Entity relation extraction in the medical domain: based on data augmentation

Anli Wang[1], Linyi Li[2,3], Xuehong Wu[2,3], Jianping Zhu[3], Shanshan Yu[1], Xi Chen[4], Jianhua Li[3], Hongtao Zhu[4]

[1]Information Center, The Third Xiangya Hospital, Central South University, Changsha, China; [2]School of Computer Science, Central South University, Changsha, China; [3]Hunan Creator Information Technology Co., Ltd., Changsha, China; [4]Information and Network Center, The Second Xiangya Hospital, Central South University, Changsha, China

Contributions: (I) Conception and design: A Wang; (II) Administrative support: A Wang; (III) Provision of study materials or patients: A Wang, H Zhu; (IV) Collection and assembly of data: L Li, J Zhu, X Wu; (V) Data analysis and interpretation: L Li, S Yu, X Chen, J Li, X Wu; (VI) Manuscript writing: All authors; (VII) Final approval of manuscript: All authors.

Correspondence to: Hongtao Zhu. Information and Network Center, The Second Xiangya Hospital, Central South University, No. 139 Renmin Middle Road, Changsha 410000, China. Email: 25266335@qq.com.

Background: Entity relation extraction is an important task in the construction of professional knowledge graphs in the medical field. Research on entity relation extraction for academic books in the medical field has revealed that there is a great difference in the number of different entity relations, which has led to the formation of a typical unbalanced data set that is difficult to recognize but has certain research value.

Methods: In this article, we propose a new entity relation extraction method based on data augmentation. According to the distribution of individual entity relation classes in the data set, the probability of whether a text is augmented during training was calculated. In text-oriented data augmentation, different augmentation methods perform differently in different language environments. The reinforcement of learning determines which data augmentation method to use in the current language environment. This strategy was applied to the entity relation extraction of the medical professional book, *Pharmacopoeia of the People's Republic of China*, and different data augmentation methods (i.e., no data augmentation, traditional data augmentation, and reinforcement learning-based data augmentation) were compared under the same neural network model.

Results: The deep-learning model using data augmentation was better than the model without data augmentation, as data augmentation significantly improved the evaluation indicators of the relation classes with low data volumes in the unbalanced data set and slightly improved the evaluation indicators of the relation classes with sufficient features and large data volumes. Additionally, the deep-learning model using reinforcement learning-based data augmentation was superior to the deep-learning model using traditional data augmentation. We found that after the application of reinforcement learning-based data augmentation, the evaluation indicators of the multiple relation classes were much better than those to which reinforcement learning-based data augmentation had not been applied.

Conclusions: For unbalanced data sets, data augmentation can effectively improve the ability of the deep-learning model to obtain data features, and reinforcement learning-based data augmentation can further enhance this ability. Our experiments confirmed the superiority of reinforcement learning-based data augmentation.

Keywords: Data augmentation; medical entity and relation extraction; reinforcement learning; unbalanced data set

Introduction

Entity relation extraction is an important task in natural language processing (NLP). Its main purpose is to determine the semantic relationship between entity pairs in unstructured text and provide data for the establishment of professional knowledge graphs. Academic books are references that are carefully written by professionals, with high knowledge integrity, and can be regarded as an unstructured professional knowledge base. When relation extraction is performed on unstructured knowledge bases, such as academic books, a knowledge atlas with relatively integrated knowledge can be obtained, which enables knowledge collection to be skipped. However, it also gives rise to a problem, as the number of entity relations in the knowledge base is unbalanced. When a knowledge base is regarded as a data set, this imbalance becomes a characteristic of the data set, making it an unbalanced data set.

Data set imbalance is a common problem in machine learning, for which there are 3 solutions (i.e., merging or removing rare entity relations, downsampling, and upsampling). If the scarce entity relation is merged or removed, the content of the final knowledge base will be incomplete; if the downsampling method is adopted, while the internal class of the data set can be balanced, it may fail to obtain rich features of a large number of entity relations that originally had rich features. Thus, upsampling becomes the only option, and data augmentation is an effective upsampling method.

In this study, we examined how to simultaneously use a variety of simple data augmentation methods to process text and how to use reinforcement learning to determine the specific data augmentation method to be used in the current language environment based on the preferences of different data augmentation methods in different language environments. Finally, a data augmentation method based on reinforcement learning was proposed to manage the data set in an unbalanced state, and entity relation extraction was performed after the internal classes were balanced.

Relevant work

There are 2 main entity relation extraction techniques; that is, template rule-based methods and eigenvector-based methods. Under the template rule-based method, the language features of entity relations are first organized by linguists, after which the corresponding entity relation rules are compiled, and finally, the entity relations are extracted through rule-based matching (1-5). The eigenvector-based methods can be divided into the following two types: traditional machine learning and deep learning. After constructing the eigenvectors, the former uses traditional machine-learning models, such as the maximum entropy classifier (6,7) or support vector machine (SVM) (8), to complete the construction of the entity relation extraction model, whose performance depends mainly on the selection or construction of eigenvectors. Machine learning is relatively simple in eigenvector selection and often only needs to embed characters (words) into a specific vector space to generate the corresponding eigenvector—word vector (9). The main factor affecting the performance of the model is the specific neural network used.

A convolutional neural network (CNN) can fully obtain language features in word vectors (10-12). However, due to the characteristics of convolution, long-distance dependency may occur, such that distant words are difficult to relate. To address this issue, a recurrent neural network (RNN) (13) has been proposed, in which each neuron is connected end-to-end; that is, the input unit of 1 neuron is the output unit of the previous neuron, and its output unit is the input unit of the next neuron. This structural design enables the establishment of a corresponding relationship between two words even if they are far away. However, as the operation process inside the neural unit is mainly performed by multiplication, gradient explosion or disappearance is unavoidable. Accordingly, the transformer neural network (TNN) (14) has been adopted. Through mechanisms, such as self-attention and multi-head self-attention, TNN not only relates two words a long distance apart but also avoids the exploding or vanishing gradients.

As a specific application scenario, more specific problems arise in health care when performing entity relation extraction. For example, when SVM are used for relation extraction, misclassification is more common for some relation classes that are not subdivided, as the process of SVM training is to find the hyperplane that results in the maximum distance between two classes. The subdivision of the relation classes followed by the use of multiple SVMs may address this issue (15).

In circumstances in which the entity pairs in the clinical conversations are not in the same text, Du *et al.* (16) used cache to store the entities, and then used the joint extraction method to reduce the complexity from $O(n^4)$ to $O(n)$. Akkasi *et al.* (17) completed a comprehensive survey of causal relation extraction from a biomedical text using

multiple mainstream deep neural models, and used a simple random oversampling technique for data augmentation to reduce class imbalance. Wu *et al.* (18) proposed a multi-head attention enhanced graph convolutional network, and used this neural network model to extract task data sets from 2 publicly available medical-biological relation extraction task data sets [i.e., Chemical-Disease Relation (CDR) and Chemical-Protein Interaction (CPI)] and to extract biomedical information from a pan-cancer pathology report corpora in multiple hospitals.

Medical entity relation extraction from unbalanced data sets

Problems

Unlike entity relation extraction in the general domain, medical entity relation extraction has to cover medical terminology in a variety of specialties. As medical terms often do not appear in commonly used texts, word vectors trained using general domain texts may not contain these specialist words, which are called out-of-vocabulary (OOV) words. OOV words do not have fixed word vectors that are independently represented. The following two processing methods are usually used: (I) the word vectors of OOV words are uniformly defined as vector 0; or (II) a new word vector is randomly generated every time an OOV word is encountered. In medical entity relation extraction, if the entity is an OOV word, the 1st method cannot effectively distinguish the entity from other types of OOV words, while the 2nd method cannot effectively use the context information of the entity vector. Thus, the word vectors used in this article were word vectors trained using the experimental data sets.

Since the objective of medical entity extraction is typically to build a credible knowledge base, the goal of entity relation extraction is a highly authoritative formal publication. These publications often feature a clear theme, and accordingly the content elucidated may focus on specific topics. Once used as the target of entity relation extraction, the class imbalance will also be highlighted. For example, in our data set (which will be further introduced in the following section), there were 961 entity relations for the class "Medications that can be used to treat a specific disease" but only 78 entity relations for the class "Medicines that can be used to treat a specific symptom." When data augmentation is not performed, deep-learning models obtain more language features of the former than the latter under the same training conditions. This is because the former enables more training sessions in the models,

allowing the models to learn more language features. Thus, to enable the latter to also acquire more language features with a smaller number, we propose data augmentation based on reinforcement learning to expand the volume of entity relations in the latter.

Unbalanced data sets

The experimental data set used in our current study was the "Indications", "Contraindications", "Adverse Reactions", and "Precautions" sections in each chapter of the "The Pharmacopoeia of the People's Republic of China—Instructions for Clinical Use of Drugs—Chemical Drugs and Biological Products Volume" (abbreviated as "Pharmacopoeia" in this article). As the amount of data in each chapter was still very large, 3,800 sentences were randomly selected for entity relation labeling.

The entity relations were divided into 8 classes (see *Table 1*).

As *Table 1* and *Figure 1* show, there are large gaps in the numbers of the different classes; the largest class had 1,230 items, accounting for 32.14% of the total, while the smallest class had only 78 items, accounting for only 2.08% of the total. The sum of the top 2 classes accounted for 58.29% of the total. From the perspective of data set design, this data set was a failure as the amount of data was seriously unbalanced; the average number of the classes was 470, but only 3 classes were above the average, and the number of items in the 2nd ranked class was 175.69% of that in the 3rd ranked class. A model trained on such an imbalanced data set will underperform in the classes with the small sample sizes, as the model learns more large-sample features and ignores the small-sample features. For large samples, it is often necessary to downsample to balance the number of samples to qualify the data set. However, given that the data set will serve relation extraction with data augmentation, the use of data augmentation will compensate for the imbalance of the samples rather than upsampling the small-sample class data. From a demand point of view, this data set just fits the demand.

Chauhan *et al.* (19) found that compared to coarser-grained annotations, finer-grained annotations are more conducive to training more effective models. Thus, when part-of-speech tagging was performed on the data set, not only was the conventional part-of-speech used to tag the words, but the words that needed to be extracted from entity relations in this article were also classified in more detail. As the entities in the entity relation extraction were nouns, the specialist nouns under part-of-speech were subdivided into "noun-symptom", "noun-disease", "noun-

Page 4 of 13

Wang et al. Relationship extraction of unbalanced data in medical field

Table 1 Number of entity relations

Class	Entity relations (n)
Medicines that can be used to treat a specific disease	961
Medicines that can be used in combination with another drug	115
Medicines that can be used to treat a specific symptom	78
Medicines that cannot be used to treat a specific disease	431
Medicines that can be used with caution to treat a specific disease	181
Medicines that can be used with caution in a specific population	216
Medicines that can cause a specific symptom	1,230
Medicines that can cause a specific disease	547
Total	3,759
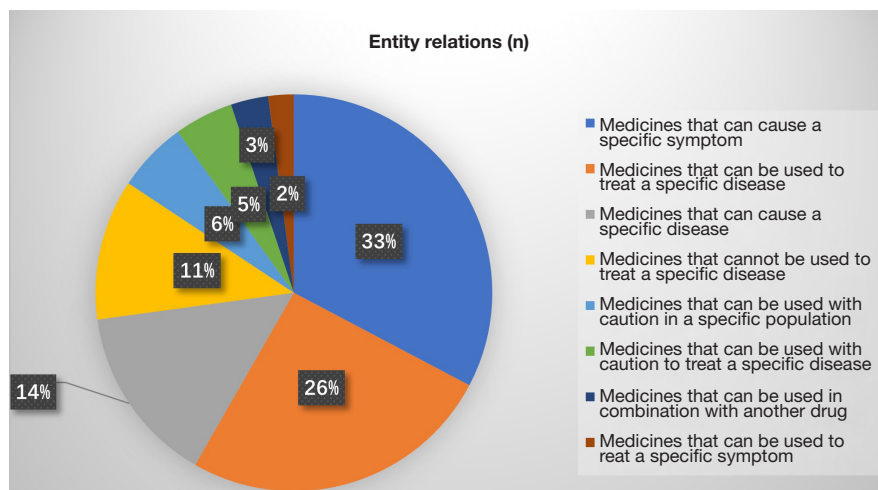
Figure 1 Pareto chart of the number of entity relations.

medicines", and "noun-population", which assisted the model to obtain the relation classification information.

Methods

Traditional data augmentation methods and their shortcomings

Data augmentation is often used for images and audio to maximize the use of each piece of data through specific processing. For example, for image data that are highly visualizable, the available augmentation techniques include cropping, flipping, rotation, scaling, and distortion, which make the image texture features more diverse. Conversely, when text is used as the research object of NLP, its features become more abstract after being extracted by the model, and it becomes difficult to verify whether the features really become more diverse after data augmentation. Thus, data augmentation for text is generally carried out from the perspective of linguistics. For example, translation can be applied to translate the text into another language and then translate it back (20). Thus, the new text has new expressions with its original semantics unchanged.

Some words have synonyms that express the same meanings, and changing the expressions of the text by replacing words with the synonyms is also a useful data augmentation method (21). Notably, the importance varies among different parts of speech in a text. The presence or absence of some words will not cause great changes in the

semantics of the text, and these unimportant words can be deleted. Thus, the random deletion of words in the text is another effective data augmentation method (22). However, few people use multiple methods for data augmentation at the same time. Wei (23) systematically sorted out 4 data augmentation methods and tested them one after another, and the results confirmed the feasibility of these data augmentation methods.

Theoretically, all the above methods can efficiently generate new texts. However, in the real world, the quality of new data generated by data augmentation using non-artificial means varies. Thus, the model may be trained by mixing the augmented texts with the non-augmented texts, and reinforcement learning can then be applied to select the appropriate texts to update the model parameters (21).

Most of the traditional data augmentation methods are relatively simple. For methods that require multiple operations in the text, most of them are operated randomly, and the specific method of data augmentation is not determined according to the text content. Further, there is a lack of feedback on the use of the data-augmented texts. Reinforcement learning is used to select the augmented text to improve the efficiency of model training; the feedback is not passed to the data augmentation part, and as a result, it neither increases the efficiency of the data augmentation nor decreases the invalid or unnecessary data augmentation operations. Thus, we proposed a reinforcement learning-based data augmentation method.

Reinforcement learning-based data augmentation

Instead of using a single data augmentation method, we chose to modify the method initially proposed by Wei (23). The new data augmentation method is as follows:

(I)   Synonym/class word replacement: replace the word in the corresponding position with its synonym/class word;

(II) Position replacement: randomly replace words in 2 positions;

(III) Synonymous/class word replacement + position replacement: replace a word with its synonym/class word and also replace its position with another word;

(IV) Word deletion: randomly delete the word at the current position.

Figure 2 illustrates the application of 4 data augmentation methods for the same word/phrase in the same text. Notably, the word/phrase is not only replaced by its synonym but also by its class word, as there may be a lack of synonyms or aliases for a specific noun, and the use of class words can increase the diversity during data augmentation.

However, the simultaneous use of multiple data augmentation methods in the same text will introduce more noise than the use of only 1 single data augmentation method. Additionally, the importance of different parts of speech in the text also differs, and it is important to define the weights of different data augmentation methods according to different parts of speech. Thus, we introduced the idea of reinforcement learning into data augmentation and proposed a reinforcement learning-based data augmentation method.

Figure 3 provides a simple flow chart of reinforcement learning-based data augmentation. This new method addresses the lack of connection between the neural network model and the data augmentation module in traditional data augmentation. Additionally, due to the increased connection between them and the timely feedback of the neural network, the generator module responsible for the data augmentation can optimize and determine the parameters of the data augmentation method in a timely manner according to the feedback, thus achieving biased data augmentation in relation to different parts of speech.

The timing of data augmentation in texts is also important. We adopted the same ratio (15%) as MASK in Bi-directional Encoder Representation from Transformers (BERT) (24) for the data augmentation. In the initial state, the probabilities of the 4 data augmentation operations are equal. During model training, the probabilities of these 4 operations are rewarded based on the results of the neural network as per Eq. [1], which states:

$$w' = w + action \qquad [1]$$

where w' represents the updated weight, w represents the original weight, and action is the reward value.

After each operation is rewarded, the balance of the original equal-probability operations is broken, such that the generator module that performs the data augmentation will have different preferences in different situations. With the deepening of model training, the data augmentation of the text will become more reasonable.

Figure 4 shows the process that the text actually undergoes in entity relation extraction during reinforcement learning-based data augmentation. First, the generator generates a data-augmentation action sequence, which allows the operation by the action sequence on the text.
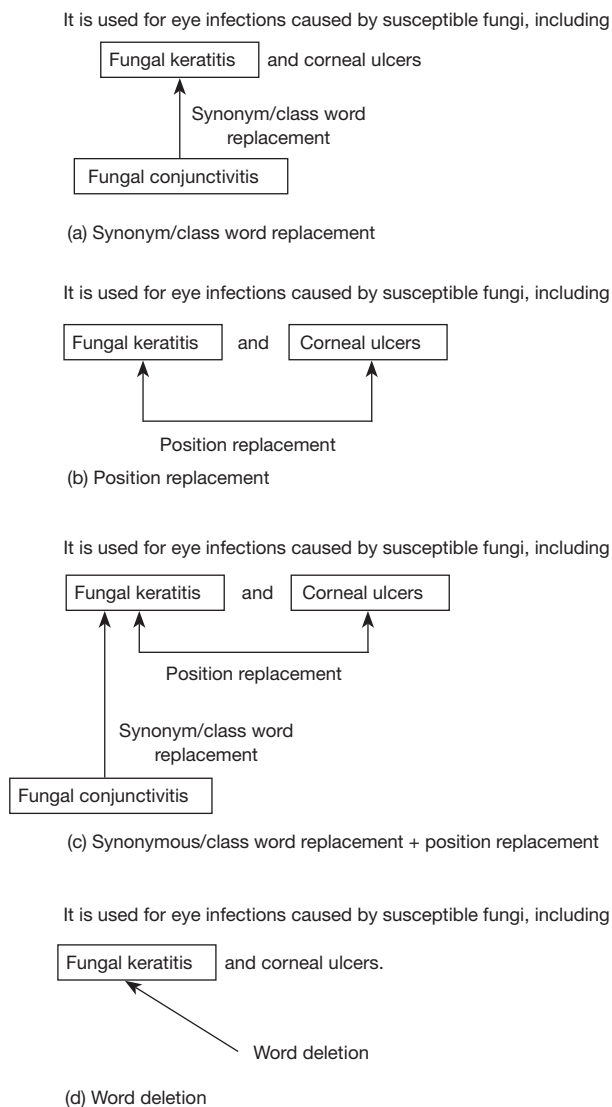
It is used for eye infections caused by susceptible fungi, including

| Fungal keratitis | and corneal ulcers

↑ Synonym/class word replacement

| Fungal conjunctivitis |

(a) Synonym/class word replacement

It is used for eye infections caused by susceptible fungi, including

| Fungal keratitis | and | Corneal ulcers |

Position replacement

(b) Position replacement

It is used for eye infections caused by susceptible fungi, including

| Fungal keratitis | and | Corneal ulcers |

Position replacement

Synonym/class word replacement

| Fungal conjunctivitis |

(c) Synonymous/class word replacement + position replacement

It is used for eye infections caused by susceptible fungi, including

| Fungal keratitis | and corneal ulcers.

Word deletion

(d) Word deletion

Figure 2 Data augmentation.

Second, 2 part-of-speech sequences are obtained from the text, which are spliced with the text sequence and then put into the neural network for training. Since a loss value representing the fitness between the text and the model will be generated during training, it is possible to know whether the text before augmentation or the text after augmentation fits the model better by comparing the loss value. The model is updated with the training parameters of the more fitting text. If the post-augmentation text fits the model better, the generator module will be rewarded according to the action sequence, and its parameters will be updated.

Compared to simple text-based data augmentation, the reinforcement learning-based data augmentation uses more data augmentation methods and also considers the connection between the generator module responsible for data augmentation and the trained model. As a result, the data augmentation is no longer conducted by random selection but is done in a more selective and biased manner. In particular, the introduction of part-of-speech vectors increases the directionality of the bias and the availability of post-augmentation text.

Statistical analysis

All Statistical analysis of medical literature data was performed using SPSS software (version 22.0, IBM Corp., Armonk, NY, USA). The quantitative data are expressed as the mean ± standard deviation (SD) for normally distributed data, or the median (interquartile range) for non-parametric data. Categorical variables of 8 valid entity relations were analyzed using the chi-square tests and Student's t-test. P value <0.05 was considered to be statistically significant.

Results

Assessment criteria

We analyzed the performance of the model from the following two perspectives: (I) the ability of the model to classify each entity relation; and (II) the overall classification performance of the multi-class model. The former is measured using precision, recall, and F score, and the latter was assessed using MACRO precision, MACRO recall, and MACRO F score.

The above assessment values were calculated from the predicted results. "True" and "False" represent whether the sample is correctly predicted, and "Positive" and "Negative" represent the prediction results of the samples. Pairwise matching yielded four combinations: true positive (TP), false positive (FP), true negative (TN), and false negative (FN).

Precision, also known as the precision rate, is calculated as the number of correct positive predictions divided by the total number of positive predictions and is calculated as follows:

$$Precision = \frac{TP}{TP + FP} \qquad [2]$$

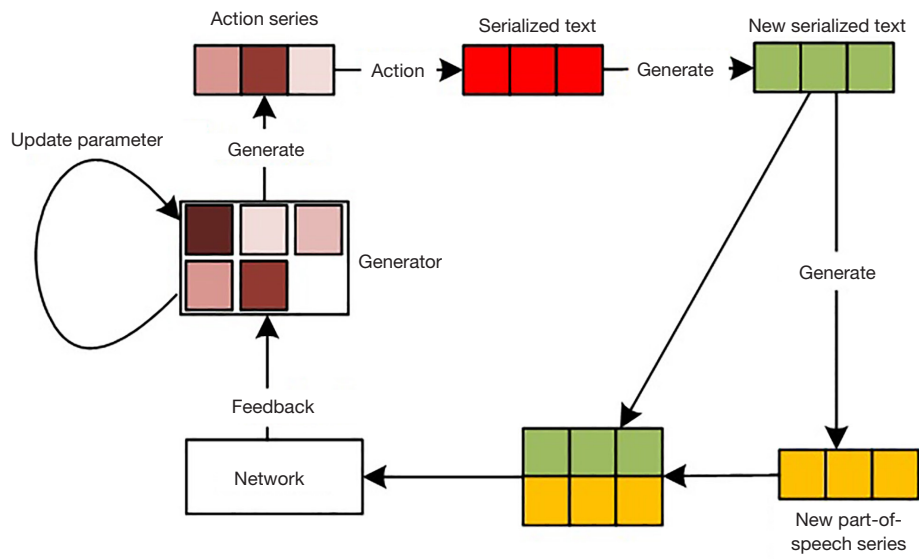Conversely, accuracy refers to how much of the entire sample was successfully predicted and is calculated as follows:
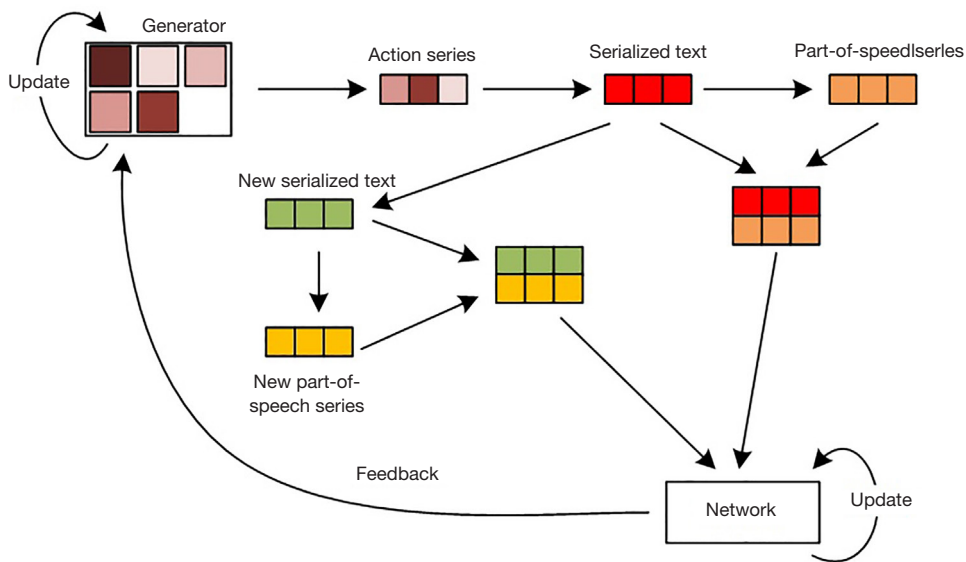
Figure 3 Reinforcement learning-based data augmentation.

Figure 4 Entity relation extraction in reinforcement learning-based data augmentation.

$$Accuracy = \frac{TP+TN}{TP+FN+FP+TN} \qquad [3]$$

$$Recall = \frac{TP}{TP+FN} \qquad [4]$$

Recall, or the true positive rate, is the measure for how many true positives get predicted out of all the positives in the data set. It is sometimes also called the sensitivity, and is calculated is follows:

Precision and recall are not sufficient to assess how good a model is. They are often negatively correlated and need to be balanced. As it is not intuitive to compare precision and recall among multiple models, a new metric, F_β-score,

was introduced to combine these two ratios to intuitively evaluate the pros and cons of a specific model.

$$F_b = \frac{(1+b^2)\,Precisiong\ Recall}{\beta^2\ Precision + Recall} \qquad [5]$$

Where β represents the different weights of precision and recall after merging, and F_β represents the importance of recall and precision. Under normal circumstances, the β value is 1 (i.e., F_1), which indicates that precision and recall are equally important.

For the multi-class models, the F_1 value of each class does not indicate the performance of the model, as each classification has different results depending on the model, making it difficult to evaluate model performance. Thus, the performance of multi-class models is assessed using MACRO precision, MACRO recall, and the MACRO F value. The formulas are as follows:

$$precision_{ma} = \frac{\sum_{i=0}^{I} precision_i}{I} \qquad [6]$$

$$recall_{ma} = \frac{\sum_{i=0}^{I} recall_i}{I} \qquad [7]$$

$$macroF_1 score = 2\,\frac{precision_{ma} \times recall_{ma}}{precision_{ma} + recall_{ma}} \qquad [8]$$

Experimental setup

In our study, the following 3 basic models for entity relation extraction were used to verify the availability of reinforcement learning-based data augmentation: TEXT CNN (textCNN) (25), long short-term memory (LSTM) (26), and gated recurrent unit (GRU) (27). Among these 3 models, the LSTM model adopted was the bi-directional long short-term memory (Bi-LSTM) model. In relation to the hyperparameters, the TEXTCNN model used 16 convolution kernels with widths of 4 and 6, while both the GRU and Bi-LSTM models used a hidden layer with a dimension of 50, the learning rate was fixed at 0.01, and the number of training rounds was 25.

As the data set was unbalanced, more data augmentation was required for the sample classes with lower data volumes; however, data augmentation should be minimized for classes with large data volume. The following formulas were used:

$$P(0) = \frac{n_i}{N} \qquad [9]$$

$$P(1) = 1 - P(0) \qquad [10]$$

where 0 and 1 represent no data augmentation and data augmentation, respectively, n_i is the number of current relation classes, and N is the total number of corpora in the corpus.

Notably, data augmentation actions should not be performed on all words. Data augmentation will not distort a sentence by excessive veter shift in the sentence, but too much data augmentation will change the meaning of the sentence, and the sentence will become a meaningless sequence of words. Inspired by the BERT (19) pre-training model, we only changed about 15% of the words in the sentences, and the remaining 85% remained unchanged. In this way, the semantics represented by the original sentences were preserved as much as possible.

Experimental results

The experiments were performed using 3 common neural networks, including textCNN, GRU, and Bi-LSTM. In the experiments, no data augmentation, traditional data augmentation, or reinforcement learning-based data augmentation were conducted independently. By comparing the evaluation indicators of the classification ability of each entity relation among different models and the MACRO evaluation indicators of each model, we tried to identify the values of the data augmentation techniques for different neural network models. In the following tables, A (Augment) represents data augmentation technology based on traditional methods and AR (Augment with Reinforcement) represents the reinforcement learning-based data augmentation.

Table 2 shows the experimental results for data augmentation on textCNN. Notably, for the entity relation classes with large data volumes, the use of a random method for data augmentation achieved certain improvements, but this was not the case for the entity relation classes with small data volumes. After the data augmentation operations, there were various degrees of classification failure, and the entity relations could not be correctly classified at all. Only the class "Medicines to be used with caution in a specific population" was correctly classified because the

Table 2 Experimental results of data augmentation in textCNN

Relation class	textCNN			A_textCNN			AR_textCNN		
	Precision	Recall	F$_1$ value	Precision	recall rate	F$_1$ value	Precision	Recall	F$_1$ value
Medicines that can cause a specific symptom	0.5795	0.904	0.7063	0.5825	0.96	0.7251	0.7659	0.916	*#0.8342
Medicines that can be used to treat a specific disease	0.4971	0.4462	0.4703	0.4824	0.841	0.6131	0.8018	0.8923	*#0.8447
Medicines that can cause a specific disease	0.1308	0.1828	0.1525	0	0	0	0.5676	0.4516	*#0.503
Medicines that cannot be used to treat a specific disease	0.4375	0.0745	0.1273	0	0	0	0.7826	0.766	*#0.7742
Medicines that can be used with caution in a specific population	0.963	0.6047	*0.7429	1	0.3256	0.4912	0.8571	0.5581	0.6761
Medicines that can be used with caution to treat a specific disease	0.2	0.119	0.1493	0	0	0	0.9063	0.6905	*#0.7838
Medicines that can be used in combination with another drug	1	0.1579	0.2727	0	0	0	0.5	0.2632	*0.3448
Medicines that can be used to treat a specific symptom	0	0	0	0	0	0	0.8462	0.55	0.6667

*, the highest F$_1$ value of different data augmentation methods for the same neural network; #, the highest F$_1$ value among all models. CNN, convolutional neural network.

entity "population" is a unique entity type. In relation to the reinforcement learning-based data augmentation, in addition to the entity relation class, "Medicines to be used with caution in a specific population", the classification ability of the textCNN was significantly improved.

Table 3 lists the experimental results of the data augmentation for GRU. The classification performance of the randomized data augmentation decreased for 4 entity relations and increased for 3 entity relations. For the class, "Medicines that can be used with caution to treat a specific disease", the classification performance was changed from "Medicines that cannot be correctly classified at all" to "Medicines that can be correctly classified but with low probability" after data augmentation. Additionally, the class "Medicines that can be used to treat a specific symptom", which had a small data volume, remained totally unclassified.

In the reinforcement learning-based data augmentation, the classification performance was slightly decreased for only 2 classes, among which the class, "Medicines that can be used with caution in a specific population" dropped by 0.004 only. Given the presence of measurement error,

the classification performance could be considered not to have changed. The classification performance of the other 6 entity relation classes was significantly improved. Notably, the F value of the class, "Medicines that can be used with caution to treat a specific disease" increased greatly from 0 (before data augmentation) to 0.7711 (after data augmentation), indicating that it had a relatively reliable classification performance. Notably, for the class, "Medicines that can be used to treat a specific symptom", the F1 value was both 0 in the no-augmentation group and randomized augmentation group but increased to 0.5 in the reinforcement learning-based data augmentation group.

Table 4 shows the experimental results of data augmentation in Bi-LSTM. For Bi-LSTM without any processing, nearly half of the entity relation classes in the unbalanced data set were completely unpredictable. After randomized data augmentation, 2 entity relation classes changed from "completely unclassifiable" to "low relation classification performance"; however, there were still classes that changed from "low relation classification performance" to "completely unclassifiable". In general, the classification performance was improved in 5 classes, decreased in 2

Page 10 of 13

Wang et al. Relationship extraction of unbalanced data in medical field

Table 3 Experimental results of data augmentation in GRU

Relation class	GRU			A_GRU			AR_GRU		
	Precision	Recall	F_1 value	Precision	Recall	F_1 value	Precision	Recall	F_1 value
Medicines that can cause a specific symptom	0.7816	0.816	0.7984	0.5915	0.944	0.7273	0.7484	0.952	*0.838
Medicines that can be used to treat a specific disease	0.4754	0.841	0.6074	0.5882	0.4615	0.5172	0.7897	0.8667	*0.8264
Medicines that can cause a specific disease	0.1795	0.1505	0.1637	0.1563	0.1075	0.1274	0.5536	0.3333	*0.4161
Medicines that cannot be used to treat a specific disease	0.25	0.0638	0.1017	0.4894	0.2447	0.3262	0.8784	0.6915	*0.7738
Medicines that can be used with caution in a specific population	1	0.6744	0.8056	0.7708	0.8605	*0.8132	0.9118	0.7209	0.8052
Medicines that can be used with caution to treat a specific disease	0	0	0	0.1905	0.1905	0.1905	0.7805	0.7619	*0.7711
Medicines that can be used in combination with another drug	0.45	0.4737	*#0.4615	0.5833	0.3684	0.4516	0.4118	0.3684	0.3889
Medicines that can be used to treat a specific symptom	0	0	0	0	0	0	0.6667	0.4	*0.5

*, the highest F_1 value of different data augmentation methods for the same neural network; #, the highest F_1 value among all models. GRU, gated recurrent unit.

classes, and was completely unclassifiable in 1 class. The last class remained completely unclassifiable even after reinforcement learning-based data augmentation.

The performance of reinforcement learning-based data augmentation was also poor in Bi-LSTM. Indeed, it only increased in 3 classes and decreased in 3 classes, and for the first time, a classification of "completely unclassifiable" was recorded. For the randomized data augmentation, the performance was improved in only 3 classes but decreased in the remaining 4 classes. Among them, the performance of the class, "Medicines that can be used in combination with another drug" declined from "low reliability" to "completely unclassifiable".

For textCNN and GRU, the improvement in classification performance by randomized data augmentation was very limited. Conversely, reinforcement learning-based data augmentation dramatically increased the classification performance of each entity relation class. As the basically bold F1 values show, the use of reinforcement learning-based data augmentation is quite effective in both network models. For Bi-LSTM, no matter which data augmentation technique was applied, the classification performance of the model was not stably elevated, and there was a possibility

of performance degradation. This may be due to the use of data shuffling in the data augmentation, which degrades the classification performance of Bi-LSTM that relies heavily on time series data.

According to the MACRO F1 score of the models (see *Table 5*), there were few improvements in the performance of the models when Bi-LSTM was used for the data augmentation. The performance of other models was significantly improved with MACRO F1 scores of around 0.60. The neural network model with the highest MACRO F1 score changed from GRU to textCNN, and the performance worsened after combination with Bi-LSTM. This may be because the time series-sensitive RNN learned a large number of noises due to the change in the order of words during the data augmentation.

Conclusions

In this article we examined the use of reinforcement learning-based data augmentation for entity relation extraction in the medical domain. Typically, a single data augmentation method is applied in NLP. Further, when multiple data augmentation methods are used, they are

Table 4 Experimental results of data augmentation in Bi-LSTM

Relation class	Bi-LSTM			A_Bi-LSTM			AR_Bi-LSTM		
	Precision	Recall	F_1 value	Precision	Recall	F_1 value	Precision	Recall	F_1 value
Medicines that can cause a specific symptom	0.7384	0.892	*0.808	0.7824	0.82	0.8008	0.6015	0.948	0.736
Medicines that can be used to treat a specific disease	0.4326	0.8564	0.5749	0.5117	0.7846	0.6194	0.7343	0.7795	*0.7562
Medicines that can cause a specific disease	0.2368	0.0968	0.1374	0.2254	0.172	*0.1951	0.1579	0.0968	0.12
Medicines that cannot be used to treat a specific disease	0	0	0	0.3014	0.234	0.2635	0.7121	0.5	*0.5875
Medicines that can be used with caution in a specific population	0.8056	0.6744	0.7342	0.8571	0.8372	*#0.8471	0.6667	0.4651	0.5479
Medicines that can be used with caution to treat a specific disease	0.5	0.0238	0.0455	0	0	0	0.5	0.0952	*0.16
Medicines that can be used in combination with another drug	0	0	0	0.3158	0.3158	*0.3158	0	0	0
Medicines that can be used to treat a specific symptom	0	0	0	0	0	0	0	0	0

*, the highest F_1 value of different data augmentation methods for the same neural network; #, the highest F_1 value among all models. Bi-LSTM, bi-directional long short-term memory.

Table 5 MACRO evaluation of each model

Model	MACRO precision	MACRO recall	MACRO F_1_SCORE
textCNN	0.476	0.3111	0.3763
A_textCNN	0.2581	0.2658	0.2619
AR_textCNN	0.7534	0.636	*#0.6897
GRU	0.3921	0.3774	0.3846
A_GRU	0.4213	0.3971	0.4088
AR_GRU	0.7176	0.6368	*0.6748
Bi-LSTM	0.3391	0.3179	0.3282
A_Bi-LSTM	0.3742	0.3955	0.3845
AR_Bi-LSTM	0.4216	0.3606	*0.3887

*, the highest F_1 value of different data augmentation methods for the same neural network; #, the highest F_1 value among all models. CNN, convolutional neural network; GRU, gated recurrent unit; Bi-LSTM, bi-directional long short-term memory.

not simultaneously applied to the same text. For the data augmentation methods that require multiple operations, the random approach is applied to the text without considering the specific content of the text. Additionally, there is no effective feedback between the data augmentation module and the model, and the training results do not improve the subsequent data augmentation and thus the usability of the text.

Thus, we proposed a reinforcement learning-based data augmentation method that adopts a variety of data augmentation methods to act on the same text. By establishing a connection between the data augmentation module and the model, this new method feeds the training results back to the data augmentation module in. a timely manner, helping the data augmentation module to continuously improve the results of the data augmentation and finally achieving the purpose of applying different data augmentation methods according to the text content.

When the traditional data augmentation and reinforcement learning-based data augmentation are applied to entity relation extraction of unbalanced data in the medical field, both data augmentation techniques effectively improve the entity relation extraction. As the various model evaluation indicators show, the reinforcement learning-based data augmentation method performed better than the traditional method.

However, reinforcement learning-based data augmentation has some limitations. Text is a kind of time series data; however, the new data augmentation method is based on the text content only and neglects the contextual information. Thus, the efficiency of the data augmentation can still be improved. In addition, as reinforcement learning is typically used in the context of Markov-Decision Process (MDP), we intend to integrate MDP into the reinforcement learning-based data augmentation method.

Acknowledgments

Footnote

Data Sharing Statement: Available at https://atm.amegroups. com/article/view/10.21037/atm-22-3991/dss

Conflicts of Interest: All authors have completed the ICMJE uniform disclosure form (available at https://atm. amegroups.com/article/view/10.21037/atm-22-3991/coif). LL, XW, JZ and JL are from Hunan Creator Information Technology Co., Ltd. The other authors have no conflicts of interest to declare.

Ethical Statement: The authors are accountable for all aspects of the work in ensuring that questions related to the accuracy or integrity of any part of the work are appropriately investigated and resolved.

References

1. Miller S, Fox H, Ramshaw L, et al. A novel use of statistical parsing to extract information from text. 1st Meeting of the North American Chapter of the Association for Computational Linguistics, 2000.
2. Aitken JS. Learning information extraction rules: An inductive logic programming approach. ECAI 2002:355-9.
3. Akkasi A, Varoğlu E, Dimililer N. ChemTok: A New Rule Based Tokenizer for Chemical Named Entity Recognition. Biomed Res Int 2016;2016:4248026.
4. Eftimov T, Koroušić Seljak B, Korošec P. A rule-based named-entity recognition method for knowledge extraction of evidence-based dietary recommendations. PLoS One 2017;12:e0179488.
5. Hu D, Zhang H, Li S, et al. Automatic Extraction of Lung Cancer Staging Information From Computed Tomography Reports: Deep Learning Approach. JMIR Med Inform 2021;9:e27955.
6. Kambhatla N. Combining lexical, syntactic, and semantic features with maximum entropy models for extracting relations. Proceedings of the ACL 2004 on Interactive poster and demonstration sessions, 2004:22-es.
7. Tratz S, Hovy E. Isi: automatic classification of relations between nominals using a maximum entropy classifier.

Proceedings of the 5th International Workshop on Semantic Evaluation, 2010:222-5.

8. Peng Y, Rios A, Kavuluru R, et al. Extracting chemical-protein relations with ensembles of SVM and deep learning models. Database (Oxford) 2018;2018:bay073. doi: 10.1093/database/bay073.

9. Khattak FK, Jeblee S, Pou-Prom C, et al. A survey of word embeddings for clinical text. J Biomed Inform 2019;100S:100057.

10. Zhao D, Wang J, Sang S, et al. Relation path feature embedding based convolutional neural network method for drug discovery. BMC Med Inform Decis Mak 2019;19:59.

11. Yang X, Bian J, Fang R, et al. Identifying relations of medications with adverse drug events using recurrent convolutional neural networks and gradient boosting. J Am Med Inform Assoc 2020;27:65-72.

12. Liimatainen K, Huttunen R, Latonen L, et al. Convolutional Neural Network-Based Artificial Intelligence for Classification of Protein Localization Patterns. Biomolecules 2021;11:264.

13. Chen T, Wu X, Li L, et al. Extraction of entity relations from Chinese medical literature based on multi-scale CRNN. Ann Transl Med 2022;10:520.

14. Alt C, Hübner M, Hennig L. Fine-tuning pre-trained transformer language models to distantly supervised relation extraction. arXiv preprint arXiv:1906.08646, 2019.

15. Cheng JY. Research on relation extraction for Chinese electronic medical records. Harbin Institute of Technology, 2016.

16. Du N, Wang M, Tran L, et al. Learning to Infer Entities, Properties and their Relations from Clinical Conversations. arXiv preprint arXiv:1908.11536, 2019.

17. Akkasi A, Moens MF. Causal relationship extraction from biomedical text using deep neural models: A comprehensive survey. J Biomed Inform 2021;119:103820.

18. Wu J, Zhang R, Gong T, et al. Bioie: Biomedical information extraction with multi-head attention enhanced graph convolutional network. 2021 IEEE International Conference on Bioinformatics and Biomedicine (BIBM). IEEE, 2021:2080-7.

19. Chauhan G, McDermott M, Szolovits P. Reflex: Flexible framework for relation extraction in multiple domains. arXiv preprint arXiv:1906.08318, 2019.

20. Yu AW, Dohan D, Luong MT, et al. Qanet: Combining local convolution with global self-attention for reading comprehension. arXiv preprint arXiv:1804.09541, 2018.

21. Kobayashi S. Contextual augmentation: Data augmentation by words with paradigmatic relations. arXiv preprint arXiv:1805.06201, 2018.

22. Zheng S, Han X, Lin Y, et al. DIAG-NRE: A neural pattern diagnosis framework for distantly supervised neural relation extraction. arXiv preprint arXiv:1811.02166, 2018.

23. Wei J, Zou K. Eda: Easy data augmentation techniques for boosting performance on text classification tasks. arXiv preprint arXiv:1901.11196, 2019.

24. Devlin J, Chang M W, Lee K, et al. Bert: Pre-training of deep bidirectional transformers for la7nguage understanding. arXiv preprint arXiv:1810.04805, 2018.

25. Kim Y. Convolutional Neural Networks for Sentence Classification. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2014:1746-51.

26. Hochreiter S, Schmidhuber J. Long short-term memory. Neural Comput 1997;9:1735-80.

27. Xu Y, Mou L, Li G, et al. Classifying Relations via long short term memory networks along shortest dependency paths. Proceedings of the 2015 Conference on Empirical Methodsin Natural Language Processing, 2015:1785-94.