applied sciences

MDPI

Article

A Message Passing Approach to Biomedical Relation Classification for Drug–Drug Interactions

Dimitrios Zaikis *,†, Christina Karalka † and Ioannis Vlahavas *

School of Informatics, Aristotle University of Thessaloniki, 541 24 Thessaloniki, Greece
* Correspondence: dimitriz@csd.auth.gr (D.Z.); vlahavas@csd.auth.gr (I.V.)
† These authors contributed equally to this work.

Featured Application: With this contribution, we aim to aid the drug development process as well as the identification of possible adverse drug events due to simultaneous drug use.

Abstract: The task of extracting drug entities and possible interactions between drug pairings is known as Drug–Drug Interaction (DDI) extraction. Computer-assisted DDI extraction with Machine Learning techniques can help streamline this expensive and time-consuming process during the drug development cycle. Over the years, a variety of both traditional and Neural Network-based techniques for the extraction of DDIs have been proposed. Despite the introduction of several successful strategies, obtaining high classification accuracy is still an area where further progress can be made. In this work, we present a novel Knowledge Graph (KG) based approach that utilizes a unique graph structure in combination with a Transformer-based Language Model and Graph Neural Networks to classify DDIs from biomedical literature. The KG is constructed to model the knowledge of the DDI Extraction 2013 benchmark dataset, without the inclusion of additional external information sources. Each drug pair is classified based on the context of the sentence it was found in, by utilizing transfer knowledge in the form of semantic representations from domain-adapted BioBERT weights that serve as the initial KG states. The proposed approach was evaluated on the DDI classification task of the same dataset and achieved a F1-score of 79.14% on the four positive classes, outperforming the current state-of-the-art approach.

Keywords: Drug–Drug Interactions; transformers; graph neural networks; language models; relation classification; domain-adaption

Citation: Zaikis, D.; Karalka, C.; Vlahavas, I. A Message Passing Approach to Biomedical Relation Classification for Drug–Drug Interactions. *Appl. Sci.* **2022**, *12*, 10987. https://doi.org/10.3390/app122110987

Academic Editors: Pavlos S. Efraimidis, Avi Arampatzis and George Drosatos

Received: 30 September 2022
Accepted: 27 October 2022
Published: 30 October 2022

1. Introduction

Drug–Drug Interactions (DDI) refer to the pharmacological action between drugs that can occur during polypharmacy, and the co-administration of more than one drug can potentially lead to harmful adverse drug reactions that have a significant impact on public health. Most DDIs are discovered during the various drug development stages or during Phase IV clinical trials conducted on already publicly available drugs [1]. The dissemination of these findings are reported at an exponential rate, rendering the task of manually finding the most relevant information very difficult and time-consuming [2]. However, the heterogeneity of the available data regarding DDIs presents new challenges in their exploration, analysis and manageability. The identification and retrieval of documented drug interactions requires gathering and analyzing data from multiple data sources, especially in the early stages of drug development.

Moreover, as the practice of medicine and scientific research increasingly produces and depends on data, addressing these issues becomes a necessity. Therefore, the automatic extraction of DDIs from biomedical literature is important in order to accelerate this time-consuming and strenuous process. Vast amounts of relevant knowledge can be extracted from various types of information sources such as scientific literature, electronic health records, online databases and many more [3]. However, these sources contain textual

information that is very diverse in terms of type, format, level of detail and differ in terms of expressiveness and semantics. Additionally, the possibility of conflicting or outdated information presents among the various sources adds to the overall complexity regarding the collection, storage and analysis of the data and consequently the extraction and exploitation of the hidden wealth of information.

DDI extraction from textual corpora is a traditional Relationship Extraction (RE) task in Machine Learning (ML) that aims to classify the interaction between drug entities [4] into specific predefined categories. Related DDI extraction studies vary based on the underlying task they aim to tackle and could be divided into pattern-based, traditional machine learning-based and deep learning-based [5]. The DDI classification task focuses on classifying the interactions between drug pairs by using gold entities with Relationship Classification (RC) techniques and are evaluated on the DDI Extraction 2013 corpus, which is considered as the benchmark dataset [6]. Similar to all underlying extraction tasks, the Deep Learning-based (DL) methods achieve the best performance and advance the state-of-the-art research in this field. Early DL-based approaches mainly utilized Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN) as their base architectures to learn better task-specific representations with the use of contextualized information incorporated into their Neural Network-based architectures.

Liu et al. [7] introduced the first CNN-based approach for the DDI task, focusing on local sentence information without defining additional features generated by Natural Language Processing (NLP) toolkits. They applied a convolutional layer that takes the input from a look-up table constructed from word and position embeddings, leveraging the neural networks ability to automatically learn features. A max pooling layer then extracts the most important feature from each feature vector before finally classifying the interactions into one of the five classes using a softmax layer. The reported results show that the position embeddings improve the classification performance but face challenges due to the different position distribution on the test set.

Similarly, Quan et al. [8] integrated multiple word embeddings in their proposed MCCNN model, to tackle the vocabulary gap, the integration of semantic information and the manual feature selection in the DDI extraction task. The proposed approach implemented a multi-channel CNN model and fused multiple versions of word embeddings that were trained on biomedical domain corpora. However, the systems performance depends greatly on the CNN's window size, leading to errors in long sentences where the relevant drug mentions are either very close or very far from each other. In an attempt to capture long distance dependencies, Liu et al. [9] utilized syntactic features in the form of dependency parsing trees and word syntax-based embeddings in their proposed DCNN approach. Due to the small number of correctly parsed long sentences, a threshold was implemented where sentences with a length smaller than the threshold were classified by the DCNN, while the rest by a CNN. Similarly, Zhao et al. [10] utilized dependency features in combination with Part-of-Speech (PoS) and position embeddings with an auto-encoder to transfer sparse bag-of-words feature vectors to dense real value feature vectors. The proposed SCNN approach additionally implemented a rule-based negative instance filtering, leading to limited generalization ability.

To alleviate the limitations of CNN-based approaches, various DDI extraction studies employed RNN-based networks that capture long sequences using an internal memory mechanism, such as Long Short Term Memory (LSTM) and Gated Recurrent Units (GRU) networks. Accordingly, Wang et al. [11] presented a three channel bidirectional LSTM (BiLSTM) architecture to capture distance and dependency-based features with their DL-STM model. To account for the imbalanced class distribution of the DDI corpus, negative instance filtering and training set sampling were employed. However, the reported results indicate that the lengths of the instances continue to adversely affect the classification performance of the model. Yi et al. [12] introduced 2ATT-RNN, a GRU architecture that leverages multiple attention layers. A word-level attention layer extracts sentence representations in combination with a sentence-level attention layer that combines other sentences containing the same drug mentions. However, the inclusion of the negative class in the

overall performance metric does not allow for a clear depiction of the effectiveness of the proposed method. Zhang et al. [13] divided the input sentence sequences into three parts according to the position of two drug entities, and applied a hierarchical BiLSTMs to integrate sentence sequences, shortest dependencies paths and attention mechanisms to classify DDIs. The experimental results show improvements over the previous approaches, but continue to underperform in cases where the two drug entities are mentioned over a long distance with each other.

Similarly, Zhou et al. [14] utilized the attention mechanism in a BiLSTM-based architecture. To improve the efficiency of the attention mechanism, the proposed PM-BLSTM system utilizes an additional position embedding to generate the attention weights. The model takes advantage of multi-task learning by predicting whether or not two drugs interact with each other, further distinguishing the types of interactions jointly. The reported results show that the position-wise attention improves the performance but continues to misclassify instances that contain multiple drug mentions. Salman et al. [15] proposed a straightforward LSTM and attention-based architecture and expanded the DDI extraction task to include sentiment-based severity prediction. The sentence-level polarity is extracted using an NLP toolkit and finally classified as either low, moderate or high level of severity for instances that contain at least on DDI. However, the Word2Vec [16] generated word embeddings are context-independent and do not account for the word positions in the sentences. Furthermore, Word2Vec learns word level embeddings, resulting in the same embedding for any learned word, independently of the surrounding context. Therefore, this type of embedding cannot generate representations for words encountered outside the initial vocabulary space, which is a major disadvantage in the DDI corpus.

Recently, Transformer-based Language Models (LM) such as ELMo [17], GPT-2 [18] and BERT [19] achieved state-of-the-art results in general domain NLP. By leveraging the capabilities of the transformers, transfer learning and the self-supervised training approach, biomedical and scientific-domain LMs, such as BioBERT [20] and SciBERT [21], were introduced in the DDI extraction task as well. Mondal [22] incorporated BioBERT as a pre-trained LM and chemical structure representations of drugs, in the form of SMILES, to extract DDIs from text. The proposed approach focused on the encoding and incorporation of the chemical structure information from external sources using a Variational AutoEncoder in an attempt to leverage both entities and sentence-level information. However, the low dimensionality of the final representations used for the Transformer initialization could potentially lead to information loss in longer sentences.

The integration and utilization of knowledge through semantic representations of data aims to mitigate the aforementioned problems [23]. Specifically, in recent years, biomedical knowledge base information represented as Knowledge Graphs (KG) tends to be preferred more and more often. KGs are powerful knowledge representation models which focus on the semantic meaning instead of only on the information structures, modeling the relationships between the graph entities [24]. As a result, KGs provide a homogenized view of data regardless of their origin, allowing for human-interpretable encoding of domain-specific information with the use of node and relation types.

Consequently, Graph Neural Networks (GNN) that take advantage of graph-based structures, in combination with Transformer-based LMs, have seen great success in various general-domain NLP tasks and have been introduced in the DDI extraction task as well. Xiong et al. [25] introduced GCNN-DDI, which utilized dependency graphs in a BiLSTM and GCN architecture to classify the interactions. Shi et al. [26], similar to GCNN-DDI, adopted a GNN and introduced a PageRank based multi-hop relevant words selection strategy for the dependency graph. These approaches rely on the construction of dependency trees (or syntax trees) from the sentences where nodes represent individual words and edges the syntactic dependency paths between words in the sentence's dependency tree. The feature vectors of the nodes are initialized by a pre-trained domain-specific LM, utilizing the POS tag of each word and a BiLSTM to update the initial word embeddings for contextual feature extraction. Both approaches utilize GNNs to improve the representations through the incorporation of dependency relations with the word embeddings. However,

while GCNN-DDI uses the raw dependency graph, DREAM additionally enhances it with long-range potential words discovered by PageRank by extending some potential relevant multi-hop neighbors, which have high information transferability.

GNN-based approaches exclusively implement dependency graphs which are complex graph structures where the number of nodes equals the number of tokens in each sentence making the application of GNNs slow and computationally expensive. Additionally, since the benchmark corpus is considered relatively small and imbalanced, proposed approaches try to overcome this limitation by incorporating complicated feature engineering or extending the available information from external sources.

In contrast to the previously reported methods, in this paper, we present a novel KG schema for the DDI classification task that is leveraged by our GNN-based architecture that includes message and non-message passing units. We constructed the KG according to the principles of the Resource Description Framework (RDF) data model where each relation is annotated by an *subject-predicate-object* triplet. The proposed graph structure is built upon the DDI corpus to model the sentence and drug mention relations and is further semantically enhanced by domain-adapting a BERT-based LM pre-trained on large-scale domain-specific corpora that generates the initial state of the graph nodes. Finally, the interactions are classified by utilizing a sub-graph, taking the context of the sentence into consideration.

We evaluated our proposed approach for the classification of DDIs according to the SemEval 2013 shared task [4] on the DDI Extraction 2013 dataset. Experimental results indicate that our KG and GNN-based classification model achieves a state-of-the-art F1-score of 79.14% on the four positive classes, outperforming other methodologies. Additionally, we show that the KG information in combination with negative instance filtering can enhance the performance of our model. Table A1 shows a comparative analysis of the related studies presented in this work and our proposed approach.

The remainder of this paper is organized as follows: in Section 2, we elaborate on the dataset used and describe our proposed approach in detail. In Section 3, we present the experimental setup and results and elaborate on the effectiveness and limitations of our proposed model. Finally, in Section 4, we present our conclusions and directions for future research.

2. Materials and Methods

In this section, we introduce the dataset and the architecture of our proposed graph neural network-based classification model for drug–drug interaction extraction, where, given a sentence containing drug mentions, each drug pair is classified into one of the five possible interaction categories. It consists of three main parts, which are the DDI Knowledge Graph, the BERT-based language model and the GNN-based classification module as shown in Figure 1. Specifically, a knowledge graph based on our proposed DDI task related schema is created where a domain-adapted BERT-based language model is then applied to generate meaningful word representations. This knowledge is then integrated into a selected part of the graph, where a GNN is trained to classify the drug pair relationship.

2.1. Dataset

The DDI–Extraction 2013 corpus [6] is a collection of biomedical texts containing sentences from the DrugBank database and MedLine abstracts. The DrugBank database focuses on providing information on medicinal substances, while MedLine is a more general database of scientific publications from health-related sectors. The corpus has been manually annotated by two expert annotators and is considered the benchmark dataset for the text-based DDI extraction task which includes the recognition of drug named entities and the interaction classification of the drug pairs.

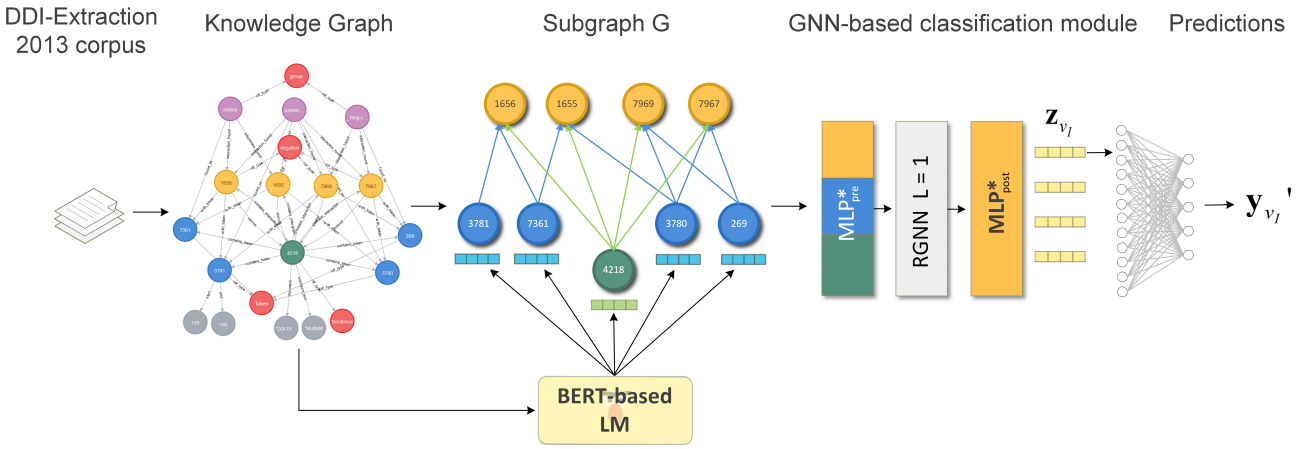


Figure 1. An overview of our proposed architecture for the classification of DDIs.

The dataset contains separate XML files where each one constitutes a document, which is further separated into its individual sentences. For each sentence, the drugs (entity) it mentions are listed and, for each possible pair of them, an interaction pair (relation pair) is defined. Therefore, n drugs define $n(n-1)/2$ pairs of interactions. Highlighted elements are characterized by unique identifiers (id) that reveal their position in the XML tree hierarchy. The corpus is split into a single training set and two separate test sets for both drug recognition and interaction classification tasks. Table 1 provides a summary of the corpus's main features and statistics for the predefined train and test datasets splits.

Drug entities and interactions are classified into categories (types) based on the context of the sentence they are mentioned in. In the majority of them, the named entities concern drugs intended for human use and are classified as "drug", "brand" and "group" types, while other substances are classified as "drug_n". Similarly, the interaction types between two drugs when administered simultaneously are categorized as follows:

- Effect: These are changes in the effect of a substance on the body, such as the appearance of symptoms and clinical findings.; The results of such effects are also referred to as pharmacodynamic properties of drugs.
- Mechanism: Refers to modifications in the absorption, distribution, metabolism and excretion of drugs, characteristics that constitute their pharmacokinetic properties. In other words, it concerns how the concentration of a substance in the body is affected by the presence of the other substances;
- Advice: Refers to descriptions containing recommendations or advice regarding the simultaneous use of two drugs;
- Int: Assigned in the case where the existence of an association between two drugs is mentioned, without any additional information indicating its type;
- Negative: It refers to the absence of interaction between two substances.

2.2. DDI Knowledge Graph

In order to model the DDI-specific Knowledge Graph, we used the RDF standard to create the proposed schema for representing the corpus knowledge. Figure 2 provides an overview of the DDI Knowledge Graph.

According to the RDF principles, the base of a knowledge graph is composed of a set of <Subject, Predicate, Object> statements, with the Subject and Object resources being respectively the initial and terminal nodes of a directed edge. The Predicate resource is considered the label of the edge in question, which is a property that associates the individual resources or serves to assign a value (Object) to some attribute of the Subject. This basic model is extended by defining classes into which the objects of the world belong.

Table 1. The DDI-Extraction 2013 corpus statistics.

		Training Set	Test Set	
			DNER	RC
	Documents	714	112	191
	Sentences	6976	665	1299
Drug Entities	Drug	9425	351	1864
	Group	3399	155	667
	Brand	1437	59	369
	Drug_n	504	120	140
DDIs	Mechanism	1322	-	303
	Effect	1700	-	363
	Advice	827	-	222
	Int	188	-	96
	Negative	23,771	-	4737

In the DDI corpus, all drug entities are annotated by providing the exact drug name and the location in the context of the specific sentence they are found in. Initially, each sentence of a document becomes an instance of the *Sentence* class, with its text preserved intact in the graph as an attribute of the specific node. Additionally, a sentence refers to a set of drugs that are modeled by the *Token* class, which is associated with the *Sentence* through the *contains_token property*. This property has a minimum cardinality of 2, filtering sentences that mention less than two drug entities (i.e., do not contain at least one drug pair). Furthermore, the set of unique drug entities in the collection is described by the *Drug_Class* class and its subclasses are the four types of drugs, which are mutually exclusive.

Finally, the concept of interaction (relationship between two drugs) is also modeled through classes. Typically an RDF statement represents a binary relationship between two resources in the form of a triplet. However, it may be necessary to add additional information regarding the statement resulting in an n–ary relationship. Each possible drug pair of a particular sentence is represented by an *Interaction* helper node. Thus, the information defining an interaction is composed centered on this node, through properties that associate it with other entities (e.g., 1 Sentence instance, 2 Drug_Class, 2 Token). Similar to the *Drug_Class*, its subclasses are based on the five predefined interaction types.

Collectively, a drug entity (Drug_Class), referred to as (found_as) Token in a particular sentence (Sentence), participates (interaction_found) in some pairs of interactions (Interaction). The sentence contains (contains_interaction) the interacting pair, while the Token reference participates (in_interaction) in it.

2.3. Modeling the DDI Relation Classification Task

In order to model the drug–drug interaction classification task and utilize DDI Knowledge Graph representations, a subgraph $G = (V, E, X, K, R)$ of the complete graph is selected. V and K denote the sets of nodes and the classes they belong to, with class instances denoted as k defining a subset of V_k. X constitutes an accompanying matrix of node characteristics (node feature vector), dimension $|V| \times d_{BERT}$, where d_{BERT} denotes the dimension of the BERT-based LM vector representation. Finally, R refers to the types of edges (properties) that associate the nodes, while E is the set of edges of the graph expressed in the form of a coordinate list (coordinate list format—COO).

Each interaction node is associated with a pair of specific drug mentions occurring in a sentence. Figure 3 shows the schema of the subgraph resulting from $K = \{Interaction(I),\ Token(T),\ Sentence(S)\}$ and $R = \{in\_interaction,\ contains\_interaction\}$. Therefore, the target is to classify the Interaction nodes V_I, or otherwise to determine the object of each triplet $< v_I, rdf{:}type, c_{v_I} >$, where c_{v_I} is a subclass of Interaction, from the three elements (i.e., the two tokens and the sentence) that determine its type.

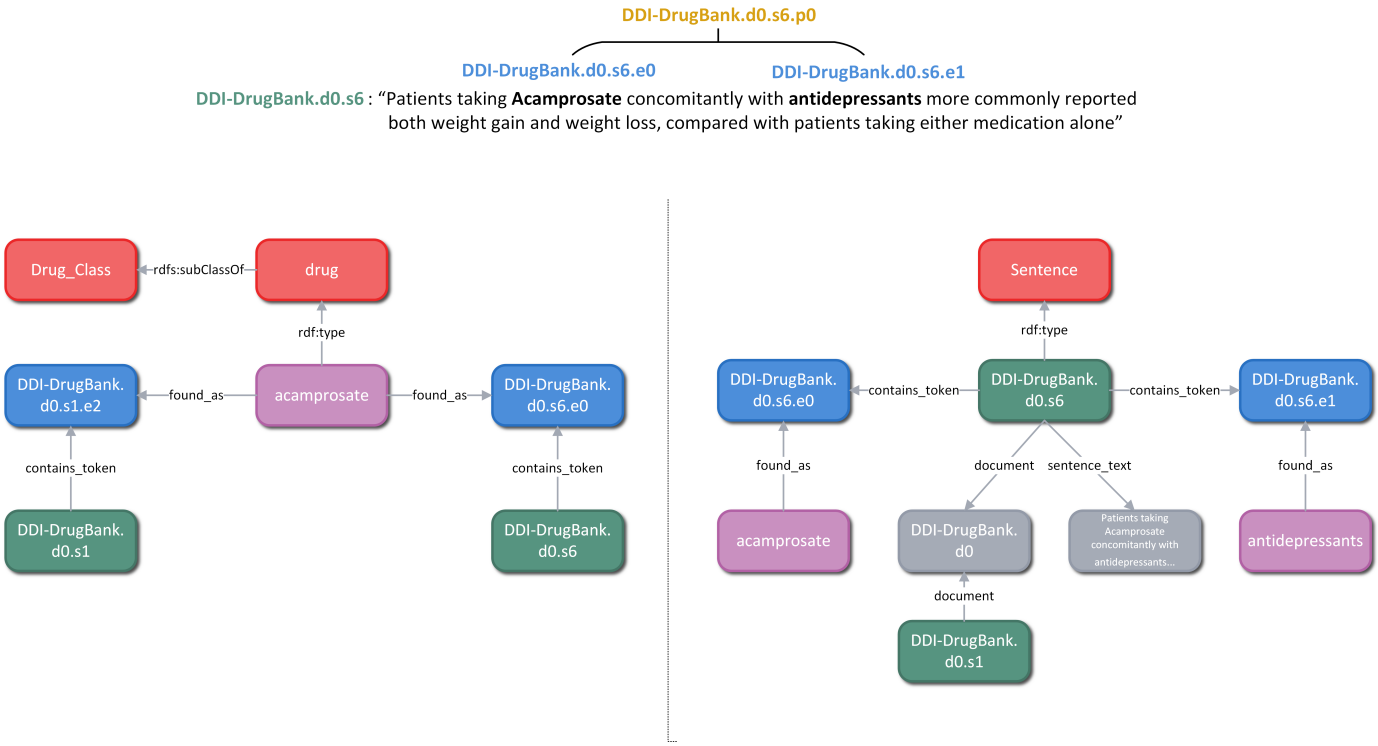


Figure 2. The DDI Knowledge Graph overview based on an example sentence. (**Left**) drug sub-graph; (**Right**) sentence sub-graph.

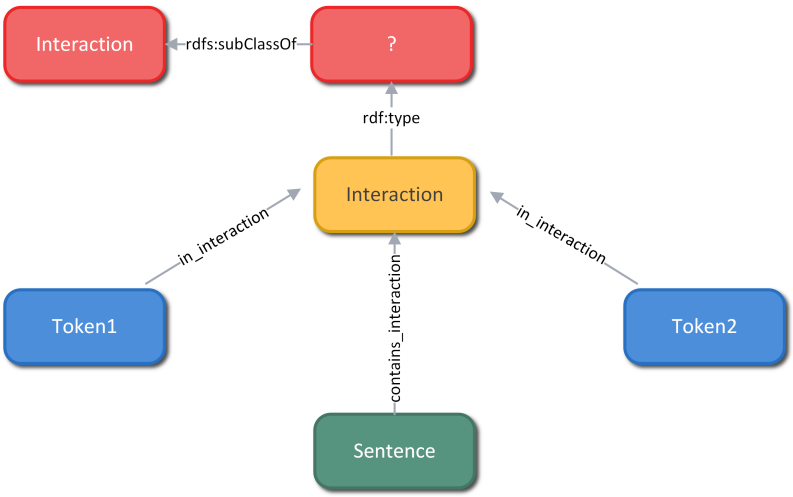


Figure 3. The schema of subgraph G which models the classification of an Interaction node.

2.4. Negative Instance Filtering

The extraction of DDIs from biomedical text is a multi-class classification problem, with *Advice*, *Effect*, *Int* and *Mechanism* being the positive classes and *Negative* being the negative class. The dataset statistics in Table 1 show the highly imbalanced nature of the corpus, in terms of both the positive and negative class distribution and within the four positive classes as well. In particular, the instances of the negative class exceed the positive classes with a ratio of 1:5.9 in the training set. It can be observed that only a part of the pairs labeled as negative explicitly express the knowledge that there is no interaction between the substances (drugs). Conversely, in the vast majority, the negative instances follow the same pattern where a number of drug–pair interactions were labeled in the same sentence, without clarifying the relationship between them. Consequently, the following set of rules was defined to detect them, as such cases can be dismissed.

- Consistent with the assumption that a substance does not interact with itself, pairs involving the same or synonymous drugs are rejected;
- There are sentences in the form of "$drug_1$ [:-]... $drug_1$ ... $drug_2$..." which provide conflicting knowledge about the type of interaction between two drug entities. Therefore, any pair involving the first occurrence of $drug_1$ will be removed;
- Regular expressions identify patterns, such as quoting a list of drugs or referring to a sub-case, broader category or abbreviation. Additionally, consecutive or overlapping matches are merged to combine regular expressions. Finally, any pairs found within a match are discarded.

The above rule set leads to the rejection of 44.5% and 0.94% of the negative and positive examples in the training set, respectively, with the ratio changing to 1:3.3. Finally, as the corresponding Interaction nodes are not taken into account, they are excluded in the results during the evaluation of our proposed system.

2.5. Language Model-Based Transfer Learning

BERT [19] is an extensively pre-trained, Transformer-based language model, capable of state-of-the-art performance in NLP tasks. The increased ability to understand the conceptual framework that characterizes it is due to the self-attention mechanism, through which the token-level importance is assigned based on the associations between the individual words (tokens) of the sentence. Additionally, due to the training as a masked language model, it is able to perceive the information of the text in a bidirectional manner, allowing the LM to produce vector representations that reflect the syntactic, semantic and grammatical relationships between words.

BERT's architecture is composed of a number of N_{enc} consecutive encoder units with d_{BERT} hidden vector dimension, where $N_{enc} = 12$, $d_{BERT} = 768$ for the base version and $N_{enc} = 24$, $d_{BERT} = 1024$ for the large version. Furthermore, multiple BERT-based variations pre-trained on specific domains have been developed which achieve better performance in domain-specific tasks compared to BERT. As an example, BioBERT [20] and SciBERT [21] are two popular variations of BERT, pre-trained on large text corpora from the biomedical and scientific field, respectively.

2.5.1. Embedding Generation

In the DDI classification task, knowledge about the interaction type of a drug pair is expressed through text. The sentence text t is an associated element of the Sentence node, or otherwise contained in $< v_S, sentence\_text, t >$ triplets. By applying a BERT-based LM to t, it becomes possible to reduce the text to a suitable vector representation, in addition to sharing information among the individual nodes of the subgraph G.

The preparation of t involves the addition of the special tokens [CLS] and [SEP], which mark the beginning and end of the sentence, respectively. Then, each drug in the sentence is replaced by *drugi*, where i is a number. Finally, WordPiece tokenization is applied, through which words outside the BERT vocabulary are broken into individual pieces (subwords) that belong to it.

Furthermore, the LM is used to initially generate word embeddings x_{v_S} (sentence embeddings) for the entire sentence and a set of $x_{v_{T_j}}$ (token embeddings) where each one is a representation of a $Token_j$ contained within the sentence. These vectors are assigned to the respective nodes v_S and v_{T_j}, constituting their feature vectors.

2.5.2. Sentence and Token Nodes Feature Generation

Each sentence contains words that reveal or indicate the type of interaction between two drugs. For example, expressions such as "should (not) be administered", "caution should be used" and "is (not) recommended", are associated with suggestions (advice) when taking more than one drug simultaneously. Therefore, although the expressions show some variety, they are characterized by a high semantic similarity and are expected to correspond to nearby points in the embeddings vector space.

When generating the sentence embeddings x_{v_S}, it is important for the LM to focus on the above type of information and therefore any reference to any drug is replaced by "drug0" (i.e., $i = 0$). Because of the repeated occurrences, the string loses its meaning within the sentence, with the BERT-based LM giving an appropriate weight. Finally, the [CLS] token embedding is chosen for x_{v_S}, which is a centralized representation of the sentence and is often used in classification tasks. However, since a sentence most likely contains more than one drug pair, the interaction classification can not be performed solely on the basis of x_{v_S}, consequently requiring the feature generation for each drug mention in the sentence.

Furthermore, the interaction type does not depend on the specific drug names in the potential drug pair, but on their position within the sentence. In line with previous studies [7], their replacement is expected to aid in noise reduction when generating the *Token* embeddings, increasing the classification performance. To this end, each unique drug entity in the sentence is assigned a sequence number i, according to the order of appearance of its first mention within the sentence. Thus, $Token_j$ references that participate in $< Drug\_Class, found\_as, Token\_j >$ triplets with a common subject are replaced by *drugi* with a common i. The final feature vector of each node v_{T_j} is obtained by pooling the embeddings of the subwords that compose it. That is, $x_{v_{T_j}} = pooling\{B_{drug\#\#}, B_{\#\#x}\}$, where the pooling method is the average, and B_w is the BERT output for the input w.

2.6. GNN-Based Classification Module

KGs are a complex and dynamic data structure where the notion of fixed ordering does not apply. However, by implementing a message passing framework, GNNs are able to better utilize the KG's underlying graph structure as well as any initially available data for its nodes, compared to other approaches [27]. Specifically, each layer first applies a transformation function on the node feature vectors. The generated messages from each node's neighbors, as well as its own message, are then aggregated to produce a new embedding that encodes additional semantic information provided by the defined relation types. These embeddings can finally be used to perform predictions for the nodes.

Given a selected subgraph G, the classification takes place on one of three types of nodes, namely the set V_I. As a sentence defines a maximum number of pairs according to the contained drug mentions, applying a BERT-based LM for their embedding generation may not be ideal. Instead, word embeddings were generated at the token level, as well as aggregated for the entire sentence that contains them. However, utilizing a GNN allows for the feature generation for each Interaction node.

The vector representation of these nodes is initialized with a null (zero) vector $h^0_{vI} = x_{v_I} = 0$, indicating no initial characterization for the Interaction nodes. However, GNNs pay special attention to the current h^{l-1}_v representation of a node when generating h^l_v from layer l. Therefore, when applying a GNN layer, its new embedding results exclusively from the topology of the graph around it, i.e., through transformation and aggregation of x_{v_S} and x_{v_T} from the one neighboring Sentence and the two Tokens nodes, respectively.

As the subgraph G is a heterogeneous graph, the use of a Relational GNN (RGNN) is required. The management of the heterogeneity is based on the logic of parameter distribution according to the type r of the edge that connects to the Interaction node (i.e., relation-specific transformations) [28]. Therefore, the embedding results from the following equation:

$$h^1_{vI} = aggr_{r \in R}\{GNN^{l=1;r}(h^0_{vI}, \{h^0_u, u \in N^r(v_I)\})\}, \quad (1)$$

where $aggr$ is an aggregation function, R the set of edge types and $N^r(v_I)$ the set of neighbors of v_I according to the triplets $< u, r, v_I >$.

The modeling capability of GNNs is determined by the expressive power of the message aggregation functions, making the choice of the appropriate GNN architecture critical to the performance of this shallow network. Therefore, with the utilization of a Graph Isomorphism Network (GIN), the architecture's deep layers are encapsulated within

the single layer GNN. Using the sum as the aggregator operation, the above relation is formulated as:

$$h_{vI}^{1} = \sum_{r \in R} MLP^{l=1;r}(h_{vI}^{0} + \sum_{u \in N^{r}(v_I)} h_{u}^{0}) \quad (2)$$

The architecture's main RGNN element is surrounded by MLP modules, which are defined according to the node classes. Specifically, through the integration of the MLP preprocessing layers, the initial representation of each node is obtained by $h_{v_k}^{0} = MLP_{pre}^{k}(x_{v_k})$, $k \in K$. Similarly, the final vector representation of the Interaction nodes is defined by $z_{v_I} = MLP_{post}^{I}(h_{v_I}^{1})$.

Conclusively, the system is now able to classify each node v_I into one of the $|C| = 5$ interaction classes. First, the probability distribution of the classes is calculated as $y'_{v_I} = softmax(z_{v_I} W^T + b)$, where W and b are the trainable weights and biases parameters, respectively. The dimensions of matrix W are $|C| \times d_{GNN}$, with the hyperparameter $d_{GNN} = \dim(z_{v_I})$ constituting the dimension of the vector space defined by the network. The final classification during the inference is obtained by $c_{v_I} = argmax_{c \in C}\{y'_{v_I}\}$.

3. Results and Discussion

3.1. Experimental Setup

Training is performed in a supervised manner on the labels of the Interaction nodes, resulting from the provided training set by merging the MedLine and DrugBank subsets. This amounts to a total of 17,176 training examples, derived from 3395 sentences and 608 documents after the construction of the knowledge graph. Similarly, the evaluation is performed on the RE task test set, where a separate graph with 3057 Interaction nodes and 604 sentences from 159 documents is created. The domain-adaption of the LMs is trained either on the training set sentences (Sentence Level Domain Adaption—SLDA) or training set paragraphs (Document Level Domain Adaption—DLDA) only, in a self-supervised manner using the Masked Language Modeling task.

Our proposed approach requires the definition of the two main elements of the architecture, the underlying BERT-based LM that will generate the word embeddings and the GNN-based classification module. First, different pre-trained BERT variants were compared, such as the base version of the general domain BERT, the scientific domain SciBERT and the biomedical domain BioBERT. Furthermore, BioBERT, the pre-training of which is in alignment with the DDI domain, was tested on both base and large versions. Additionally, since recent studies show that domain-adapting a LM by pre-training it on the downstream task can potentially offer large gains in the task performance [29], we aligned both SciBERT and BioBERT base to the DDI task corpus and compared their performance.

Having the features of the nodes generated by the BERT-based LMs, it is then necessary to define and train the classification unit. GIN [30] was chosen as the GNN framework, with node embeddings dimensions $d_{GNN} = 256$ for $d_{BERT} = 768$ and $d_{GNN} = 512$ for $d_{BERT} = 1024$, with the internal MLP unit consisting of $l_{GIN} = 3$ consecutive layers. Its performance is also compared to the mean GraphSAGE framework [31]. Additionally, the contribution of a single-level MLP_{pre} of size d_{GNN} and two-level MLP_{post} of sizes $d_{GNN}/2$ and $d_{GNN}/4$ with a drop-rate of 0.4 and ReLU activations are evaluated.

Adam was chosen as the optimizer with a learning rate and weight decay equal to 5×10^{-5} and 5×10^{-4}, respectively. Furthermore, the mini-batch training approach is followed, where the Interaction nodes of the training set are divided into 53 batches of size 324, while the number of epochs is equal to 170. Finally, the cross entropy loss function is used in the context of the multi-class classification problem.

The experiments were conducted on a computer with a single RTX 3090 24 GB graphics card an a 24-core Intel CPU and the LM domain-adaption on a computer with two RTX A6000 48 GB graphics cards and were implemented using the Pytorch library and the Python programming language.

3.2. Evaluation Metrics

Similar to the related studies, the performance of the system was evaluated based on the Precision (P), Recall (R) and micro F1-score ($F1_{mirco}$) metrics on the test set with the four positive classification targets C^+. The ratio of correctly classified instances c to all instances that were classified as c or actually belong to c constitutes the Precision and Recall of class c, respectively. The micro F1-score, which is the harmonic mean of P and R, provides an overall picture of the system without focusing on the individual performance of each class. The metrics are defined by the following formulas, where the number of corresponding cases is denoted by the combination of T (true) or F (false) and P (positive) or N (negative):

$$P_{micro} = \frac{\sum_{c \in C} + TP}{\sum_{c \in C} + (TP_c + FP_c)} \qquad R_{micro} = \frac{\sum_{c \in C} + TP}{\sum_{c \in C} + (TP_c + FN_c)} \qquad F1_{micro} = \frac{2 \times P_{micro} \times R_{micro}}{P_{micro} + R_{micro}}$$

3.3. Overall Comparison

In Table 2, we show the results of our method in comparison to baseline and state-of-the-art DDI classification approaches which reported their overall performance metric based on the four positive classes. These approaches are trained on the same training set and evaluated on the same test set provided by the DDI corpus and follow the same experimental setting without the inclusion of external information. These approaches can effectively be divided into two categories: traditional methods that utilize extensive feature engineering and state-of-the-art neural network-based approaches that aim to learn feature representations automatically based on different architectures. We compare our proposed approach to the traditional method "FBK-irst" presented in [32], which used linear features, path-enclosed tree kernels and linguistic features. For the NN-based approaches, we compared our approach to the following methods:

- "SCNN" [10]—CNN-based architecture with manually designed features;
- "MCCNN" [8]—CNN with multichannel word embeddings;
- "ASDP-LSTM" [13]—Hierarchical RNNs with shortest dependency paths;
- "PM-BLSTM" [14]—Bidirectional LSTM with position-aware attention;
- "GCNN-DDI" [25]—Bidirectional LSTM with GNN that utilized entire dependency graphs;
- "DREAM" [26]—Bidirectional LSTM with GNN that utilized PageRank enhanced dependency graphs.

The experimental results show that our Knowledge Graph-based approach that utilized BioBERT LM achieves the best overall performance for the classification of DDIs. The proposed KG schema with the domain-adapted pre-trained weights and the non-message passing MLPs are the main contributing factors, which will be analyzed in the following subsections. In the four positive classes, our approach achieves the best results in the *Advice*, *Effect* and *Mechanism* classes and a similar score in the *Int* class. In the following sections, we additionally analyze and discuss the various components of our method and their contribution to the overall performance.

Table 2. Overall performance comparison of our proposed method. All values are $F1$ scores (%) and '-' denotes the value was not provided in the published paper. $F1_{micro}$ denotes the overall score on the four positive classes. The highest values are shown in bold.

Method	System	Advice	Effect	Int	Mechanism	$F1_{Micro}$
SVM	FBK-irst	69.20	62.80	54.70	67.90	65.10
CNN	SCNN	-	-	-	-	68.60
CNN	MCCNN	78.20	68.20	51.00	72.20	70.21
LSTM	ASDP-LSTM	80.30	71.80	54.30	74.00	72.90
LSTM	PM-BLSTM	81.60	71.28	48.57	74.42	72.99
GNN	GCNN-DDI	83.50	75.80	51.40	79.40	77.00
GNN	DREAM	84.80	76.10	**55.10**	81.60	78.30
Our method	BioBERT-GIN	**86.45**	**78.46**	54.80	**82.27**	**79.14**

3.4. The Importance of the Pre-Trained Language Model Domain

The initially generated graph node features are inextricably linked to the performance of the GNN [33]. Accordingly, we evaluated the effects of the various BERT-based LMs for the task of relationship classification from biomedical text, by comparing the models that are trained with their respective BERT variant word embeddings, as shown in Table 3.

Based on the overall performance metrics, we define BERT (M9.1), SciBERT (M6.1) and BioBERT (M2.1) as the three baseline approaches with BioBERT (M2.1) achieving the best baseline results. This further validates the fact that domain-specific LMs tend to outperform general-domain LMs on the domain-specific task. However, the general-domain BERT achieves significantly better performance in the underrepresented class Int. As a reminder, sentences that contain the interactions of type Int indicate that there is a relationship between two drugs but no additional information about the relation type. In this context, the performance increase of general-domain BERT can be attributed to the use of non-scientific language when describing these types of interactions.

The best performing model is M3.3, which makes use of the SLDA BioBERT base and the GIN framework surrounded by pre- and post-processing MLPs. Furthermore, it is the only one that achieves R_{micro} and $F1_{micro}$ scores greater than 75 in addition to the maximum value of P_{micro} among all the models. Moreover, it achieves the best F1 scores in most of the classes (4 out of 5), unlike other models that usually excel in just one class (M9.1-3).

At the same time, M8.2 that utilized DLDA SciBERT achieves a comparable score in R_{micro} and $F1_{micro}$ to the best performing models and surpasses 70% of the models in P_{micro} but significantly underperforms on the Int class. We observe an improvement over the SciBERT baseline approach (M6.1-3) proving that domain-adapting SciBERT to the DDI domain leads to a performance gain in the relationship classification task. Similarly, the performance improved significantly when domain-adapting BioBERT (M3.1-3) to the same task with SLDA, indicating that the biomedical-domain pre-trained LM model may benefit from adapting to other tasks in the same domain. Conversely, adapting same LM with DLDA (M4.1-3), a significant performance degradation can be observed, suggesting that the LM benefits mostly from the context of individual sentences and not larger paragraphs.

The overall results show the effectiveness of the biomedical-domain pre-trained BioBERT base LM, especially compared to the general-domain BERT. Furthermore, aligning the BioBERT to the DDI corpus did yield significant improvement and led to performance increase. Similarly, domain-adapting SciBERT with DLDA produced improved task performance. Noticeably, BioBERT large (M5.1-3) performs worse than its base counterparts, especially in the Int class, which warrants further investigation as no interpretable patterns could be found. However, recent findings [34] suggest that fine-tuning noise increases with model size and that instance-level accuracy has momentum leading to larger models having higher variance due to the fine-tuning seed.

3.5. Effectiveness of the GIN Message Aggregation Function Layers

A basic hyperparameter of the GIN module is the number of layers of the internal MLP network (l_{GIN}). Figure 4 shows the system's behavior on the test set as a function of the MLP network depth (number of layers). For $l_{GIN} = 1$, GIN shows comparable performance to GraphSAGE (M1.1—Table 3), which is significantly lower than the best performing model, validating the limiting factor of shallow aggregation functions. However, as l_{GIN} increases beyond four layers, the $F1_{micro}$ score displays a sharp decrease as it detects a higher percentage of existing interactions, combined with an increase in false positives, evidenced by the significant difference in the recall–precision curve slopes.

Therefore, the intermediate values $l_{GIN} = 2$ and 3 are compared. The change in $F1_{micro}$ is negligible, while the recall improvement and precision drop is approximately 4% when increasing the layer depth by one. However, at $l_{GIN} = 3$, a better compromise is made between the two metrics, with a difference of only 1.7%, compared to the corresponding 9.6% for $l_{GIN} = 2$. Moreover, considering the risk of not being able to detect an existing

interaction, the behavior of the system with the best recall is considered more appropriate for the current task.

Table 3. Performance comparison of the pre-trained LM. SAGE denotes the models that utilize the GraphSAGE framework. SLDA and DLDA denote the Sentence and Document Level Domain Adaption, respectively, and x denotes the inclusion of the corresponding pre/post-processing MLP unit. The highest values are shown in bold.

Model		MLP		Metrics per Classification Target				Overall Metrics		
M	**BERT-Based LM**	**Pre**	**Post**	**Advice**	**Effect**	**Int**	**Mech.**	P_{micro}	R_{micro}	$F1_{micro}$
1.1	SAGE BioBERT base			70.51	67.04	48.21	58.39	71.59	57.37	63.70
1.2			x	74.78	72.92	51.61	70.72	72.42	69.47	70.92
1.3		x	x	65.59	70.02	50.00	68.69	75.00	60.39	66.90
2.1	BioBERT base			75.00	74.96	52.71	71.56	72.91	71.18	72.04
2.2			x	76.84	76.09	52.71	71.49	73.16	72.58	72.87
2.3		x	x	82.45	73.46	52.80	74.27	73.42	74.87	74.14
3.1	BioBERT SLDA			79.14	76.19	52.71	71.46	72.81	71.28	72.04
3.2			x	84.45	75.46	53.80	76.27	75.81	74.13	74.96
3.3		x	x	**86.45**	**78.46**	54.80	**82.27**	**84.33**	**75.55**	**79.14**
4.1	BioBERT DLDA			72.30	73.14	40.88	67.94	66.67	70.53	68.54
4.2			x	72.53	73.36	40.88	70.02	66.91	71.84	69.29
4.3		x	x	71.94	70.43	39.37	69.22	63.78	72.76	67.98
5.1	BioBERT large			69.23	67.83	15.50	70.66	62.97	66.45	64.66
5.2			x	70.86	69.54	20.97	71.52	65.48	67.63	66.54
5.3		x	x	66.67	67.19	12.17	71.85	62.58	66.45	64.45
6.1	SciBERT			72.48	71.68	51.24	73.00	70.51	70.79	70.65
6.2			x	73.51	72.82	52.31	73.75	70.03	73.16	71.56
6.3		x	x	73.63	69.86	50.38	70.25	66.67	72.11	69.28
7.1	SciBERT SLDA			80.12	69.43	47.06	67.11	67.16	71.32	69.18
7.2			x	77.35	69.18	43.94	68.60	68.45	69.08	68.76
7.3		x	x	79.06	67.15	40.35	67.10	70.55	65.26	67.81
8.1	SciBERT DLDA			78.86	73.24	51.91	70.46	72.52	71.18	71.85
8.2			x	80.89	74.15	47.93	73.28	72.74	74.08	73.40
8.3		x	x	77.97	71.26	46.55	73.94	72.70	70.79	71.73
9.1	BERT base			73.24	68.23	58.06	67.24	69.19	67.37	68.27
9.2			x	72.24	67.62	**60.00**	70.74	68.25	69.87	69.05
9.3		x	x	68.15	68.42	55.74	68.91	64.25	71.18	67.54

3.6. Effectiveness of Non–Message Passing Units

In addition to adopting the GIN framework to increase the expressiveness of the shallow GNN network, it has been further proposed to incorporate non-message passing units MLP_{pre} and MLP_{post} into the architecture [35]. The contribution of increased model complexity to the performance is confirmed by the significant improvement in the otherwise underperforming GraphSAGE (M1.1). Although not as pronounced, GIN models also appear to benefit.

The performance metrics in Table 3 show the advantage of including the MLP_{post} unit in the M1-9.2 models over the basic M1-9.1 models. Adding the unit yields an average increase of 0.9% in each metric, with eight models achieving better P_{micro}, R_{micro} scores, and seven models achieving better $F1_{micro}$ scores. At the same time, the rest of the models where either precision or recall is affected, the opposite metric ($P_{micro} \iff R_{micro}$) shows an improvement in the order of 2.6% on average, which is always superior to the corresponding drop.

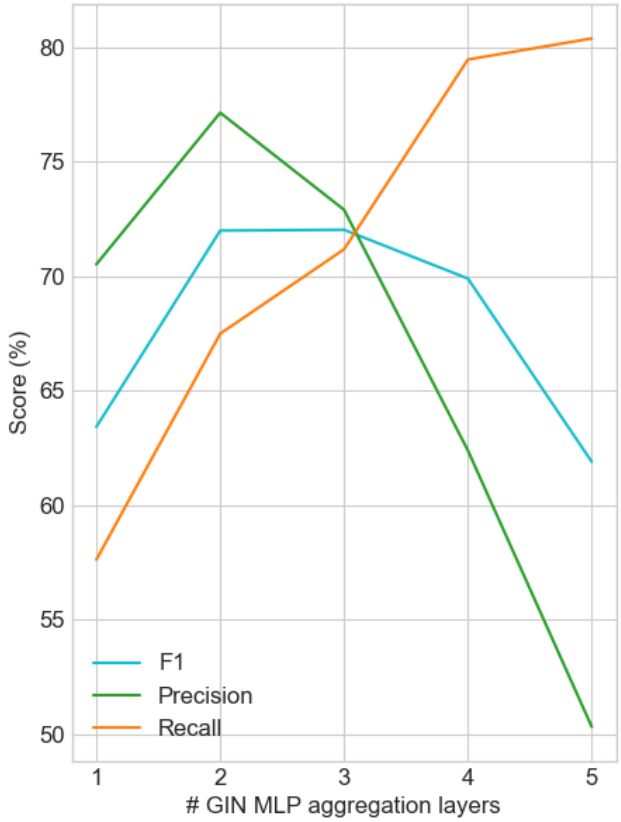


Figure 4. GIN MLP performance compared to the number of aggregation layers using the baseline BioBERT and no additional MLP units.

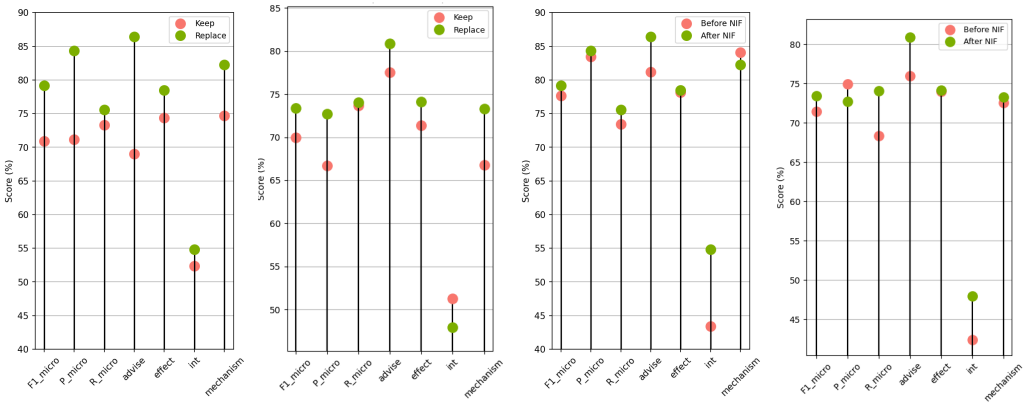
Conclusively, the three best performing models (M2.3, M3.3 and M8.2) include this unit. Each presents an improvement over its base version (M2.1, M3.1 and M8.1, respectively), while improving on the individual class level as well. Particularly, in at least three of the five classes, there is a clear improvement or at least some stability (decrease ≤ 0.08). This behavior is partially confirmed by the other (underperforming) GIN models, although not universally.

In contrast, the effect of the combination of the two units appears to be negative in the vast majority of models. Through MLP_{pre}, the word embeddings x_v produced by the BERT-based LMs (d_{BERT}) are projected into a smaller dimensional space d_{GNN} before the execution of message passing by GNN. This results in reduced performance in at least two instances. However, obvious exceptions are the models which make use of the BioBERT base architecture. Especially in combination with GIN, the models M2-3.3 outperform their M2-3.1-2 counterparts in every metric, with the exception of the $F1_{Effect}$ metric.

3.7. Effectiveness of Preprocessing

Focusing on the best performing model and its base (M3.3 and M8.2 respectively), the contribution of the preprocessing steps to the data was studied. Particularly, Figure 5 shows the effects of Drug Name Replacement (DNR, Figure 5a,b) and Negative Instance Filtering (NIF, Figure 5c,d) on the $F1$ metric of each positive class and the overall $F1_{micro}$ score.

First, reducing imbalance greatly benefits the *Advice* and *Int* classes in both models. Especially in M3.3 (Table 3), applying NIF improved the Recall by 7.6% and 10.9%, respectively, and the Precision by 3.4% and 6.2% respectively. Simultaneously, a relative robustness is demonstrated in the *Effect* class when restoring the rejected pairs; however, their inclusion appears to favor the *Mechanism* class. Conclusively, the overall $F1_{micro}$ is improved in each case through the rejection of trivial negative instances; however, M3.3 maintains a better balance between Recall and Precision than M8.2, where although no class is perceptibly affected, their removal leads to an increase in false positives.



(**a**) (**b**) (**c**) (**d**)

Figure 5. Performance comparison based on Drug Name Replacement (DNR) and Negative Instance Filtering (NIF), (**a**) M3.3 with DNR, (**b**) M8.2 with DNR, (**c**) M3.3 with NIF, (**d**) M8.2 with NIF.

In contrast, the use of *drugi* over the original drug names provides a clear performance improvement when generating word embeddings with BERT-based LM. This fact confirms the hypothesis that the type of interaction is determined by their syntactic role within the sentence and its content and not by specific substances' names. This method of replacing drug names, also known as entity blinding or entity masking, supports the generalization of the model. Therefore, through these preprocessing steps, noise is reduced in the feature vectors of the Token nodes.

3.8. Error Analysis

To analyze the advantages and limitations of our proposed approach, we compare and analyze the classification results of the best performing model (M3.3) on a few indicative cases (Table 4). Figure 6 shows the confusion matrix, with a total of 330 errors that were made, representing 11% of the 3057 test cases. In addition, 273 (83%) originate from the DrugBank instances and the remaining 57 (17%) from MedLine abstracts, corresponding to 9.8% and 20% of the total interactions of their respective collection.

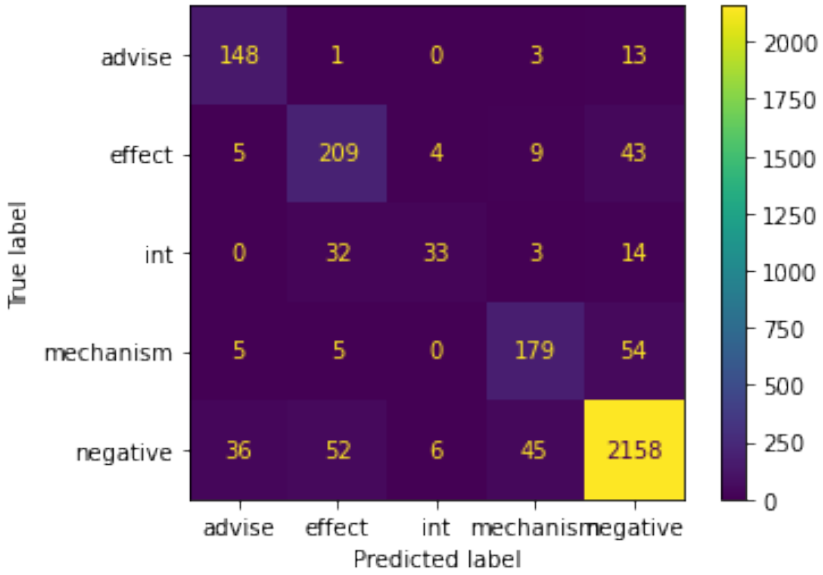


Figure 6. Confusion matrix of our proposed model.

First, misclassifying an existing interaction is the least common type of error, accounting for 20% of the total errors. Furthermore, 48% of these correspond to the case where instances of Int are classified as Effect, which makes this the main source of confusion in

this category. The contributing drug pairs were found in only five sentences, with one of them giving 28 of the total 32. This is S1, where although the system can recognize the existence of interactions in the long list of drugs, it fails to predict the types. This is probably due to the presence of "enhance", a word regularly associated with drug pairs with the interaction type Effect. Furthermore, 42% of the errors involve false positive pairs that can occur due to the following cases:

- The possibility of an annotation error is not excluded. Indicatively, in S2, every pair containing e_0 is of type Effect, as predicted by the model. However, any such relationship other than (e_0, e_1) is marked as negative in the data set.
- A substance appears multiple times in the sentence, such as *mebendazole* in S3. The pairs e_0, e_1 and e_0, e_2 are the Mechanism and Negative types, respectively, but both are predicted as Mechanism. An attempt was made to limit these occurrences when rejecting cases, but, as S3 indicates, they do not follow a specific pattern that can be expressed.
- Similar to the case of S1, confusion is caused when the description is made with expressions that refer to another type. In S4, e_0 is the subject of the sentence and declares the simultaneous administration of the mentioned drugs as safe, since they do not interact. However, due to the wording, the system perceives the relationships as Advice.
- Cases where drug mentions are listed and are not covered by regular expressions. e.g., in S5, the existence of "the" excludes the match and makes it impossible to locate the negative pairs (e_0, e_i), $i = 1, \ldots, 5$. However, as a large number of instances has been discarded from the corpus, the model is unable to handle these underrepresented patterns.

However, the most serious form of error concerns the inability to detect existing interactions, or else the existence of false negatives. An obvious source of error is the particularly long sentences, where descriptions are highly complex. The same applies to long sentences that could have been separated into smaller sentences, occurring in 53 related instances. We define sentences as long when they have a length of ≥ 40 tokens (the number of words separated by a space, having replaced drugs with "drug"). S6 is an example of a sentence with a length of 40, where three interactions of the Mechanism type were not detected.

However, there are several instances where misclassification can be attributed to system errors. For example, the interaction in S7 is not found, even though the sentence contains characteristic expressions that suggest that an interaction is being described and does not include any redundant information that might cause any confusion. The existence of such phenomena causes difficulty in the holistic interpretation of the results.

3.9. Data Uncertainty

The main point of uncertainty that may arise in our proposed approach, that is shared with all related works using the DDI corpus, is input-dependent data uncertainty. In this case, the observation noise varies based on the input and is commonly introduced during the data generation process [36]. In order to address this issue, we attempt to deal with the observed inconsistencies during the the pre-processing stage.

Initially, simple entity name transformations are applied by changing the plural form to the singular form when the same substance is referenced and both cases are found in the the Drug_Class set (e.g., "penicillin" and "penicillins"). Consequently, this leads to a total of 122 cases of identified cases.

An additional point of uncertainty concerns the classification of drugs into the four classes as each drug mention generally belongs to a single class. However, cases were observed where the same drug was labeled with different classes throughout the instances found in the dataset. Although the percentage of entities in which this is observed is relatively small, the drug mentions in question participate in a large number of interaction pairs. Specifically, 6023 pairs are identified in which at least one entity with multiple types is found.

Table 4. Indicative cases of misclassified instances. Drug names are denoted in bold and underlined words describe the interaction in the sentence. Subscripts denote the drug name (entity) index in the sentence.

	Sentence
S1	Other drugs which may enhance the neuromuscular blocking action of **nondepolarizing agents** such as **MIVACRON** include certain **antibiotics** e.g., ***antibiotics_group***).
S2	**Thalidomide**$_{e0}$ has been reported to enhance the sedative activity of **barbiturates**$_{e1}$, **alcohol**$_{e2}$, **chlorpromazine**$_{e3}$, and **reserpine**$_{e4}$.
S3	Preliminary evidence suggests that **cimetidine**$_{e0}$ inhibits **mebendazole**$_{e1}$ metabolism and may result in an increase in plasma concentrations of **mebendazole**$_{e2}$.
S4	**Pyrimethamine**$_{e0}$ may be used with **sulfonamides**$_{e1}$, **quinine**$_{e2}$ and other **antimalarials**$_{e3}$, and with other **antibiotics**$_{e4}$.
S5	**Dopamine antagonists**$_{e0}$, such as the **neuroleptics**$_{e1}$ (**phenothiazines**$_{e2}$, **butyrophenones**$_{e3}$, **thioxanthines**$_{e4}$) or **metoclopramide**$_{e5}$, ordinarily should not be administered concurrently with **Permax**$_{e6}$ (a **dopamine agonist**$_{e7}$)
S6	The bioavailability of **SKELID**$_{e0}$ is decreased 80% by **calcium**$_{e1}$, when **calcium**$_{e2}$ and **SKELID**$_{e3}$ are administered at the same time, and 60% by some **aluminum**$_{e4}$- or **magnesium**$_{e5}$-containing **antacids**$_{e6}$, when administered 1 hour before **SKELID**$_{e7}$.
S7	**Anticholinergics**$_{e0}$ antagonize the effects of **antiglaucoma agents**$_{e1}$.

Thus, the management of differentiation is sought without rejecting them. For example, in the case of "corticosteroid" where the vast majority of occurrences belong to a specific class, the assumption can be made that all instances should be labeled based on the majority class. In contrast, in the case of "tetracycline", where the class distribution is not clearly in favor of a single class, no such assumption can be made without introducing more uncertainty in the dataset.

The class of a drug entity is defined by a <drug-name, rdf:type, drug-class> triplet, where the object takes its value from the set of the four positive drug classes which should be unique for each instance of the Drug_Class class. However, a small amount of drug names cannot be classified to a single class. Moreover, each individual case is characterized by the name of the drug, as well as the set of the additional classes it was labeled in its various occurrences in the dataset. Therefore, for each one of these classes *c*, a blank node <_:substance-name_c> and a statement in the form of <_:substance-name_c, rdf:type, c> are included in the KG. Furthermore, in order to emphasize that those individual entities are not independent, the property name is defined which participates in <_:substance-name_c, name, substance-name> triplets. Therefore, entities that share the same value in this attribute are referring to the same substance.

An example of how our proposed approach performs on instances that contain uncertainties can be seen in the example sentence S2 (Table 4) in Section 3.8.

4. Conclusions

In this paper, we propose a Knowledge Graph schema in a Graph Neural Network-based architecture for the classification of Drug–Drug Interactions from biomedical literature, which achieves state-of-the-art performance. Specifically, we presented a Graph Isomorphism Network-based architecture with message passing and non-message passing units that leverage the proposed DDI-specific graph structure that models the knowledge (drug identifiers, names, types and interactions) from the DDI corpus. Token and sentence embeddings are generated for the drug named entities and sentences, respectively, and are passed to the graph, populating the Token and Sentence nodes, taking advantage of the underlying BERT-based LM.

Although our approach achieves state-of-the-art performance in the DDI classification task, the experimental results show that the individual class scores are greatly affected by the underlying LM, indicating that further improvements can be achieved. Based on the

results, future work could be directed towards exploring a combination of general and domain-specific corpora to pre-train or domain-adapt a LM to further improve the performance of each positive class. Another direction for future work is to extend our approach with multi-task learning for extracting the entities in combination with interactions that could potentially improve generalization by using the domain information contained in the training signals of the related tasks as an inductive bias.

Author Contributions: Conceptualization, D.Z.; methodology, D.Z. and C.K.; formal analysis, D.Z. and C.K.; investigation, D.Z. and C.K.; software, D.Z. and C.K.; resources, D.Z.; validation, D.Z. and C.K.; writing—original draft preparation, D.Z.; writing—review and editing, D.Z., C.K. and I.V.; visualization, D.Z. and C.K.; supervision, D.Z. and I.V. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable

Data Availability Statement: The DDI Extraction 2013 corpus is available at https://github.com/isegura/DDICorpus, (accessed on 25 October 2022).

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

DDI	Drug–Drug Interaction
NN	Neural Networks
GNN	Graph Neural Network
GCN	Graph Convolutional Network
RGNN	Relational Graph Neural Network
KG	Knowledge Graphs
ML	Machine Learning
RDF	Resource Description Framework
RE	Relation(ship) Extraction
PoS	Part-of-Speech
CNN	Convolutional Neural Network
RNN	Recurrent Neural Network
LSTM	Long Short-Term Memory
BiLSTM	Bidirectional Long Short-Term Memory
GRU	Gated Recurrent Unit
DGF	Dependency Graph Features
XML	Extensible Markup Language
LM	Language Model
SLDA	Sentence Level Domain Adapted
DLDA	Document Level Domain Adapted
GIN	Graph Isomorphism Network
MLP	Multi-Layer Perceptron
DNR	Drug Name Replacement
NIF	Negative Instance Filtering
CPU	Central Processing Unit

Appendix A

Table A1. Comparative analysis of related studies and our proposed approach. NIF and DNR denote Negative Instance Filtering and Drug Name Replacement, respectively. Values that are not reported in the published work are denoted with '-'. DGF denotes Dependency Graph Features. Y and N denote Yes and No, respectively.

Reference	Method	Embeddings	Emb. dim.	Features	NIF	DNR	Highlights	Review
Liu et al. [7]	CNN	Order	300	Position	Y	Y	Position embeddings improve performance	Dependent on position distributions of the input
Quan et al. [8]	CNN	CBOW	200	Position	N	Y	Multiple embeddings capture better representations	Errors in long sentences
Liu et al. [9]	CNN	Order	300	Position, DGF	N	Y	Syntactic features for long distance dependencies	Only a small set of large sentences parsed correctly
Zhao et al. [10]	CNN	Word2Vec	-	Position, PoS, DGF	Y	N	Dependency features and position embeddings	Filtering rules lead to limited generalization ability
Wang et al. [11]	LSTM	Word2Vec	100	Distance, DGF	Y	Y	Captures distance and dependency-based features	Low performance on long sentences
Yi et al. [12]	RNN	GloVe	100	Position	N	Y	Multiple attention layers to capture better representations	Semantic ambiguity leads to misclassifications
Zhang et al. [13]	LSTM	Word2Vec	200	PoS, DGF	N	N	Integration of sentence sequences, shortest dependency paths and attention layers	Errors in long sentences
Zhou et al. [14]	LSTM	Word2Vec	300	Position	Y	Y	Additional position embeddings to generate the attention weights	Misclassification of instances containing multiple drug mentions
Salman et al. [15]	LSTM	Word2Vec	100	Position, DGF	N	N	Task expansion to sentiment-based severity prediction	Word positions in the sentences are not taken into account
Mondal [22]	BERT-VAE	BioBERT	300	Chemical Structures	N	N	Utilizes chemical structure representations of drugs	Information loss in longer sentences due to low representation dimensionality
Xiong et al. [25]	LSTM- GCNN	Word2Vec	200	PoS, DGF	Y	N	Dependency features with graph neural network	Complex graph structures impact performance
Shi et al. [26]	LSTM-GCNN	Word2Vec	200	PoS, DGF, PageRank	Y	N	Dependency features with graph neural network and PageRank	Added complexity with feature generation from complex graph structures
Our approach	BERT-GIN	BioBERT	512	KG, DAPT	Y	Y	Novel DDI task-based Knowledge Graph leveraged by a graph neural network without relying on manual feature engineering	Nodes are initialized with domain-adapted representations to better capture sentence context

References

1. Percha, B.; Altman, R.B. Informatics confronts Drug–Drug Interactions. *Trends Pharmacol. Sci.* **2013**, *34*, 178–184. [CrossRef]
2. Hunter, L.; Cohen, K.B. Biomedical Language Processing: What's Beyond PubMed? *Mol. Cell* **2006**, *21*, 589–594. [CrossRef]
3. Wang, Y.; Wang, L.; Rastegar-Mojarad, M.; Moon, S.; Shen, F.; Afzal, N.; Liu, S.; Zeng, Y.; Mehrabi, S.; Sohn, S.; et al. Clinical information extraction applications: A literature review. *J. Biomed. Inform.* **2018**, *77*, 34–49. [CrossRef] [PubMed]
4. Segura-Bedmar, I.; Martínez, P.; Herrero-Zazo, M. SemEval-2013 Task 9: Extraction of Drug–Drug Interactions from Biomedical Texts (DDIExtraction 2013). In *Proceedings of the Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*; Association for Computational Linguistics: Atlanta, GA, USA, 2013; pp. 341–350.
5. Zhang, T.; Leng, J.; Liu, Y. Deep learning for Drug–Drug Interactions extraction from the literature: A review. *Briefings Bioinform.* **2019**, *21*, 1609–1627. [CrossRef] [PubMed]
6. Herrero-Zazo, M.; Segura-Bedmar, I.; Martínez, P.; Declerck, T. The DDI corpus: An annotated corpus with pharmacological substances and Drug–Drug Interactions. *J. Biomed. Inform.* **2013**, *46*, 914–920. [CrossRef] [PubMed]
7. Liu, S.; Tang, B.; Chen, Q.; Wang, X. Drug–Drug Interaction Extraction via Convolutional Neural Networks. *Comput. Math. Methods Med.* **2016**, *2016*, 6918381. [CrossRef]
8. Quan, C.; Hua, L.; Sun, X.; Bai, W. Multichannel Convolutional Neural Network for Biological Relation Extraction. *BioMed Res. Int.* **2016**, *2016*, 1850404. [CrossRef] [PubMed]
9. Liu, S.; Chen, K.; Chen, Q.; Tang, B. Dependency-based convolutional neural network for drug–drug interaction extraction. In Proceedings of the 2016 IEEE international conference on bioinformatics and biomedicine (BIBM), Shenzhen, China, 15–18 December 2016; pp. 1074–1080.
10. Zhao, Z.; Yang, Z.; Luo, L.; Lin, H.; Wang, J. Drug drug interaction extraction from biomedical literature using syntax convolutional neural network. *Bioinformatics* **2016**, *32*, 3444–3453. [CrossRef]
11. Wang, W.; Yang, X.; Yang, C.; Guo, X.; Zhang, X.; Wu, C. Dependency-based long short term memory network for drug–drug interaction extraction. *BMC Bioinform.* **2017**, *18*, 99–109. [CrossRef]
12. Yi, Z.; Li, S.; Yu, J.; Tan, Y.; Wu, Q.; Yuan, H.; Wang, T. Drug-drug interaction extraction via recurrent neural network with multiple attention layers. In Proceedings of the International Conference on Advanced Data Mining and Applications, Foshan, China, 12–15 November 2017; pp. 554–566.
13. Zhang, Y.; Zheng, W.; Lin, H.; Wang, J.; Yang, Z.; Dumontier, M. Drug–drug interaction extraction via hierarchical RNNs on sequence and shortest dependency paths. *Bioinformatics* **2017**, *34*, 828–835. [CrossRef]
14. Zhou, D.; Miao, L.; He, Y. Position-aware deep multi-task learning for drug–drug interaction extraction. *Artif. Intell. Med.* **2018**, *87*, 1–8. [CrossRef] [PubMed]
15. Salman, M.; Munawar, H.S.; Latif, K.; Akram, M.W.; Khan, S.I.; Ullah, F. Big Data Management in Drug–Drug Interaction: A Modern Deep Learning Approach for Smart Healthcare. *Big Data Cogn. Comput.* **2022**, *6*, 30. [CrossRef]
16. Mikolov, T.; Chen, K.; Corrado, G.; Dean, J. Efficient Estimation of Word Representations in Vector Space. *arXiv* **2013**. arXiv:1301.3781.
17. Peters, M.E.; Neumann, M.; Iyyer, M.; Gardner, M.; Clark, C.; Lee, K.; Zettlemoyer, L. Deep Contextualized Word Representations. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, New Orleans, AK, USA, 1–6 June 2018; Volume 1 (Long Papers), pp. 2227–2237. [CrossRef]
18. Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; Sutskever, I. Language models are unsupervised multitask learners. *OpenAI Blog* **2019**, *1*, 9.
19. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. BERT: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Minneapolis, MN, USA, 2–7 June 2019 . Available online: https://aclanthology.org/N19-1423 (accessed on 20 September 2022).
20. Lee, J.; Yoon, W.; Kim, S.; Kim, D.; Kim, S.; Thus, C.H.; Kang, J. BioBERT: A pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics* **2019**, *36*, 1234–1240. [CrossRef] [PubMed]
21. Beltagy, I.; Lo, K.; Cohan, A. SciBERT: A 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 (EMNLP-IJCNLP), Hong Kong, China, 3–7 November 2019; pp. 3615–3620. [CrossRef]
22. Mondal, I. BERTChem-DDI : Improved Drug–Drug Interaction Prediction from text using Chemical Structure Information. In Proceedings of the Knowledgeable NLP: The First Workshop on Integrating Structured Knowledge and Neural Networks for NLP, Suzhou, China, 7 December 2020; pp. 27–32.
23. Gu, W.; Yang, X.; Yang, M.; Han, K.; Pan, W.; Zhu, Z. MarkerGenie: An NLP-enabled text-mining system for biomedical entity relation extraction. *Bioinform. Adv.* **2022**, *2*, vbac035. [CrossRef]
24. Ren, Z.H.; You, Z.H.; Yu, C.Q.; Li, L.P.; Guan, Y.J.; Guo, L.X.; Pan, J. A biomedical knowledge graph-based method for Drug–Drug Interactions prediction through combining local and global features with deep neural networks. *Briefings Bioinform.* **2022**, *23*, bbac363. [CrossRef]

25. Xiong, W.; Li, F.; Yu, H.; Ji, D. Extracting Drug–Drug Interactions with a dependency-based graph convolution neural network. In Proceedings of the IEEE International Conference on Bioinformatics and Biomedicine (BIBM), San Diego, CA, USA, 18–21 November 2019; pp. 755–759.
26. Shi, Y.; Quan, P.; Zhang, T.; Niu, L. DREAM: Drug-drug interaction extraction with enhanced dependency graph and attention mechanism. *Methods* **2022**, *203*, 152–159. [CrossRef]
27. Hamilton, W.L.; Ying, R.; Leskovec, J. Representation learning on graphs: Methods and applications. *arXiv* **2017**, arXiv:1709.05584.
28. Schlichtkrull, M.; Kipf, T.N.; Bloem, P.; van den Berg, R.; Titov, I.; Welling, M. Modeling Relational Data with Graph Convolutional Networks. In *The Semantic Web*; Gangemi, A., Navigli, R., Vidal, M.E., Hitzler, P., Troncy, R., Hollink, L., Tordai, A., Alam, M., Eds.; Springer: Berlin/Heidelberg, Germany, 2018; pp. 593–607.
29. Gururangan, S.; Marasović, A.; Swayamdipta, S.; Lo, K.; Beltagy, I.; Downey, D.; Smith, N.A. Don't Stop Pretraining: Adapt Language Models to Domains and Tasks. *arXiv* **2020**, arXiv:2004.10964.
30. Xu, K.; Hu, W.; Leskovec, J.; Jegelka, S. How Powerful are Graph Neural Networks? In Proceedings of the 7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, 6–9 May 2019.
31. Hamilton, W.; Ying, Z.; Leskovec, J. Inductive representation learning on large graphs. *Adv. Neural Inf. Process. Syst.* **2017**, *30* . Available online: https://proceedings.neurips.cc/paper/2017/file/5dd9db5e033da9c6fb5ba83c7a7ebea9-Paper.pdf (accessed on 20 September 2022).
32. Chowdhury, M.F.M.; Lavelli, A. FBK-irst : A Multi-Phase Kernel Based Approach for Drug–Drug Interaction Detection and Classification that Exploits Linguistic Information. In Proceedings of the Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013), Atlanta, GA, USA, 14–15 June 2013; pp. 351–355.
33. Duong, C.T.; Hoang, T.D.; Dang, H.T.H.; Nguyen, Q.V.H.; Aberer, K. On node features for graph neural networks. *arXiv* **2019**, arXiv:1911.08795.
34. Zhong, R.; Ghosh, D.; Klein, D.; Steinhardt, J. Are larger pretrained language models uniformly better? comparing performance at the instance level. *arXiv* **2021**, arXiv:2105.06020.
35. You, J.; Ying, R.; Leskovec, J. Design Space for Graph Neural Networks. In Proceedings of the 34th International Conference on Neural Information Processing Systems (NIPS'20), Vancouver, BC, Canada, 6–12 December 2020; Curran Associates Inc.: Red Hook, NY, USA, 2020.
36. Xiao, Y.; Wang, W.Y. Quantifying uncertainties in natural language processing tasks. In Proceedings of the AAAI Conference on Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2019; Volume 33, pp. 7322–7329.