DDI-MuG: Multi-aspect Graphs for Drug-Drug Interaction Extraction

Jie Yang[1], Yihao Ding [1], Siqu Long[1], Josiah Poon[1], Soyeon Caren Han[1, 2*]
[1]The University of Sydney, [2] The University of Western Australia
{jyan4704, ydin0771, slon6753}@uni.sydney.edu.au,
{josiah.poon, caren.han}@sydney.edu.au,
caren.han@uwa.edu.au

Abstract

Drug-drug interaction (DDI) may lead to adverse reactions in patients, thus it is important to extract such knowledge from biomedical texts. However, previously proposed approaches typically focus on capturing sentence-aspect information while ignoring valuable knowledge concerning the whole corpus. In this paper, we propose a Multi-aspect Graph-based DDI extraction model, named DDI-MuG. We first employ a bio-specific pre-trained language model to obtain the token contextualized representations. Then we use two graphs to get syntactic information from input instance and word co-occurrence information within the entire corpus, respectively. Finally, we combine the representations of drug entities and verb tokens for the final classification. It is encouraging to see that the proposed model outperforms all baseline models on two benchmark datasets. To the best of our knowledge, this is the first model that explores multi-aspect graphs to the DDI extraction task, and we hope it can establish a foundation for more robust multi-aspect works in the future.

1 Introduction

According to statistics from the U.S. Centers of Disease Control and Prevention, from 2015 to 2018, 48.6 % of Americans used at least one prescription drug in 30 days[1]. More seriously, 20% of the elderly took more than 10 drugs simultaneously (Zhang et al., 2020). However, drug-drug interaction (DDI) may occur when patients take multiple drugs, resulting in reduced drug effectiveness or even, possibly, adverse drug reactions (ADRs) (Zhu et al., 2020). Therefore, the study of DDI extraction can be considerably important to patients' healthcare, as well as clinical research. Currently, a number of drug databases, such as DailyMed (Barrière and Gagnon, 2011), TWOSIDES (Tatonetti et al., 2012) and DrugBank (Wishart et al., 2017) can be used for retrieving DDI knowledge directly. However, with the exponential growth in biomedical literature, huge amounts of the most current and valuable knowledge remain hidden in biomedical literature (Zhang et al., 2020). Thus, the development of an automatic tool to extract DDI is an urgent need.

During the past few years, various deep learning-based approaches, such as (Liu et al., 2016; Zhang et al., 2018; Li and Ji, 2019; Ren et al., 2019; Mondal, 2020; Asada et al., 2020; Fatehifar and Karshenas, 2021; Shi et al., 2022) have been proposed to extract DDI knowledge. It is worth noting that compared with Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM), which are sequential-based architectures, Graph Neural Networks (GNNs) can better deal with complex structural knowledge. Based on this, Li and Ji (2019) combined a Bio-specific BERT (Devlin et al., 2019) and Graph Convolutional Network (GCN) (Kipf and Welling, 2017) to capture contextualized representation together with syntactic knowledge. Shi et al. (2022) adopted the Graph Attention Network (GAT) (Veličković et al., 2018) on an enhanced dependency graph to obtain higher-level drug representations for DDI extraction. However, as examples in Table 1, all the previous models only pay attention to the sentence-aspect features, and do not even exploit the corpus knowledge, which could cause essential clues to be overlooked.

To alleviate the issues mentioned above, in this work, we propose a multi-aspect graphs-based DDI extraction model, DDI-MuG, which can make use of the information in both sentence and corpus aspects. First, we use PubMedBERT to obtain sentence semantic representation. We then apply a GCN with an average pooling layer to capture syntactic features from the input instance, and another GCN with average pooling is employed to model

*Corresponding Author (caren.han@sydney.edu.au)
[1]https://www.cdc.gov/nchs/data/hus/2019/039-508.pdf

Proceedings of the 13th International Workshop on Health Text Mining and Information Analysis (LOUHI), pages 127 - 137

Table 1: Summary of previous neural network-based models and our proposed model

Model	Sentence (semantic)	Sentence (syntactic)	Corpus
AB-LSTM (Sahu and Anand, 2018)	GloVe (Pennington et al., 2014)	No	No
DCNN(Liu et al., 2016)	Order embedding(Lai et al., 2016)	No	No
ASDP-LSTM (Zhang et al., 2018)	Word2Vec(Mikolov et al., 2013)	Dependency parse	No
RHCNN (Sun et al., 2019)	Bio-word emb.(Pyysalo et al., 2013)	Dependency parse	No
GCNN-DDI (Xiong et al., 2019)	Bio-word emb.(Pyysalo et al., 2013)	Dependency parse	No
BERTChem-DDI(Mondal, 2020)	BioBERT(Jinhyuk et al., 2019)	No	No
BERTDesc-DDI(Asada et al., 2020)	SciBERT(Beltagy et al., 2019)	No	No
DDI-MuG (Ours)	PubMedBERT(Gu et al., 2021)	Dependency parse	PMI

the word co-occurrence in the corpus level simultaneously. After that, an attentive pooling is used to integrate and obtain the optimal feature from the output of PubMedBERT and both sentence-aspect and corpus-aspect graphs. Finally, we employ a fully connected neural network in the output layer for the classification. Our proposed model is evaluated on two benchmark datasets: DDIExtraction-2013 (Herrero-Zazo et al., 2013) and TAC 2018 corpora (Demner-Fushman et al., 2018). Experimental results show that our proposed model improves the performance of DDI extraction effectively.

To recap, the main contributions of our work can be summarized as follows:

- We propose a novel neural model, named DDI-MuG, to exploit information from sentence-aspect and corpus-aspect graph. As far as we know, this is the first model that utilizes multi-aspect graphs for the DDI extraction task.
- We explore the effectiveness of different components in DDI-MuG. Experimental results indicate that knowledge from multi-aspect graphs are complementary, and their effective combination can largely improve the performance.
- We evaluate the proposed model on two benchmark datasets, and achieve new state-of-the-art performance on both of them.

The rest of the paper is organized as follows. First, we introduce the background in Section 1. Then, several related works are introduced in Section 2. Next, in Section 3, we explain the framework in the proposed model in detail. We then describe the two benchmark datasets, evaluation metrics, and parameters setting in Section 4. Section 5 presents the experimental results and discussion, and finally, we conclude this work in Section 6.

2 Related Works

Knowledge in many applications is exceedingly complex for a single-aspect network to learn robust representations. Multi-aspect networks have thus emerged naturally in different fields. Khan and Blumenstock (2019) developed a multi-aspect GCNs model to consider different aspects of phone networks for poverty research. They employed subspace analysis and a manifold ranking procedure in order to merge multiple views and prune the graph, respectively. Liu et al. (2020) first constructed semantic-based, syntactic-based, and sequential-based text graphs, and then utilized an inter-graph propagation to coordinate heterogeneous information among graphs. In order to exploit richer sources of graph edge information, Gong and Cheng (2019) resorted to multi-dimensional edge weights to encode edge directions. Similarly, Huang et al. (2020) used multi-dimensional edge weights to exploit multiple attributes, adapting the edge weights before entering into the next layer.

3 Methods

The architecture of the proposed model is illustrated in Figure 1. First, we obtain the contextual semantic representation of the input instances by PubMedBERT. Then, a sentence-aspect graph is constructed to encode the syntactic feature from the dependency path, while a corpus-aspect graph is used to explore word co-occurrence within the entire corpus. Based on the vocabulary and instances analysis, we find that the part-of-speech (POS) tag of words, especially words corresponding to verbs, might be helpful for the final representation. Therefore, we subsequently feed the representations of verbs and drug entities from PubMedBERT, together with the two graphs, into an

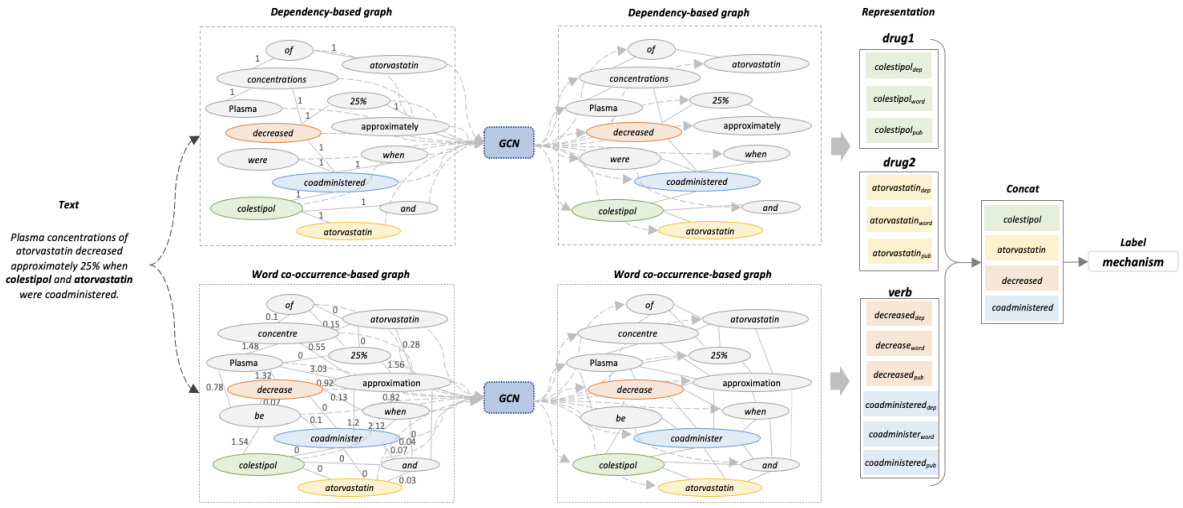

Figure 1: The proposed model architecture. This example is selected from DDIExtraction-2013 dataset. Two drugs are labeled in bold. As the space is limited, only part of the edges are shown in the word co-occurrence-based graph.

attentive pooling layer, to distinguish important features from all representations. Finally, a fully connected layer with softmax is employed to perform the classification. The process is described in the following subsections in detail.

3.1 Encoding sentences with PubMedBERT

PubMedBERT was pre-trained on 14 million biomedical abstracts with 3.2 billion words from scratch. Given an input sentence $S = [w_1, w_2, ..., w_n, ..., w_t]$ with drug entities d_1 and d_2, we convert each word w_i into word pieces and then feed them into PubMedBERT. After the PubMedBERT calculation, we employ average pooling to aggregate vectorial representations of word pieces as the word representations. We denote the two drugs and verbs representations by $drug1_{pub}$, $drug2_{pub}$, and $verbs_{pub}$ respectively.

3.2 Graph construction

Considering a graph with *n* nodes, the node *i* at the *l*-th layer is updated based on the representation of all neighborhood nodes in the *(l-1)*-th layer as follows:

$$H^l = \sigma\left(\hat{A}H^{l-1}W^l\right) \quad (1)$$

Here, $\hat{A} = \widetilde{D}^{-\frac{1}{2}}\widetilde{A}\widetilde{D}^{\frac{1}{2}}$ represents the normalized adjacency matrix, and $\widetilde{A} = A + I$ is the adjacency matrix with added self-connections. $\widetilde{D}$ is the diagonal node degree matrix with $\widetilde{D}(i,i) = \sum_j \widetilde{A}(i,j)$. $H^l \in R^{n*d_l}$ is the node embedding matrix at the l-th layer, n is the number of nodes, d_l indicates the dimension of the node features. Finally, $W^l \in R^{d_l * d_{l+1}}$ denotes a layer-specific trainable weight matrix, and σ is a nonlinear function.

For each input instance, we encode a dependency graph from the current instance and a word co-occurrence over the entire corpus.

3.2.1 Sentence-aspect dependency graph

Dependency parser is widely used in relation classification tasks with the aim of exploring syntactic information of sentence. We apply the Stanford dependency parser (Chen and Manning, 2014) to extract dependency syntactic information. Figure 2 shows the dependency relation of the input text in Figure 1. The connection from *coadministered* to *colestipol* means that *coadministered* is the head word of *colestipol*, and *"nsubjpass"* denotes the *"passive nominal subject"* dependency relation between the two words. We use the word embedding from PubMedBERT as the initial node representations, and set edge weights as 0 or 1 to indicate if two nodes are connected in the dependency path.

Let the node representations in *l*-th layer of the dependency graph be M^l. We apply two graph convolutional layers to update each node, thus the updated M^2 is expressed as follows:

$$M^2 = \sigma\left(\hat{A}M^1W^2\right) \quad (2)$$

Then, an average pooling layer is applied to get the syntactic-based sentence embedding. Let $d_1, d_2, ..., d_n, ..., d_t$ be the updated node representations obtained from graph convolutional layers,

the output of dependency graph, G_{Dep}, is shown as:

$$G_{Dep} = \underset{1 \leq i \leq t}{avg} [d_i] \quad (3)$$

We denote the outputs of drug and verbs representations as $drug1_{dep}$, $drug2_{dep}$, and $verbs_{dep}$, respectively.

3.2.2 Corpus-aspect word co-occurrence graph

Information on the co-occurrence of words indicates the connection between them, such as whether they form as a common phrase or provide clues for classification tasks. Firstly, we first lemmatize each word with Natural Language Toolkit (NLTK) [2]. Then we connect all word pairs in graph, and employ point-wise mutual information (PMI) (Turney, 2001), a word associations measure, to store the word correlation information as an edge weight as follows:

$$A_{ij} = \begin{cases} 1, & i = j \\ PMI(i,j), & i \neq j, PMI(i,j) > 0 \\ 0, & i \neq j, PMI(i,j) \leq 0 \end{cases} \quad (4)$$

The PMI between any two words is calculated as:

$$PMI(i,j) = log \frac{p(i,j)}{p(i)p(j)}, \quad (5)$$

$$p(i,j) = \frac{\#W(i,j)}{\#W}, p(i) = \frac{\#W(i)}{\#W}. \quad (6)$$

where *i, j* are words, *#W(i,j)* is the number of examples in a fixed sliding window that contains both words, *#W(i)* is the number of instances in the sliding window that contain word *i*, and *#W* is the total number of sliding windows. It is worth noting that the entire input sentence is set as the sliding window. Suppose there are 31,738 instances in the corpus, and the word of *"decrease"* and *"coadminister"* appear 1,821 and 953 times respectively, and that they occur 27 times together in the whole corpus. Based on Formula 5 to 6, the PMI between this two words is -4.8. A positive PMI value corresponds to a high correlation between two words, while a negative value means that the two words have a small probability or no probability of occurrence. When two words have a negative PMI value, we view them as non-co-occurring and set their edge weight as 0.

Suppose the node representations in *l*-th layer is N^l. Similar to the dependency graph, the updated N^2 is shown as:

$$N^2 = \sigma(\hat{A}N^1W^2) \quad (7)$$

After an average pooling layer was utilized to get the word co-occurrence-based embedding, the G_{Word} graph is expressed as:

$$G_{Word} = \underset{1 \leq i \leq t}{avg} [w_i] \quad (8)$$

where w_i is the updated *l*-th node representation from graph convolutional layers.

Drug and verbs representations, denotes by $drug1_{word}$, $drug2_{word}$, and $verbs_{word}$, are extracted from G_{Word} and used as input for the next layer.

3.3 Attentive pooling layer

So far, given two drug entities and verbs, we have obtained rich feature representations from PubMedBERT and two graphs. As each instance has a different number of verbs, we apply an attentive pooling to get a fixed-length representation for verbs. In detail, this pooling mechanism computes the weights of feature vectors by using an attention mechanism, allowing it to learn the most significant feature effectively. Let A_{drug1} and A_{drug2} be the combined representation of drug entities from PubMedBERT and the two graphs, and A_{verbs} be the corresponding verbs representation:

$$A_{drug1} = [drug1_{pub}; drug1_{dep}; drug1_{word}] \quad (9)$$

$$A_{drug2} = [drug2_{pub}; drug2_{dep}; drug2_{word}] \quad (10)$$

$$A_{verbs} = [verbs_{pub}; verbs_{dep}; verbs_{word}] \quad (11)$$

where [;] denotes concatenation. These three representations are fed into the attentive pooling layer separately as follows:

$$H_{drug1} = tanh(A_{drug1}) \quad (12)$$

$$\alpha = Softmax(w^a H_{drug1}) \quad (13)$$

$$z_{drug1} = \alpha A_{drug1} \quad (14)$$

where w^a is the learning parameter, α is the attention weights. z_{drug1}, z_{drug2} and z_{verbs} are the representation of the two drugs and verbs, as the output of the attentive pooling layer.

[2] https://www.nltk.org/

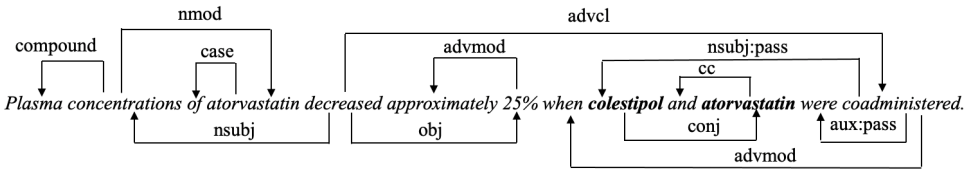

Figure 2: An example of dependency relation. Two drugs are labeled in bold.

3.4 Fully connected and softmax layer

In this layer, the updated representation of two drugs and verbs are concatenated as z_{total}, and a nonlinear activation functions *tanh* is then applied over z_{total} into a fully connected layer. Finally, we deploy a softmax with a dropout layer to get the probability score for each class. The process is expressed as follows:

$$z_{total'} = tanh(z_{total}) \quad (15)$$

$$p(y|x) = Softmax(W^s z_{total'} + b^s) \quad (16)$$

where $z_{total'}$ is the output of the fully connected layer, W^s and b^s are the softmax matrix and the bias parameter, respectively.

4 Experiments

In our experiments, two public DDI extraction corpora, i.e., DDIExtraction-2013 and TAC 2018, were used to evaluate the proposed model. This section introduces the two corpora in detail and then presents the evaluation metrics and parameters setting.

4.1 DDIExtraction-2013 dataset

We obtained the corpus from the challenge SemEval-2013 Task 9 (Segura-Bedmar et al., 2013). This corpus is the major dataset that can be used to evaluate and compare the performance of DDI extraction models. It contains manually annotated sentences from 175 abstracts in MedLine[3], and 730 abstracts in DrugBank[4]. There are four kinds of positive interaction types: *Advice, Effect, Mechanism, Int*. If the two drugs are unrelated, their relations are labeled as *Negative*. The definitions of the five types are as follows:

- **Advice**: a recommendation or advice regarding the simultaneous use of two drugs is described between two drugs.
- **Effect**: an effect or a pharmacodynamic mechanism is described between two drugs.
- **Mechanism**: a pharmacokinetic mechanism is described between two drugs.
- **Int**: a DDI occurs between two drugs, but no additional information is provided.
- **Negative**: there is no interaction between two drugs.

The original corpus suffers from a serious data imbalance problem. For example, the ratio of *Int* to *Negative* instances in the training set is 1:123.7, which heightens the difficulty of classifying drug pairs that hold *Int* relations, and continually affect the overall performance. To alleviate this data imbalance issue, many negative examples are filtered out in earlier studies, e.g., (Kim et al., 2015; Liu et al., 2016; Zhao et al., 2016; Wang et al., 2017; Sahu and Anand, 2018; Zhu et al., 2020). To ensure that the experiment results can be compared fairly with other baseline models, we adopted three rules in (Liu et al., 2016) to remove negative instances:

- If both drugs have the same name, remove the corresponding instances. The assumption is that drug will not interact with itself.
- If one drug is a particular case or an abbreviation of the other, filter out the corresponding instances. Several patterns, such as *"DRUG-A (DRUG-B)"* and *"DRUG-A such as DRUG-B"*, are used to identify such cases.
- If both drugs appear in the same coordinate structure, filter out the corresponding instances. Also, we use some pre-defined patterns, like *"DRUG-A,* $(DRUG - N)^+$*, DRUG-B"*, to filter out such instances.

Table 2 summarizes the statistics and divisions of this corpora.

4.2 TAC 2018 corpus

One of the tasks in "Drug-Drug Interaction Extraction from Drug Labels" track of the Text Analysis

[3]https://www.nlm.nih.gov/bsd/medline.html
[4]https://go.drugbank.com/

Table 2: The statistics of DDIExtraction-2013 corpus.

		Training		Test	
		Original	**Filtered**	**Original**	**Filtered**
Positive	Advice	826	824	221	221
	Effect	1,687	1,676	360	358
	Mechanism	1,319	1,309	302	301
	Int	188	187	96	96
Negative		23,772	19,342	4,737	3,896
Overall		27,792	23,338	5,716	4,872

Conference (TAC) 2018[5] was to detect and extract DDIs from structured product labelings (SPLs). The organizers provided a set of 22 SPLs for training (Training-22). Two other datasets containing 57 and 66 SPLs were provided as test sets. The organizers also provided an additional 180 SPLs (NLM-180) to supplement the training set. Interactions in this corpus are classified into one of the following three types:

- **Pharmacokinetic**: This type includes phrases that demonstrate changes in physiological functions (Demner-Fushman et al., 2018), such as *decrease exposure, increased bioavailability*.
- **Pharmacodynamic**: This type includes phrases that describe the effects of the drugs, e.g., *blood pressure lowering*.
- **Unspecified**: This type corresponds to caution phrases, e.g., *avoid use*.

As the original corpus is in .XML format, we use the dataset in the KLncLSTMsentClf model (Baruah and Kolla, 2018) to train and evaluate our proposed model. In total, we obtain 6,436 training sentences by merging the training-22 and NLM-180 corpora. The two test sets contain 8,205 and 4,256 sentences, respectively.

4.3 Evaluation metrics

precision(P), recall(R) and *F-score(F)* are the major evaluation metrics in the DDI extraction task. In this paper, we adopt the standard micro-average *precision, recall* and *F-score* to evaluate the performance and the formulas are listed as follows:

$$Precision = \frac{TP}{(TP + FP)}, \tag{17}$$

$$Recall = \frac{TP}{(TP + FN)}, \tag{18}$$

$$F - score = \frac{2 * P * R}{(P + R)}. \tag{19}$$

TP(true positive) represents the number of correctly classified positive instances, FP(false positive) denotes the number of negative instances that are misclassified as positive instances, and FN(false negative) is the number of positive instances that are misclassified as negative ones.

4.4 Parameters setting

In our experiment, PyTorch library (Paszke et al., 2019) is used as the computational framework. As there is no development or validation set in the original corpus, we randomly select 20% of the training dataset as the validation set to adjust the model parameters, and the remaining 80% as the training set. The parameters used are shown as follow:

- Maximal length n = 128.
- Embedding size of PubMedBERT m_1 = 768.
- Hidden layer dimension of dependency and co-occurrence graph m_2 & m_3 = 200.
- Mini-batch size = 32.
- Dropout rate p = 0.1.
- Learning rage lr = 0.0001.
- Number of epoch = 10.

5 Results and Discussion

5.1 Results on DDIExtraction-2013

5.1.1 Comparison with baseline methods

We compare the performance of our DDI-MuG with 11 baseline methods. The comparison results of different models are showed in Table 3. The highest value is labeled in bold, and the second highest value is marked underline. In general, deep neural network-based approaches achieve better performance than statistical ML-based methods. It demonstrates the capability and potential of utilizing neural network in DDI extraction task. A notable exception is that the F1-score of SVM-DDI (Kim et al., 2015) is slightly higher than the AB-LSTM model (Sahu and Anand, 2018). This might be due to SVM-DDI (Kim et al., 2015) benefiting from rich and complex lexical and syntactic handcraft features. It can be seen that our DDI-MuG obtains the best overall performances in view of precision and F1-score. In terms of the performances for all four types, DDI-MuG performs best

[5]https://tac.nist.gov/2018/

Table 3: Performance Comparisons on DDIExtraction-2013 Corpus. The highest value is labeled in bold, and the second highest value is marked underline.

Methods	Breakdown F1				Overall performance		
	Advice	Effect	Mechanism	Int	Precision	Recall	F1
Statistical ML-based methods							
UTurKu(Björne et al., 2013)	0.630	0.600	0.582	0.507	0.732	0.499	0.594
WBI(Thomas et al., 2013)	0.632	0.610	0.618	0.510	0.642	0.579	0.609
FBK-irst(Chowdhury and Lavelli, 2013)	0.692	0.628	0.679	0.547	0.646	0.656	0.651
SVM-DDI(Kim et al., 2015)	0.725	0.662	0.693	0.483	-	-	0.670
Deep neural network-based methods							
AB-LSTM(Sahu and Anand, 2018)	0.697	0.683	0.681	0.542	0.678	0.659	0.669
DCNN(Liu et al., 2016)	0.777	0.693	0.702	0.464	0.757	0.647	0.698
Joint AB-LSTM(Sahu and Anand, 2018)	0.794	0.676	0.763	0.431	0.734	0.697	0.715
ASDP-LSTM (Zhang et al., 2018)	0.803	0.718	0.740	0.543	0.741	0.718	0.729
RHCNN (Sun et al., 2019)	0.805	0.734	0.782	0.589	0.773	0.737	0.754
GCNN-DDI (Xiong et al., 2019)	0.835	0.758	0.794	0.514	0.801	0.740	0.770
DREAM(Shi et al., 2022)	0.848	0.761	0.816	0.551	0.823	0.747	0.783
Our methods							
DDI-MuG(with word. graph)	0.893	0.812	0.871	0.599	0.868	0.805	0.835
DDI-MuG(with dep. graph)	0.900	**0.826**	0.865	0.583	0.842	**0.835**	0.839
DDI-MuG	**0.907**	0.823	**0.893**	**0.606**	**0.870**	0.824	**0.847**

on *Advice*, *Mechanism* and *Int*, and obtain the second best performance on *Effect*. It is worth noting that all methods achieve relatively low performance on *Int*. This discrepancy might be caused by the insufficient training samples of *Int*, which leads to these models to be underfitting.

Then, we find out the contributions of multi-aspect graphs to the proposed model. By removing in turn the sentence-aspect dependency graph and corpus-aspect word co-occurrence graph, our method reduces to DDI-MuG(with word. graph) and DDI-MuG(with dep. graph), respectively. From Table 3, we can see that the F1-score of DDI-MuG(with dep. graph) is higher than the F1-score of DDI-MuG(with word. graph), which proves that the syntactic features are indeed valuable for identifying the interaction relation between two drugs. Overall, it can be seen that the F1-score of DDI-MuG surpass the DDI-MuG(with word. graph) and DDI-MuG(with dep. graph) by 0.012 and 0.008, seperately. This indicates that multi-aspect graphs are complementary to each other, and together can serve as an appropriate supplement to contextual information.

5.1.2 Impact of pre-trained embedding

To evaluate the efficiency of the pre-trained language model, we conduct the experiments of replacing PubMedBERT with other similar models. As shown in Table 4, the four bio-specific models, i.e., BioBERT, SciBERT, ouBioBERT(Wada et al., 2020), and PubMedBERT, leading to improvement over standard BERT. DDI-MuG by PubMedBERT achieves the best result for the reason that it was pre-trained on biomedical texts from scratch.

5.1.3 Error analysis

In addition to present the above achievements, it is necessary to discuss the limitations of our approach. One common type of error is that the four kinds of positive instances are often misclassified as negative instances. This is due to the imbalanced data that small instances categories being misclassified as large instance categories. There is another notable error that 34.4% of *Int* type instances are misclassified as *Effect* type. This is because that some *Int* instances have similar semantics to *Effect* instances. For example, in the following two instances:

- *"**arbiturates** may decrease the effectiveness of oral contraceptives, certain antibiotics, quinidine, **theophylline**, corticosteroids, anticoagulants, and beta blockers."*
- *"**sulfoxone** may increase the effects of **barbiturates**, tolbutamide, and uricosurics."*

The words *decrease* and *increase* are the clues for identifying interactions in the two semantically close sentences. However, the first instance belongs to the *Int* type, while the second belongs to *Effect*. The number of *Int* instances is far smaller than the number of *Effect* instances, which also leads to the occurrence of this kind of mistake.

Table 4: The effect of pre-trained embedding. The highest value is labeled in bold.

Pre-trained embedding	P	R	F1
DDI-MuG(by BERT)	0.801	0.790	0.795
DDI-MuG(by BioBERT)	0.843	0.816	0.829
DDI-MuG(by SciBERT)	0.839	0.825	0.832
DDI-MuG(by ouBioBERT)	0.850	**0.826**	0.838
DDI-MuG(by PubMedBERT)	**0.870**	0.824	**0.847**

5.1.4 Are verb representations really helpful?

In our previous vocabulary and instances analysis, we found that in the DDIExtraction-2013 corpus, when instances contain the words *inhibit, increased, decreased*, there is a great possibility that the drug pair has the *Mechanism* relation. On the other hand, when instances contain *avoided, recommended* or *administered*, the drug pair is likely to have the *Advice* relation.

Thus, to further investigate how the verbs are important for the final classification, we studied the effect of extracting DDI only from the drug information, without using the verbs knowledge. Table 5 shows the comparison of the performance with and without the verbs information. This result indicates verbs representation can serve as a supplement to improve the model performance.

Table 5: The comparison of with or without verbs information. The highest value is labeled in bold.

	Precision	Recall	F-score
DDI-MuG(drug-only)	0.863	0.823	0.843
DDI-MuG(all)	**0.870**	**0.824**	**0.847**

5.2 Results on TAC 2018

5.2.1 Comparison with baseline model

Since we use the same dataset as KLncLSTMsentClf (Baruah and Kolla, 2018), we view it as the baseline model. From Table 6, we can see that our proposed model achieves better results in both two test sets, which indicates the transferability of our proposed model.

Table 6: Comparison with baseline models on the TAC 2018 corpus. The highest value is labeled in bold.

Dataset	Model	P	R	F1
Test1	KLncLSTMsentClf	0.470	0.620	0.530
Test1	DDI-MuG(with word. graph)	0.717	0.712	0.715
Test1	DDI-MuG(with dep. graph)	0.688	0.718	0.703
Test1	DDI-MuG(all)	**0.721**	**0.728**	**0.723**
Test2	KLncLSTMsentClf	0.490	0.670	0.567
Test2	DDI-MuG(with word. graph)	0.710	0.726	0.718
Test2	DDI-MuG(with dep. graph)	0.713	0.730	0.721
Test2	DDI-MuG(all)	**0.717**	**0.743**	**0.729**

6 Conclusions

In this paper, we propose DDI-MuG, a novel multi-aspect graphs framework for DDI extraction task. Concretely, a bio-specific pre-trained language model, PubMedBERT, is firstly employed to encode the context information of each word from the aspect of sentence semantic information. Then, two graphs are utilized to explore sentence syntactic and corpus word co-occurrence information, respectively. After that, attentive pooling mechanism is employed to update the representations of drug entities and verbs. Finally, by feeding the concatenated representation of the two drugs and verbs into a fully connected and softmax classifier, the interaction between two drugs is obtained. Extensive comparison experiments with baseline models on two public datasets verify the effectiveness of utilizing multi-aspect graphs in the DDI extraction task.

For the future work, there are at least two directions could be considered. Firstly, the performance on categories with small training samples, like *Int* in the DDIExtraction-2013 corpora, is unsatisfactory. The solution of contrastive learning can be explored. Secondly, drug knowledge from external databases could be integrated in the architecture for richer drug representations.

References

Masaki Asada, Makoto Miwa, and Yutaka Sasaki. 2020. Using drug descriptions and molecular structures for drug–drug interaction extraction from literature. *Bioinformatics*, 37(12):1739–1746.

Caroline Barrière and Michel Gagnon. 2011. Drugs and disorders: From specialized resources to web data. In *Workshop on Web Scale Knowledge Extraction, 10th International Semantic Web Conference*.

Gaurav Baruah and Maheedhar Kolla. 2018. Klicklabs at the TAC 2018 drug-drug interaction extraction from drug labels track. In *Proceedings of the 2018 Text Analysis Conference, TAC 2018, Gaithersburg, Maryland, USA, November 13-14, 2018*. NIST.

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. Scibert: Pretrained language model for scientific text. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pages 3615–3620, Hong Kong, China. Association for Computational Linguistics.

Jari Björne, Suwisa Kaewphan, and Tapio Salakoski. 2013. UTurku: Drug named entity recognition and drug-drug interaction extraction using SVM classification and domain knowledge. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 651–659, Atlanta, Georgia, USA. Association for Computational Linguistics.

Danqi Chen and Christopher Manning. 2014. A fast and accurate dependency parser using neural networks. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 740–750, Doha, Qatar. Association for Computational Linguistics.

Md. Faisal Mahbub Chowdhury and Alberto Lavelli. 2013. FBK-irst : A multi-phase kernel based approach for drug-drug interaction detection and classification that exploits linguistic information. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 351–355, Atlanta, Georgia, USA. Association for Computational Linguistics.

Dina Demner-Fushman, Kin Wah Fung, Phong Do, Richard David Boyce, and Travis R. Goodwin. 2018. Overview of the tac 2018 drug-drug interaction extraction from drug labels track. *Theory and Applications of Categories*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Mohsen Fatehifar and Hossein Karshenas. 2021. Drug-drug interaction extraction using a position and similarity fusion-based attention mechanism. *Journal of Biomedical Informatics*, 115:103707.

Liyu Gong and Qiang Cheng. 2019. Exploiting edge features for graph neural networks. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9203–9211.

Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. 2021. Domain-specific language model pretraining for biomedical natural language processing. *ACM Trans. Comput. Healthcare*, 3(1).

María Herrero-Zazo, Isabel Segura-Bedmar, Paloma Martínez, and Thierry Declerck. 2013. The ddi corpus: An annotated corpus with pharmacological substances and drug–drug interactions. *Journal of Biomedical Informatics*, 46(5):914 – 920.

Zhichao Huang, Xutao Li, Yunming Ye, and Michael K. Ng. 2020. Mr-gcn: Multi-relational graph convolutional networks based on generalized tensor product. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, pages 1258–1264. International Joint Conferences on Artificial Intelligence Organization.

Lee Jinhyuk, Yoon Wonjin, and Kim. 2019. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Muhammad Raza Khan and Joshua E. Blumenstock. 2019. Multi-gcn: Graph convolutional networks for multi-view networks, with applications to global poverty. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01):606–613.

Sun Kim, Haibin Liu, Lana Yeganova, and W. John Wilbur. 2015. Extracting drug–drug interactions from literature using a rich feature-based linear kernel approach. *Journal of Biomedical Informatics*, 55:23–30.

Thomas N. Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks. In *5th International Conference on Learning Representations*, Toulon, France. OpenReview.net.

Siwei Lai, Kang Liu, Shizhu He, and Jun Zhao. 2016. How to generate a good word embedding. *IEEE Intelligent Systems*, 31(6):5–14.

Diya Li and Heng Ji. 2019. Syntax-aware multi-task graph convolutional networks for biomedical relation extraction. In *Proceedings of the Tenth International Workshop on Health Text Mining and Information Analysis (LOUHI 2019)*, pages 28–33, Hong Kong. Association for Computational Linguistics.

Shengyu Liu, Buzhou Tang, Qingcai Chen, and Xiaolong Wang. 2016. Drug-drug interaction extraction via convolutional neural networks. *Computational and Mathematical Methods in Medicine*, 2016.

Xien Liu, Xinxin You, Xiao Zhang, Ji Wu, and Ping Lv. 2020. Tensor graph convolutional networks for text classification. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):8409–8416.

Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. 2013. Linguistic regularities in continuous space word representations. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 746–751, Atlanta, Georgia. Association for Computational Linguistics.

Ishani Mondal. 2020. BERTChem-DDI : Improved drug-drug interaction prediction from text using chemical structure information. In *Proceedings of Knowledgeable NLP: the First Workshop on Integrating Structured Knowledge and Neural Networks for NLP*, pages 27–32, Suzhou, China. Association for Computational Linguistics.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. *PyTorch: An Imperative Style, High-Performance Deep Learning Library*. Curran Associates Inc., Red Hook, NY, USA.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

Sampo Pyysalo, F Ginter, Hans Moen, T Salakoski, and Sophia Ananiadou. 2013. Distributional semantics resources for biomedical text processing. *Proceedings of Languages in Biology and Medicine*, pages 39–44.

Yafeng Ren, Hao Fei, and Donghong Ji. 2019. Drug-drug interaction extraction using a span-based neural network model. In *2019 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 1237–1239.

Sunil Kumar Sahu and Ashish Anand. 2018. Drug-drug interaction extraction from biomedical texts using long short-term memory network. *Journal of Biomedical Informatics*, 86:15–24.

Isabel Segura-Bedmar, Paloma Martínez, and María Herrero-Zazo. 2013. SemEval-2013 task 9 : Extraction of drug-drug interactions from biomedical texts (DDIExtraction 2013). In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 341–350, Atlanta, Georgia, USA. Association for Computational Linguistics.

Yong Shi, Pei Quan, Tianlin Zhang, and Lingfeng Niu. 2022. Dream: Drug-drug interaction extraction with enhanced dependency graph and attention mechanism. *Methods*, 203:152–159.

Xia Sun, Ke Dong, Long Ma, Richard Sutcliffe, Feijuan He, Sushing Chen, and Jun Feng. 2019. Drug-drug interaction extraction via recurrent hybrid convolutional neural networks with an improved focal loss. *Entropy*, 21(1).

Nicholas P. Tatonetti, Patrick P. Ye, Roxana Daneshjou, and Russ B. Altman. 2012. Data-driven prediction of drug effects and interactions. *Science translational medicine*, 4(125).

Philippe Thomas, Mariana Neves, Tim Rocktäschel, and Ulf Leser. 2013. WBI-DDI: Drug-drug interaction extraction using majority voting. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 628–635, Atlanta, Georgia, USA. Association for Computational Linguistics.

Peter Turney. 2001. Mining the web for synonyms: Pmi-ir versus lsa on toefl. *Proceedings of the Twelfth European Conference on Machine Learning*, pages 491–502.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph attention networks. In *International Conference on Learning Representations*.

Shoya Wada, Toshihiro Takeda, Shiro Manabe, Shozo Konishi, Jun Kamohara, and Yasushi Matsumura. 2020. A pre-training technique to localize medical bert and to enhance biomedical bert.

Wei Wang, Xi Yang, Canqun Yang, Xiaowei Guo, Xiang Zhang, and Chengkun Wu. 2017. Dependency-based long short term memory network for drug-drug interaction extraction. *BMC bioinformatics*, 18(Suppl 16):578–578.

David S Wishart, Yannick D Feunang, An C Guo, Elvis J Lo, Ana Marcu, Jason R Grant, Tanvir Sajed, Daniel Johnson, Carin Li, Zinat Sayeeda, Nazanin Assempour, Ithayavani Iynkkaran, Yifeng Liu, Adam Maciejewski, Nicola Gale, Alex Wilson, Lucy Chin, Ryan Cummings, Diana Le, Allison Pon, Craig Knox, and Michael Wilson. 2017. DrugBank 5.0: a major update to the DrugBank database for 2018. *Nucleic Acids Research*, 46(D1):D1074–D1082.

W. Xiong, F. Li, H. Yu, and D. Ji. 2019. Extracting drug-drug interactions with a dependency-based graph convolution neural network. In *2019 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 755–759, Los Alamitos, CA, USA. IEEE Computer Society.

Tianlin Zhang, Jiaxu Leng, and Ying Liu. 2020. Deep learning for drug–drug interaction extraction from the literature: a review. *Briefings in bioinformatics*, 21(5):1609–1627.

Yijia Zhang, Wei Zheng, Hongfei Lin, Jian Wang, Zhihao Yang, and Michel Dumontier. 2018. Drug–drug interaction extraction via hierarchical RNNs on sequence and shortest dependency paths. *Bioinformatics*, 34(5):828–835.

Zhehuan Zhao, Zhihao Yang, Ling Luo, Hongfei Lin, and Jian Wang. 2016. Drug drug interaction extraction from biomedical literature using syntax convolutional neural network. *Bioinformatics (Oxford, England)*, 32(22):3444–3453.

Yu Zhu, Lishuang Li, Hongbin Lu, Anqiao Zhou, and Xueyang Qin. 2020. Extracting drug-drug interactions from texts with biobert and multiple entity-aware attentions. *Journal of Biomedical Informatics*, 106:103451.