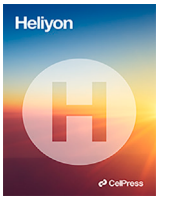
Heliyon 9 (2023) e16819

Contents lists available at ScienceDirect

Heliyon

journal homepage: www.cell.com/heliyon



Research article

MTMG: A multi-task model with multi-granularity information for drug-drug interaction extraction

Haohan Deng, Qiaoqin Li, Yongguo Liu *, Jiajing Zhu

Knowledge and Data Engineering Laboratory of Chinese Medicine, School of Information and Software Engineering, University of Electronic Science and Technology of China, Chengdu 610054, China

ARTICLE INFO

Keywords:
Drug-drug interactions
Drug named entity recognition
Multi-task framework
Multi-granularity

ABSTRACT

Drug-drug interactions (DDIs) extraction includes identifying drug entities and interactions between drug pairs from the biomedical corpus. The discovery of potential DDIs aids in our understanding of the mechanisms underlying adverse reactions or combination therapy to improve patient safety. The manual extraction of DDIs is very time-consuming and expensive; therefore, computer-aided extraction of DDIs is vital. Many neural network-based methods have been proposed and achieved good efficiency in the extraction of DDIs over the years. However, most studies improved the performance of DDIs extraction with various external drug features while directly using golden drug entities, leading to error propagation and low universality in practical application. In this paper, we propose a new multi-task framework called MTMG, which changes DDIs extraction from a sentence-level classification task to a sequence labeling task named Drug-Specified Token Classification (DSTC). The proposed approach, MTMG, jointly trains DSTC with drug named entity recognition (DNER) and two sentence-level auxiliary tasks we designed. We aim to improve the performance of the entire DDIs extraction pipeline by better using the correlation between entities and relationships and, to the extent possible, using the information of varying granularity implied in the dataset. Experimental results show that MTMG can both improve the accuracy of DNER and DDIs extraction and outperforms state-of-the-art technique.

1. Introduction

In multi-drug combinations, the effect of one medication may be enhanced or diminished by another, which may also cause severe side effects or lead to death [1]. These interactions are known as drug-drug interactions (DDIs), and for these reasons, the study of DDIs has attracted widespread attention in the biomedical community.

As a result of the exponential growth of biomedical literature, manually extracting DDIs becomes impractical. At the same time, several databases provide DDIs information for medical-related professionals or researchers, such as DrugBank, SFINX, and PharmGKB. However, most of these databases are only updated every 1-3 years [2], which is far from keeping pace with the growth of the medical literature—resulting in significant amounts of DDIs information hidden in biomedical publications. Thus, automatic extraction of drug entities and DDIs is critical for clinical medical research and patient health management.

* Corresponding author.
E-mail address: liuyg@uestc.edu.cn (Y. Liu).

https://doi.org/10.1016/j.heliyon.2023.e16819
Received 8 May 2023; Received in revised form 29 May 2023; Accepted 30 May 2023

Available online 1 June 2023

Thanks to the DDI2013 challenge [3], It clarified that the goal of the DDI task is to identify drug-named entities and classify their interactions in the literature. More importantly, it provides a gold standard dataset containing entity classes and interaction types, greatly accelerating the development of automatic extraction methods on DDI tasks.

In current years, as neural networks have performed so well in other fields, a variety of neural network-based techniques have been proposed for DDIs extraction, divided into the pipelined and joint extraction approaches. For the traditional pipelined approach, Drug Name Entity Recognition (DNER) and DDIs classification are considered two independent tasks, with most studies focusing solely on the latter. In other words, most of the work construct various neural network structures and used additional drug knowledge to classify the relation of drug pairs pre-labeled in biomedical texts. In this regard, early researchers encoded input text sequences into the contextualized embedding using convolutional neural networks (CNNs) or recurrent neural networks (RNNs) to extract DDIs based on learned semantic features [4,2]. The development of CNNs provides a powerful method for extracting DDIs, the researchers attempt to extract additional features from some drug-related graph data to help complete DDIs extraction [5]. In addition, the widely used technologies in recent years, such as the pre-training model BERT [6], have also been successfully applied to the DDIs extraction task.

Although considerable research has been performed on individual DDI classification task, additional tools are needed to identify drug entities in text first in practical applications, and the primary drawbacks of such pipeline methods are: (1) error propagation between the DNER and DDIs classification. (2) potentially helpful knowledge from one task is not used by the other (e.g., classifying a relationship between two drugs may assist in identifying the type of two drugs and vice versa). (3) the performance in practice is too dependent on the drug entity recognition results of other natural language processing tools and the quality of additional drug features. In fact, in a more general field, recent studies proposed joint approaches to identify entities and their relationships overcoming the aforementioned problems and showing excellent application prospects [7,8]. Then, the researchers learned the idea of the joint model applied in the general field, made improvements according to the structure and characteristics of biomedical texts, and proposed more advanced joint extraction methods of drug entities and interactions [9]. Its limitation is to complete the joint extraction of drug entities and relationships by designing a complex tagging scheme, the difficulty of model learning and the time complexity is increased compared with the standard method of linear complexity.

In this paper, to deal with the aforementioned limitations, we design multiple tasks framework called MTMG to improve the entire DDIs extraction pipeline, including DNER and DDIs classification, by exploiting the multi-granularity information (token-level granularity and sentence-level granularity) implied in the DDI extraction 2013 dataset without relying on any additional external drug knowledge and complex feature engineering. First, we use the pre-trained model BioBERT as a shared encoding layer because of its good performance in the biomedical domain. Then, as a main sequence labeling task, DNER aims to identify the boundaries of drug named entities and classify their classes. Besides the DNER, we design another main sequence labeling task called Drug-Specified Token Classification (DSTC) based on the DDIs classification task, to classify the relationship between each token and a given drug entity. Our primary motivation is to mine useful information between DNER and DDIs classification. Finally, to provide more robust coarse-grained information and useful training signals for the main tasks, we employ two related sentence-level auxiliary binary classification tasks, i.e. a binary classification task for predicting whether a sentence contains more than two entities or not (ENC, Entity Number Classification) and another binary classification task for predicting whether a sentence contains positive drug-drug interactions or not (PDC, Positive DDI Classification). Examples of task labels in MTMG are shown in Table 1.

Table 1
Examples of labels for tasks. The token-level tasks are the main tasks. In addition to DNER, we designed DSTC to predict the relationship between each token in the input text and a specified given drug. In DSTC, A token belonging to a drug entity is directly labeled with the type of interaction between the entity and the specified drug. Otherwise, it is assigned an 'O' label. Furthermore, we designed two sentence-level auxiliary tasks to help improve the main tasks. Concretely, the first sentence-level task ENC determines whether there are more than two drug entities in a sentence; the second sentence-level task PDC predicts if a sentence contains a drug pair with positive drug-drug interactions. It is obvious that more labels have been added to our data to support the auxiliary task training. The extra labels, however, could be extracted from the original dataset and do not require extra manual annotations. In summary, the goal of this research was to exploit the multi-granularity information implicit in the raw DDI dataset to enhance the efficiency of the entire DDIs extraction pipeline.

Text	**Acetazolamide** may increase the effects of other **folic acid antagonists**.
Token-level tasks	DNER: B-drug O O O O O O B-group I-group I-group O
	DSTC:
	Acetazolamide as the specified drug: O O O O O O O effect effect effect O
	folic acid antagonists as the specified drug: effect O O O O O O O O O O O
Sentence-level tasks	ENC: False
	PDC: True
Text	Co-administration with **antifungal agents** such as **ketoconazole** or **ltraconazole** is not recommended.
Token-level tasks	DNER: O O B-group I-group O O B-drug O B-drug O O O O
	DSTC:
	antifungal agents as the specified drug: O O O O O O false O false O O O O
	ketoconazole as the specified drug: O O false false O O O O false O O O O
	itraconazole as the specified drug: O O false false O O false O O O O O O
Sentence-level tasks	ENC: True
	PDC: False

This paper contributes the following:

(1) We design the DDIs classification as a sequence labeling task to fully explore the mutual benefit between DNER and DDIs classification and mitigate the error propagation problem. The complete drug entities and their relationships, including overlapping relationships, can be parsed from the output of both main tasks.
(2) In order to maximize the value of implicit information contained in the dataset, we design two sentence-level auxiliary tasks to provide coarse-grained information for training a more robust token representation for the main tasks. And we implement this work by jointly learning sentence-level and token-level labels without additional drug knowledge and complex feature engineering.
(3) We conduct experiments of the whole pipeline on the DDI extraction 2013 dataset, and the results show that MTMG outperforms existing pipelined or joint methods, achieving best performance in both the drug named entity recognition and the DDIs classification.

The structure of the remaining parts of the paper is as follows. In section 2, related work is summarized. Section 3 discusses materials and the methodology. In section 4, the experimental settings and results are discussed. Finally, conclusions are drawn in section 5.

2. Related work

Drug-drug interactions extraction has recently become a popular research area in pharmaceutical and biomedical research. The whole task of DDIs extraction is divided into the pipelined method and the joint method. Below we summarize the innovation and development of DDIs extraction in recent years.

CNNs-based or RNNs-based were first applied to DDIs extraction task. Liu et al. [4] pioneered the application of CNNs to the DDIs extraction, who used position embeddings and word embeddings to capture the semantic feature. Zhao et al. [10] proposed a new embedding vector called syntactic word embedding and combined the syntactic information with traditional features to extract DDIs using CNN. Sun et al. [11] proposed a novel recursive hybrid convolution neural network (RHCNN) for DDIs extraction, a recurrent structure first learned the complete semantic features. Then, a hybrid convolutional neural network is used to obtain dependency features and local context features of consecutive words to form sentence-level features.

In addition to using CNNs to learn the features of input, some researchers used RNNs to solve this problem because of the intrinsic sequential features inherent in the input text. Zhang et al. [2] introduced a hierarchical RNN-based method, learning the feature representation of the subsequences and shortest dependency paths (SDP) to extract DDIs. Wang et al. [12] constructed linear, depth first search (DFS), and breadth first search (BFS) channels based on BiLSTM for sentences and their SDP, learned distance-based and dependency-based features, and combined them for the final output. Jiang et al. [13] proposed a skeleton long short term memory (skeleton-LSTM) network to grasp the skeleton structure of DDI instances and learn the representation of the commonness of DDI instance structure. Mostafapour and Dikenel [14] combined different attention mechanisms with BiLSTM layers, The first level attention mechanism used to highlight the words that are more related to the target entities and the top attention mechanism learned the weight for the sentence-level features that are related to the annotated relationship. Similarly, Fatehifar and Karshenas [15] also proposed a novel attention mechanism based on BiLSTM, but to effectively integrate word similarity and position information for improving the discrimination of important words. Zhang et al. [16] proposed a hybrid model that uses RNN to learn features of sentence sequences and CNN to learn features of dependency sequences and then combines the outputs of the two networks to complete the extraction of relations. Another hybrid model, called SGRU-CNN, was proposed by Wu et al. [17], which applies stacked BiGRU for lexical information and CNN for entity position information.

The above neural network-based approaches parse the features inherent in the input text, such as semantic features, entity position and part-of-speech tagging information, through different network structures. However, some studies in recent years have shown that neural networks achieve better results if supplied with more impactful information. Zhu et al. [18] presented an approach based on BioBERT [19], a fine-tuned BERT (Bidirectional Encoder Representations from Transformers) model trained on a biomedical corpus. They used Doc2Vec to learn the feature of drug description information and a attention layer to fuse three different entity information, and obtained a better result than using BioBERT only. In a recent study, Huang et al. [20] proposed a text-augmented method based on BioBERT. They used additional markers to highlight drug entities in the text by adding their boundary and type information. They also used BioGPT2 (Biomedical Generative Pre-Training 2.0) [21] to produce meaningful samples that could help in the discovery of new biomedical interactions. Based on PubMedBert [22], Jin et al. [23] proposed a method of integrating neural network and knowledge graph to fuse three-level semantic features, aiming to alleviate the problem of misclassification of DDIs. At the same time, various graph representation learning-based methods have been applied to DDI extraction tasks to obtain features from drug-related graph structure knowledge. Asada et al. [5] proposed a method using GCN [24] to learn the feature of molecular information and extract DDIs using CNN. In another model, Asada [25] used drug description and molecular structure information simultaneously and improved the efficiency of DDIs extraction significantly. Feng et al. [26] used GCN to learn topological relationships between each drug node and their neighbor drugs from existing DDIs networks. More recently, Wang et al. [27] used GCN and a bond-aware attentive message propagating approach to encode drugs' DDIs and molecular structure features and integrated multi-view information through a novel contrast learning component, and this method outperformed the state-of-the-art methods consistently. Recently, 3D structural information of drugs was also exploited by scholars for DDIs extraction. He et al. [28] proposed a model consisting of a pre-trained text attention mechanism and a 3D graph neural network and to improve the extraction ability for DDIs.

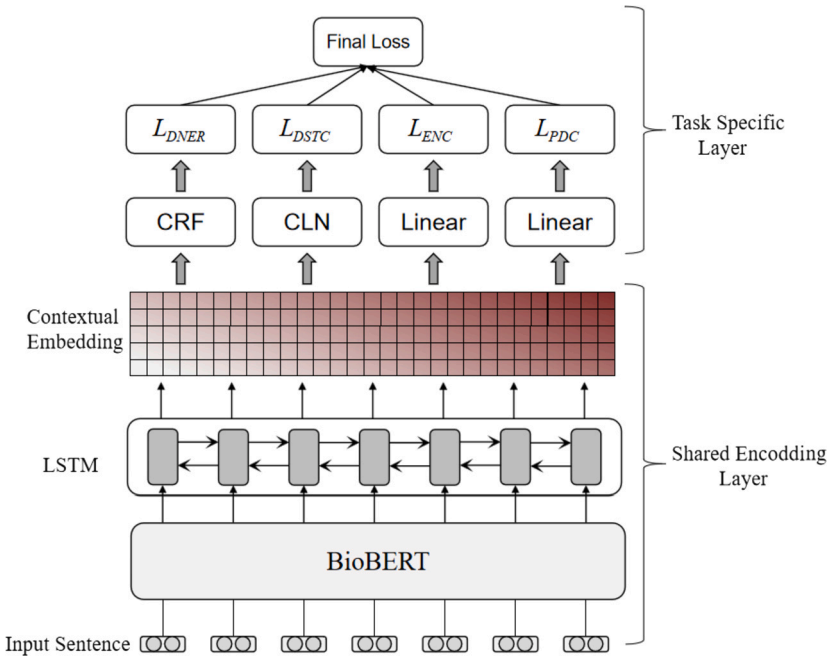


Fig. 1. The architecture of MTMG. DNER and DSTC are two main tasks for learning token-level labels, and PDC and EDC are two auxiliary tasks for learning sentence-level labels. Linear means linear layers, connecting every input neuron to every output neuron. CRF means conditional Random Fileds, a type of discriminative undirected probabilistic graphical model. And CLN is Conditional layer normalization, which can incorporate the context features into a certain variable as a condition, so as to integrate the context information into the normalized word vector space.

In general, most computational methods achieve satisfactory results by extracting chemical and biological properties of drugs from different drug-related databases, they still have many limitations in practical applications. First, previous methods calculate the drug-drug similarity based on multiple features to predict whether an interaction occurs between drug pairs, which require a time-consuming entity linking and feature engineering due to multiple names of the same drug, confusing use of drug names in different countries or regions, frequent abbreviations and acronyms [29–31]. Meanwhile, some features are missing in most drug databases resulting in the properties of some kinds of drugs that may not be readily available, which are crucial to these models to predict DDIs. One solution is to simply move out these drugs that lack a particular property. However, this is impractical in the real scenario [32]. Therefore, these models are deficient in scalability and generalizability, and their performance in practical is heavily dependent on the quality of chemical or pharmacological features.

It is also worth noting that the mentioned DDIs prediction methods focus only on classifying the relationship between given drug pairs in the input text while ignoring the critical predecessor task in the pipeline, Drug Named Entity Recognition (DNER). This separated framework allows the overall task of DDIs extraction to be more flexible to handle. However, in practical applications, this ignores the correlation between the two subtasks and often requires external NLP toolkits to identify drug entities, leading to error propagation problem that affect the results of DDIs classification. In fact, early studies on joint modeling named entity recognition and relation extraction in a more general field show excellent application prospects [7,8]. However, due to the different types of two tasks and still training under the pipeline sequence, the correlation between the two tasks is not fully exploited. It is informative that Zhao et al. [33] transformed the named entity normalization task into a sequence-labeling task, trained in parallel with the NER task, and their experimental results confirmed that the mutual benefits between the two tasks were fully exploited.

Regarding the task of DDIs extraction, there are few studies that focus on the joint extraction of drug entities and DDIs or provide an evaluation of the whole pipeline. Suárez-Paniagua et al. [34] proposed a two-stage pipelined method and evaluated the two subtasks completely for the first time. Zaikis and Vlahavas [35] proposed an end-to-end pipeline method based on the Transformer architecture and set up a set of drug entity extraction rules to alleviate the error propagation problem. The method proposed by Luo et al. [9] is the most advanced in the joint method, they transformed the extraction of drug entities and relationships into a single sequence labeling problem based on the idea of Zheng et al. [36], and optimized the labeling strategy and the extraction rule of triplets, which can effectively extract the overlapping relationship from the input text, and greatly improve the performance by using the ELMo (Embedding from language Model) embedding.

3. Materials and method

In this section, the dataset and the architecture of MTMG are described, and an overview of the method to accomplish DNER and DDI triplets extraction is provided. The structure of MTMG is shown in Fig. 1. In the training phase, to complete the training of all tasks and achieve the best model results, we provide golden labels for each of the designed tasks, and of course, all the additional

Table 2
DDI corpus statistics.

Corpus		Training set			Test set		
		DrugBank	MedLine	Overall	DrugBank	MedLine	Overall
Entity types	*Drug*	8, 197	1, 228	9, 425	1, 518	346	1, 864
	Group	3, 206	193	3, 399	626	41	667
	Brand	1, 423	14	1, 437	347	22	369
	Drug_n	103	401	504	21	119	140
DDIs types	*Mechanism*	1, 257	62	1, 319	278	24	302
	Effect	1, 535	152	1, 687	298	62	360
	Advice	818	8	826	214	7	221
	Int	178	10	188	94	2	96
	False	22, 118	1, 547	23, 665	4, 381	356	4, 737

labels in our data are derived from the original dataset without any additional manual annotation. The following sections provide a comprehensive description of the process.

3.1. Dataset

The DDI Extraction 2013 corpus [3] is a valuable dataset that provides standard training and test data to automatically extract DDIs from biomedical texts. The corpus includes 233 chosen abstracts about DDIs from MedLine as well as 784 documents describing drug interactions from the DrugBank database. For each sentence in the dataset, pharmacological substances and their types, offsets (start and end positions in the sentence), all possible drug pairs and their interactions were manually annotated.

The pharmacological substances in the DDI Extraction 2013 corpus are divided into four kinds for drug named entity recognition. Generic drugs are defined as *drug*, those active substances that are not approved for human use are defined as *drug_n* and branded drugs and drug groups are defined as *brand* and *group* respectively. And the interactions between drug pairs are labeled as the following types:

- mechanism: This type is designated when describing pharmacokinetic mechanisms, such as changes in the concentration of drugs in the blood.
- effect: This type is designated when describing an effect of DDIs or a pharmacodynamic mechanism, It can be a pharmacological effect or changes in the patient's symptoms caused by the simultaneous use of drugs.
- advice: This type is designated when describing recommendations about taking two drugs simultaneously.
- int: This type is designated When a pair of drugs interact, but no additional information is provided in the sentence to determine the kind of interaction.
- false: When there is no interaction between two drugs, this type is assigned.

The original data set provides separate training sets for each of the two tasks—DNER and DDIs classifications—as well as a single test set for both tasks. Since there is no test data for DDIs classification in the test set of DNER, MTMG only uses the test set of DDIs classification that contains the same test data of drug entity in the evaluation stage in order to jointly evaluate the two tasks. At the same time, to ensure that each test sample has drug entity instances and DDI instances, we delete the test sample with the number of drug entities less than 2 in the sentence. Table 2 shows the statistical information in the corpus.

According to the statistics, the data for both tasks are incredibly unbalanced. The type *Drug* makes up 64% of instances for DNER and 23% of instances for the type *group*, while the types *brand* and *drug_n* make up just 10% and 3% of occurrences, respectively. The data imbalance is considerably worse for DDI classification, with just 15% of the samples being positive and the other 85% being negative. Moreover, the number of labels in the positive is also unbalanced. The type *int* only accounts for 4.6% of the total number of positive, far less than the other three positive types. In the subsequent sections, we will introduce the mitigation of the data imbalance by an optimized loss function.

3.2. Tasks

As described in section 1, MTMG simultaneously utilizes the dataset's latent multi-granularity information and includes two main tasks and two auxiliary tasks. In addition to DNER being a primary task, we design another sequence labeling task Drug-Specified Token Classification (DSTC), to better exploit the mutual benefit between entities and relations. At the same time, two auxiliary tasks are introduced to provide more robust sentence-level information for the main tasks and improve the performance of the main tasks. Fig. 2 depicts the input to MTMG and the result of each task. Given a input sentence $X = \{x_1, \ldots, x_i, \ldots, x_n\}$, where n denotes the length of X, x_i represents a token. The detailed descriptions of tasks are as follows.

DNER: Given the input sentence X, the purpose of DNER is to predict the set of labels $Y = \{y_1, \ldots, y_i, \ldots, y_n\}$, where y_i is the entity type label corresponding to token x_i, which is predefined according to the BIO (Beginning, Inside, Outside) scheme.

DSTC: Drug-Specified Token Classification is a token-level classification task as shown in Fig. 3. Given X contains the drug entity set $E = \{e_1, \ldots, e_i, \ldots, e_n\}$. Each drug entity will be treated as the specified drug one time. When entity e_i acts as the specified drug,

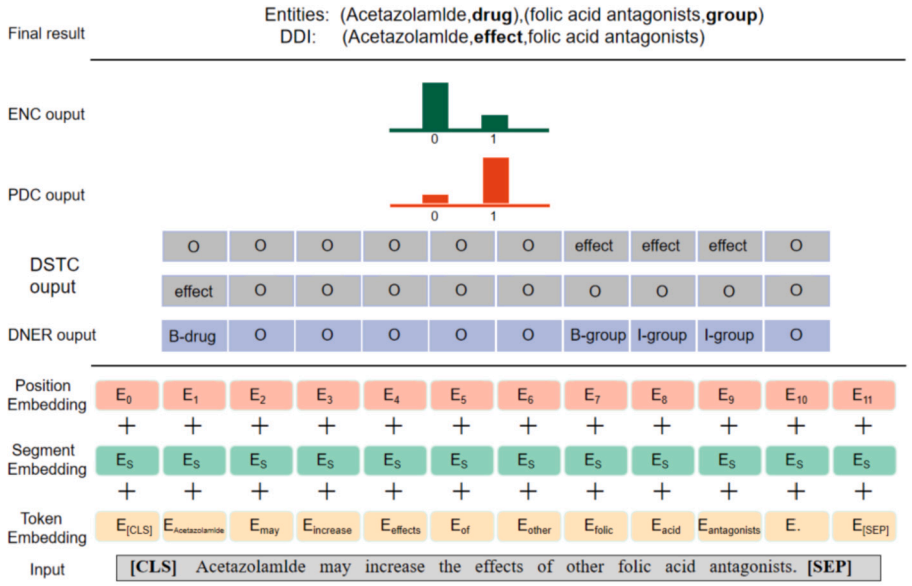


Fig. 2. Description of the input and output of MTMG model. Like previous studies using the BERT model, we use [CLS] and [SEP] to represent the beginning and end of the input text, respectively, and there are three primary input embeddings. The output shows the prediction labels of four tasks, and we parse the output of DNER and DSTC tasks to get the final result.

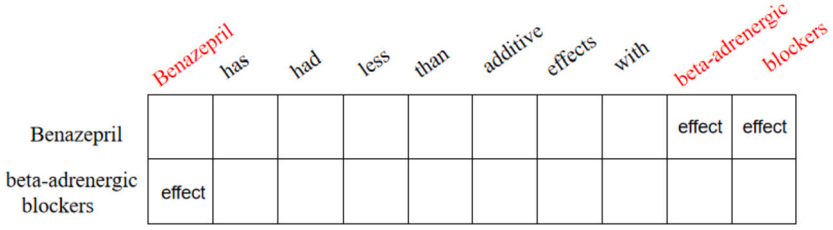


Fig. 3. An example of Drug-Specified Token Classification. Transverse is the token set of a sentence, and column is the set of drug entities. In this example, the token **blocker** is assigned the label *effect* combined with the specified drug **Benazepril**, indicating that the **blocker** belongs to a drug entity that has drug interaction *effect* with **Benazepril**.

the purpose of this task is to predict the relationship between other tokens in the sentence and e_i. We did not use the BIO scheme to delineate entity boundaries but directly used the five DDI types and 'O' as labels to reduce the training difficulty of DSTC. For example, when a token is labeled as *effect*, it means that the token belongs to an entity and the DDI between that entity and the specified drug is *effect*, and when it is predicted as 'O', it shows that the token does not belong to any entity. We transform DDIs classification task into this token-level main task with the aim of better mining the correlation between DDIs classification and DNER by keeping them at the same granularity.

ENC: Entity Number Classification is a sentence-level binary classification task. In our dataset, all sentences contain at least two entities. Given X, the goal of this task is to determine whether the number of entities in X is greater than 2. We design the ENC task with the hope that the model will know the approximate number of drug entities to alleviate the problem of under- or over-recognition entities.

PDC: Positive DDI Classification is a sentence-level binary classification task. In the dataset, none of the drug pairs in the partial sentences have a positive drug-drug interaction. Given X, this task aims to predict whether X contains a positive example of DDI. We introduce this task because the model can give DSTC global guiding information if it knows whether the context describes the positive DDIs.

3.3. Shared input representation layer

3.3.1. Encoding sentences with BioBERT

BERT [6] has received much attention in the NLP field since it was used as an encoder to generate language representations. This is a pre-trained context-sensitive word representation model that uses Transformers [37] trained on the English Wikipedia and BooksCorpus to obtain a contextual representation of each word. The general pre-training model fares poorly in biomedical tasks because of the huge number of domain-specific nouns found in the biomedical corpus that are absent from the general corpus and complicated terms made up of sub-words.

For the above reason, we use BioBERT [19] as feature extractor and hard shared its parameters. This is a pre-trained model on a large-scale biomedical corpus, which has experimentally achieved the best results on most NLP tasks in the biomedical domain.

Given an input sentence X from the DDIs corpus, where x_i represents the $i-th$ token. We limit the length of X to n_t and then use BERT tokenizer to words in X into tokens. We employ the average vector of the final three layers of BioBERT as the embedding representation of each x_i to increase the output vector's comprehensiveness $V = \left\{ v_1, \ldots, v_i, \ldots, v_{n_t} \right\} \in \mathbb{R}^{n_t \times d}$, where d is the dimension of hidden states.

3.3.2. Bidirectional LSTM layer

RNNs can collect both temporal information and semantic information in the sequence, which achieves good applications in various NLP studies in recent years [38,39]. In this work, we employ BiLSTM [40] to encode the output of the BioBERT. We can model the relationship between the previous and subsequent words in this type of LSTM, and the complete sentence information is available at each time step, thereby enhancing the contextual representation of sentences in the DDI corpus. The representation of the i-th token encoded by BiLSTM h_i can be gained by Eq. (1).

$$\begin{aligned} \overrightarrow{h_i} &= \overrightarrow{LSTM}(v_i, \overrightarrow{h_{i-1}}), \\ \overleftarrow{h_i} &= \overleftarrow{LSTM}(v_i, \overleftarrow{h_{i+1}}), \\ h_i &= [\overrightarrow{h_i}; \overleftarrow{h_i}], \quad i = 0, \ldots, n_t. \end{aligned} \tag{1}$$

Then we get the BiLSTM encoded sentence sequence $H = \left\{ h_1, \ldots, h_i, \ldots, h_{n_t} \right\} \in \mathbb{R}^{n_t \times d_t}$, where d_t is the dimension of the output vector, which is twice as large as the LSTM cell.

3.4. Task-specific layers

In the task-specific layer, we use the output of the shared layer as the input for both main tasks and auxiliary tasks. Each task has its own independent parameters, including a project layer and a classifier, which are described in detail as follows.

3.4.1. The main tasks

As described in subsection 3.2, DNER and DSTC are two sequence labeling tasks. For DNER, similar to previous works [8,38], using BIO encoding scheme aims to identify the entity position and its type by assigning a tag for every token. We use a common sequence annotation algorithm linear-chain Conditional Random Fields (CRFs) for DNER. CRFs [41] are a type of discriminative undirected probabilistic graphical model. It is applied to encode correlations between observations and build consistent interpretations. The linear-chain CRF, which base each prediction solely on its close neighbors, are popular in natural language processing. The linear-chain CRFs score is defined as Eq. (2).

$$S(X, Y) = \sum_{i=1}^{n_t} s_{i, y_i} + \sum_{i=0}^{n_t} A_{y_i, y_{i+1}}, \tag{2}$$

where $Y = \left\{ y_1, \ldots, y_i, \ldots, y_n \right\}$ is a sequence of predictions. $S \in \mathbb{R}^{n_t \times m}$, m is the numbers of tags, s_{i, y_i} is the score of the predicted tag for token x_i. And $A \in \mathbb{R}^{m \times m}$ is a square transition matrix where $A_{i,j}$ represents the transition score from the tag i to tag j. Then, the probability of the sequence y is obtained using a softmax over all possible sequences of tags as Eq. (3).

$$P(Y|X) = \frac{e^{S(X,Y)}}{\sum_{\tilde{Y} \in Y_X} e^{S(X, \tilde{Y})}}, \tag{3}$$

where Y_X is the all possible tag sequences for sentence X. $\tilde{Y}$ is the tag sequence with the highest score obtained by applying the Viterbi algorithm [41].

For DSTC, our goal is to predict the relationship between a given drug entity and other tokens in a sentence, so the most important thing is how to jointly represent the embedding vector of a specified drug and other tokens. Firstly, for a specified drug entity e consisting of C tokens, we average it to obtain its embedding vector $h_e = \frac{1}{C} \sum_c^C h_c$. Meanwhile, we are inspired by Liu [42] and use Conditional Layer Normalization (CLN) to get the representation of other tokens conditional on the vector of the specified drug h_e. For the i-th token in the sentence, its representation conditional on h_e can be updated by Eq. (4).

$$h_i^e = \gamma_i^e \odot \left(\frac{h_i - \mu}{\sigma} \right) + \lambda_i^e, \tag{4}$$

where $\gamma_i^e = W_\alpha h_e + b_\alpha$ and $\lambda_i^e = W_\beta h_e + b_\beta$ are the parameters and biases of layer normalization generated with h_e being the condition, respectively. μ and σ are the mean and standard deviation across the elements of h_i, as shown in Eq. (5).

$$\mu = \frac{1}{d_t} \sum_{k=1}^{d_t} h_{ik}, \sigma = \sqrt{\frac{1}{d_t} \sum_{k=1}^{d_t} \left(h_{ik} - \mu \right)^2}, \tag{5}$$

where h_{ik} means the k-th dimension of h_i, h_e and h_i^e are then fed to a biaffine classifier as shown in Eq. (6).

Algorithm 1: Training our model.

Input: DDI2013 Dataset
Initialize: Max epochs, batch size, learning rate, etc. Model parameter θ of shared layers and task-specific layer randomly.
for *each epoch* **in** Max epochs **do**
  for *each mini-batch* **in** DDI2013 **do**
    $L(\theta)_{ENC}$ = Eq.3 for Entity Number Classification
    $L(\theta)_{PDC}$ = Eq.3 for Positive DDI Classification
    $L(\theta)_{DNER}$ = Eq.9 for Drug named entity recognition
    $L(\theta)_{DSTC}$ = Eq.9 for Drug-Specified Token Classification
    Compute the final loss: $L(\theta) = w_1 L(\theta)_{ENC} + w_2 L(\theta)_{PDC}$
    $+w_3 L(\theta)_{DNER} + w_4 L(\theta)_{DSTC}$
    Compute gradient ∇_θ
    Update model $\theta = \theta - \epsilon \nabla_\theta$
  end
end

$$s_{i,e} = h_e^{\top} \mathrm{U} h_i^e + W \left[h_e; h_i^e\right] + b, \tag{6}$$

where $s_{i,e}$ is the predicted scores of the pre-defined relation classes. U, W and b are all trainable parameters. We also use softmax to get the final classification probability $y_{i,e} = softmax\left(s_{i,e}\right)$.

For DSTC and DNER, it is known from Table 2 that there is a serious problem of class imbalance, about which we apply the loss function proposed by Menon et al. [43] instead of the softmax cross-entropy to optimize this problem, which utilizes the class-based prior information. The loss function of the main task is shown in Eq. (7).

$$L_{main} = -\frac{1}{M} \sum_{i=1}^{M} \log(1 + \sum_{y' \neq y} e^{f_{y'}(x_i) - f_y(x_i) - \log \pi_y}), \tag{7}$$

where M is the number of training samples, x_i is the i-th sample, y is the predicted label, y' is the ground truth label corresponds to x_i. And $f_y(x)$ denotes the output of the last layer of the neural network, π_y is the class prior estimate of the label y. Compared with the standard cross entropy loss, this function adds a class prior-based offset to the output of the neural network, which can fit the mutual information between classes in the training process.

3.4.2. The auxiliary tasks

ENC and PDC are two sentence-level classification tasks that aim to optimize the representation of tokens in the shared layer by learning sentence-level labels. Related study [44] have proved that using pooling operation on the output of BERT is better than directly using [CLS] token as the sentence embedding vector for classification [45]. Therefore, we applied a mean pooling over H to generate a fixed-size sentence representation vector h_s, where $h_s \in \mathbb{R}^{d_t}$. Finally, we can obtain the probability of label 0 or 1 through a linear layer and softmax as shown in Eq. (8).

$$P(tag|X) = softmax(W_a h_s + b_a), \tag{8}$$

where $P(tag|X) \in \mathbb{R}^m$, m is the number of tags, which is set to 2 for binary classification task, $W_a \in \mathbb{R}^{d_t \times m}$ and $b_a \in \mathbb{R}^m$ are trainable weight matrix and bias. The loss function of the auxiliary task is written as Eq. (9).

$$L_{auxiliary} = -\sum_{m} q_{tag} \, log(P(tag \mid X)), \tag{9}$$

where q_{tag} is Dirac delta which equals to 1 if predicted class tag is the right ground-truth label, and 0 otherwise.

The joint training process of MTMG is provided by Algorithm 1, where w_1, w_2, w_3, w_4 are weight hyper-parameters that need to be set in advance.

3.5. The extraction rules

After getting the sequence labeling results of the main task, the type and boundary of the drug entity can be obtained directly from the DNER results because the BIO encoding format contains all this information. We extract DDI triplets (s, r, o) from DNER and DSTC results based on the following rules:

(1) Based on the DNER result, we can get the position of all the drug entities in the sentence. In the DSTC results, each specified drug as the subject s, other drug entities as the object o. We only query what kind of drug-drug interaction label is assigned to the token of object entity, and the result is used as the relation r, then the triplet (s, r, o) is extracted.
(2) Each specified drug searches only backward for tokens that have interrelationships with it based on its position in the sentence to avoid extracting duplicate triplets.
(3) If an object contains 1 or 2 tokens, the relation label r is determined strictly according to the prediction result of the head token in DSTC.

Table 3
Performance comparison with the other existing methods.

Method		DNER			DDI			
		P	R	F1	P	R	F1	Overall F1
Golden entities	BiLSTM	-	-	-	0.684	0.665	0.674	-
	HRNN [2]	-	-	-	0.741	0.718	0.729	-
	RHCNN [11]	-	-	-	0.773	0.738	0.755	-
	BioBERT [11]-BiLSTM	-	-	-	0.799	0.785	0.792	-
Pipelined methods	BiLSTM-CRF+BiLSTM [9]	0.932	0.861	0.895	0.648	0.630	0.639	0.767
	BiLSTM-CRF+HRNN [9]	0.932	0.861	0.895	0.692	0.707	0.692	0.794
	BioBERT-BiLSTM	0.909	0.917	0.913	0.746	0.722	0.734	0.824
Joint methods	Graph Tagging [47]	0.817	0.813	0.815	0.587	0.571	0.573	0.694
	Att-BiLSTM+ELMo [9]	0.905	0.939	0.922	0.750	0.752	0.751	0.836
	MTMG	0.919	0.931	**0.925**	0.772	0.782	**0.777**	**0.851**

(4) If an object contains 3 or more tokens, the prediction label class of the head token is added 1.5 points, and the prediction label class of other tokens is added 1 point. Finally, the relation label r is determined as the label class with the highest score.

4. Results and discussion

4.1. Experimental setup

We conducted experiments on the dataset presented in Section 3.1 to evaluate the performance of MTMG. We use the predefined train and test splits to complete the training and experiment of the overall model, including 6976 sentences in 714 abstracts from DrugBank and MedLine. Micro-averaged Precision (P), Recall (R) and F1-score (F1) are used to evaluate the results. And we sum and average the F1-scores of DNER and DDIs extraction to get the overall F1-score to evaluate the overall task performance.

In our experiment, for any method using BioBERT, the pre-trained weights come from BioBERT-base v1.1[1] with 12 stacked encoder layers and a hidden layer size of 768. The dataset are trained with batch size of 16, a dropout with the probability of 0.1 after the Shared input representation layer, and sentences are padded to the maximum sentence length of 256. We use AdamW [46] as an adaptive optimizer with a learning rate $1e^{-}3$ and decay of $1e^{-}2$ per epoch and the model was trained for 30 epochs. Finally, the experiments were conducted on a computer with a single NVIDIA RTX 3090 24 GB graphics card and a 16-core Intel CPU and we implemented our MTMG model with the Pytorch library. Our code has been released at https://github.com/HHDeng/MTMG.

4.2. Performance comparison

To further demonstrate the effectiveness of MTMG, we provide an evaluation of the full pipeline including DNER and DDIs classifications and compare it to several different state-of-the-art approaches. None of the methods compared in this paper use additional drug-related knowledge to improve the performance of DDIs extraction. When compared with the baseline model, the improvement by the MTMG is statistically significant ($P < 0.005$, McNemar test).

First, we compare some recent studies on DDIs extraction based on golden entities. (1) BiLSTM: a bidirectional LSTM model using only word embedding and position embedding as our baseline model. (2) HRNN [2]: a hierarchical RNN-based approach to combine sentence sequence and the shortest dependency paths for DDIs extraction task. (3) RHCNN [11]: a recurrent hybrid CNN for DDIs extraction first obtains the complete semantic embedding through a recurrent structure and then learns sentence-level information through a hybrid convolutional neural network and uses focal loss to mitigate class imbalance problem. (4) BioBERT [19]+BiLSTM: a model architecture like our shared layer, feeding the output of BioBERT into a bidirectional LSTM network to complete the DDIs extraction.

Second, for the pipelined methods, we complete DNER to obtain the entity set and then classify the interaction between drug entity pairs. Meanwhile, we directly use some experimental data from Luo et al. [9]. (1) BiLSTM-CRF+BiLSTM: The sequence labeling task of entities is completed using BiLSTM-CRF, and then DDIs extraction is completed for the mentioned drug entities based on a BiLSTM. (2) BiLSTM-CRF+HRNN: The DNER is completed by a BiLSTM-CRF model, and then a HRNN [2] model completes the DDIs extraction. (3) BioBERT-BiLSTM: DNER is completed using BioBERT-BiLSTM architecture, and then DDIs are extracted using another BioBERT-BiLSTM model.

For the joint extraction method, MTMG is compared with the most advanced joint extraction method in the DDIs extraction task, and we also apply some approaches that have good results in the general field to the DDI dataset. (1) Graph Tagging [47]: a transition-based approach which can model interdependence between relations as well as between entities and relations by designing a graph scheme. (2) Att-BiLSTM+ELMo [9]: the previous state-of-the-art joint DDIs extraction method which proposed a novel tagging scheme for DDI and greatly improved the performance by combining ELMo pre-trained embedding.

According to the experimental results in Table 3, MTMG achieves the best individual performance in DNER and DDIs extraction and the best overall performance. We believe that BioBERT pre-trained weights are a significant contributing factor. Among the

[1] https://github.com/naver/biobert-pretrained.

Table 4
Comparison of F1 score on different DDI types.

Methods	Advice	Effect	Mechanism	Int
HRNN [2]	80.30%	71.80%	74.00%	54.30%
RHCNN [11]	80.54%	73.49%	78.25%	58.90%
BioBERT-BiLSTM	84.82%	79.34%	82.37%	53.49%
MTMG	83.56%	78.39%	82.15%	46.15%

methods using golden entities, the structure of BioBERT-BiLSTM can achieve the best results compared to other methods due to the fact that BioBERT is pre-trained on a large-scale biomedical corpus, and the Transformer-based mechanism can capture long-distance dependencies, which is highly adaptable to the DDIs extraction task.

Of course, in the pipelined approach, all methods achieved poorer DDIs extraction results compared to the same model architecture but using golden entities. For instance, if the recognition results of drug entities are first obtained using a BiLSTM-CRF model, then the DDI extraction is done using BiLSTM or HRNN, and the DDI extraction F1 scores are reduced by 3.5% and 3.7%, respectively, compared with the direct use of golden entities. Similarly, the BioBERT+BiLSTM method obtained a 5.8% decrease. It indicates that in the pipelined approach, DNER identification errors greatly affect the results of DDIs extraction, i.e., the error propagation problem. In particular, the BioBERT + BiLSTM architectures with higher performance can still achieve the best results in the pipelined method, but it seems more sensitive to the influence of incorrect drug entities. The possible reason is that more correct predictions become incorrect due to wrong entity boundaries. And MTMG uses the same BioBERT+BiLSTM model, outperforming the pipelined approach in terms of overall F1 score by 2.7%, which indicates that MTMG can exploit the dependencies between entities and relations to mitigate the error propagation problem further.

Compared with the joint extraction methods, we found that the methods with a strong performance in the traditional field such as Graph tagging did not perform well in both DNER and DDIs extraction tasks. That may be because biomedical texts have more special nouns and complex nouns composed of multiple sub-words, and the sentence structure is more complex (like more overlapping relationships). And compared with the state-of-the-art joint extraction method on the DDI problem, we achieve a 2.6% improvement in F1 score on the DDIs extraction task with a close performance of DNER, indicating that MTMG can dig out the mutual benefits between two tasks and thus improving the performance of both tasks.

4.3. Performance on DDIs extraction

Correctly extracting the hidden drug pair interactions from the vast biomedical literature is the ultimate goal of our work, which can significantly facilitate the relevant medical practitioners and provide more convenient support for the construction of drug-related databases.

Table 4 displays the effectiveness of MTMG on different DDI types and compares it with several DDI extraction methods using gold entities mentioned above. We can see that the performance is basically positively correlated with the amount of data, but the type *effect* is an exception. It has the largest number of samples, but its performance is not the best. This may be because many sentences describing the type *effect* have no obvious characteristics, making it difficult for the model to make predictions and often incorrectly predicting into other categories. In addition, the performance of MTMG in type *int* is particularly poor compared with other models. A possible reason is that we design the DDIs extraction as a sequence labeling task, which leads to more sparse label distribution and more serious class imbalance. This will be the focus of our subsequent work for improvement.

The confusion matrix of the MTMG's DDIs extraction results is shown in Fig. 4. We regularize the prediction results of each DDI type to emphasize the classification results, and the greater the proportion, the deeper the color. It can be seen from the Fig. 4 that there are two main problems: (1) A large number of positive DDIs are wrongly predicted as *False*. (2) Worse performance on type *int*, and instances of *int* are frequently mistakenly classed as type *effect*. Regarding the first problem, the main reason is still the class imbalance problem. The number of negative samples is several times the total number of positive samples, leading to some samples of positive DDIs inevitably misclassifying as negative instances. Some studies have also obtained similar results with the second problem [2]. We found that the model does not fit the type *int* well except because the number of samples is minimal. Another important reason is that the instances of type *int* and the instances of type *effect* have high semantic similarity, resulting in a considerable number of *int* instances being wrongly predicted as *effect* and also reducing the type *effect*'s precision. For example, in the sentence *"conversely, diethylpropion may interfere with antihypertensive drugs (i.e., guanethidine, a-methyldopa)"* (type *int*) and *"Cyclopentolate may interfere with the anti-glaucoma action of carbachol or pilocarpine."* (type *effect*), the main relations between drugs are all "may interfere". While in the sentence *"Trilostane may interact with aminoglutethimide or mitotane (causing too great a decrease in adrenal function)."* (type *int*) and *"Ethopropazine may interact with alcohol or other CNS depressants, causing increased sedative effects."* (type *effect*), these two sentences have a similar semantic structure and "may interact" is a common keyword. Consequently, the two DDI types' semantic similarity causes poor performance on the type *int* and is one of the reasons why the type *effect*, which has the largest number of instances in the positive DDIs, does not perform the best.

4.4. Effect of auxiliary tasks on performance

To evaluate the impact of the auxiliary tasks we designed on the main tasks, we conduct ablation studies and showed in Table 5.

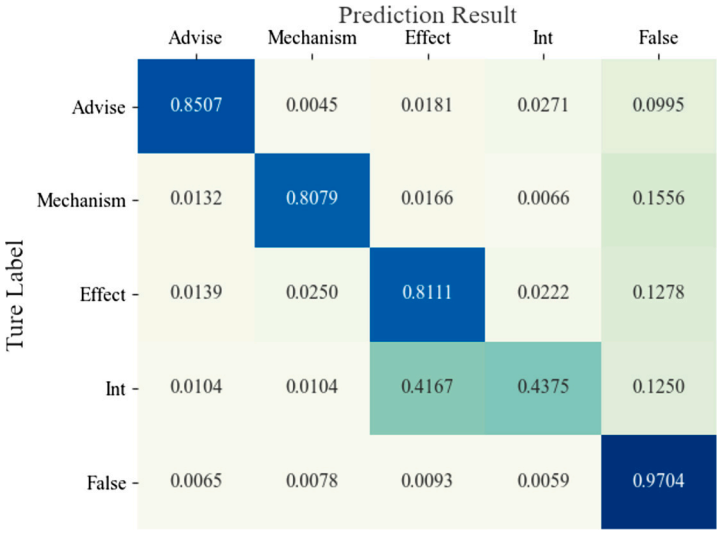


Fig. 4. The confusion matrix of the DDIs extraction results of MTMG.

Table 5
Ablation study results about auxiliary tasks.

Model	DNER			DDI		
	P	R	F1	P	R	F1
Joint	91.89%	93.13%	92.50%	77.22%	78.24%	77.72%
w/o ENC	90.97%	93.09%	92.02%	77.19%	78.14%	77.67%
w/o PDC	91.79%	93.32%	92.55%	76.80%	77.43%	77.11%
w/o ENC&PDC	91.25%	92.53%	91.88%	76.54%	77.32%	76.93%

The ablation experiments results showed that when the ENC task is not involved in the joint training, the results of the DNER task decreased by 0.48%, and the DDIs extraction was also affected, which once again indicates the existence of error propagation problems. When the PDC task does not participate in joint training, DNER has the best results due to the involvement of the EDC task. However, the performance of DDIs extraction decreases significantly, which indicates that the sentence-level DDI information provided by the PDC is helpful for DDIs extraction. In addition, when both auxiliary tasks are not involved in joint training, the F1 scores of the two main tasks decrease by 0.67% and 0.79%, respectively, which indicates that the robust sentence-level information provided by our designed auxiliary tasks for the main tasks during training can indeed help the main tasks to improve their performance and make good use of the prior knowledge of the dataset itself.

4.5. Analysis of main task design

In this article, we designed DDIs classification as a sequence labeling task called Drug-Specified Token Classification to fully explore the mutual benefit between DNER and DDIs classification. In this subsection, we conduct comparison experiment to show how effective this approach is. Without any auxiliary tasks, two sets of main tasks are designed and jointly trained using the same dataset partitioning and the same model architecture. And the result of the experiment is evaluated with F1 score. The two sets are: (1) training DNER and DSTC jointly; (2) training DNER and traditional drug-drug interaction classification task jointly. The training curves of different main task combinations are shown in Fig. 5.

In Fig. 5, the blue curve is the training curve of set (1), and the yellow curve is the training curve of set (2). Node • represents the result of DNER, and node + represents the result of DDIs extraction.

As seen from Fig. 5, the F1 scores of DNER or DDIs extraction in set (2) are lower than those in set (1) in the overall training process. Specifically, the training curve of DNER in set (2) is significantly lower than in set (1) at the beginning of training. Two curves gradually converge to a similar curve after 10 epochs, but the performance in set (2) is still slightly lower. And the performance of DDIs extraction in set (2) is significantly lower than that in set (1) in the whole training process. It demonstrates that the Drug-Specified Token Classification task can effectively exploit the correlation between DNER and DDIs classification and alleviate the error propagation problem.

4.6. Analysis of weights in loss function

In Algorithm 1, w_1 and w_2 are the loss weights for the main tasks, while w_3 and w_4 are for the auxiliary tasks. In the implementation of MTMG, w_1 and w_2 are set to 1. To investigate how the different loss weights of the auxiliary tasks affect the main tasks,

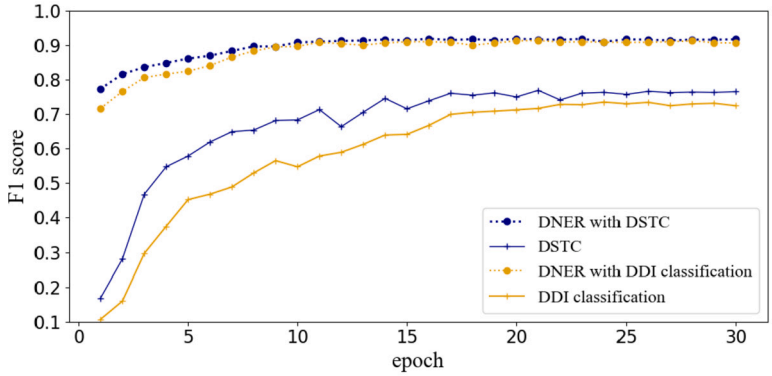


Fig. 5. Training curves of different main task combinations.

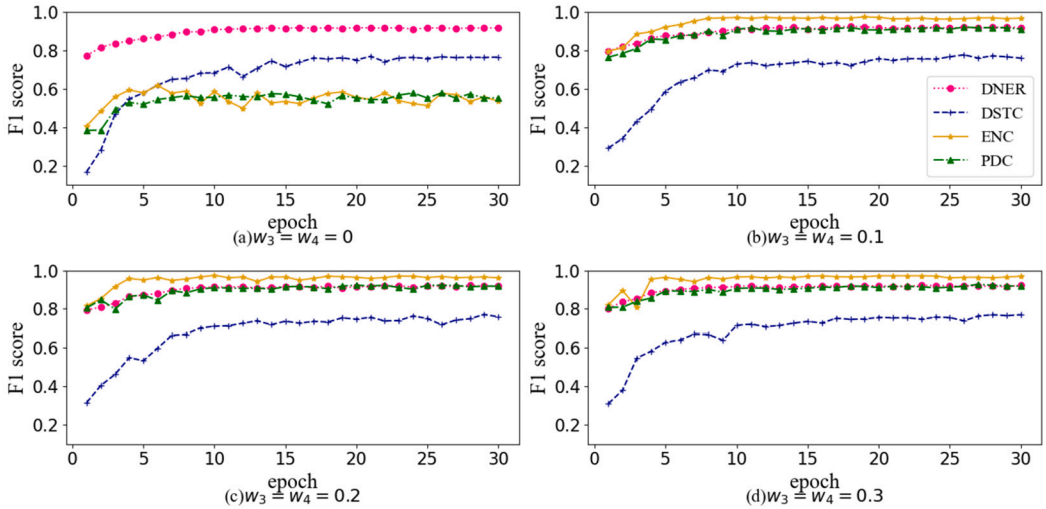


Fig. 6. Training curves of auxiliary tasks' different loss weights.

we set w_3 and w_4 with different values and conduct comparison experiments in this subsection. Fig. 6 shows the training curves in different cases and use the F1 score as a comprehensive evaluation metric.

In Fig. 6, w_3 and w_4 are set to 0, 0.1, 0.2, and 0.3, respectively. From Fig. 6 (b) we find that when w_3 and w_4 are set to 0.1, the training curves of all four tasks are the smoothest and the convergence speed is the best, which indicates that the auxiliary task can provide the most effective training guidance to the main tasks when w_3 and w_4 are set to 0.1. Fig. 6 (c) and (d) show that the performance of MTMG is greatly reduced when w_3 and w_4 are too large. The cause may be that the effects of other terms on the objective function are lessened by auxiliary tasks with a large weight, and the learned features are unrelated to the main tasks.

In addition, when w_3 and w_4 are set to 0, the losses of the auxiliary tasks do not participate in backpropagation, and the training of the auxiliary tasks relies only on information learned from the main tasks. Fig. 6 (a) shows that the F1 scores of the two main tasks are significantly lower in the early training period, which again indicates that the auxiliary task can help accelerate the convergence of the main tasks. At the same time, only relying on the loss backpropagation of the main tasks, the F1 scores of the two auxiliary tasks can still maintain a level of more than 0.5 after a certain epoch, which indicates that the main tasks can provide some useful information to the auxiliary tasks.

4.7. Case study

We compared the prediction results of the pipelined method of BioBERT+BiLSTM with those of MTMG to analyze the benefits and drawbacks of MTMG, and chose three cases examples as shown in the Table 6.

For sentence 1, which describes the relationship between three drug entities in a parallel structure and another drug entity, the pipelined approach tends to screen out reasonable drug pairs by external tools and then classify the relationships individually without considering the dependencies of parallel entities. In contrast, MTMG correctly extracts all drug triplets, indicating that MTMG can fully consider the dependencies between all entities and implicitly learn the influence of syntactic structure on relationships, which has better universality for sentences with overlapping relationships.

Table 6
A case study on the DDI2013 dataset shows examples of extraction results for different methods. Each example has four rows: the sentence, the gold standard answer, the extraction result of the BioBERT+BiLSTM pipelined approach, and the extraction result of MTMG. The red text indicates the wrong result.

Sentence 1	Cytochrome P450 inducers, such as **rifampin**, **carbamazepine**, and **phenytoin**, induce metabolism and caused a markedly decreased C max and AUC of oral **midazolam** in adult studies.
Golden truth	DNER:{(rifampin, *drug*), (carbamazepine, *drug*), (phenytoin, *drug*), (midazolam, *drug*)} DDI:{(rifampin, *mechanism*, midazolam), (carbamazepine, *mechanism*, midazolam), (phenytoin, *mechanism*, midazolam), (rifampin, *false*, carbamazepine), (rifampin, *false*, phenytoin), (carbamazepine, *false*, phenytoin)}
Pipeline	DNER:{(rifampin, *drug*), (carbamazepine, *drug*), (phenytoin, *drug*), (midazolam, *drug*)} DDI:{(rifampin, *mechanism*, midazolam), (carbamazepine, *mechanism*, midazolam), (phenytoin, *mechanism*, midazolam)}
MG	DNER:{(rifampin, *drug*), (carbamazepine, *drug*), (phenytoin, *drug*), (midazolam, *drug*)} DDI:{(rifampin, *mechanism*, midazolam), (carbamazepine, *mechanism*, midazolam), (phenytoin, *mechanism*, midazolam), (rifampin, *false*, carbamazepine), (rifampin, *false*, phenytoin), (carbamazepine, *false*, phenytoin)}
Sentence 2	Careful monitoring of prothrombin time in patients receiving **DIFLUCAN** and **coumarin-type anticoagulants** is recommended.
Golden truth	DNER:{(DIFLUCAN, *brand*), (coumarin-type anticoagulants, *group*)} DDI:{(DIFLUCAN, *advise*, coumarin-type anticoagulants)}
Pipeline	DNER:{(DIFLUCAN, *brand*), (anticoagulants, *group*)} DDI:{(DIFLUCAN, *advise*, anticoagulants)}
MTMG	DNER:{(DIFLUCAN, *brand*), (coumarin-type anticoagulants, *group*)} DDI:{(DIFLUCAN, *advise*, coumarin-type anticoagulants)}
Sentence 3	This is typical of the interaction of **meperidine** and **MAOIs**.
Golden truth	DNER:{(meperidine, *drug*), (MAOIs, *group*)}; DDI:{(meperidine, *int*, MAOIS)}
Pipeline	DNER:{(meperidine, *drug*), (MAOIs, *drug*)}; DDI:{(meperidine, *false*, MAOIS)}
MTMG	DNER:{(meperidine, *drug*), (MAOIs, *group*)}; DDI:{(meperidine, *effect*, MAOIS)}

For sentence 2, the pipelined model incorrectly predicts the boundary of a drug entity, which directly leads to the propagation of the error to the DDI classification. Even if the DDI class is correctly predicted, the wrong DDI triplet is extracted. In contrast, MTMG obtains the correct entity recognition results and the DDI triplet, indicating that MTMG is more sensitive to the boundary of drug entities and can alleviate the error propagation problem to some extent.

Sentence 3 describes two drug entities with *int* relationship. The pipelined method misclassifies both entity class and DDI class, and MTMG obtains the correct entity but also mispredicts DDI class, which proves that our approach also has insufficient prediction ability to deal with small sample categories like *int*. How to optimize the class imbalance problem is still an important direction for improvement. In addition, although both MTMG and the pipelined method extract wrong DDI class label, the pipelined method predicts a negative label *false*, and we get a positive label *effect*. We believe the PDC task provides the model with positive DDI information, confirming that supervised objectives at different granularities can be combined to help get more accurate predictions.

5. Conclusions

In this paper, we propose a multi-task approach that exploits different granularity information for the whole DDI extraction process, including DNER and DDI extraction. We design the regular sentence-level DDI classification task as a sequence labeling task like DNER and demonstrate that this can better exploit the correlation between entities and relations to improve the performance. Two auxiliary tasks related to the main task are also designed to provide a priori coarse-grained information about the dataset itself to guide more robust and generalized training. We experimentally explore the relevance of the auxiliary tasks to the main task. The effectiveness of MTMG is demonstrated by the experimental findings on the DDI2013 datasets.

In our work, the class imbalance problem remains significant despite introducing a better loss function, such as poor identification of DDI class *int*. In future work, we will consider techniques related to data enhancement to optimize this problem. In addition, using more knowledge to improve the overall DDI extraction performance is also one of our following studies.

CRediT authorship contribution statement

Haohan Deng: Conceived and designed the experiments; Performed the experiments; Wrote the paper.
Qiaoqin Li: Analyzed and interpreted the data; Contributed reagents, materials, analysis tools or data.
Yongguo Liu, **Jiajing Zhu**: Analyzed and interpreted the data; Wrote the paper.

Declaration of competing interest

The authors declare no conflict of interest.

Data availability

Data will be made available on request.

Acknowledgements

This research was supported in part by the National Natural Science Foundation of China (NSFC) under grant 62202084, the Sichuan Science and Technology Program under grants 2023YFS0325, 2022YFS0059 and 2023YFQ0010, the Natural Science Foundation of Sichuan under grant 2022NSFSC0883, the Fundamental Research Funds for the Central Universities under grant ZYGX2021J020, and the China Postdoctoral Science Foundation under grant 2021M690028.

References

[1] V. Miranda, A. Fede, M. Nobuo, V. Ayres, A. Giglio, M. Miranda, R.P. Riechelmann, Adverse drug reactions and drug interactions as causes of hospital admission in oncology, J. Pain Symptom Manag. 42 (3) (2011) 342–353.
[2] Y. Zhang, W. Zheng, H. Lin, J. Wang, Z. Yang, M. Dumontier, Drug–drug interaction extraction via hierarchical RNNs on sequence and shortest dependency paths, Bioinformatics 34 (5) (2017) 828–835.
[3] The ddi corpus: an annotated corpus with pharmacological substances and drug–drug interactions, J. Biomed. Inform. 46 (5) (2013) 914–920.
[4] S. Liu, B. Tang, Q. Chen, X. Wang, Drug-drug interaction extraction via convolutional neural networks, Comput. Math. Methods Med. 2016 (2016) 1–8.
[5] M. Asada, M. Miwa, Y. Sasaki, Enhancing drug-drug interaction extraction from texts by molecular structure information, 2018, pp. 680–685.
[6] J. Devlin, M. Chang, K. Lee, K. Toutanova, BERT: pre-training of deep bidirectional transformers for language understanding, in: NAACL-HLT (1), Association for Computational Linguistics, 2019, pp. 4171–4186.
[7] S. Zheng, Y. Hao, D. Lu, H. Bao, J. Xu, H. Hao, B. Xu, Joint entity and relation extraction based on a hybrid neural network, Neurocomputing 257 (2017) 59–66.
[8] M. Miwa, M. Bansal, End-to-end relation extraction using lstms on sequences and tree structures, in: ACL (1), The Association for Computer Linguistics, 2016.
[9] L. Luo, Z. Yang, M. Cao, L. Wang, Y. Zhang, H. Lin, A neural network-based joint learning approach for biomedical entity and relation extraction from biomedical literature, J. Biomed. Inform. 103 (2020) 103384.
[10] Z. Zhao, Z. Yang, L. Luo, H. Lin, J. Wang, Drug-drug interaction extraction from biomedical literature using syntax convolutional neural network, Bioinformatics 32 (22) (2016) 3444–3453.
[11] X. Sun, K. Dong, L. Ma, R. Sutcliffe, F. He, S. Chen, J. Feng, Drug-drug interaction extraction via recurrent hybrid convolutional neural networks with an improved focal loss, Entropy 21 (1) (2019).
[12] W. Wang, X. Yang, C. Yang, X.-W. Guo, X. Zhang, C. Wu, Dependency-based long short term memory network for drug-drug interaction extraction, BMC Bioinform. 18 (2017).
[13] Z. Jiang, G. Liang, Q. Jiang, Drug-drug interaction extraction from literature using a skeleton long short term memory neural network, in: IEEE International Conference on Bioinformatics and Biomedicine, 2017.
[14] V. Mostafapour, O. Dikenelli, Attention-wrapped hierarchical blstms for ddi extraction, arXiv:1907.13561, 2019.
[15] M. Fatehifar, H. Karshenas, Drug-drug interaction extraction using a position and similarity fusion-based attention mechanism, J. Biomed. Inform. 115 (2021) 103707.
[16] Y. Zhang, H. Lin, Z. Yang, J. Wang, S. Zhang, Y. Sun, L. Yang, A hybrid model based on neural networks for biomedical relation extraction, J. Biomed. Inform. 81 (2018) 83–92.
[17] H. Wu, Y. Xing, W. Ge, X. Liu, J. Zou, C. Zhou, J. Liao, Drug-drug interaction extraction via hybrid neural networks on biomedical literature, J. Biomed. Inform. 106 (2020) 103432.
[18] Y. Zhu, L. Li, H. Lu, A. Zhou, X. Qin, Extracting drug-drug interactions from texts with biobert and multiple entity-aware attentions, J. Biomed. Inform. 106 (2020) 103451.
[19] J. Lee, W. Yoon, S. Kim, D. Kim, S. Kim, C.H. So, J. Kang, BioBERT: a pre-trained biomedical language representation model for biomedical text mining, Bioinformatics 36 (4) (2019) 1234–1240.
[20] L. Huang, J. Lin, X. Li, L. Song, Z. Zheng, K. Wong, EGFI: drug-drug interaction extraction and generation with fusion of enriched entity and sentence information, Briefings Bioinform. 23 (1) (2021).
[21] Y. Papanikolaou, A. Pierleoni, DARE: data augmented relation extraction with GPT-2, CoRR, arXiv:2004.13845 [abs].
[22] Y. Gu, R. Tinn, H. Cheng, M. Lucas, N. Usuyama, X. Liu, T. Naumann, J. Gao, H. Poon, Domain-specific language model pretraining for biomedical natural language processing, ACM Trans. Comput. Healthc. 3 (1) (2022) 2:1–2:23.
[23] X. Jin, X. Sun, J. Chen, R.F.E. Sutcliffe, Extracting drug-drug interactions from biomedical texts using knowledge graph embeddings and multi-focal loss, in: M.A. Hasan, L. Xiong (Eds.), Proceedings of the 31st ACM International Conference on Information & Knowledge Management, Atlanta, GA, USA, October 17-21, 2022, ACM, 2022, pp. 884–893.
[24] T.N. Kipf, M. Welling, Semi-supervised classification with graph convolutional networks, arXiv:1609.02907, 2017.
[25] M. Asada, M. Miwa, Y. Sasaki, Using drug descriptions and molecular structures for drug–drug interaction extraction from literature, Bioinformatics 37 (12) (2020) 1739–1746.
[26] Y. Feng, S. Zhang, J. Shi, DPDDI: a deep predictor for drug-drug interactions, BMC Bioinform. 21 (1) (2020) 419.
[27] Y. Wang, Y. Min, X. Chen, J. Wu, Multi-view graph contrastive representation learning for drug-drug interaction prediction, in: WWW, ACM / IW3C2, 2021, pp. 2921–2933.
[28] H. He, G. Chen, C.Y. Chen, 3DGT-DDI: 3D graph and text based neural network for drug-drug interaction prediction, Briefings Bioinform. 23 (3) (2022).
[29] J. Kolář, T. Ambrus, V. Špringer, Drug nomenclature in view of pharmacopoeial names, Chemicke Listy 104 (1) (2010) 27–32.
[30] S. Liu, B. Tang, Q. Chen, X. Wang, Drug name recognition: approaches and resources, Information 6 (4) (2015) 790–810.
[31] L. Merchant, R. Lutter, S. Chang, Identical or similar brand names used in different countries for medications with different active ingredients: a descriptive analysis, BMJ Quality Safety 29 (12) (2020), bmjqs–2019–010316.
[32] X. Yue, Z. Wang, J. Huang, S. Parthasarathy, S. Moosavinasab, Y. Huang, S.M. Lin, W. Zhang, P. Zhang, H. Sun, Graph embedding on biomedical networks: methods, applications and evaluations, Bioinformatics 36 (4) (2020) 1241–1251.
[33] S. Zhao, T. Liu, S. Zhao, F. Wang, A neural multi-task learning framework to jointly model medical named entity recognition and normalization, in: AAAI, AAAI Press, 2019, pp. 817–824.
[34] V. Suárez-Paniagua, R.M.R. Zavala, I. Segura-Bedmar, P. Martínez, A two-stage deep learning approach for extracting entities and relationships from medical texts, J. Biomed. Informatics 99 (2019).
[35] D. Zaikis, I.P. Vlahavas, TP-DDI: transformer-based pipeline for the extraction of drug-drug interactions, Artif. Intell. Med. 119 (2021) 102153.

[36] S. Zheng, F. Wang, H. Bao, Y. Hao, P. Zhou, B. Xu, Joint extraction of entities and relations based on a novel tagging scheme, in: ACL (1), Association for Computational Linguistics, 2017, pp. 1227–1236.
[37] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, L. Kaiser, I. Polosukhin, Attention is all you need, in: NIPS, 2017, pp. 5998–6008.
[38] G. Lample, M. Ballesteros, S. Subramanian, K. Kawakami, C. Dyer, Neural architectures for named entity recognition, in: HLT-NAACL, The Association for Computational Linguistics, 2016, pp. 260–270.
[39] S.K. Sahu, A. Anand, Drug-drug interaction extraction from biomedical texts using long short-term memory network, J. Biomed. Inform. 86 (2018) 15–24.
[40] M. Schuster, K.K. Paliwal, Bidirectional recurrent neural networks, IEEE Trans. Signal Process. 45 (11) (1997) 2673–2681.
[41] J.D. Lafferty, A. McCallum, F.C.N. Pereira, Conditional random fields: probabilistic models for segmenting and labeling sequence data, in: ICML, Morgan Kaufmann, 2001, pp. 282–289.
[42] R. Liu, J. Wei, C. Jia, S. Vosoughi, Modulating language models with emotions, in: Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, Association for Computational Linguistics, 2021.
[43] A.K. Menon, S. Jayasumana, A.S. Rawat, H. Jain, A. Veit, S. Kumar, Long-Tail Learning via Logit Adjustment, 2020.
[44] N. Reimers, I. Gurevych, Sentence-bert: sentence embeddings using Siamese bert-networks, in: EMNLP/IJCNLP (1), Association for Computational Linguistics, 2019, pp. 3980–3990.
[45] C. Sun, X. Qiu, Y. Xu, X. Huang, How to fine-tune BERT for text classification?, in: CCL, in: Lecture Notes in Computer Science, vol. 11856, Springer, 2019, pp. 194–206.
[46] I. Loshchilov, F. Hutter, Decoupled weight decay regularization, in: 7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019, OpenReview.net, 2019.
[47] S. Wang, Y. Zhang, W. Che, T. Liu, Joint extraction of entities and relations based on a novel graph scheme, in: Proceedings of the 27th International Joint Conference on Artificial Intelligence, IJCAI'18, AAAI Press, 2018, pp. 4461–4467.