on global word co-occurrence, which is calculated as pointwise mutual information (PMI) values and so the word representations are based on the global context. It is often built for the entire corpus level, and does not change with the sentence-level context. Generally, the semantics of words may differ according to their context, so the word embedding features need to be more dynamic and flexible. We use the graph construction method of [6], which builds individual graphs for each document. In this way, the word representations can be dynamically adjusted on the local context.

B. Multi-Granularity Information Fusing

In general, characters, words, entities, sentences, and even paragraphs represent different granularities of knowledge, each containing different levels of semantics. Fine-grained information-based models, such as characters and words, are very common. Pre-trained language models trained on large-scale corpora, such as BERT [22] and XLNet [23], and medical domain models, such as MC-BERT [24], are typical examples of word-based models for fine-grained information learning. Considering the characteristics of Chinese, some works [25], [26] have introduced multi-granularity pre-trained language models that combine characters and words to utilize the word information for character-based models. The introduction of word-level information allows the model to learn from coarser-grained knowledge, reducing the difficulty of model learning. In addition, for Chinese named entity recognition (NER) tasks, although character-based models, which are the finest-grained representations, have shown superior performance compared with word-based models [27], the integration of word information into characters can further benefit the final prediction [4], [7]. These multi-granularity information benefits are also observed in information retrieval tasks [3], [9].

These works have shown that different granularities of knowledge in text are both useful and complementary. Moreover, the coarser information contains more complete and abstract semantic information, and can be regarded as the introduction of prior knowledge [21], [28].

III. Method

Our multi-granularity learning model consists of a graph construction module, an inter-granularity aggregation module, an intra-granularity aggregation module, and an output module. The graph construction module parses the input text and converts it into three graphs at the character, word, and phrase granularity. The inter-granularity aggregation module and intra-granularity aggregation module present two learning stages, as illustrated in Fig. 2. These stages learn inter-granularity features and fuse granular features via graph neural networks. Specifically, two cross-graph message passing mechanisms are proposed in the intra-granularity aggregation module. Finally, the output module is designed to convert the output embeddings to the final prediction.

A. Graph Construction

For a given text segment $S = \{c_1, c_2, \ldots, c_n\}$, we perform text parsing at various levels of granularity. First, we consider the character level, which is the smallest semantic unit in Chinese. Each distinct character within the segment is treated as a node, and the co-occurrence relations between characters within a fixed-length sliding window form the edges. This character graph is denoted as $G_{char} = (V_{char}, E_{char})$, where V_{char} is the set of nodes with $|V_{char}| <= n$, and where E_{char} represents the set of edges. Next, we proceed to word-level parsing. The words existing in the lexicon are selected as word-level nodes. To mitigate the negative impact of different word segmentation methods, we include all potential words present in the given text segment S. The words are then reordered on the basis of the index of the first character. The resulting word graph is denoted as $G_{word} = (V_{word}, E_{word})$, where V_{word} represents the set of nodes, and where E_{word} is the set of edges. The edges in word-graph are also from the co-occurrence of words, which can be identified within local sentences or calculated as PMI values [5] via the global corpus. In our model, we incorporate both of these methods and present a comparison in the experimental section. Finally, similar to the selection of potential words in word-level parsing, phrases that match the domain-specific phrase dictionary are collected as nodes. Owing to the sparsity of phrases, we connect them on the basis of their semantic similarity. Specifically, the edge between two phrases is set to the value of their cosine similarity, and if this value falls below a threshold, the edge is set to zero. This phrase graph is denoted as $G_{phrase} = (V_{phrase}, E_{phrase})$, where V_{phrase} represents the set of nodes, and where E_{phrase} represents the set of edges. All three graphs are undirected graphs.

Another necessary graph is the multi-granularity graph. This graph incorporates all the nodes from the character, word, and phrase graphs. In this comprehensive graph, we consider the containment relationships between these different granularities. Specifically, an edge between a character-granularity node, v_{char_i}, and a word-granularity node, v_{word_j}, is assigned a value of 1 if the character $char_i$ appears in the word $word_j$, where $char_i$ is the i-th character and $word_j$ is the j-th word. A similar approach is applied to the relationships between words and phrases. Additionally, for character-granularity nodes, we introduce edges between adjacent characters, following their forward and backward sequence order in the text. This multi-granularity graph is denoted as $G_{cwp} = (V_{cwp}, E_{cwp})$, where

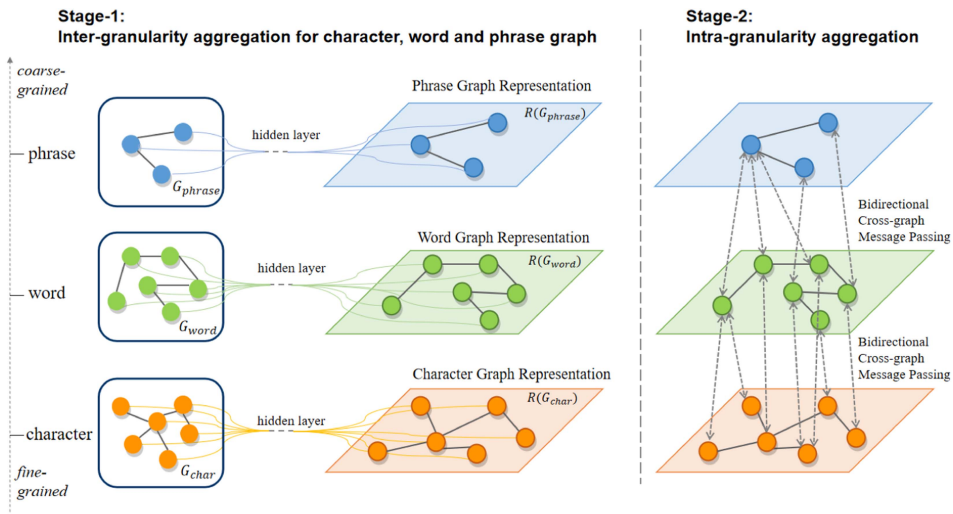

Fig. 2. Overview of the learning framework of the multi-granularity graph neural network (MG-GNN). (a) shows the inter-graph aggregation of three single-granularity graphs, and (b) shows the two message passing modes of the intra-granularity graphs.

$V_{cwp} = \{V_{char}, V_{word}, V_{phrase}\}$ is the set of nodes, and E_{cwp} is the set of edges.

B. Inter-Granularity Aggregation

Given the individual single-granularity graphs, the initial node embeddings and the adjacent matrix are denoted as $X_{gr} \in \mathbb{R}^{|V_{gr}| \times d^w}$ and $A_{gr} \in \mathbb{R}^{|V_{gr}| \times |V_{gr}|}$, respectively, where gr represents the different granularities $[c, w, p]$ (character, word, phrase), d^w is the dimension of word embeddings, and $|V_{gr}|$ denotes the size of the node set.

To learn the embeddings of nodes in each graph, we employ the gated graph neural networks (GGNN) [19]. Initially, the information from direct neighbor nodes is collected as $a_{gr}^{(1)}$. By stacking the t-layer GGNN, we can obtain the information from the t-step neighbors.

$$H_{gr}^{(0)} = X_{gr}, \tag{1}$$

$$a_{gr}^{(t)} = A_{gr} H^{(t-1)} W_t, \tag{2}$$

where X_{gr} is the initial embedding for a specific granularity "gr" (character-level, word-level, or phrase-level), and A_{gr} represents the adjacent matrix.

Then, we aggregate the neighborhood information $a_{gr}^{(t)}$ with the current node's self-information $H^{(t-1)}$. The node representation is updated according to the formulas (3)–(6), which illustrate how the GGNN uses the initial embedding and neighborhood information to derive the final representation:

$$z_{gr}^{(t)} = \sigma\left(W_{gr}^z a_{gr}^t + U_{gr}^z H_{gr}^{(t-1)} + b_{gr}^z\right), \tag{3}$$

$$r_{gr}^{(t)} = \sigma\left(W_{gr}^r a_{gr}^t + U_{gr}^r H_{gr}^{(t-1)} + b_{gr}^r\right), \tag{4}$$

$$\tilde{H}_{gr}^{(t)} = leaky\_relu\left(W_{gr}^a a_{gr}^t + U_{gr}^a \left(r_{gr}^{(t)} \odot H_{gr}^{(t-1)}\right) + b_{gr}^h\right), \tag{5}$$

$$H_{gr}^{(t)} = \tilde{H}_{gr}^{(t)} \odot z_{gr}^{(t)} + H_{gr}^{(t-1)} \odot \left(1 - z_{gr}^{(t)}\right), \tag{6}$$

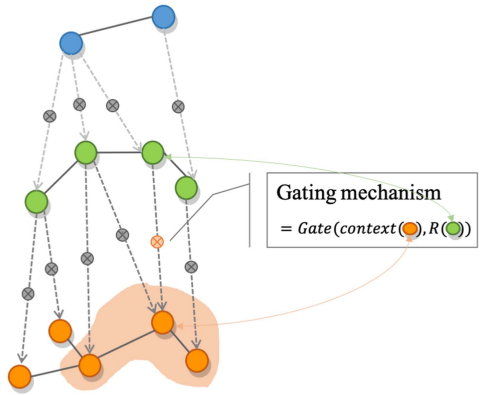

Fig. 3. Context-based gating mechanism in top-down mode. The blue nodes represent phrase-level nodes, the green nodes represent word-level nodes, and the orange nodes represent character-level nodes. Best viewed in color.

where $H_{gr}^{(t)}$ is the updated representation after t-layer. We stack two layers here, that is $t = 2$. $z_{gr}^{(t)}$ and $r_{gr}^{(t)}$ are the update and reset gates in GGNN. $\odot$ is element-wise multiplication.

C. Intra-Granularity Aggregation

Initial Embeddings: After aggregating the neighborhood information within the inter-graphs, we obtain the t-layer outputs of the character graph $H_c^{(t)}$, the word graph $H_w^{(t)}$, and the phrase graph $H_p^{(t)}$. These outputs are then concatenated to form the input for the intra-graph aggregation process.

$$H_{cwp}^{(0)} = \left[H_c^{(t)}; H_w^{(t)}; H_p^{(t)}\right], \tag{7}$$

Context-based Gating Mechanism: To better control the message flow between graphs, we use a context-based gating mechanism to update the adjacent matrix $A_{cwp} \in \mathbb{R}^{|V_{cwp}| \times |V_{cwp}|}$.

To effectively enhance the coarse-grained knowledge of the fine-grained nodes, a context-based gating mechanism, depicted in Fig. 3, is introduced. This mechanism aims to convey the

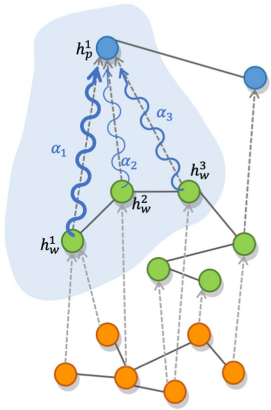

Fig. 4. Graph attention mechanism in bottom-up mode. The meanings of the node colors are the same as those in Fig. 3. The thicker the curve connecting a word-level node to a phrase-level node is, the higher the weight assigned to that edge. Best viewed in color.

meaning of entire phrases to individual words and characters while minimizing the deviations caused by segmentation errors or incomplete word meanings. Specifically, for each character, only the information of the coarse-grained nodes that are directly connected to the character is collected. The update formulas for this process are denoted as (8)–(9).

$$a'_{ij} = a_{ij} \odot g_{ij}, \tag{8}$$

$$g_{ij} = Simi(AGG(v_{before\_i}, v_i, v_{after\_i}), R(v_j)), \tag{9}$$

where a_{ij} is the item of i-th row and j-th column in A_{cwp}, and a'_{ij} is the item of i-th row and j-th column in new adjacent matrix A'_{cwp}, v_j is the neighbors of v_i, v_{before_i} is the previous character of v_i, v_{after_i} is the character after v_i, $Simi$ is the cosine similarity, and $R(v_j) = H^{(0)}_{cwp}(v_j)$, $H^{(0)}_{cwp}$ is treated as the embedding lookup table.

Now, we obtain the new adjacent matrix A'_{cwp} and the initial embeddings $H^{(0)}_{cwp}$, and the intra-graphs aggregation is updated in the same way as the inter-graph aggregation. To capture information from a larger neighborhood, we again stack two layers of the graph neural network.

Graph Attention: In addition to the top-down mode, we further explore a bottom-up mode, which leverages graph attention to propagate semantically richer fine-grained information to the upper layers. This attention mechanism serves as a selector, filtering out unimportant messages, as not all characters contribute equally to the meaning of words or final phrases. Fig. 4 illustrates the abstract representation of this bottom-up mode. The weights of edges are calculated using graph self-attention, as proposed in [29]. The specific update formulas are denoted as (10)–(12):

$$e_{ij} = \sigma(Wh_i, Wh_j) \tag{10}$$

where W is the shared weight matrix, h_i and h_j are nodes and for nodes $j \in \mathcal{N}_i$, where $\mathcal{N}_i$ is neighborhood of node i in the graph. σ is the leaky relu activation function.

$$a_{ij} = \text{softmax}(e_{ij}) = \frac{exp(e_{ij})}{\sum_{k \in \mathcal{N}_i} exp(e_{ik})}, \tag{11}$$

Based on the attention weight a_{ij}, the final embedding for node i is denoted as:

$$h_i = \sigma\left(\sum_{j \in \mathcal{N}_i} a_{ij} W h_j\right), \tag{12}$$

Overall, during the intra-granularity aggregation process, each character's neighbors encompass not only the immediately preceding and following characters but also the words and phrases that encompass it. This approach facilitates interaction between fine-grained characters and coarse-grained knowledge, effectively integrating their respective information into a character representation. The output of this module is denoted as $H^{(t)}_{cwp}$.

D. Output Module

To date, we have obtained a multi-granularity representation, which can be used for various downstream tasks. The specific output module is determined by the requirements of each task. Here, we demonstrate its adaptability to two common tasks: sequence labeling and classification.

Text Sequence Labeling Tasks: Since sequence labeling tasks require preserving the original character order, the final character representations need to be retrieved from the entire embedding set and reorganized according to their original order. This process is denoted as:

$$R_s = \left\{R^{(t)}_{c_1}, R^{(t)}_{c_2}, \ldots, R^{(t)}_{c_i}, \ldots, R^{(t)}_{c_n}\right\}, \tag{13}$$

where $R^{(t)}_{c_i} = H^{(t)}_{cwp}(c_i)$. The matrix $H^{(t)}_{cwp}$ is treated as a embedding lookup table and $R^{(t)}_{c_i}$ is the character embedding of the i-th character in the text.

Bidirectional long short-term Memory (BiLSTM) is applied to capture the sequence information.

$$\begin{bmatrix} f_\tau \\ i_\tau \\ o_\tau \\ \tilde{c}_\tau \end{bmatrix} = \begin{bmatrix} \sigma \\ \sigma \\ \sigma \\ \tanh \end{bmatrix} \left(W \begin{bmatrix} R^{(t)}_{c_\tau} \\ h_{\tau-1} \end{bmatrix} + b \right) \tag{14}$$

$$c_\tau = f_\tau \odot c_{\tau-1} + i_\tau \odot \tilde{c}_\tau \tag{15}$$

$$h_\tau = o_\tau \odot tanh(c_\tau) \tag{16}$$

After that, a standard conditional random field (CRF) layer is exploited on $\{h_1, h_2, \ldots, h_\tau, \ldots, h_n\}$ for the sequence labeling tasks.

Text Classification Tasks: For text classification, we utilize both the max pooling layer and the average pooling layer to extract the final features. Specifically, we independently apply max pooling and average pooling to the phrase-level representations, and subsequently concatenate the results into a single one-dimensional vector. This vector is then projected onto the label space via a multilayer perceptron (MLP). Finally, a Softmax classifier is employed to perform the classification, yielding the predicted category for the given text.

$$out = \left[Pool_{mean}\left(H^{(t)}_p\right), Pool_{max}\left(H^{(t)}_p\right)\right] \tag{17}$$

$$Output = MLP(out), \tag{18}$$

where $H_p^{(t)}$ is the entity-section features of $H_{cwp}^{(t)}$.

E. Decoding and Training

For the sequence labeling tasks, we have the probability of the label sequence $y = \{l_1, l_2, \ldots, l_\tau, \ldots, l_n\}$, which is calculated as formula (19):

$$P(y|S) = \frac{exp\left(\Sigma_i \left(W_{(l_{i-1},l_i)} h_i + b_{(l_{i-1},l_i)}\right)\right)}{\Sigma_{y'} exp\left(\Sigma_i \left(W_{(l'_{i-1},l'_i)} h_i + b_{(l'_{i-1},l'_i)}\right)\right)}, \tag{19}$$

where y' is the arbitrary label sequence, and $W_{(l_{i-1},l_i)}$, $W_{(l'_{i-1},l'_i)}$, $b_{(l_{i-1},l_i)}$, $b_{(l'_{i-1}}$ are model parameters.

The loss function is a negative log-likelihood loss with L2 regularization.

$$\mathcal{L} = -\Sigma_{i=1}^{N} \log(p(y_i|s_i)) + \lambda||\Theta||^2, \tag{20}$$

where N is the number of a set of training data, λ is the L_2 regularization parameter and Θ is the parameter set.

$$\hat{y} = \text{SoftMax}(W_{final} Output + b_{final}), \tag{21}$$

The loss function is cross-entropy loss with L2 regularization.

$$\mathcal{L} = -\Sigma y_i \log(\hat{y_i}) + \lambda||\Theta||^2, \tag{22}$$

Moreover, we add L2 regularization to the cross-entropy to avoid overfitting.

IV. Experiments

A. Task Definition

We selected two kinds of common medical tasks to verify the effectiveness of our framework. **(1) The diagnosis prediction task**, which is commonly regarded as a document classification task, is vital in medical treatment. Traditionally, diagnosis relies on doctors' expertise, and is time-consuming. Diagnosis prediction aims to assist doctors in decision-making by analyzing electronic medical records (EMRs). **(2) Medical named entity recognition (NER)**, a character-level sequence labeling task, aims to identify specific entities such as diagnoses, drugs, and symptoms within medical texts.

These two tasks, one that predicts a diagnosis on the basis of semantic information of coarse-grained information, and the other that focuses on semantics at a fine-grained level, are therefore suitable for validating our proposed model. The NER task is used to verify the fine-grained representations, especially with our top-down mode, whereas the diagnosis prediction task focuses on coarse-grained semantics, aiming to demonstrate the necessity of obtaining accurate coarse representations from our bottom-up mode. Examples from each dataset are provided in the Appendix A.

B. Datasets

To better validate our method, we construct two datasets for the two tasks in Section IV-A. The motivation for using self-constructed datasets lies in their advantages related to real-world data sources, high knowledge density, and wide disease coverage. (1) Data Sources. Our data are derived from real EMRs and manually labeled by physicians, ensuring alignment with the actual application scenarios of the tasks. This enhances the relevance and applicability of our experiments. (2) Knowledge Density. EMR data have high knowledge density and include sufficient medical phrases to meet the multi-granularity requirements of our model. (3) Disease Coverage. Our datasets cover a wide range of diseases, minimizing the bias that might arise from limited disease types in experiments.

TABLE I
Detailed Statistics Information for DiagPrediction Dataset

Statistics	Quantity
# Train Samples	10000
# Test Samples	2439
# Total Labels	153
# Average characters per sample	62.4
# Average words per sample	39.9
# Max characters per sample	99
# Max words per sample	69
Vocabulary Size - character	1982
Vocabulary Size - words	7160

TABLE II
Detailed Statistics Information for EMR-NER Dataset

Entity Type	# Train	# Test
Position	899	181
Disease Nature	2457	59
Dosage	305	67
Medication Frequency	101	24
Start Time	1000	254
Duration	98	22
Disease Diagnosis	709	176
Treatment Effect	346	88
Treatment Drug	800	197
Negative Symptoms	2188	474
Intake Pattern	264	57
Total # of entities	9167	1599

First, we construct a dataset for the diagnosis prediction task, namely, the DiagPrediction dataset. Each sample in the dataset contains the chief complaint and the history of the present illness from the electronic medical records, with a single diagnosis label. In total, we collected 12,439 samples covering 153 general disease diagnoses; more detailed statistics are reported in Table I.

Second, we gather 2,506 Chinese electronic medical records from hospitals and construct a dataset for medical named entity recognition, namely, the EMR-NER dataset. In this dataset, we annotate 11 kinds of common medical entities, totaling 10,766 labeled entities. Table II provides detailed statistics for the EMR-NER dataset.

C. Parameters and Metrics

The optimizer employed is Adam [30] with a learning rate of 0.005. The word embeddings are pre-trained via Glove [15] with dimensions of 200, and the training corpus is from an extensive collection of medical texts. The dropout rate is 0.5. We employ 2 layers of the GGNN, and the dimension of the GGNN output layer is 200. The LSTM hidden dimension used in the NER task is 128. The batch sizes of the diagnosis prediction task and NER task are 512 and 50, respectively. In addition, in the graph

TABLE III
EXPERIMENTAL RESULTS ON DIAGPREDICTION DATASET

Model	Accuracy	Macro		
		Precision	Recall	F1
SWEM [31]	84.72	81.33	73.52	75.22
FastText [32]	85.24	76.69	72.33	72.54
TextCNN [16]	85.77	81.62	74.04	75.47
RNN [33]	79.05	60.46	61.70	59.08
DPCNN [34]	82.62	68.84	61.49	62.91
LEAM [35]	85.53	78.76	74.03	74.73
Attention-LSTM [18]	82.62	68.84	61.60	62.33
Transformer [36]	73.64	57.30	48.03	49.05
TextGCN [5]	83.35	79.24	69.77	71.97
TextING [6]	83.48	75.46	69.19	69.61
ProtoPatient [37]	74.14	72.49	68.61	69.06
Gzip [38]	72.78	57.90	50.84	51.69
MCBERT [24]	85.86	79.55	76.32	76.01
MG-GNN + local co-oc (Proposed)	**86.68**	**82.28**	**77.62**	**79.31**
MG-GNN + global PMI (Proposed)	**87.07**	**84.40**	**78.10**	**79.83**

The last block reports our proposed MG-GNN. 'local co-oc' refers to single granularity graphs are constructed by the co-occurrence in each sample. 'global PMI' refers to the graphs are constructed by the PMI value calculated on the whole corpus.

construction stage, the length of the sliding window is 3, and the cosine similarity threshold is set at 0.3. All the parameters are determined by the development set which is split from training set, and the final model is trained on the whole training set.

We use the micro precision, micro recall, and micro F1 score as metrics for the NER task. For the diagnosis prediction task, the metrics include accuracy, macro precision, macro recall, and the macro F1 score.

D. Baselines

To better evaluate our framework, we select several kinds of methods for comparison in diagnosis prediction task. **(1)** Simple compositional methods: SWEM [31], FastText [32]; **(2)** CNN-based and RNN-based methods: TextCNN [16], TextRNN [33], DPCNN [34]; **(3)** Attention-based methods: LEAM [35], Attention-LSTM [18], Transformer [36]; **(4)** GNN-based methods: TextGCN [5], TextING [6]; **(5)** Other state-of-the-art methods. A non-parametric compressor-based method named Gzip [38], and a prototypical network-based method named ProtoPatient [37]. The max text length is set to 100 tokens for all the models, and the remaining parameters are kept up with the original settings.

The baselines for the NER task include bidirectional LSTM with a CRF layer, where the character-based model and bicharacter-based model [17] are both evaluated. In addition, we compare our results with those of lattice LSTM [7], LGN [4], and SoftLexicon [10] since these methods integrate the coarse-grained information into characters.

E. Main Experimental Results of the Proposed Model

The experimental results on the DiagPrediction dataset are reported in Table III. Overall, our proposed methods outperform all the above methods in terms of accuracy, precision, recall, and F1 score. The MG-GNN with global PMI is superior to that with local co-occurrence. Specifically, the accuracy of the MG-GNN with a global PMI reaches 87.07%.

TABLE IV
THE EXPERIMENTAL RESULTS ON EMR-NER DATASET

Model	Micro		
	Precision	Recall	F1
BiLSTM+CRF (Char-based)	81.42	69.60	75.05
BiLSTM+CRF (Bichar-based)	81.51	70.30	75.49
Lattice LSTM [7]	78.74	79.81	79.27
LGN [4]	77.99	78.87	78.43
SoftLexicon [10]	80.06	78.40	79.22
MCBERT [24]	65.17	77.28	70.71
MG-GNN + local co-oc (Proposed)	**82.31**	**78.31**	**80.26**
MG-GNN + global PMI (Proposed)	**83.05**	**80.20**	**81.60**

In the first block of Table III, we present two simple compositional methods, with FastText [32] achieving an 85.24% accuracy. Among the classical neural network-based models in the second block, TextCNN [16] exhibits the best performance, achieving an 85.77% accuracy. The third block shows the attention-based methods, where the LEAM [35] achieves an 85.53% accuracy. The two models based on graph neural networks achieves a similar performance. Compared with those of the baselines, the accuracy, macro precision, macro recall, and macro F1 score of the MG-GNN with global PMI are improved by 1.3%, 2.78%, 4.06%, and 4.36%, respectively.

Furthermore, we conducted experiments on the EMR-NER dataset, and the results are reported in Table IV. Our proposed model achieves a micro-F1 score of 80.26%, which improves the F1 score by 0.99% compared with the best baseline model.

In the first block of Table IV, the BiLSTM-based models achieved a micro-F1 score of 75.49%. Moving to the second block, three state-of-the-art models integrating word-level information into the character representations were presented. Among them, lattice LSTM and SoftLexicon exhibited similar best performances at 79.27% and 79.22%, respectively. It is obvious that the methods in Blocks 2 and 3 (our proposed

TABLE V
THE RESULTS OF ABLATION STUDY ON DIAGPREDICTION DATASET

Model	Accuracy	Macro		
		Precision	Recall	F1
MG-GNN + local co-oc	**86.68**	**82.28**	**77.62**	**79.31**
MG-GNN + global PMI	**87.07**	**84.40**	**78.10**	**79.83**
w/o word and phrase	84.64	75.92	72.24	71.42
w/o character and phrase	85.94	77.94	76.11	75.18
w/o character and word	86.08	72.05	74.97	71.38
w/o phrase	85.07	74.23	72.85	71.60
w/o word	86.28	78.51	73.08	75.98
w/o character	86.45	81.55	76.87	77.41
w/o context-based gating	86.35	82.63	76.67	77.54
w/o graph attention	86.57	82.12	77.03	78.39
w/o gating and attention	85.49	78.49	75.17	75.34

TABLE VI
THE RESULTS OF ABLATION STUDY ON EMR-NER DATASET

Model	Micro		
	Precision	Recall	F1
MG-GNN + local co-oc	**82.31**	**78.31**	**80.26**
MG-GNN + global PMI	**83.05**	**80.20**	**81.60**
w/o word and phrase	78.89	74.55	76.66
w/o phrase	80.82	74.86	77.73
w/o context-based gating	81.71	78.78	80.22
w/o graph attention	81.27	78.73	79.98
w/o gating and graph attention	80.56	76.26	78.35

method) enhance the recall immensely since they introduce extra word and phrase information.

V. ANALYSIS AND DISCUSSION

A. Ablation Experiments on Multi-Granularity Information

Our model integrates knowledge from multiple granularities. To validate the positive impact of each granularity on the final performance, we conducted ablation experiments, with the results reported in Tables V and VI.

For the diagnosis prediction task, as illustrated in Table V, we first evaluated single granularity models, i.e., characters, words, and phrases, and the results are listed in Rows 2-4. Among these, word-granularity knowledge performs the best in single granularity learning, achieving an accuracy of 85.94%. Moreover, we integrate knowledge from two granularities in Rows 5-7, and the highest accuracy of 86.45% is achieved by combining word and phrase knowledge. These findings align with our initial expectations, namely, that coarse-grained information contributes significantly more to document classification tasks.

For the NER task, character granularity is necessary; thus, we conducted ablation experiments only on the character granularity graph and the character-word granularity graph. As illustrated in Table VI, the character-based single-granularity graph model achieves a micro-F1 score of 76.66%. Compared with the character-based Bi-LSTM (reported 75.05% at Table IV), our single-granularity graph model incorporates inter-graph aggregation, effectively enhancing the character representations. The two-granularity graph model uses the same intra-graph aggregation method to update the final character representations, resulting in a 1.07% improvement in the F1 score compared with the single-granularity graph model. Moreover, the inclusion of phrase information also proves effective, as evidenced by an increase in the F1 score from 77.73% to 80.26%. Thus, each granularity contributes unique semantic information, and combining them allows for a better knowledge representation.

B. Ablation Experiments on Message Passing Mechanisms

To validate the effectiveness of the two message passing mechanisms, we conducted ablation experiments, with the results reported in the bottom blocks of Tables V and VI. Removing either the context-based gating mechanism or the graph attention mechanism results in a significant decline in the performance of the MG-GNN across all the metrics. Compared with the model without both context-based gating and graph attention, our MG-GNN improves the accuracy from 85.49% to 87.07% on the DiagPrediction dataset, improves the micro-F1 score from 78.35% to 81.60% on the EMR-NER dataset. This demonstrates the effectiveness of the message passing mechanisms.

C. Analysis of Similarity Between Phrases

The phrase granularity graph is built via cosine similarity between phrases, with values ranging from 0 to 1. A threshold is set to decide whether there is an edge between two phrases, effectively helping each phrase select its neighbors. We investigate various threshold values: 0.2, 0.3, 0.4 and 0.6, leading to two important observations:

1) The best performance is achieved with a threshold of 0.3. Increasing the threshold does not always improve performance since the graph becomes sparser as the threshold increases. Lower thresholds, such as 0.2, result in most phrase nodes being connected because their cosine similarity is generally no less than 0.2. However, a sparse graph significantly impacts performance negatively.
2) As shown in Fig. 5(a) and (b), the similarity between different symptoms of the same disease is high. For example, in Fig. 5(a), the similarity between "cough" and "productive cough" is 0.81, and in (b), the similarity between "urinary urgency" and "frequent urination" reaches 0.98. In other words, the final phrase-granularity graph often consists of a few closely connected subgraphs representing several related symptoms.

D. Analysis of PMI

The key to effectively applying GNNs to text is constructing an appropriate graph, which involves defining nodes and their connections. We demonstrated two methods for creating a word-level graph: local co-occurrence and global PMI. The results in Tables III and IV indicate that the model using PMI values outperforms the one using local co-occurrence in both the diagnosis prediction task and the medical NER task. For the disease prediction task, the introduction of global PMI increased the model accuracy from 86.68% to 87.07%, and raised the micro-F1 score from 80.26% to 81.60% in the NER task. This improvement can be attributed to the fact that the word relationships in the graph are based on the global context, thus incorporating more comprehensive information.

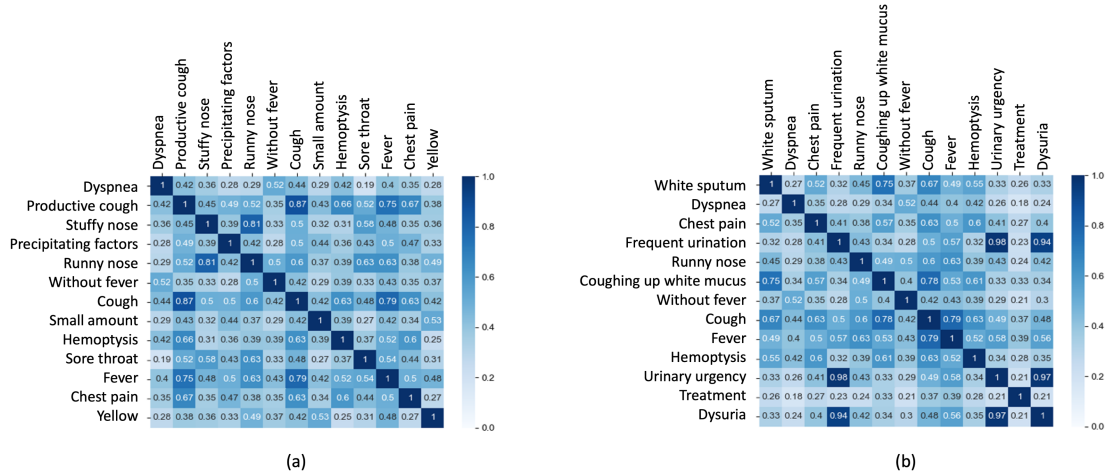

Fig. 5. Two Examples of Phrase Similarity Matrices: The darker the color, the more similar the corresponding phrases. Similarity values range from 0 to 1.

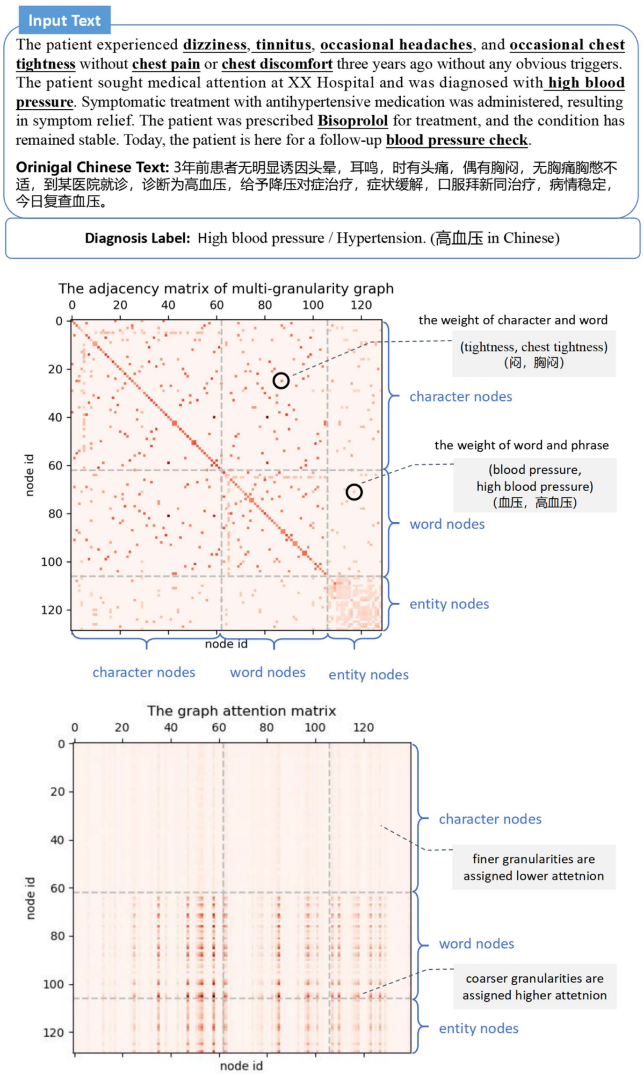

Fig. 6. An example from the diagnosis prediction dataset. Visualization of the adjacent matrix of the multi-granularity graph and the graph attention matrix.

VI. Case Study

To demonstrate the effects of different granularities of knowledge, we provide an example from the DiagPrediction dataset in Fig. 6.

First, we visualized the adjacency matrix used in the intra-graph learning in Fig. 6. The x-axis and y-axis both represent node ids. Starting from the origin, first, character nodes are identified, followed by word nodes, and finally, entity nodes. The multi-granularity graph is dense, with connections established between nodes of different granularities through co-occurrence relationships, inclusion relationships, and semantic similarity relationships. The weights on the edges effectively reflect the relationships between different granularities.

Furthermore, we visualize the graph attention matrix of our model in Fig. 6. Darker colors represent higher weights. Intuitively, word and phrase nodes are assigned greater importance in the disease prediction task, indicating a greater influence on model prediction.

VII. Conclusion

In this paper, we propose a novel learning framework, the multi-granularity graph neural network (MG-GNN), to integrate multi-granularity knowledge. Applied within the medical domain, our framework leverages professional domain knowledge effectively. By constructing character, word, and phrase graphs and employing graph neural networks for inter-graph and intra-graph message passing, we utilize a context-based gating mechanism and graph attention to capture fused semantic knowledge. Specifically, our framework accommodates three granularity levels of knowledge. It initially learns single granularity information independently, and then uses two message passing mechanisms to fuse different granularities, ensuring that the fine-grained and coarse-grained knowledge complement each other.

Furthermore, we evaluate our proposed MG-GNN model across two medical NLP applications, both of which demonstrate

TABLE VII
AN EXAMPLE IN ELECTRONIC MEDICAL RECORD NER DATASET

	English Version	Chinese Version
Text	During the illness, in fever 10 days ago, the highest body temperature up to 39, accompanied by chills, and body temperature returned to normal after self-administered paracetamol. No coughing, and expectoration, no bleeding gums, no mouth ulcers, no abdominal pain, and diarrhea. Hair loss. Raynaud's phenomenon. Normal stool, general sleep and diet,and recent weight gain.	病程中10天前发热，体温最高39度，伴畏寒，自服扑热息痛后体温正常。无咳嗽咳痰，无牙龈出血，无口腔溃疡,无腹痛、腹泻，有脱发、雷诺现象,小便量少，大便正常，睡眠饮食一般，近期体重有所增加。
Annotations	Negative Symptoms: { '5-5': 'fever', '20-20': 'chills', '33-33': 'coughing', '36-36': 'expectoration', '39-39': 'bleeding', '44-44': 'ulcers','53-54': 'hair loss', '56-57': 'Raynaud's phenomenon'} Position: {'40-40': 'gums','43-43': 'mouth'} Treatment Drug: {'30-30': 'paracetamol'} Treatment Effect: {'23-27': 'body temperature returned to normal'}	阴性症状: {'7-8': '发热', '18-19': '畏寒', '34-35': '咳嗽', '36-37': '咳痰', '42-43': '出血', '48-39': '溃疡', '59-60': '脱发', '62-65': '雷诺现象'} 部位: {'40-41': '牙龈', '45-47': '口腔'} 治疗药物: {'23-26': '扑热息痛'} 治疗效果: {'28-31': '体温正常' }

Best viewed in color.

TABLE VIII
AN EXAMPLE IN AUTOMATIC DISEASE DIAGNOSIS DATASET

	English Version	Chinese Version
Chief Complaint	Intermittent chest tightness for one year.	间断胸闷1年。
History of Present Illness	One year ago, the patient developed precordial pain after exertion, which was paroxysmal squeezing pain and relieved spontaneously after 3-5 minutes. It was accompanied by chest tightness, and there was no fall or disturbance of consciousness. Come to see the doctor today.	患者1年前每逢劳累后出现心前区疼痛，呈阵发性压榨样疼痛，持续3-5分钟可自行缓解，伴胸闷，无摔倒及意识障碍。今日前来就诊。
Diagnosis	Coronary Heart Disease	冠心病

superior performances compared with the baselines. We also conducted extensive ablation experiments to verify that the information at each granularity is useful, and under our multi-granularity learning, they can promote the final performance of the model together. In addition, we verify the effectiveness of the two message passing mechanisms.

In future work, we plan to increase the flexibility of the framework by accommodating more granularity levels, such as the subsentence-level and sentence-level information. This would allow for even richer representations and potentially improve performance on more complex tasks. These directions will help refine the MG-GNN and extend its applicability across a broader range of NLP applications.

APPENDIX
SAMPLES FOR OUR TWO DATASETS

To make a better demonstration of our dataset, we provide sample data for each dataset in Tables VII and VIII. The text, annotation, and labels have all been translated into English.

REFERENCES

[1] H. Jin, T. Wang, and X. Wan, "Multi-granularity interaction network for extractive and abstractive multi-document summarization," in *Proc. 58th Annu. Meeting Assoc. Comput. Linguistics*, 2020, pp. 6244–6254. [Online]. Available: https://www.aclweb.org/anthology/2020.acl-main.556

[2] P. Rajpurkar, E. Chen, O. Banerjee, and E. J. Topol, "AI in health and medicine," *Nature Med.*, vol. 28, pp. 31–38, Jan. 2022, doi: 10.1038/s41591-021-01614-0.

[3] X. Ma, J. Guo, R. Zhang, Y. Fan, and X. Cheng, "Pre-train a discriminative text encoder for dense retrieval via contrastive span prediction," in *Proc. 45th Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, 2022, pp. 848–858, doi: 10.1145/3477495.3531772.

[4] T. Gui et al., "A lexicon-based graph neural network for Chinese NER," in *Proc. Conf. Empirical Methods Natural Lang. Process. 9th Int. Joint Conf. Natural Lang. Process.*, 2019, pp. 1040–1050. [Online]. Available: https://www.aclweb.org/anthology/D19-1096

[5] L. Yao, C. Mao, and Y. Luo, "Graph convolutional networks for text classification," in *Proc. AAAI Conf. Artif. Intell.*, 2019, pp. 7370–7377, doi: 10.1609/aaai.v33i01.33017370.

[6] 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*, 2020, pp. 334–339. [Online]. Available: https://www.aclweb.org/anthology/2020.acl-main.31

[7] Y. Zhang and J. Yang, "Chinese NER using lattice LSTM," in *Proc. 56th Annu. Meeting Assoc. Comput. Linguistics*, 2018, pp. 1554–1564. [Online]. Available: https://www.aclweb.org/anthology/P18-1144

[8] C. Xia et al., "Multi-grained named entity recognition," in *Proc. 57th Annu. Meeting Assoc. Comput. Linguistics*, 2019, pp. 1430–1440. [Online]. Available: https://aclanthology.org/P19-1138

[9] S. Li, X. Xu, F. Shen, and Y. Yang, "Multi-granularity separation network for text-based person retrieval with bidirectional refinement regularization," in *Proc. ACM Int. Conf. Multimedia Retrieval*, 2023, pp. 307–315, doi: 10.1145/3591106.3592253.

[10] R. Ma, M. Peng, Q. Zhang, Z. Wei, and X. Huang, "Simplify the usage of lexicon in Chinese NER," in *Proc. 58th Annu. Meeting Assoc. Comput. Linguistics*, 2020, pp. 5951–5960. [Online]. Available: https://www.aclweb.org/anthology/2020.acl-main.528

[11] A. Zhu et al., "DSSL: Deep surroundings-person separation learning for text-based person retrieval," in *Proc. 29th ACM Int. Conf. Multimedia*, 2021, pp. 209–217, doi: 10.1145/3474085.3475369.

[12] L.-H. Lee and Y. Lu, "Multiple embeddings enhanced multi-graph neural networks for chinese healthcare named entity recognition," *IEEE J. Biomed. Health Inform.*, vol. 25, no. 7, pp. 2801–2810, Jul. 2021, doi: 10.1109/JBHI.2020.3048700.

[13] L. Yao, C. Mao, and Y. Luo, "Clinical text classification with rule-based features and knowledge-guided convolutional neural networks," *BMC Med. Inform. Decis. Mak.*, vol. 19, pp. 31–39, 2019, doi: 10.1186/s12911-019-0781-4.

[14] M. Tomas, S. Ilya, C. Kai, S. C. Gregory, and D. Jeffrey, "Distributed representations of words and phrases and their compositionality," in *Proc. Conf. Neural Inf. Process. Syst.*, 2013, pp. 3111–3119. [Online]. Available: https://dl.acm.org/doi/10.5555/2999792.2999959

[15] J. Pennington, R. Socher, and C. Manning, "GloVe: Global vectors for word representation," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2014, pp. 1532–1543. [Online]. Available: https://www.aclweb.org/anthology/D14-1162

[16] Y. Kim, "Convolutional neural networks for sentence classification," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2014, pp. 1746–1751. [Online]. Available: https://www.aclweb.org/anthology/D14-1181

[17] X. Chen, X. Qiu, C. Zhu, P. Liu, and X. Huang, "Long short-term memory neural networks for Chinese word segmentation," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2015, pp. 1197–1206. [Online]. Available: https://www.aclweb.org/anthology/D15-1141

[18] P. Zhou et al., "Attention-based bidirectional long short-term memory networks for relation classification," in *Proc. 54th Annu. Meeting Assoc. Comput. Linguistics*, 2016, pp. 207–212. [Online]. Available: https://www.aclweb.org/anthology/P16-2034

[19] Y. Li, D. Tarlow, M. Brockschmidt, and R. S. Zemel, "Gated graph sequence neural networks," in *Proc. 4th Int. Conf. Learn. Representations*, Y. Bengio and Y. LeCun, Eds., San Juan, Puerto Rico, May 2016. [Online]. Available: http://arxiv.org/abs/1511.05493

[20] H. Peng et al., "Large-scale hierarchical text classification with recursively regularized deep graph-CNN," in *Proc. World Wide Web Conf. World Wide Web*, Lyon, France, 2018, pp. 1063–1072, doi: 10.1145/3178876.3186005.

[21] S. Vashishth, M. Bhandari, P. Yadav, P. Rai, C. Bhattacharyya, and P. Talukdar, "Incorporating syntactic and semantic information in word embeddings using graph convolutional networks," in *Proc. 57th Annu. Meeting Assoc. Comput. Linguistics*, 2019, pp. 3308–3318. [Online]. Available: https://www.aclweb.org/anthology/P19-1320

[22] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics: Hum. Lang. Technol.*, 2019, pp. 4171–4186. [Online]. Available: https://www.aclweb.org/anthology/N19-1423

[23] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. Salakhutdinov, and Q. V. Le, *XLNet: Generalized Autoregressive Pretraining for Language Understanding*, Red Hook, NY, USA: Curran Associates Inc., 2019.

[24] N. Zhang, Q. Jia, K. Yin, L. Dong, F. Gao, and N. Hua, "Conceptualized representation learning for chinese biomedical text mining," 2020, *arXiv:2008.10813*.

[25] X. Zhang, P. Li, and H. Li, "AMBERT: A pre-trained language model with multi-grained tokenization," in *Proc. Findings Assoc. Comput. Linguistics: ACL-IJCNLP*, C. Zong, F. Xia, W. Li, and R. Navigli, Eds., Aug. 2021, pp. 421–435. [Online]. Available: https://aclanthology.org/2021.findings-acl.37

[26] S. Diao, J. Bai, Y. Song, T. Zhang, and Y. Wang, "ZEN: Pre-training chinese text encoder enhanced by N-gram representations," in *Proc. Findings Assoc. Comput. Linguistics: EMNLP*, T. Cohn, Y. He, and Y. Liu, Eds., Nov. 2020, pp. 4729–4740. [Online]. Available: https://aclanthology.org/2020.findings-emnlp.425

[27] H. Li, M. Hagiwara, Q. Li, and H. Ji, "Comparison of the impact of word segmentation on name tagging for Chinese and Japanese," in *Proc. Ninth Int. Conf. Lang. Resour. Eval.*, 2014, pp. 2532–2536. [Online]. Available: http://www.lrec-conf.org/proceedings/lrec2014/pdf/358_Paper.pdf

[28] A. Prakash et al., "Condensed memory networks for clinical diagnostic inferencing," in *Proc. 31st AAAI Conf. Artif. Intell.*, 2017, pp. 3274–3280. [Online]. Available: https://dl.acm.org/doi/10.5555/3298023.3298044

[29] P. Velikovi, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, "Graph attention networks," in *Proc. Int. Conf. Learn. Representations*, 2018, pp. 1–12. [Online]. Available: https://openreview.net/forum?id=rJXMpikCZ

[30] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. 3rd Int. Conf. Learn. Representations*, Y. Bengio and Y. LeCun, Eds., San Diego, CA, USA, May 2015. [Online]. Available: http://arxiv.org/abs/1412.6980

[31] D. Shen et al., "Baseline needs more love: On simple word-embedding-based models and associated pooling mechanisms," in *Proc. 56th Annu. Meeting Assoc. Comput. Linguistics*, 2018, pp. 440–450. [Online]. Available: https://www.aclweb.org/anthology/P18-1041

[32] A. Joulin, E. Grave, P. Bojanowski, and T. Mikolov, "Bag of tricks for efficient text classification," in *Proc. 15th Conf. Eur. Chapter Assoc. Comput. Linguistics*, 2017, pp. 427–431. [Online]. Available: https://www.aclweb.org/anthology/E17-2068

[33] P. Liu, X. Qiu, and X. Huang, "Recurrent neural network for text classification with multi-task learning," in *Proc. 25th Int. Joint Conf. Artif. Intell.*, 2016, pp. 2873–2879.

[34] R. Johnson and T. Zhang, "Deep pyramid convolutional neural networks for text categorization," in *Proc. 55th Annu. Meeting Assoc. Comput. Linguistics*, 2017, pp. 562–570. [Online]. Available: https://www.aclweb.org/anthology/P17-1052

[35] G. Wang et al., "Joint embedding of words and labels for text classification," in *Proc. 56th Annu. Meeting Assoc. Comput. Linguistics*, 2018, pp. 2321–2331. [Online]. Available: https://www.aclweb.org/anthology/P18-1216

[36] A. Vaswani et al., "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., 2017, vol. 30. [Online]. Available: https://proceedings.neurips.cc/paperfiles/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf

[37] B. van Aken et al., "This patient looks like that patient: Prototypical networks for interpretable diagnosis prediction from clinical text," in *Proc. 2nd Conf. Asia-Pacific Chapter Assoc. Comput. Linguistics 12th Int. Joint Conf. Natural Lang. Process.*, 2022, pp. 172–184. [Online]. Available: https://aclanthology.org/2022.aacl-main.14

[38] Z. Jiang, M. Yang, M. Tsirlin, R. Tang, Y. Dai, and J. Lin, ""low-resource" text classification: A parameter-free classification method with compressors," in *Proc. Findings Assoc. Comput. Linguistics*, 2023, pp. 6810–6828. [Online]. Available: https://aclanthology.org/2023.findings-acl.426