
This is an electronic reprint of the original article.
This reprint may differ from the original in pagination and typographic detail.

Gao, Ya; Ji, Shaoxiong; Marttinen, Pekka

Knowledge-augmented Graph Neural Networks with Concept-aware Attention for Adverse Drug Event Detection

Published in:

Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)

Published: 25/05/2024

Document Version

Publisher's PDF, also known as Version of record

Published under the following license:

CC BY-NC

Please cite the original version:

Gao, Y., Ji, S., & Marttinen, P. (2024). Knowledge-augmented Graph Neural Networks with Concept-aware Attention for Adverse Drug Event Detection. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)* (pp. 9787–9798). (LREC proceedings), (International conference on computational linguistics). European language resources distribution agency.

This material is protected by copyright and other intellectual property rights, and duplication or sale of all or part of any of the repository collections is not permitted, except that material may be duplicated by you for your research use or educational purposes in electronic or print form. You must obtain permission for any other use. Electronic or print copies may not be offered, whether for sale or otherwise to anyone who is not an authorised user.

Knowledge-augmented Graph Neural Networks with Concept-aware Attention for Adverse Drug Event Detection

Yao Gao^{1*}, Shaoxiong Ji^{2*}, Pekka Marttinen¹

¹ Aalto University ² University of Helsinki
{yao.gao; pekka.marttinen}@aalto.fi, shaoxiong.ji@helsinki.fi

Abstract

Adverse drug events (ADEs) are an important aspect of drug safety. Various texts such as biomedical literature, drug reviews, and user posts on social media and medical forums contain a wealth of information about ADEs. Recent studies have applied word embedding and deep learning-based natural language processing to automate ADE detection from text. However, they did not explore incorporating explicit medical knowledge about drugs and adverse reactions or the corresponding feature learning. This paper adopts the heterogeneous text graph, which describes relationships between documents, words, and concepts, augments it with medical knowledge from the Unified Medical Language System, and proposes a concept-aware attention mechanism that learns features differently for the different types of nodes in the graph. We further utilize contextualized embeddings from pretrained language models and convolutional graph neural networks for effective feature representation and relational learning. Experiments on four public datasets show that our model performs competitively to the recent advances, and the concept-aware attention consistently outperforms other attention mechanisms.

Keywords: Adverse Drug Event Detection, Graph Neural Networks, Knowledge Augmentation, Attention Mechanism

1. Introduction

Pharmacovigilance, i.e., drug safety monitoring, is a critical step in drug development (Wise et al., 2009). It detects adverse events and safety issues and promotes drug safety through post-market assessment; therefore, it promotes safe drug development and shows significant promise in better healthcare service delivery. A drug-related negative health outcome is referred to as an Adverse Drug Event (ADE) (Donaldson et al., 2000). Given the significant harm caused by ADEs, it is essential to detect them for pharmacovigilance purposes.

Clinical trials are the common way to detect ADEs. However, some ADEs are hard to investigate through clinical trials due to their long latency (Sultana et al., 2013). Additionally, regular trials cannot cover all aspects of drug use. Through the voluntary Post-marketing Drug Safety Surveillance System (Li et al., 2014), users report their experiences with drug usage and related safety issues. Nevertheless, the system suffers several limitations, such as incomplete reporting, under-reporting, and delayed reporting.

Recent advances in automated pharmacovigilance are based on collecting large amounts of text about adverse drug events from various platforms, such as medical forums (e.g., AskAPatient), biomedical publications, and social media, and training Natural Language Processing (NLP) models to automatically detect whether a given textual

record contains information about adverse drug reactions, which is usually framed as a binary classification task. Text mentions of adverse drug events include a plethora of drug names and adverse reactions. Figure 1 shows an example annotated with concepts from the Unified Medical Language System (UMLS), where each identified medical concept in the text is assigned a Concept Unique Identifier (CUI) from UMLS. To understand the drug information and corresponding adverse reactions, the NLP model needs to capture abundant medical knowledge and be able to do relational reasoning.

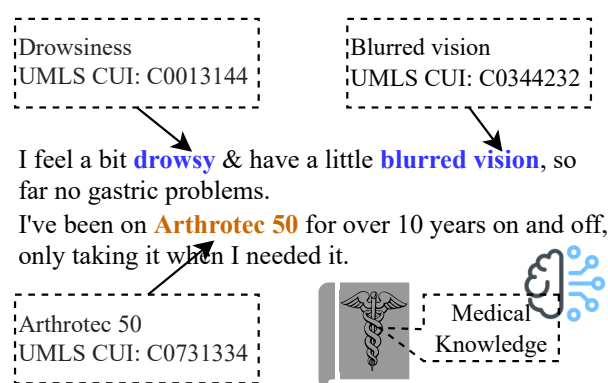


Figure 1: An example of a text mentioning an adverse drug event from Karimi et al. (2015). The recognition of drugs and adverse reactions requires medical knowledge and relational reasoning.

Early studies used rule-based methods (Xu et al., 2010; Sohn et al., 2014) with manually built rules or applied machine learning algorithms such as

* Equal contribution. Corresponding author: Shaoxiong Ji. Work done while affiliated with Aalto University.

conditional random fields (Nikfarjam et al., 2015; Wang et al., 2022), support vector machine (Bologala et al., 2018), and neural networks (Cocos et al., 2017; Huynh et al., 2016). These approaches can process text with manual feature engineering or enable automated feature learning with deep learning methods, allowing for automated ADE detection. However, they are limited in capturing rich contextual information and relational reasoning.

Graphs are expressive and can represent various data. For example, nodes in a graph for a collection of texts can represent various entities, such as words, phrases, and documents, while edges represent relationships between them. Such text graphs, together with graph neural networks, are widely used in NLP applications such as sentiment classification and review rating (Yao et al., 2019; Lin et al., 2021; Zhang et al., 2020b). Recently, graphs have been used for text representation with graph boosting (Shen et al., 2020) or contextualized graph embeddings (Gao et al., 2022) for ADE detection. Other works have applied knowledge graph embeddings and link prediction to ADE prediction in drug-effect knowledge graphs (Kwak et al., 2020; Joshi et al., 2022). However, medical knowledge plays an important role in ADE detection from text, and so far, there are no studies that incorporate medical knowledge in a text graph and learn concept-aware representations that inject medical concepts (e.g., the UMLS concepts as illustrated in Figure 1) into the text embeddings.

A recent model called CGEM (Gao et al., 2022), which is close to our work, applied a heterogeneous text graph, embodying word and document relations for an ADE corpus, to learn contextualized graph embeddings for ADE detection. Here, we extend the work by Gao et al. (2022) in two useful ways. First, we show how the graph can be augmented with medical knowledge from the UMLS metathesaurus (Bodenreider, 2004) and how different similarity measurement methods can contribute to better knowledge infusion. Second, we deploy concept-aware self-attention that applies different feature learning for various types of nodes. We name our model as KnowCAGE (Knowledge-augmented Concept-Aware Graph Embeddings). Our contributions are thus summarized as follows:

- We introduce medical knowledge, i.e., the UMLS metathesaurus, to augment the contextualized graph embedding model for representation learning on drug adverse events.
- A concept-aware self-attention is devised to learn discriminable features for the concept (from the medical knowledge), word, and document nodes.
- Experimental results evaluated in four public datasets from medical forums, biomedical pub-

lications, and social media show our approach outperforms recent advanced ADE detection models in most cases.

2. Related Work

Recent advances in adverse drug event detection use word embeddings and neural network models to extract text features and capture the drug-effect interaction. Many studies deploy recurrent neural networks to capture the sequential dependency in text. For example, Cocos et al. (2017) utilized a Bidirectional Long Short-Term Memory (BiLSTM) network, and Luo (2017) proposed to learn sentence- and segment-level representations based on LSTM. To process entity mentions and relations for ADE detection and extraction, pipeline-based systems (Dandala et al., 2019) and jointly learning methods (Wei et al., 2020) are two typical approaches.

Several recent publications studied graph neural networks for ADE detection. Kwak et al. (2020) built a drug-disease graph to represent clinical data for adverse drug reaction detection. GAR (Shen et al., 2021) uses graph embedding-based methods and adversarial learning. CGEM (Gao et al., 2022) combines contextualized embeddings from pretrained language models with graph convolutional neural networks.

Some other studies also adopted other neural network architectures, such as capsule networks and self-attention mechanisms. Zhang et al. (2020a) proposed the gated iterative capsule network (GICN) using CNN and a capsule network to extract the complete phrase information and deep semantic information. The gated iterative unit in the capsule network enables the clustering of features and captures contextual information. The attention mechanism prioritizes representation learning for the critical parts of a document by assigning them higher weight scores. Ge et al. (2019) employed multi-head self-attention, and Wunnava et al. (2020) developed a dual-attention mechanism with BiLSTM to capture semantic information in the sentence.

Another direction of related work is knowledge augmentation for deep learning models. Many publications adopt knowledge graph to guide the representation learning in various applications (Ji et al., 2022). For example, Ma et al. (2018) injected commonsense knowledge into a long short-term memory network, and Liang et al. (2022) enhanced the graph convolutional network with affective knowledge to improve aspect-based sentiment analysis. Knowledge injection is also used for other applications such as hate speech detection (Pamungkas et al., 2021), mental healthcare (Yang et al., 2022), and personality detection (Poria et al., 2013). Some other works focus on the construction

of medical knowledge graphs and data augmentation with knowledge graphs (Sun et al., 2020; Shi et al., 2022).

3. Methods

This section introduces the proposed graph-based model with knowledge augmentation, i.e., Knowledge-augmented Concept-Aware Graph Embeddings (KnowCAGE), as illustrated in Figure 2. The model consists of four components: 1) Knowledge-augmented Graph Construction, 2) Heterogeneous Graph Convolution, 3) Concept-aware Attention, and 4) Ensemble-based ADE classification layers. Following TextGCN (Yao et al., 2019), we construct a heterogeneous text graph, which contains three types of nodes: words, documents, and concepts, and we augment it with medical knowledge from the Unified Medical Language System metathesaurus. Heterogeneous graph convolution networks are then used to encode the text graph and learn rich representations. We use the contextualized embeddings from pretrained BERT (Bidirectional Encoder Representations from Transformers) (Devlin et al., 2019) to represent the node features in the heterogenous text graph. The adjacency matrix and feature matrix obtained from the embedding layers are inputs to graph neural network encoders, which take into account the relationships and information between and within the nodes. Considering different types of nodes, we use a concept-aware self-attention, inspired by the entity-aware representation learning (Yamada et al., 2020), which treats the different types of nodes differently, allowing the most significant content to have the largest contribution to the final prediction. To boost the prediction of ADE even further, we follow the BertGCN model (Lin et al., 2021) and apply an ensemble classifier with contextualized embeddings on the one hand and the graph networks on the other, and learn a weight coefficient to balance these two prediction branches.

3.1. Knowledge-augmented Graph Construction

We firstly build the heterogeneous text graph for the whole document collection by using the external knowledge source - UMLS - to augment the word/document graph with concept information. Representing text in a heterogeneous graph can provide different perspectives for text encoding and improve ADE detection. In the UMLS metathesaurus, different words or phrases are assigned CUIs, where each CUI represents one concept class. Every concept class has an attribute “preferred name” which is a short description or a synonym of this concept. Given a document, our initial

step involves employing the longest string match to map words or phrases in the document to CUIs. Within the UMLS, numerous concepts may be associated with a single word or phrase. We utilize the first concept returned by the system. Note that we also filter out concepts falling under certain semantic types, such as those related to plants and animals, which are irrelevant to the task. As a result of this mapping strategy, only a small subset of concepts are extracted for each dataset.

After our model leverages UMLS to retrieve the preferred names corresponding to concepts in the dataset, these preferred names are added to the graph as the concept nodes. Therefore, the augmented graph also contains concept nodes in addition to the word and document nodes. The number of total nodes $n = n_d + n_w + n_c$, where n_d , n_w and n_c are the numbers of documents, words, and concepts, respectively. There are five types of edges, i.e., word-word, word-concept, document-concept, concept-concept, and document-word edges. The weights of document-word edges and document-concept edges are calculated as the term frequency-inverse document frequency (TF-IDF), while the weights of the other edges are defined as the similarity or association between two nodes.

Depending on the characteristics of the data set, the most suitable way of knowledge infusion varies. Therefore, we explore different measurement methods (represented as SIM): L1 distance, L2 distance, Cosine distance, and Pointwise Mutual Information. They provide distinct insights about node relations, such as co-occurrence information or knowledge about semantic distances.

The word encoding for distance computation is derived from fine-tuning language model on ADE classification. Specifically, the weight between the node i and the node j is computed as:

$$\mathbf{A}_{ij} = \begin{cases} \text{SIM}(i, j), & \text{SIM} > 0; i, j: \text{word/concept} \\ \text{TF-IDF}_{ij}, & i: \text{document}, j: \text{word/concept} \\ 0, & \text{otherwise} \end{cases}$$

We use pretrained contextualized embeddings from language models. Given the dimension of embeddings denoted as d , the pooled output of contextualized document encoding is denoted as $\mathbf{H}_{doc} \in \mathbb{R}^{n_d \times d}$. We initialize word and concept nodes with a zero matrix to get the initial feature matrix which is used as input to the graph neural network:

$$\mathbf{H}^{[0]} = \begin{pmatrix} \mathbf{H}_{doc} \\ \mathbf{0} \end{pmatrix}, \quad (1)$$

where $\mathbf{H}^{[0]} \in \mathbb{R}^{(n_d+n_w+n_c) \times d}$ and $[0]$ denotes the initial layer.

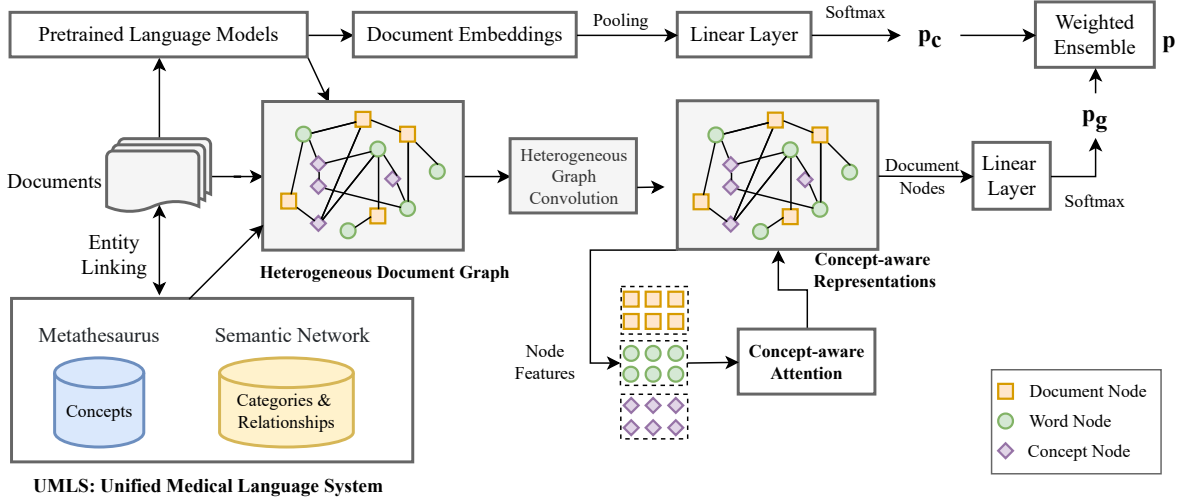


Figure 2: An illustration of the model architecture with knowledge-augmented graph embeddings and concept-aware representations

3.2. Heterogeneous Graph Convolution

We adopt graph neural networks over the heterogeneous text graph to learn complex relations between words, concepts, and documents. Specifically, given the initial input features $\mathbf{H}^{[0]}$ obtained from pretrained language models and the adjacency matrix \mathbf{A} , we update the representations via graph convolution. A forward pass of the i -th layer of a convolutional graph network can be denoted as:

$$\mathbf{H}^{[i+1]} = f(\hat{\mathbf{A}}\mathbf{H}^{[i]}\mathbf{W}^{[i]}), \quad (2)$$

where $\hat{\mathbf{A}}$ is the normalized adjacency matrix, $\mathbf{H}^{[i]}$ are the hidden representations of i -th layer, $\mathbf{W}^{[i]}$ is the weight matrix, and $f(\cdot)$ is an activation function. The KnowCAGE framework can adopt various types of convolutional graph neural networks. Our experimental study chooses three representative models, i.e., Graph Convolutional Network (GCN) (Kipf and Welling, 2017), Graph Attention Network (GAT) (Veličković et al., 2018), and Deep Graph Convolutional Neural Network (DGCNN) (Zhang et al., 2018). GCN is a spectral-based model with a fixed number of layers where different weights are assigned to layers and the update of node features incorporates information from the node's neighbors. It employs convolutional architectures to get a localized first-order representation. Graph attention layers in GAT assign different attention scores to one node's distant neighbors and prioritize the importance of different types of nodes. DGCNN concatenates hidden representations of each layer to capture rich substructure information and adopt a SortPooling layer to sort the node features.

3.3. Concept-aware Attention Mechanism

Different types of nodes have various impacts on the prediction of adverse drug events. Inspired by the contextualized entity representation learning from the knowledge supervision of knowledge bases (Yamada et al., 2020), we propose to use a concept-aware attention mechanism that distinguishes the types of nodes, especially the concept nodes, and better captures important information related to the positive or negative ADE classes.

Two types of nodes may not have the same impact on each other. Thus, we use different transformations for different types of nodes in the concept-aware attention mechanism in order to learn concept-aware attentive representations. We obtain key and value matrices $\mathbf{K} \in \mathbb{R}^{l \times d_h}$ and $\mathbf{V} \in \mathbb{R}^{l \times d_h}$ similarly to the key and value in the self-attention of transformer network (Vaswani et al., 2017). Concept-aware attention has nine different query matrices \mathbf{Q} for concept nodes c , word nodes w and document nodes d , i.e., \mathbf{Q}_{ww} , \mathbf{Q}_{cc} , \mathbf{Q}_{dd} , \mathbf{Q}_{cw} , \mathbf{Q}_{wc} , \mathbf{Q}_{wd} , \mathbf{Q}_{dw} , \mathbf{Q}_{dc} , and $\mathbf{Q}_{cd} \in \mathbb{R}^{l \times d_h}$. Then, we use \mathbf{Q} , \mathbf{K} and \mathbf{V} to compute the attention scores. For example, for i -th document and j -th concept nodes, i.e., \mathbf{x}_i and $\mathbf{x}_j \in \mathbb{R}^{d_h}$, we calculate the attention score as:

$$\alpha_{ij} = \text{Softmax} \left(\frac{(\mathbf{K}\mathbf{x}_j)^T \mathbf{Q}_{cd}\mathbf{x}_i}{\sqrt{l}} \right) \quad (3)$$

The concept-aware representation $\mathbf{h}_i \in \mathbb{R}^l$ for the i -th document is obtained as:

$$\mathbf{h}_i = \sum_{j=1}^n \alpha_{ij} \mathbf{V}\mathbf{x}_j \quad (4)$$

We can obtain the representations of word and concept nodes in the same way. These concept-aware representations are fed to the graph network

as the node features in the next iteration of model updating.

3.4. Classification Layers and Model Training

We apply the two linear layers and a softmax function over the concept-aware document embeddings \mathbf{h}_i to compute the probability of classifying the document in each class \mathbf{p}_g , representing the presence or absence of mentions of ADE in the document. Besides, the interpolation of the prediction probability of two classifiers is adopted to combine the prediction of graph-based modules and pretrained language model-based predictions (Lin et al., 2021). We use a similar classification module to process the contextualized embeddings from the pretrained language model (the upper branch in Fig. 2) and denote the corresponding classification probabilities by \mathbf{p}_c . A weight coefficient $\lambda \in [0, 1)$ is introduced to balance the results from graph-based encoding and contextualized models:

$$\mathbf{p} = \lambda \mathbf{p}_g + (1 - \lambda) \mathbf{p}_c. \quad (5)$$

This interpolation strategy can also be viewed as a weighted ensemble of two classifiers.

ADE detection is a binary classification task, and the classes are highly imbalanced in most datasets. To complicate the matter further, most datasets contain only a small number of samples, making the downsampling method to balance the classes inappropriate. This study applies the weighted binary cross-entropy loss function to alleviate this problem. The weighted loss function is denoted as:

$$\mathcal{L} = \sum_{i=1}^N [-w_+ y_i \log(p_i) - w_- (1 - y_i) \log(1 - p_i)], \quad (6)$$

where $w_+ = \frac{N_1}{N_0 + N_1}$ and $w_- = \frac{N_0}{N_0 + N_1}$ are weights of documents predicted as positive or negative samples respectively, N_0 and N_1 are the numbers of negative/positive samples in the training set, and y_i is the ground-truth label of a document. The Adam optimizer (Kingma and Ba, 2015) is used for model optimization.

4. Experimental Setup

Our goal is to conduct experiments on four ADE datasets and answer the following research questions.

RQ1: How does the proposed model perform in ADE detection on texts from various sources, compared to other methods?

RQ2: How does the heterogeneous graph convolution with knowledge augmentation improve the accuracy of ADE detection?

RQ3: Does the concept-aware attention improve the accuracy of the heterogeneous graph convolution to detect ADE?

In this section, we will describe the setup of the experiments, and in the next section, we will present the results of the experiments.

4.1. Data and Preprocessing

We used four datasets from the medical forum, biomedical publications, and social media, as summarized in Table 1, for evaluation. We preprocess data by removing stop words, punctuation, and numbers. For the data collected from Twitter, we use the tweet-preprocessor Python package¹ to remove URLs, emojis, and some reserved words for tweets.

Table 1: A statistical summary of datasets

Dataset	Documents	ADE	non-ADE
SMM4H	2,418	1,209	1,209
Twimed-Pub	1,000	191	809
Twimed-Twitter	625	232	393
CADEC	7,474	2,478	4,996

Twimed (Twimed-Twitter and Twimed-Pub)² The Twimed dataset (Alvaro et al., 2017) includes two sets collected from different domains, i.e., Twimed-Twitter from social media and Twimed-Pub for biomedical publications. In each document, people from various backgrounds annotate diseases, symptoms, drugs, and their relationships. A document annotated as outcome-negative is regarded as an adverse drug event. Models are tested using 10-fold cross-validation.

SMM4H³ This dataset from Social Media Mining for Health Applications (#SMM4H) shared tasks (Magge et al., 2021) is collected from Twitter with a description of drugs and diseases. We use the official validation set to evaluate the model performance for a fair comparison with baseline models developed in the SMM4H shared task.

CADEC⁴ The CSIRO Adverse Drug Event Corpus contains patient-reported posts from a medical forum called AskaPatient (Karimi et al., 2015). It includes extensive annotations on drugs, side effects, symptoms, and diseases. We use 10-fold cross-validation to evaluate the model’s performance.

¹<https://pypi.org/project/tweet-preprocessor/>

²<https://github.com/nestoralvaro/TwiMed>

³<https://healthlanguageprocessing.org/smm4h-2021/task-1/>

⁴<https://data.csiro.au/collection/csiro:10948>

4.2. Baselines and Evaluation

We compare the performance of our method with two sets of baseline models: 1) models designed for ADE detection and 2) pretrained contextualized models, and report Precision (P), Recall (R), and F1-score.

Customized models for ADE detection are as follows. **CNN-Transfer** (Li et al., 2020) (CNN-T for short) and **ATL** (Li et al., 2020) both exploited adversarial transfer learning, but they applied various feature extractors. Variants of attention mechanisms are designed and utilized in **HTR-MSA** (Wu et al., 2018), **IAN** (Alimova and Solovyev, 2018), **MSAM** (Zhang et al., 2019), and **ANNSA** (Zhang et al., 2021). **CGEM** (Gao et al., 2022) is a predecessor of our work, which developed a contextualized graph-based model.

The previously mentioned ADE detection baselines did not use the SMM4H dataset in their experiments. Therefore, we compare our model with pretrained language models. We use the base version of pretrained models for a fair comparison. Yaseen and Langer (2021) combined the LSTM network with the BERT text encoder (Devlin et al., 2019) for ADE detection. We denote it as BERT-LSTM. Pimpalkhute et al. (2021) introduced a data augmentation method and adopted the RoBERTa text encoder with additional classification layers (Liu et al., 2019) for ADE detection, denoted as RoBERTa-aug. Kayastha et al. (2021) utilized the domain-specific BERTweet (Nguyen et al., 2020) that is pretrained with English Tweets using the same architecture as BERT-base and classified ADE with a single-layer BiLSTM network, denoted as BERTweet-LSTM.

Furthermore, Large Language Models (LLMs) have demonstrated remarkable proficiency across various NLP tasks. In this study, we specifically investigate the zero-shot capabilities of **GPT-4 OpenAI** (2023) across four distinct datasets. We utilize the following prompt: *"Here are documents from [name and a short description about the dataset]. Determine if the following text contains adverse drug events (ADEs), with the answer being either 'yes' or 'no'."*

4.3. Hyper-parameters

Table 2 shows the hyper-parameters we tuned in our experiments, where LR is the learning rate. When the number of iterations exceeds a certain threshold, the learning rate scheduler decays the learning rate by the parameter γ . In our experiment, we set γ and the iteration milestone to 0.1 and 30, respectively.

Table 2: Choices of hyper-parameters

Hyper-parameters	Choices
LR for text encoder	$2e^{-5}$, $3e^{-5}$, $1e^{-4}$
LR for classifier	$1e^{-4}$, $5e^{-4}$, $1e^{-3}$
LR for graph-based models	$1e^{-3}$, $3e^{-3}$, $5e^{-3}$
Hidden dimension for GNN	200, 300, 400
Weight coefficient λ	0, 0.1 0.3, 0.5, 0.7, 0.9

5. Results

5.1. Comparison with Baselines in Different Domains (RQ1)

We compare our model's predictive performance with baseline models on the TwiMed (Table 3), SMM4H (Table 4) and CADEC (Table 5) datasets. The results of GPT-4 are obtained in a zero-shot setting, while the results of other baselines are taken directly from the original papers. However, some of the baselines did not conduct experiments on all four datasets. Our proposed model outperforms baseline models, in most cases, demonstrating its effectiveness in ADE detection from texts in various domains (RQ1). It is worth noting that while GPT-4 shows robust natural language understanding capabilities, its zero-shot performance on this task falls short of the effectiveness demonstrated by our fine-tuned knowledge-augmented model.

We observed that GPT-4 exhibits a tendency to erroneously identify the presence of ADEs in this task, resulting in a notably low recall value. Our model achieves a more balanced precision-recall trade-off, leading to higher F1 scores. Table 5 shows that our model consistently outperforms the baselines. Our proposed model can capture rich features to identify a document containing ADEs.

5.2. Usefulness of the Knowledge Augmented Graph Convolution (RQ2)

Here, we investigate in more detail how the heterogeneous graph convolution with knowledge augmentation can help with ADE detection (RQ2). In Table 3, most models such as HTR-MSA, IAN, CNN-T and ATL perform worse on TwiMed-Twitter dataset, showing that it is difficult to process informal tweets with colloquial language. However, the graph-based encoder in our model helps in effectively encoding information from the informal text, resulting in a better ability to capture the relationships between different entities, improving performance in most cases. Table 4 compares our model with several pretrained BERT-based models. Our model differs from the pretrained models by additionally employing the GNN architectures in addition to the pretrained embeddings, and the results suggest

Table 3: Results for two TwiMed datasets, i.e., TwiMed-Pub and TwiMed-Twitter. Scores are reported with the mean of 10-fold cross validation following the setup of baselines for our approach. The results of customized models are from the corresponding publications. Results of GPT-4 are derived in a zero-shot setting. **Bold** text indicates the best performance.

Models	TwiMed-Pub			TwiMed-Twitter		
	P (%)	R (%)	F1 (%)	P (%)	R (%)	F1 (%)
HTR-MSA (Wu et al., 2018)	75.0	66.0	70.2	60.7	61.7	61.2
IAN (Alimova and Solovyev, 2018)	87.8	73.8	79.2	83.6	81.3	82.4
CNN-T (Li et al., 2020)	81.3	63.9	71.6	61.8	60.0	60.9
MSAM (Zhang et al., 2019)	85.8	85.2	85.3	74.8	85.6	79.9
ATL (Li et al., 2020)	81.5	67.0	73.4	63.7	63.4	63.5
CGEM (Gao et al., 2022)	88.4	85.0	86.7	84.2	83.7	83.9
GPT-4 (OpenAI, 2023)	89.2	85.4	87.0	76.1	85.3	80.1
KnowCAGE (GCN)	88.8	85.8	87.3	84.1	84.0	84.0
KnowCAGE (GAT)	89.6	83.4	86.4	84.8	84.1	84.4
KnowCAGE (DGCNN)	88.7	83.7	86.1	83.5	84.1	83.8

Table 4: Results of SMM4H dataset. For our approach, scores are reported for the best-performing results, which follows the setup of baselines. Results from GPT-4 are obtained from a zero-shot setting, while results of other baselines are from the corresponding publications. **Bold** text indicates the best performance.

Models	P (%)	R (%)	F1 (%)
BERTweet-LSTM (Kayastha et al., 2021)	81.2	86.2	83.6
RoBERTa-aug (Pimpalkhute et al., 2021)	82.1	85.7	84.3
BERT-LSTM (Yaseen and Langer, 2021)	77.0	72.0	74.0
CGEM (Gao et al., 2022)	86.7	93.4	89.9
GPT-4 (OpenAI, 2023)	62.4	96.7	75.9
KnowCAGE (GCN)	85.3	95.9	90.3
KnowCAGE (GAT)	85.4	94.6	89.8
KnowCAGE (DGCNN)	87.2	97.0	91.8

Table 5: Results for CADEC dataset. For our approach, scores are reported with the mean of 10-fold cross validation following the setup of baselines. Results of GPT-4 are obtained in a zero-shot setting, while results of other baselines are from the corresponding publications. **Bold** text indicates the best performance.

Models	P (%)	R (%)	F1 (%)
HTR-MSA (Wu et al., 2018)	81.8	77.6	79.7
CNN-T (Li et al., 2020)	84.8	79.4	82.0
ATL (Li et al., 2020)	84.3	81.3	82.8
ANNSA (Zhang et al., 2021)	82.7	83.5	83.1
GPT-4 (OpenAI, 2023)	68.6	83.0	75.1
KnowCAGE (GCN)	86.6	90.8	88.7
KnowCAGE (GAT)	87.1	89.7	88.4
KnowCAGE (DGCNN)	87.1	93.9	90.4

the GNN can further improve models' performance on this task. Compared with another graph-based model, CGEM, our method applies knowledge augmentation to incorporate concept information into graph learning, which is also seen to improve the performance of ADE detection in most cases.

During the graph construction process, we apply four different computational methods, L1 distance,

L2 distance, Cosine distance, and Pointwise Mutual Information (PMI), to derive weights of word-word edges, word-concept edges, and concept-concept edges. These methods measure the association between words from various perspectives. PMI focuses on the co-occurrence of each word pair, while others consider the words' semantic distances. Figure 3 shows the effects of these measurements. For TwiMed-Pub and TwiMed-Twitter, the utilization of PMI when graph construction yields the best performance. It shows that the incorporation of word co-occurrence information achieves better graph representation for these two datasets. In contrast, distance-based computational methods of edge weights produce the optimal results for datasets with more training samples, such as SMM4H and CADEC datasets, suggesting that the choice of measurement should be conditional upon the specific characteristics of the datasets.

Additionally, we examine three graph architectures to study which one is most suitable for the ADE detection task. Similar to the selection of edge weights, the sample size of the dataset affects the choice of graph architectures. For datasets containing more training samples (i.e., SMM4H and CADEC datasets), DGCNN performs better. When the number of training samples is small, GCN and GAT achieve better performance. Hence, we conclude that the graph-based encoding method improves the performance. However, we also notice that none of the examined graph architectures consistently outperforms the others on all three datasets from different domains.

5.3. Effectiveness of the Concept-Aware Attention (RQ3)

We examine the effectiveness of concept-aware attention by comparing it with two other attention mechanisms, i.e., simple dot-product attention

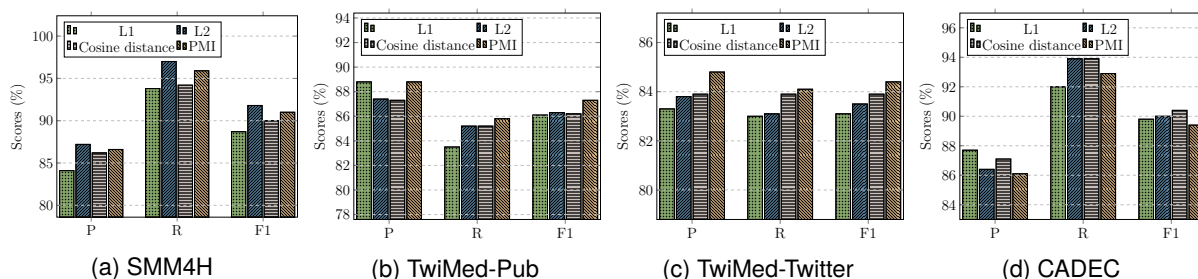


Figure 3: The effect of different computational methods for weights of edges.

(Gao et al., 2022) and structured self-attention (Lin et al., 2017). Table 6 shows that concept-aware attention consistently achieves the best F1 score on four datasets. The concept-aware attention distinguishes different types of nodes from the heterogeneous graph and makes the overall model better utilize the knowledge augmentation from the ULMS.

5.4. Case Study

By adopting Concept-Aware attention, crucial words and concepts can be effectively highlighted, which ultimately improves the classification performance of the model. Meanwhile, this attention mechanism also improves the model’s explainability, as it provides explicit insight into the importance of words and concepts through their corresponding attention scores. In this section, we present a case study to demonstrate the effect of Concept-Aware Attention. We select one positive sample and one negative sample from the SMM4H dataset, respectively. They are both accurately classified by the proposed model, where Concept-Aware Attention allows important words and concepts to be highlighted.

We visualize it by employing the node cloud. In Figure 4, distinct colors distinguish word nodes from concept nodes. Node sizes correspond to their contribution when classifying the document; thus, nodes that are more important to the document will be shown in larger sizes. We set a threshold to ensure that only nodes of sufficient importance will be in the figure.

Figure 4a presents the node cloud for the document “**debating on taking a trazodone and literally passing out for the day**”. The document is a positive sample containing the implications of the adverse effects. The figure’s blue-shaded word nodes can be grouped into five categories: words containing the meaning of “side effects”; names of medicines for depression, adverse effects; depression symptoms; and others. The model learns enough knowledge related to the symptoms and side effects of a certain disease. Moreover, the integration of concept-aware attention enhances

the model’s focus on concepts indicating adverse effects. Red concept nodes shown in the figure are identified to be closely related to the document by the model. They are centered around side effects and depression, emphasizing that the incorporation of concept-aware attention facilitates a better understanding of concept-level information by the model.

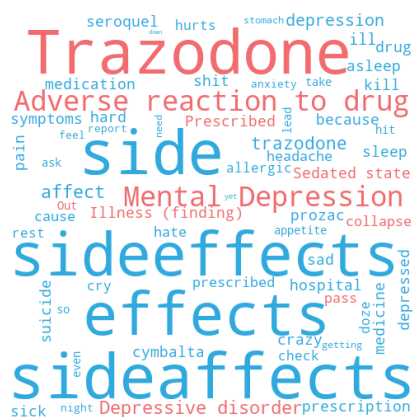
The node cloud for the negative document “**i don’t get why they make stuff like cymbalta to decrease the suicide rate, when there are bunch of precautions about how it may kill you.**” is visualized in Figure 4b. Negative documents are more likely to be misclassified in this task than positive documents. Because negative documents often contain: (a) descriptions of disease symptoms, which are easily mistaken for adverse drug effects; (b) negative sentiments towards drugs, without descriptions of adverse effects. The example in Figure 4b corresponds to case (b). However, with the incorporation of concept-aware attention, the model focuses more on the word nodes and concept nodes that represent the meaning of the document itself without mistakenly associating information related to adverse effects. Therefore, contrary to the example in Figure 4a, most nodes shown in Figure 4b lack a direct relation to “adverse effects”. Instead, these nodes mainly correspond to symptoms of depression and the term “suicide”.

6. Conclusion

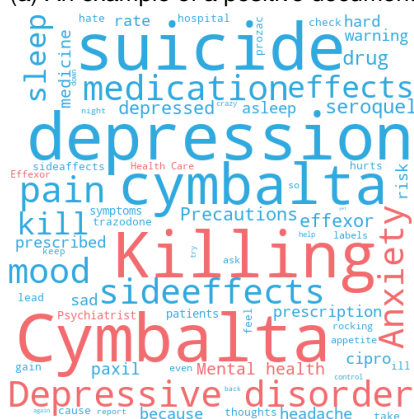
The automated detection of adverse drug events from social media content or biomedical literature requires the model to encode text information and capture the relation between drugs and adverse effects efficiently. This paper utilizes knowledge-augmented contextualized graph embeddings to learn contextual information and capture relations for ADE detection. We equip different graph convolutional networks with pretrained language representations over the knowledge-augmented heterogeneous text graph and develop concept-aware attention to optimally process the different types of nodes in the graph. By comparing our model with other baseline methods, experimental results show

Table 6: Comparison on the choices of attention mechanisms

Datasets	dot-product attention			structured attention			concept-aware attention		
	P (%)	R (%)	F1 (%)	P (%)	R (%)	F1 (%)	P (%)	R (%)	F1 (%)
SMM4H	86.0	94.2	89.9	85.0	95.4	89.3	89.9	97.0	91.8
Twimed-Pub	87.9	84.5	86.2	88.9	82.9	85.8	88.8	85.8	87.3
Twimed-Twitter	84.5	82.2	83.4	83.0	81.8	82.4	84.8	84.1	84.4
CADEC	86.7	89.1	87.9	84.3	88.2	86.2	87.1	93.9	90.4



(a) An example of a positive document



(b) An example of a negative document

Figure 4: A visualization of the concept-aware attention in the form of node cloud

that graph-based embeddings incorporating concept information from the UMLS can inject medical knowledge into the model and the concept-aware attention can learn richer concept-aware representations, leading to better detection performance.

7. Acknowledgements

We acknowledge the computational resources provided by the Aalto Science-IT project and CSC - IT Center for Science, Finland. This work was supported by the Research Council of Finland (Flagship programme: Finnish Center for Artificial Intelligence FCAI, and grants 336033, 352986, 358246) and EU (H2020 grant 101016775 and NextGener-

ationEU). We thank Volker Tresp, Zhen Han, Ruotong Liao, and Zhiliang Wu for their valuable discussions.

8. Bibliographical References

- Ilseay Alimova and Valery Solovyev. 2018. Interactive attention network for adverse drug reaction classification. In *Conference on Artificial Intelligence and Natural Language*, pages 185–196. Springer.
- Nestor Alvaro, Yusuke Miyao, and Nigel Collier. 2017. Twimed: Twitter and PubMed comparable corpus of drugs, diseases, symptoms, and their relations. *JMIR Public Health and Surveillance*, 3(2):e6396.
- Olivier Bodenreider. 2004. *The Unified Medical Language System (UMLS): Integrating Biomedical Terminology*. *Nucleic Acids Research*, 32:D267–70.
- Danushka Bollegala, Richard Sloane, Simon Maskell, Joanna Hajne, and Munir Pirmohamed. 2018. Learning causality patterns for detecting adverse drug reactions from social media. *Journal of Medical Internet Research*.
- Anne Cocos, Alexander G Fiks, and Aaron J Masino. 2017. Deep learning for pharmacovigilance: recurrent neural network architectures for labeling adverse drug reactions in twitter posts. *Journal of the American Medical Informatics Association*, 24(4):813–821.
- Bharath Dandala, Venkata Joopudi, and Murthy Devarakonda. 2019. Adverse drug events detection in clinical notes by jointly modeling entities and relations using neural networks. *Drug Safety*, 42(1):135–146.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*.
- Molla S Donaldson, Janet M Corrigan, Linda T Kohn, et al. 2000. To err is human: building a safer health system.

- Ya Gao, Shaoxiong Ji, Tongxuan Zhang, Prayag Tiwari, and Pekka Marttinen. 2022. Contextualized graph embeddings for adverse drug event detection. In *Proceedings of ECML-PKDD*.
- Suyu Ge, Tao Qi, Chuhan Wu, and Yongfeng Huang. 2019. Detecting and extracting of adverse drug reaction mentioning tweets with multi-head self attention. In *Proceedings of the Fourth Social Media Mining for Health (# SMM4H) Workshop and Shared Task*, pages 96–98.
- Trung Huynh, Yulan He, Alistair Willis, and Stefan R ger. 2016. Adverse drug reaction classification with deep neural networks. In *COLING*.
- Shaoxiong Ji, Shirui Pan, Erik Cambria, Pekka Marttinen, and Philip S Yu. 2022. A survey on knowledge graphs: Representation, acquisition and applications. *IEEE Transactions on Neural Networks and Learning Systems*, 33.
- Pratik Joshi, V Masilamani, and Anirban Mukherjee. 2022. A knowledge graph embedding based approach to predict the adverse drug reactions using a deep neural network. *Journal of Biomedical Informatics*, page 104122.
- Sarvnaz Karimi, Alejandro Metke-Jimenez, Madonna Kemp, and Chen Wang. 2015. Cadec: A corpus of adverse drug event annotations. *Journal of Biomedical Informatics*, 55:73–81.
- Tanay Kayastha, Pranjal Gupta, and Pushpak Bhat-tacharyya. 2021. BERT based Adverse Drug Effect Tweet Classification. In *Proceedings of the Sixth Social Media Mining for Health (# SMM4H) Workshop and Shared Task*, pages 88–90.
- Diederik P Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *ICLR*.
- Thomas N Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations*.
- Heeyoung Kwak, Minwoo Lee, Seunghyun Yoon, Jooyoung Chang, Sangmin Park, and Kyomin Jung. 2020. Drug-disease graph: predicting adverse drug reaction signals via graph neural network with clinical data. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 633–644. Springer.
- Hui Li, Xiao-Jing Guo, Xiao-Fei Ye, Hong Jiang, Wen-Min Du, Jin-Fang Xu, Xin-Ji Zhang, and Jia He. 2014. Adverse drug reactions of spontaneous reports in shanghai pediatric population. *PLoS One*, 9(2):e89829.
- Zhiheng Li, Zhihao Yang, Ling Luo, Yang Xiang, and Hongfei Lin. 2020. Exploiting adversarial transfer learning for adverse drug reaction detection from texts. *Journal of Biomedical Informatics*, 106:103431.
- Bin Liang, Hang Su, Lin Gui, Erik Cambria, and Ruifeng Xu. 2022. Aspect-based sentiment analysis via affective knowledge enhanced graph convolutional networks. *Knowledge-Based Systems*, 235:107643.
- Yuxiao Lin, Yuxian Meng, Xiaofei Sun, Qinghong Han, Kun Kuang, Jiwei Li, and Fei Wu. 2021. BertGCN: Transductive Text Classification by Combining GCN and BERT. *arXiv preprint arXiv:2105.05727*.
- Zhouhan Lin, Minwei Feng, Cicero Nogueira dos Santos, Mo Yu, Bing Xiang, Bowen Zhou, and Yoshua Bengio. 2017. A structured self-attentive sentence embedding. In *International Conference on Learning Representations*.
- Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.
- Yuan Luo. 2017. Recurrent neural networks for classifying relations in clinical notes. *Journal of Biomedical Informatics*, 72:85–95.
- Yukun Ma, Haiyun Peng, and Erik Cambria. 2018. Targeted aspect-based sentiment analysis via embedding commonsense knowledge into an attentive lstm. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.
- Arjun Magge, Ari Klein, Antonio Miranda-Escalada, Mohammed Ali Al-Garadi, Ilseyar Alimova, Zulfat Miftahutdinov, Eulalia Farre, Salvador Lima-L pez, Ivan Flores, Karen O’Connor, et al. 2021. Overview of the sixth social media mining for health applications (# SMM4H) shared tasks at NAACL 2021. In *Proceedings of the Sixth Social Media Mining for Health (# SMM4H) Workshop and Shared Task*, pages 21–32.
- Dat Quoc Nguyen, Thanh Vu, and Anh Tuan Nguyen. 2020. Bertweet: A pre-trained language model for english tweets. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 9–14.
- Azadeh Nikfarjam, Abeed Sarker, Karen O’connor, Rachel Ginn, and Graciela Gonzalez. 2015. Pharmacovigilance from social media: mining adverse drug reaction mentions using sequence

- labeling with word embedding cluster features. *Journal of the American Medical Informatics Association*, 22(3):671–681.
- R OpenAI. 2023. Gpt-4 technical report. *ArXiv*, 2303.
- Endang Wahyu Pamungkas, Valerio Basile, and Viviana Patti. 2021. A joint learning approach with knowledge injection for zero-shot cross-lingual hate speech detection. *Information Processing & Management*, 58(4):102544.
- Varad Pimpalkhute, Prajwal Nakhate, and Tausif Diwan. 2021. IIITN NLP at SMM4H 2021 Tasks: Transformer Models for Classification on Health-Related Imbalanced Twitter Datasets. In *Proceedings of the Sixth Social Media Mining for Health (# SMM4H) Workshop and Shared Task*, pages 118–122.
- Soujanya Poria, Alexandar Gelbukh, Basant Agarwal, Erik Cambria, and Newton Howard. 2013. Common sense knowledge based personality recognition from text. In *Mexican International Conference on Artificial Intelligence*, pages 484–496. Springer.
- Chen Shen, Zhiheng Li, Yonghe Chu, and Zhongying Zhao. 2021. GAR: Graph adversarial representation for adverse drug event detection on Twitter. *Applied Soft Computing*, 106:107324.
- Chen Shen, Hongfei Lin, Zhiheng Li, Yonghe Chu, Zhengguang Li, and Zhihao Yang. 2020. A graph-boosted framework for adverse drug event detection on twitter. In *2020 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 1129–1131. IEEE.
- Binbin Shi, Lijuan Zhang, Jie Huang, Huilin Zheng, Jian Wan, and Lei Zhang. 2022. Mda: An intelligent medical data augmentation scheme based on medical knowledge graph for chinese medical tasks. *Applied Sciences*, 12(20):10655.
- Sunghwan Sohn, Cheryl Clark, Scott R Halgrim, Sean P Murphy, Christopher G Chute, and Hongfang Liu. 2014. Medxn: an open source medication extraction and normalization tool for clinical text. *Journal of the American Medical Informatics Association*, 21(5):858–865.
- Janet Sultana, Paola Cutroneo, and Gianluca Trifirò. 2013. Clinical and economic burden of adverse drug reactions. *Journal of Pharmacology & Pharmacotherapeutics*, 4(Suppl1):S73.
- Haixia Sun, Jin Xiao, Wei Zhu, Yilong He, Sheng Zhang, Xiaowei Xu, Li Hou, Jiao Li, Yuan Ni, Guotong Xie, et al. 2020. Medical knowledge graph to enhance fraud, waste, and abuse detection on claim data: model development and performance evaluation. *JMIR Medical Informatics*, 8(7):e17653.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008.
- Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph attention networks. In *International Conference on Learning Representations*.
- Jin Wang, Liang-Chih Yu, and Xuejie Zhang. 2022. Explainable detection of adverse drug reaction with imbalanced data distribution. *PLoS Computational Biology*, 18(6):e1010144.
- Qiang Wei, Zongcheng Ji, Zhiheng Li, Jingcheng Du, Jingqi Wang, Jun Xu, Yang Xiang, Firat Tiryaki, Stephen Wu, Yaoyun Zhang, et al. 2020. A study of deep learning approaches for medication and adverse drug event extraction from clinical text. *Journal of the American Medical Informatics Association*, 27(1):13–21.
- Lesley Wise, John Parkinson, June Raine, and Alasdair Breckenridge. 2009. New approaches to drug safety: a pharmacovigilance tool kit. *Nature Reviews Drug Discovery*, 8(10):779–782.
- Chuhan Wu, Fangzhao Wu, Junxin Liu, Sixing Wu, Yongfeng Huang, and Xing Xie. 2018. Detecting tweets mentioning drug name and adverse drug reaction with hierarchical tweet representation and multi-head self-attention. In *Proceedings of the Third Social Media Mining for Health (# SMM4H) Workshop and Shared Task*, pages 34–37.
- Susmitha Wunnava, Xiao Qin, Tabassum Kakar, Xiangnan Kong, and Elke Rundensteiner. 2020. A dual-attention network for joint named entity recognition and sentence classification of adverse drug events. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, pages 3414–3423.
- Hua Xu, Shane P Stenner, Son Doan, Kevin B Johnson, Lemuel R Waitman, and Joshua C Denny. 2010. Medex: a medication information extraction system for clinical narratives. *Journal of the American Medical Informatics Association*, 17(1):19–24.
- Ikuya Yamada, Akari Asai, Hiroyuki Shindo, Hideaki Takeda, and Yuji Matsumoto. 2020. LUKE: Deep Contextualized Entity Representations with

- Entity-aware Self-attention. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6442–6454.
- Kailai Yang, Tianlin Zhang, and Sophia Ananiadou. 2022. A mental state knowledge-aware and contrastive network for early stress and depression detection on social media. *Information Processing & Management*, 59(4):102961.
- Liang Yao, Chengsheng Mao, and Yuan Luo. 2019. Graph convolutional networks for text classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 7370–7377.
- Usama Yaseen and Stefan Langer. 2021. Neural Text Classification and Stacked Heterogeneous Embeddings for Named Entity Recognition in SMM4H 2021. In *Proceedings of the Sixth Social Media Mining for Health (# SMM4H) Workshop and Shared Task*, pages 83–87.
- Muhan Zhang, Zhicheng Cui, Marion Neumann, and Yixin Chen. 2018. An end-to-end deep learning architecture for graph classification. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32.
- Tongxuan Zhang, Hongfei Lin, Yuqi Ren, Liang Yang, Bo Xu, Zhihao Yang, Jian Wang, and Yijia Zhang. 2019. Adverse drug reaction detection via a multihop self-attention mechanism. *BMC Bioinformatics*, 20(1):1–11.
- Tongxuan Zhang, Hongfei Lin, Bo Xu, Yuqi Ren, Zhihao Yang, Jian Wang, and Xiaodong Duan. 2020a. Gated iterative capsule network for adverse drug reaction detection from social media. In *2020 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 387–390. IEEE.
- Tongxuan Zhang, Hongfei Lin, Bo Xu, Liang Yang, Jian Wang, and Xiaodong Duan. 2021. Adversarial neural network with sentiment-aware attention for detecting adverse drug reactions. *Journal of Biomedical Informatics*, 123:103896.
- Yufeng Zhang, Xueli Yu, Zeyu Cui, Shu Wu, Zhongzhen Wen, and Liang Wang. 2020b. Every document owns its structure: Inductive text classification via graph neural networks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 334–339.