End-to-end drug entity recognition and adverse effect relation extraction via principal neighbourhood aggregation network

Luyue Kong 1, Qinghan Lai 2* and Song Liu 2*

1Department of Medicine, Shandong Medical College, Jinan, Shandong Province, 250002, P.R.China
2School of Computer Science and Technology, Qilu University of Technology (Shandong Academy of Sciences), Jinan, Shandong Province, 250353, P.R.China
*Corresponding author’s e-mail: 1043119228@stu.qlu.edu.cn; liusong@qlu.edu.cn

Abstract. Drug entity and adverse effect relation extraction is a critical task that aims to recognize drug entity and extract adverse effect relation from the unstructured medical text. Many works have used statistical learning approach, traditional or new deep learning approach to solve similar pharmacovigilance problem. Recent works tended to employ the graph convolutional network to enhance the ability of drug and adverse effect information extraction. However, the injective problem generated by a single aggregator and the weak generalization of the summation aggregator cause the graph neural network lack the ability to extract sufficient relation information. To solve these problems, we propose a new end-to-end model named recognizing Drug entities and extracting Adverse effect relations via the Principal Neighbourhood Aggregation network (DAPNA). Moreover, we compared the DAPNA with baseline models on the Adverse Drug Effect (ADE) dataset using multiple metrics. The experimental results demonstrate the method proposed in this paper achieves state-of-the-art results and can be applied to other drug and adverse effect information extraction tasks.

1. Introduction

Adverse drug effects are harmful to patients and can easily lead to illness or death [1], which is an essential problem that doctors try to avoid in their daily medical decision-making. In medical domain, drug entities and adverse drug reactions usually exist in unstructured obscure medical texts, which is inefficient for researchers to manually recognize drug entities and adverse drug effect relations. Therefore, it is of great significance to design a reliable end-to-end model to identify drug entities and extract adverse drug effect relations automatically.

With the rapid development of deep learning, researchers proposed some end-to-end drug entity recognition and adverse drug effect relation extraction models in recent years. Li et al. (2016) proposed an end-to-end drug entity recognition and adverse effect relation extraction model [2] based on Convolutional Neural Networks (CNN) [3]. Using Bidirectional Long Short-Term Memory neural networks (BiLSTM) [4], Bekoulis et al. (2018) put forward to transform the drug entity recognition and adverse effect reaction extraction into a multi-head selection problem [5]. To enhance model robustness, Bekoulis et al. (2018) further introduced Adversarial Training (AT) [6] into the multi-head selection model (MHS) [7]. Although the CNN, BiLSTM, and AT have improved the performance of the models, it is challenging for them to extract complex drug and adverse drug effect information in one simple network structure. Then Tran et al. (2019) introduced neural metric learning (NML) to improve the ability to recognize drug entities and extract adverse effect relations [8]. Moreover, Eberts et al. (2019)
introduced the Bert [9] model as the core module for drug entity recognition and adverse effect relation extraction task [10]. The complexity of the model structure improves the ability to extract drug and adverse drug effect information to some extent, but those models ignore building the specific model for the adverse effect relation information, which is non-Euclidean spatial data different from traditional Euclidean one. Based on this concept, Sun et al. (2019) and Fu et al. (2019) introduced Graph Convolutional Networks (GCN) [11] to extract entity and relation information, respectively [12,13]. Furthermore, to distinguish the aggregated node information, Lai et al. (2020) proposed Entity-Relations via Improved Graph Attention networks (ERIGAT), which introduced the improved graph attention networks (IGAT) [14] to enhance the accuracy of drug entity recognition and adverse effect relation extraction [15]. However, the GCNs currently applied to the domain of entity recognition and relation extraction are all using a single aggregator (the sum aggregator) to aggregate node information, which can easily cause the injective problem, resulting in an insufficient understanding of the distribution of the messages received from the neighbourhood. Additionally, the generation of summation aggregators is inadequate because messages and gradients can amplify or attenuate exponentially with small degree changes.

As mentioned above, we design a new end-to-end drug entity recognition and adverse effect relation extraction model considering the advantages of CNN, LSTM, and GCN in node embedding, semantic extraction, and relation information extraction, respectively. To solve the problem that individual aggregation is prone to produce injectivity, and the summation aggregator can not learn the distribution of messages effectively, we introduce the Principal Neighbourhood Aggregation network (PNA) [16] to improve the entity recognition and relation extraction model. Using multiple aggregators to aggregate neighbourhood information and designing a degree-based information scaler, the PNA can reduce the injective problem of aggregation of neighbourhood information, which can effectively improve the ability to extract drug entity and adverse effect relation information.

The ADE dataset [17] is a standard corpus, which systematically annotates drug entities, adverse effect entities, and adverse effect relations based on case reports. At present, the ADE dataset has become a common dataset in drug entity recognition and adverse effect relation extraction. Accordingly, there are many studies on ADE dataset that provides adequate comparison models.

In the experiments, we tested the DAPNA and designed contrast experiments on ADE dataset using multiple metrics (precision, recall, F1-value, and overall-value). The main contributions of this paper are as follows:

- We designed a new end-to-end drug entity and adverse effect relation extraction model that improves the capability to recognize drug entities and extract adverse effect relations.
- We introduced the PNA into drug entity recognition and adverse effect relation extraction domain for the first time, which significantly improved the effect of drug entity and adverse effect Relation extraction task, and expanded the research in the current medical domain.

2. Model

As shown in Figure 1, given a medical text, the DAPNA recognize drug entities and extract adverse effect relations in an end-to-end manner.

![Figure 1. A marked sentence in ADE.](image)

In this section, we describe the structure and implementation details of various modules of the DAPNA model. The structure of DAPNA is shown in Figure 2. Among the modules, the introduction of PNA is the core innovation. And the rest modules mostly use existing methods.
2.1. Word embedding module

In the word embedding module, we combine the word-level embedding with the char-level embedding to obtain the single vector representation of words, the details of this module is shown in Figure 3 below.

First we employ the pre-trained language model GloVe [18] to obtain a pre-trained word vector, then we use the word2vec model [19] to handle those pre-trained word vectors and obtain word-level embedding of words as usual. But in the drug entity and adverse effect relation extraction task, the char-level information is critical at the same time. For instance, the prefix "hypo" can specify an adverse-effect entity (such as hyponatremia or hypophosphatemia). Accordingly, the character information can complement the semantics of the word-level embedding in a fine-grained manner. Moreover, the CNN is adopted to extract char-level information, which can capture word morphological features. Finally, we concatenate the word-level embedding with char-level embedding to get a uniform word vector representation.

2.2. Entity span recognition module

We use the BIEOU scheme to annotate entity span, where "B," "I," "E," and "O" is the beginning, inside, end, and outside of the entity span, respectively. And the "U" represents a single word entity. Moreover, the CNN is adopted to extract char-level information, which can capture word morphological features. Finally, we concatenate the word-level embedding with char-level embedding to get a uniform word vector representation.
vector representation of $w_i$. We employ the BiLSTM to extract semantic information, and we use Softmax to predict the entity span tags, then we get the target entity span tags $t = \{t_1, t_2, \cdots, t_{|w|}\}$. The entity span loss is calculated as below:

$$L_{\text{span}} = -|W| \sum_{i=1}^{|w|} \log P(t_i = t_i | w).$$  \hfill (1)

2.3. Graph embedding module

To extract drug entity and adverse effect relation information using the PNA, we should first obtain a adjacency matrix of the entity and relation graph from the entities obtained above.

2.3.1. Entity and relation node embedding

We define $e$ and $\hat{e}$ as an entity and entities set in a text, respectively. For example, the $\hat{e}_{\text{exam}} = \{e_1, e_2, e_3\}$ is an entity set with $e_1, e_2, e_3$ according to the predicted entity span tags $\hat{t}$. Then, we employ a Multi-Layer Perceptron (MLP) \cite{20} and a CNN to obtain a single vector representation of entities $N_e \in \mathbb{R}^{d_{\text{node}}}$. Next, assuming a candidate relation between any two entities in a text, we define $r_{jk}$ as the relation between entity $e_j$ and $e_k$. Motivated by the impact of contextual information on predictive relations, we input contextural information (the left and right word information of entity) with each entity span information into another CNN and MLP. Then, the extracted information of entities and context are concatenated to a single vector. Finally, we adopt a MLP to get the $d_{\text{node}}$-dimensional vector $N_{r, jk} \in \mathbb{R}^{d_{\text{node}}}$.

2.3.2. Obtaining adjacency matrix

To obtain the adjacency matrix of the graph, we need to predict the possibility of candidate relation nodes. We define a binary relation tag $b \in \{0,1\}$, where 1 means a relation between the two entities and 0 means no relation between the two entities. The predicted relation tag is formulated as below:

$$P(\hat{b} = b | r_{jk}, w) = \text{softmax}(W_{\text{adj}} N_{r, jk}),$$  \hfill (2)

where $\hat{b}$ is the predicted existence tag of relation $r_{jk}$ and $W_{\text{adj}}$ is the weight matrix. Then, we can calculate the relation existence loss value as follows:

$$L_{\text{rel}} = - \sum_{r_{jk}} \log P(\hat{b} = b | r_{jk}, w) \text{ # candidate relation } r_{jk}. \hfill (3)$$

Finally, we define the following rules to get the adjacent matrix $A$:

- If $P(\hat{b} = 1 | r_{jk}, w) \geq 0.5$, we assume that $N_{e_j}$ and $N_{e_k}$ have a edge with $N_{r, jk}$, respectively, i.e. the corresponding position element in $A$ is 1.
- As a common method, we add a self-loop to the graph, i.e. the diagonal element in $A$ is 1.
- All the remaining locations are set to 0.

2.4. Principal neighbourhood aggregation network

The traditional graph neural network approach uses a single aggregator to aggregate neighbourhood information (such as GCN and IGAT), which is prone to produce injective problems, resulting in the loss of specific information of neighbourhood and inefficiency in extracting the relation between drug
entity and adverse effect. Moreover, most graph neural network approach uses summation aggregators, where a small degree change may cause the gradient and aggregated information to be amplified or attenuated exponentially, which weakens the generalization ability of the GCNs.

To solve these problems, we introduce the PNA into our proposed model. When aggregating neighbourhood information, the PNA adopt four aggregators (Mean, maximum, minimum, and standard deviation aggregation) that can guarantee that at least one aggregator works, which greatly alleviates the injective problem. Moreover, the PNA model combines multiple aggregators with degree-scalers to improve the generalization of summation aggregator. Assuming we set $d$ as the degree of node and $\gamma$ as a variable parameter, the degree-scaler is formulated as follows:

$$S(d, \gamma) = \left( \frac{\log(d+1)}{\delta} \right)^\gamma, -1 \leq \gamma \leq 1, d > 0$$

(4)

where $\delta = |\text{train}| \sum_{i \in \text{train}} \log(d_i + 1)$ and $S(d, \gamma)$ is an injective function for $d > 0$. As shown in Figure 4, the scalers are combined with multiple aggregators to aggregate neighbourhood information, which is formulated as follows:

$$\oplus = \left[ \begin{array}{c} S(D, \gamma = 1) \\ S(D, \gamma = -1) \end{array} \right] \otimes \left[ \begin{array}{c} \text{mean} \\ \text{max} \\ \text{min} \\ \text{std} \end{array} \right]$$

(5)

where $I$ means no scaling and $\otimes$ is the tensor product. Additionally, we define $\text{Edge}$ as the edge of two nodes in graph and $I$ as the current layer in the PNA. Accordingly, the PNA can be formulated as follows:

$$N_i^{(l+1)} = \text{MLP}\left( N_i^{(l)} \oplus \text{MLP} \left( N_i^{(l)}, N_{ij}^{(l)} \right) \right).$$

(6)

### 2.5. Entity and relation prediction

The final feature vector $F$ is employed to calculate classification probability, which is obtained by concatenating the input and output of the PNA. The predicted entity and relation tags can be formulated as follows:

$$P(\hat{y_e} | e_j, w) = \text{softmax}(W_{ew} F_{ej}),$$

(7)
\[ P(\hat{y}_e | r_{jk}, w) = \text{softmax}(W_{rel}F_{r_{jk}}), \] (8)

where \( \hat{y}_e \) and \( \hat{y}_r \) are the predicted entity tag and relation tag with the corresponding weight matrix \( W_{ent} \) and \( W_{rel} \), respectively. Then, we can calculate the entity and relation loss functions as follows:

\[ L_{eT} = -|e|^{-1} \sum_{e_j \in e} \log P(\hat{y}_e = y_e | e_j, w), \] (9)

\[ L_{rT} = -\sum_{r_{jk} \in \# \text{candidate relation } r_{jk}} \log P(\hat{y}_r = y_r | r_{jk}, w). \] (10)

Finally, the DAPNA model loss (\( L_{model} = L_{span} + L_{rel} + L_{eT} + L_{rT} \)) is defined to update model parameters through backpropagation.

2.6. Implementation details
In the DAPNA model, we set the char-level and word-level embedding dimension as 50 and 100, respectively. Then, a 1-layer BiLSTM is employed to extract information with hidden layer dimensions of 128. Afterwards, to embed entity and relation nodes, we adopt a CNN with two kernels (the size of the kernels are 2 and 3) and set the output channels as 25. Besides, the output dimension of MLP is 128. Next, the 2-layers PNA is leveraged in our model with the hidden dimensions of 128. Finally, adopting the AdaDelta [21], we optimize the parameters with the learning rate of 0.3.

3. Experiments

3.1. Datasets and data preprocessing
We conducted experiments on the ADE dataset, which includes 6821 sentences. However, many sentences are repeated, and there are a few overlapping entities redundant for the experiments. Therefore, we unified the entities and relations in repeated sentences into one unique sentence, and removed 130 pairs of overlapping entities. After data preprocessing, the ADE contains 4271 sentences, 10652 entities (entity types are "Adverse_effects" and "Drugs"), and 6682 relations (relation type is "Effect"). Similar to Bekoulis et al. (2018) [5], we divided the ADE dataset with 3417 for training, 427 for validation, and 427 for testing.

3.2. Evaluation metrics and criteria
In our experiments, the "Strict" standard method was employed to evaluate our proposed model. Specifically, only when both the entity span and the entity type were correct can we determine that the predicted entity was correct. Moreover, the predicted relation was judged as correct only if both the entity span and the relation type were correct.

As for the metrics, the recall (R), precision (P), and F1-value were adopted to evaluate the model results. Additionally, we defined Overall-score as the average of entity and relation F1-scores, which can comprehensively evaluate the results of the models.

3.3. Baseline models
We compared DAPNA with several baseline models for drug entity recognition and adverse effect relation extraction, namely CNN-based [2], MHS [5], MHS-AT [7], NML-based [8], Bert-based [10], GCN-based [12], and ERIGAT [15], which are named based on their model architectures.

3.4. Results and analysis
Table 1 listed the results of the DAPNA model and the baseline models on the ADE dataset.
Compared with the models (CNN-based, MHS, and MHS-AT) using a single network, the DAPNA significantly improved the accuracy of drug entity recognition and adverse effect relation by approximately 3~13% in the Overall-score.

Table 1. The results of the DAPNA and baseline models on ADE dataset.

| Models     | Named Entity Recognition |  | Relation Extraction |  | Overall |
|------------|--------------------------|--|---------------------|--|---------|
|            | P  | R   | F1  | P  | R   | F1  |       |
| CNN-based  | 79.50 | 79.60 | 79.55 | 64.00 | 62.90 | 64.35 | 71.55 |
| MHS        | -   | -   | 86.40 | -   | -   | 74.58 | 80.49 |
| MHS-AT     | -   | -   | 86.73 | -   | -   | 75.52 | 81.13 |
| NML-based  | 86.16 | 88.08 | 87.11 | 77.36 | 77.25 | 77.29 | 82.20 |
| Bert-based | 89.26 | 89.26 | 89.25 | 78.09 | 80.43 | 79.24 | 84.25 |
| GCN-based  | 87.32 | 81.93 | 84.54 | 82.39 | 75.44 | 78.76 | 81.65 |
| ERIGAT     | 90.73 | 85.92 | 88.27 | 84.81 | 75.86 | 80.09 | 84.17 |
| DAPNA      | **90.81** | 86.15 | 88.42 | **84.97** | 76.13 | **80.31** | **84.37** |

Afterwards, although the NML-based improved the performance by introducing the neural metric learning, there is still about 2% difference in the Overall-score compared with the DAPNA.

Moreover, the Bert-based model containing huge semantic information reaches the highest value in entity recognition recall, entity recognition F1-value, and relation extraction recall, respectively. However, we can found that the DAPNA still has a slight advantage in Overall-score, which reflects that the DAPNA can still effectively extract drug and adverse effect information by using less semantic information.

Additionally, compared with the GCN-based and ERIGAT models, the DAPNA still achieves the best values in entity recognition precision (90.81%), relation extraction precision (84.97%), relation extraction F1-value (80.31%), and Overall-score (84.37%), respectively.

On the whole, the DAPNA has achieved better overall performance compared with other baseline models because the DAPNA reduces the injective problem by adopting multiple aggregators and degree-based scaler, which can learn the specificity message distribution of neighbourhood nodes and thus strengthen the performance in extracting drug entity and adverse effect relation.

4. Conclusion

In this paper, we proposed a novel end-to-end entity recognition and relation extraction model introducing the PNA into the entity and relation extraction for the first time, and applied in drug entity and adverse effect relation domain for the first time too. Using multiple aggregators and degree-based scalers, we reduce the injective problem in aggregating neighbourhood node message and strengthen the generalization ability, which improves the ability of the DAPNA to extract drug and adverse effect information. Compared with the baseline models, the DAPNA has achieved the best result in several metrics and high performance on the ADE dataset. However, the structure information of drug and adverse effect graph was ignored in this paper, which is an important issue in our future research.

Acknowledgments

This work was supported by the Science and Technology Development Plan for Higher Education in the Shandong Province under Grant J18KA360 and the Key Research and Development Program in Shandong Province under Grant 2019GGX105010.

References

[1] Pirmohamed M, James S, Meakin S, et al. Adverse drug reactions as cause of admission to
hospital: prospective analysis of 18 820 patients[J]. Bmj, 2004, 329(7456): 15-19.

[2] Li F, Zhang Y, Zhang M, et al. Joint Models for Extracting Adverse Drug Events from Biomedical Text[C]//IJCAI. 2016, 2016: 2838-2844.

[3] Krizhevsky A, Sutskever I, Hinton G E. Imagenet classification with deep convolutional neural networks[J]. Advances in neural information processing systems, 2012, 25: 1097-1105.

[4] Huang Z, Xu W, Yu K. Bidirectional LSTM-CRF models for sequence tagging[J]. arXiv preprint arXiv:1508.01991, 2015.

[5] Bekoulis G, Deleu J, Demeester T, et al. Joint entity recognition and relation extraction as a multi-head selection problem[J]. Expert Systems with Applications, 2018, 114: 34-45.

[6] Goodfellow I J, Shlens J, Szegedy C. Explaining and harnessing adversarial examples[J]. arXiv preprint arXiv:1412.6572, 2014.

[7] Bekoulis G, Deleu J, Demeester T, et al. Adversarial training for multi-context joint entity and relation extraction[C]//Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. 2018: 2830-2836.

[8] Tran T, Kavuluru R. Neural metric learning for fast end-to-end relation extraction[J]. arXiv preprint arXiv:1905.07458, 2019.

[9] Devlin J, Chang M W, Lee K, et al. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding[C]//Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers). 2019: 4171-4186.

[10] Eberts M, Ulges A. Span-based joint entity and relation extraction with transformer pre-training[J]. arXiv preprint arXiv:1909.07755, 2019.

[11] Kipf T N, Welling M. Semi-supervised classification with graph convolutional networks[J]. arXiv preprint arXiv:1609.02907, 2016.

[12] Sun C, Gong Y, Wu Y, et al. Joint type inference on entities and relations via graph convolutional networks[C]//Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. 2019: 1361-1370.

[13] Fu T J, Li P H, Ma W Y. GraphRel: Modeling text as relational graphs for joint entity and relation extraction[C]//Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. 2019: 1409-1418.

[14] Veličković P, Cucurull G, Casanova A, et al. Graph attention networks[J]. arXiv preprint arXiv:1710.10903, 2017.

[15] Lai Q, Zhou Z, Liu S. Joint Entity-Relation Extraction via Improved Graph Attention Networks[J]. Symmetry, 2020, 12(10): 1746.

[16] Corso G, Cavalleri L, Beaini D, et al. Principal Neighbourhood Aggregation for Graph Nets[J]. Advances in Neural Information Processing Systems, 2020, 33.

[17] Gurulingappa H, Rajput A M, Roberts A, et al. Development of a benchmark corpus to support the automatic extraction of drug-related adverse effects from medical case reports[J]. Journal of biomedical informatics, 2012, 45(5): 885-892.

[18] Pennington J, Socher R, Manning C D. Glove: Global vectors for word representation[C]//Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP). 2014: 1532-1543.

[19] Mikolov T, Chen K, Corrado G, et al. Efficient estimation of word representations in vector space[J]. arXiv preprint arXiv:1301.3781, 2013.

[20] Wang W, Chau K, Qiu L, et al. Improving forecasting accuracy of medium and long-term runoff using artificial neural network based on EEMD decomposition[J]. Environmental research, 2015, 139: 46-54.

[21] Zeiler M D. AdaDelta: an adaptive learning rate method[J]. arXiv preprint arXiv:1212.5701, 2012.