# Context-Aware End-To-End Relation Extracting From Clinical Texts With Attention-Based Bi-Tree-GRU

Dehua Chen[1] , Yunying Wu[1] , Jiajin Le[1] , Qiao Pan[1]

[1] Donghua University, Shanghai 201600, China
{chendehua, lejiajin, panqiao}@dhu.edu.cn,wyy2016@foxmail.com

**Abstract.** Extracting clinical entities and their relations from clinical texts is a preliminary task for constructing medical knowledge graph. Existing end-to-end models for extracting entity and relation have limited performance in clinical text because they rarely take both latent syntactic information and the effect of context information into account. Thus, this paper proposed a context-aware end-to-end neural model for extracting relations between entities from clinical texts with two level attention. We show that entity-level attention effectively acquires more syntactic information by learning a weighted sum of child word nodes rooted at the target entities. Meanwhile, sub sentence-level attention in an effort to capture the interactions between the target entity pairs and context entity pairs by assigning weights of each context representation within one sentence. Experiments on real-world clinical texts from Shanghai Ruijin Hospital demonstrate that our model significantly gains better performance in the application of clinical texts compared with existing joint models.

**Keywords:** Clinical Text, Clinical Entity Recognition, Relation Classification, Joint Model, Attention Mechanism.

## 1 Introduction

Clinical texts such as ultrasound reports and CT reports provide a wealth of clinical factual knowledge, mainly embodied in various clinical entities, such as organizational entities, location entities, index entities and attribute entities. Extracting these entities and their relation from unstructured reports are standard tasks of clinical information extraction and the foundation of constructing medical knowledge graphs. Named Entity Recognition (NER) and Relation Extraction (RE) as information extraction techniques are critical to natural language understanding and knowledge acquisition. Tradition work usually treat them as two steps to perform information extraction in a pipeline model, where the final performance will be hurt by the preceding errors generated in NER. To resolve this problem, joint models for NER and RE have been proposed and found effective for alleviating the problem of error propagation and utilizing the interactions between the two sub-tasks [1,2,3,4].

Recently, deep learning has attracted much attention and become an alternative work because it employs automatic feature learning without handcrafted and complicated feature engineering. Several neural models have dominated the joint models of RE due to the better results. Miwa M. and Bansal M. [5] proposed a neural network-based method using the shortest dependency path (SDP) between a given entity pair to incorporate the linguistic structures. Fei L. et al [6] applied similar methods into the field of medicine to extract the entities like drug names or disease names. Zhou P. et al [7] proposed an attention-based bidirectional long short-term memory (Bi-LSTM) model for RE to capture the most important semantic information in a sentence and Sorokin D. et al [8] combined the context representations with the attention mechanism. These previous models mostly did not consider both latent syntactic information and the effect of context information when modeling, which take on importance in informative and complex clinical texts because they are full of the disease description.

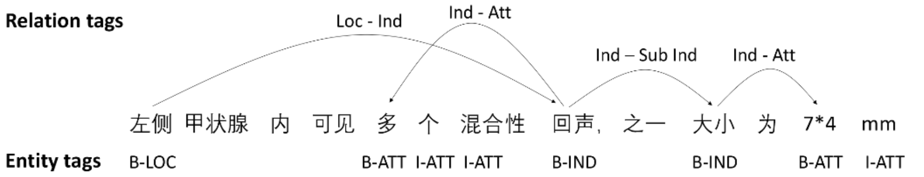

**Fig. 1.** The annotation of an example sentence from the thyroid ultrasound reports. Based on the word segmentation with correction by completing the vocabulary set, words are annotated as Anatomical Location (LOC), Index (IND) or Attribute (ATT) in BIO (Beginning, Insider, Outsider) tagging scheme. The relation tags are annotated as Location-Index (Loc-Ind), Index-Attribute (Ind-Att), Index-Sub Index (Ind-Sub Ind) or Unknown.

Given a sentence “左侧[left]甲状腺[thyroid]内[inside]可见[indicate]多个[multiple]混合性[mixed]回声[echos]，之一[one of which]大小[size]为[is]7*4mm” [1], whose detail of annotation is shown in Figure 1, “大小” is an index and it is related to another index “回声”. In another case, “甲状腺[thyroid]左叶[left lobe]大小[size]及[and]形态[appearance]正常[normal]”[1], “大小” is also an index, but it is directly related to two anatomical locations “左” and “右叶”, not related to the other indexes. Tree-structured relation extraction models [5,9] have been demonstrated the effectiveness of building the paths between words relying on the dependency parsing trees to handle with complex syntax information. As shown in Figure 2, “回声” and “大小” are parsed in two sub trees, whose relation is coordinating relation (“并列关系” in Chinese) and contributes to identifying whether it exists a relation between both indexes (“大小” is related to the prior index). However, only including the syntactical relation in models is not enough, the latent information within entities themselves could also be beneficial in RE. Taking the example in Figure 2 again, the child word nodes “多” (an adjective), “个” (a quantifier) and “混合型” (a distinctive word) are under the word node “回声” (a norm). It implicitly indicates that the sub-tree means the description of “回声” and could be made use of to predict their relation as “Index-Attribute”. In addition, a

[1] The words in square brackets are Chinese segmented words in English. The whole sentences of two examples in English are as followed: (1) There are multiple mixed echoes in the left thyroid, one of which is 7*4 mm big. (2) The size and appearance of the thyroid are normal.

sentence in general contains at least one anatomic location, one index and one attribute. We need to find the related indexes for the anatomic locations, the related attributes for the indexes and detect whether the links between the indexes exist or not, so the interactions between multiple relations (at least two) within one sentence should be captured but mostly be neglected in the tradition models.

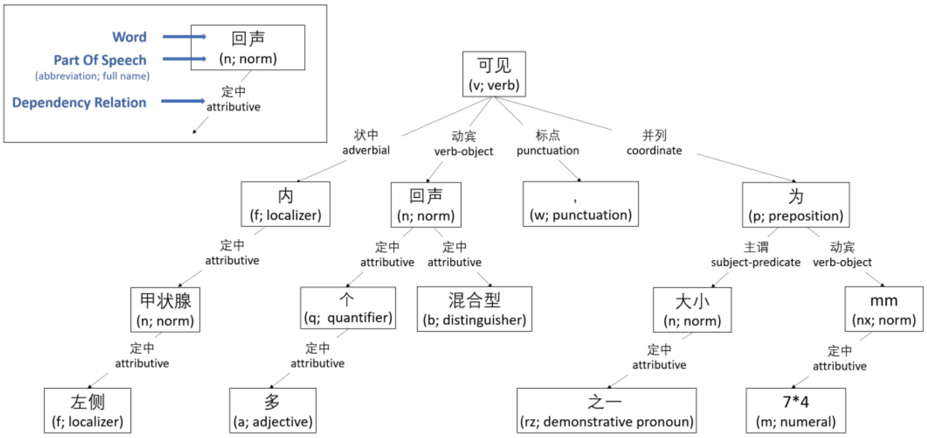

**Fig. 2.** An example of dependency parsing tree. A rectangular indicates a word node and part of speech tag (POS) is under the word, the strings covering the arrows denote the syntactic dependency relation.

In this paper, we propose a novel context-aware end-to-end neural RE model with entity-level and sub sentence-level attention mechanism to boost performance in clinical information extraction. We introduce entity-level attention to capture richer syntactical information by adopting a weighted sum of different child word nodes under the entities instead of the direct sum in traditional tree-structured models. Inspired by the previous work [8] incorporating context representations in RE, we additionally stack sub-sentence level attention layers on the original tree-structured RE models so that the other entity pairs within one sentence are taken as context information to facilitate the prediction of the target entity pairs.

Experimental results show that the proposed model significantly achieves better performance than state-of-the-art methods. We also analysis the contribution of different attention methods at entity-level and sub sentence-level based on the tests of composition change. We find entity-level attention's performance of F1 no worse than no attention mechanism and it generally obtains 6-9% improvement of F1 when syntactic rules are more complex. Sub sentence-level attention boosts F1 by at least 4.2% compared with no attention mechanism and achieves an 3.1% improvement compared with joint models with simple attention.

## 2 Related Work

As for named entity recognition, traditional models are based on Hidden Markov Models (HMM) [10] and Conditional Random Fields (CRF) [11]. In recent years, a great mass of deep learning methods has been presented because they are able to take low dimensional and dense embeddings as inputs to represent the words or other natural language processing (NLP) features and save time and effort by learning features from trained data automatically. Huang Z. et al [12] and Lample G. et al [13] created a representative model based on Bi-LSTM with CRF on top, which is able to capture memory of historical and future information in long dependency through gating units and the transition probability of the past and next tags thanks to CRF. Notice that both LSTM and Gated Recurrent Unit (GRU) belong to the family of recurrent neural network (RNN). LSTM addresses the vanishing and exploding gradient problems of conventional RNNs in long-term dependencies [14] and GRU is a variant of LSTM without separate memory cells to modulate information flows through units [15].

For relation classification, besides classic kernel-based models [16,17], several neural models have also been proposed, such as convolutional neural network (CNN)-based models [18,19], RNN-based models [20], hybrid models combining RNN and CNN [21]. Recently, neural RE models investigated syntax information to facilitate performance by tree-structured LSTM(Tree-LSTM), Tai K. S. et al [9] firstly introduced and Miwa M. and Bansal M. [5] improved it by using Bi-Tree-LSTM instead of one direction LSTM and handling an arbitrary number of children nodes in a dependency tree instead of a fixed number.

For end-to-end/joint extraction, Roth D. and Yih W. [1] proposed a joint inference framework via integer linear programming, Kate R. J. and Mooney R. J. [2] gave a graph-based method called card-pyramid parsing, and Li Q. and Ji H. [3] presented an incremental joint framework to accomplish entity recognition and relation classification simultaneously. Miwa M. and Bansal M. [5] applied joint extraction into Neural Network model, which also was adopted in biomedical texts [6].

Zhou P. et al [7] proposed an attention-based Bi-LSTM model for RE and demonstrated that attention mechanism is able to capture the important semantic information. Sorokin D. et al [8] introduce attention mechanism at sentence level to take context information into account and obtain great improvements. Among these, there is still room for further improvements, specially in information-rich clinical texts. We propose a novel end-to-end neural model with attention mechanism to assign different weights for child word nodes in the sub-trees under the target entities at entity-level and the context entity pairs in the same sentences at sub sentence-level and finally demonstrate that the model have the ability to capture the syntactic and context information from clinical texts.

## 3 Model

In this section, we introduce the proposed context-aware end-to-end model of entity and relation extraction with entity-level and sub sentence-level attention. The framework comprises of one preliminary task and two primary tasks: (1)input representation: segment the sentences into words, retrieve Part Of Speech(POS) tags, parse the dependency of the sentences and convert them into vectors as inputs of the whole network; (2)clinical entity extraction: Bi-GRU with CRF layer to identify the entity types, as shown in Figure 3; (3)clinical relation extraction: Bi-Tree-GRU with attention mechanism at entity-level and sub sentence-level to extract relationships between the entity pairs recognized in step 2 and details are described in Figure 4. The input vectors are shared and affected by both clinical entity and relation extraction so that the interactions between the two steps could be exploited and the error delivery from the previous step could be reduced.

## 3.1 Clinical Entity Recognition

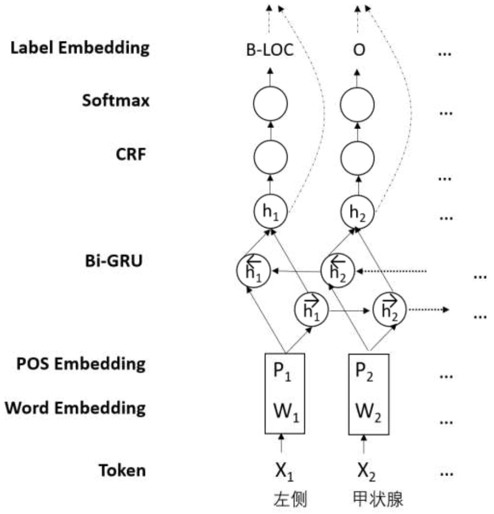

**Fig. 3.** The architecture of clinical entity recognition. The concatenations of words and POS tags are fed into Bi-GRU with CRF on top and softmax layer outputs the predictions of entity types.

### Input Representation.

For an input sentence consisting of n words $\mathrm{X} = \{x_1, x_2, \cdots, x_n\}$, each word $x_i \in \mathbb{R}^{\mathrm{V}}$ is represented by one-hot encoding and need to be transformed into a dense vector $w_i$ by learned embedding matrix $U_w \in \mathbb{R}^{\mathrm{V} \times d_w}$, where V is the total size of vocabularies in the dictionary and $d_w$ is a hyper-parameter to be chosen.

$$w_i = U_w\, x_i \tag{1}$$

$$emb_i^E = [w_i\, ,\ p_i\, ] \tag{2}$$

Where $w_i$ and $p_i$ denote the embeddings of the i−th word and the POS tag embeddings of $w_i$, respectively. Learned POS embedding matrix $U_1 \in \mathbb{R}^{L_1 \times d_1}$ reduces the dimensionality of original representations of POS tags, where $L_1$ is the numbers of POS tags in our biomedical datasets and $d_1$ is the dimension of dense POS vectors. The concatenation of the word embedding and POS tag embedding is taken as the input for the task of clinical entity recognition and its dimension is $d_E = d_1 + d_w$.

**Bi-GRU Layer.**

Given the input vectors $emb_t^E$ and the previous hidden state $\mathrm{h}_{t-1}^{(E)}$, we mainly use GRU model as the following implementation:

$$\mathrm{z}_t^{(E)} = \sigma(\mathrm{W}_z^{(E)} emb_t^E + \mathrm{U}_z^{(E)} \mathrm{h}_{t-1}^{(E)} + \mathrm{b}_z^{(E)}) \tag{1}$$

$$\mathrm{r}_t^{(E)} = \sigma(\mathrm{W}_r^{(E)} emb_t^E + \mathrm{U}_r^{(E)} \mathrm{h}_{t-1}^{(E)} + \mathrm{b}_r^{(E)}) \tag{4}$$

$$\widetilde{h_t^{(E)}} = \tanh(\mathrm{W}_h^{(E)} emb_t^E + \mathrm{U}_h^{(E)}(\mathrm{r}_t^{(E)} \odot \mathrm{h}_{t-1}^{(E)}) + \mathrm{b}_h^{(E)}) \tag{5}$$

$$\mathrm{h}_t^{(E)} = (1 - \mathrm{z}_t^{(E)}) \odot \mathrm{h}_{t-1}^{(E)} + \mathrm{z}_t^{(E)} \odot \widetilde{h_t^{(E)}} \tag{6}$$

Where σ is the sigmoid function and the ⊙ is element-wise multiplication. $\mathrm{z}_t^{(E)}$ is the upgrade vector, $\mathrm{r}_t^{(E)}$ is the reset vector and $\mathrm{h}_t^{(E)}$ is the output vector. $W_*^{(E)}$ and $U_*^{(E)}$ are the parameter matrices and $b_*^{(E)}$ is the bias vectors. The left and right output vectors $\overrightarrow{\mathrm{h}_t^{(E)}}$and $\overleftarrow{\mathrm{h}_t^{(E)}}$, referring to the forward and backward GRU, are concatenated to represent a word as $\mathrm{h}_t^{(E)} = [\overrightarrow{\mathrm{h}_t^{(E)}}; \overleftarrow{\mathrm{h}_t^{(E)}}]$, which are effective for numerous sequential tagging tasks because of the bidirectional information included.

**CRF Layer.**

Without CRF layer, the model will predict the tags based on the independence assumptions and ignore the strong dependencies (e.g, I-LOC cannot follow B-IND) between the adjacent outputs. Thus, we take the dependence between the two labels into consideration through stacking CRF layer on Bi-GRU. Given by an input sentence, the model could jointly decode the best chain of labels effectively by adopting Viterbi algorithm after learning the distribution of annotated datasets.

First, we compute the score of prediction sequences $\mathrm{y}^E = \{\mathrm{y}_1^E, \mathrm{y}_2^E, \cdots, \mathrm{y}_n^E\}$ of the input sentence X as followed, where $A_{\mathrm{y}_i^E, \mathrm{y}_{i+1}^E}$ denotes the transition score from tag $\mathrm{y}_{i+1}^E$ to $\mathrm{y}_i^E$ and $P_{i,\mathrm{y}_i^E}$ is the score of tag $\mathrm{y}_i^E$ of word $x_i$:

$$\mathrm{S}(\mathrm{X}, \mathrm{y}^E) = \sum_{i=0}^{n} A_{\mathrm{y}_i^E, \mathrm{y}_{i+1}^E} + \sum_{i=1}^{n} P_{i,\mathrm{y}_i^E} \tag{7}$$

Afterwards, we maximize the conditional probability of the prediction sequences via softmax layer. In practice, we usually the log-likelihood to choose the parameters during CRF training. Where Y(X) denotes all the possible prediction sequences including those go against the BIO tagging scheme:

$$\wp(\, y^E|\mathrm{X}) = \frac{\exp(\mathrm{S}(\mathrm{X},\mathrm{y}^E))}{\sum_{\widetilde{y^E}\in Y(X)}\exp(\mathrm{S}(\mathrm{X},\widetilde{\mathrm{y}^E}))} \tag{8}$$

$$L^E = log\big(\wp(\, y^E|\mathrm{X})\big) = \mathrm{S}(\mathrm{X},\mathrm{y}^E) - \log(\textstyle\sum_{\widetilde{y^E}\in Y(X)}\exp(\mathrm{S}(\mathrm{X},\widetilde{\mathrm{y}^E}))) \tag{9}$$

While decoding, we predict the outputs by searching the posteriori sequences with the highest score:

$$\widehat{y^E} = \mathrm{argmax}_{\widetilde{y^E}\in Y(X)}\mathrm{S}(\mathrm{X},\widetilde{\mathrm{y}^E}) \tag{10}$$

## 3.2 Context-Aware RE With Attention-Based Bi-Tree-GRU

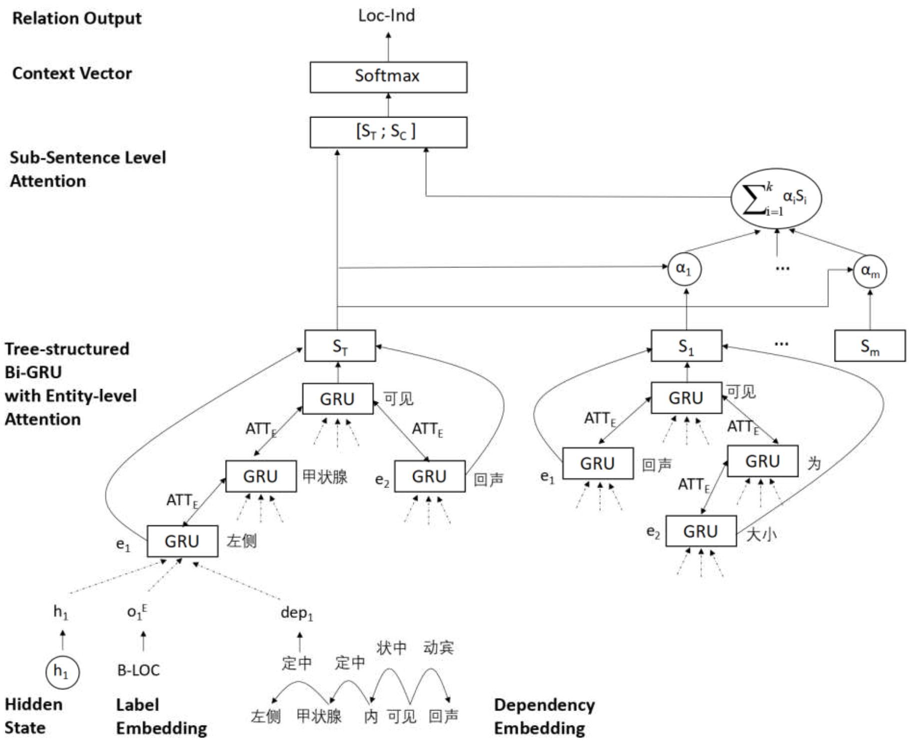

**Fig. 4.** The architecture of relation classification. The concatenations of hidden states from Bi-GRU, the label embeddings from clinical entity recognition and the syntactic dependency embeddings from dependency parsing tree are fed into Bi-Tree-GRU with entity-level and sub sentence-level attention and finally softmax predicts the relations between entity pairs.

**Input Representation.**

$$emb_i^R = [h_i\,,\ o_i^E\,,\ dep_i\,] \tag{11}$$

Where $h_i$ denotes the hidden state which has been explained in Eq.(6), $o_i^E$ denotes the label embeddings converted from the outputs of clinical entity recognition, the representation of the entity tags. After dependency-parsing our clinical documents, we convert the dependency types between the words and their parents in the parsing tree into the dense vectors $dep_i$ representing the syntactic information. For example, as shown in Figure 2, the syntactical dependency relationship between the word "左侧[left]" and its parent "甲状腺[thyroid]" is attributive relation("定中关系" in Chinese). We directly concatenate the three vectors and take them as the input vectors of relation extraction task.

**Entity-Level Attention**.

Supposed the t-th word node has N children, matrix $R_t$ consists the representation vectors of the children nodes, $\mathrm{W_h}$ and $\mathrm{W}_\alpha^T$ are shared parameters to be learned, $\alpha$ is the attention weight and the representation vector $ch_t$ is generated by a weighted sum of its children embeddings.

$$R_t = [r_{t1}, r_{t2}, \dots r_{tN}] \tag{12}$$

$$M_t = \tanh(\mathrm{W_h} H_t) \tag{13}$$

$$\alpha = softmax(\mathrm{W}_\alpha^T M_t) \tag{14}$$

$$ch_t = H_t \alpha^T \tag{15}$$

Then, we calculate the hidden state vector of t-th word node in the GRU unit:

$$\mathrm{z}_t^{(R)} = \sigma(\mathrm{W}_z^{(R)} emb_t^R + \mathrm{U}_z^{(R)} ch_t + \mathrm{b}_z^{(R)}) \tag{16}$$

$$\mathrm{r}_t^{(R)} = \sigma(\mathrm{W}_r^{(R)} emb_t^R + \mathrm{U}_r^{(R)} ch_t + \mathrm{b}_r^{(R)}) \tag{17}$$

$$\widetilde{\mathrm{h}_t^{(R)}} = \tanh(\mathrm{W}_h^{(R)} emb_t^R + \mathrm{U}_h^{(R)} \left(\mathrm{r}_t^{(R)} \odot \mathrm{h}_{t-1}^R\right) + \mathrm{b}_h^{(R)} \tag{18}$$

$$\mathrm{h}_t^{(R)} = \left(1 - \mathrm{z}_t^{(R)}\right) \odot \mathrm{h}_{t-1}^{(R)} + \mathrm{z}_t^{(R)} \odot \widetilde{\mathrm{h}_t^R} \tag{19}$$

Finally, we obtain bidirectional output vector:

$$\mathrm{S}_p = [\uparrow \mathrm{h}_p^{(R)}; \downarrow \mathrm{h}_{e\prime t}^{(R)}; \downarrow \mathrm{h}_{e\prime\prime t}^{(R)}] \tag{20}$$

We adopt the concatenation strategy to represent a sub sentence $\mathrm{S}_t$, which is similar with the entity recognition, but we call the two directions as bottom-up and top-down in the tree-structured network.

Where $\uparrow \mathrm{h}_t^{(R)}$ denotes the hidden state vector in bottom-up direction for the lowest common ancestor of the focus entity pair, and $\downarrow \mathrm{h}_{e\prime t}^{(R)}, \downarrow \mathrm{h}_{e\prime\prime t}^{(R)}$ denote the hidden state vector in top-down direction for the first and second entities. If the entity is comprised of more than one word, we take the word node which is less near to the root.

**Sub Sentence-Level Attention With Context.**

One sub-sentence incorporates the words of the target pairs and other words on the path to their lowest common ancestor. For example, as shown in Figure 2, the entity “左侧[left]” pairs up with “回声[echos]”, and “甲状腺[thyroid]” is the word on the path to their lowest common ancestor “可见[indicate]”, so these four words count as one sub-sentence. The sub-sentences may have overlaps. Take the above-mentioned example again, the word “回声[echos]” is also contained in another sub-sentence.

Supposed that $\mathrm{S} = \{S_1, S_2, \dots, S_k\}$ is comprised of k context sub-sentences, which are related to target entity pairs, we use a weighted sum of the context sub-sentences as followed:

$$G(S_i, S_T) = {S_i}^T \mathrm{W}_\alpha^{(S)} S_T \tag{21}$$

$$\alpha_i^{(S)} = softmax(G(S_i, S_T)) \tag{22}$$

$$S_c = \textstyle\sum_{i=1}^{k} \alpha_i^{(S)} S_i \tag{23}$$

$$S^* = [S_T; S_c] \tag{24}$$

Unlike the ways of simply getting a weighted sum of sub-sentences, we use $G_i$ to capture the relationship between the target sub-sentence $S_T$ and it context sub-sentences $S_i (1 \le i \le k)$ using the weight matrix $\mathrm{W}_\alpha^{(S)}$ to be learned. $S_c$ denotes the resulting representation of $S_T$’s context sub-sentences, a weighted sum of each single context representation $\alpha_i^{(S)}$. Finally, we obtain $\mathrm{S}^*$ as a context vector, the concatenation of $S_T$ and $S_c$, and feed it into the softmax layer.

We maximize the conditional probability of the relation sequences via softmax, compute log-likelihood and predict the outputs of relations via the highest conditional probability.

$$\wp\big(\, y^R | S^*; W^{(C)}, b^{(C)}\big) = softmax(W^{(C)} S^* + b^{(C)}) \tag{25}$$

$$L^R = \log(\wp\big(\, y^R | S^*; W^{(C)}, b^{(C)}\big)) \tag{26}$$

$$\widehat{y^R} = \ \mathrm{argmax}_{y^R \in Y(S^*)} \wp\big(\, y^R | S^*; W^{(C)}, b^{(C)}\big) \tag{27}$$

### 3.3 The Adjusted Optimization Function.

To increase the valid entity or relation prediction and reduce the impact of large amounts of O(outsider) tags in NER task and U(unknown) tags in RE task, we introduce the hyper-parameter $\gamma$ to adjust the imbalance of data. The smaller $\gamma$ is, the less impact of ‘O’ tags in the model. We take adjusted $L^E$ as the example:

$$L^E = \ \sum log\big(\wp(\, y^E|\mathrm{X})\big) \cdot (1 - I(O)) + \gamma log\big(\wp(\, y^E|\mathrm{X})\big) \cdot I(O) \tag{28}$$

$$I(O) = \begin{cases} 1, if\ tag = 'O' \\ 0, if\ tag \neq 'O' \end{cases}$$

Finally, we combine two loss function into one and use a hyper-parameter $\beta$ to balance them.

$$Loss = Loss^R + \beta Loss^E = -L^E - \beta L^R \tag{29}$$

# 4 Experiment

## 4.1 Experimental Setting

**Dataset.**
The experiment datasets in this paper are collected from Ruijin Hospital. We evaluated the model on the ultrasonic reports, X-ray/CT reports, Puncture reports, pathology reports of thyroid and mammary gland.

**Table 1.** Statistics of datasets

| Dataset | Number of sentences | Number of entities | Number of relations |
|---|---|---|---|
| Thyroid | 53,850 | 664,681 | 511,345 |
| Mammary Gland | 33,373 | 318,077 | 263,907 |

**Metrics.**
Precision(P), recall(R) and F1-score are used to compare the performance of the models.

Where True Positive (TP) denotes the number of entity types are identified as correct and boundaries are matched in NER or the numbers of correct relation types in RE. False Positive (FP) denotes the number of incorrectly identified entities or relations that do not meet the above conditions. False Negative (FN) denotes the number of unidentified entities or relations.

$$P = \frac{TP}{TP+FP}\ ; \quad R = \frac{TP}{TP+FN}\ ; \quad F1 = \frac{2*P*P}{P+R} \tag{30}$$

**Preprocessing.**
We pretrain our datasets by word2vec to initialize embeddings and adopt Chinese Natural Language Processing Tool HanLP [22] to conduct word segmentation, POS tagging and dependency parsing on the clinical texts from Ruijin Hospital. Then, we manually checkout the outputs and finally add some medicine-specific words and rules to complete annotations.

**Hyper-Parameters.**
While training, we use Adagrad optimization to tune the models with three-fold validation by adopting grid search for optional hyper-parameters and randomized search for scoped ones. Initially, we try {0.01/0.05/0.1/0.2} for learning rate, {50/100/150/200} for batch size, {25/50/100/150/200} for the dimension of variant

embeddings and 0.0-1.0 for drop probability and importance β and γ. During experiments, we find a set of effective configuration, which is shown in Table 2.

**Table 2.** Settings of hyper-parameters

| Type | Hyper-parameter | Value |
|---|---|---|
| Training | Learning Rate α | 0.05 |
| | Batch Size | 100 |
| | Drop Probability | 0.5 |
| Loss | Relation Importance β | 0.5 |
| | 'O' tag importance γ | 0.1 |
| Embedding | Word Dimension | 200 |
| | POS Dimension | 25 |
| | Dependency Dimension | 25 |
| Entity Recognition | GRU Hidden Unit Dimension | 100 |
| Relation Classification | Tree-structured GRU Hidden Unit Dimension | 100 |

## 4.2 Results

**Comparison with the baseline model.**

Table 3 compares our model with the baseline model of Miwa M. and Bansal M. [5] on the different reports from the thyroid and mammary gland dataset. Results show that our model significantly achieve the better performance, due to the statistical significance($p<0.01$) using the Wilcoxon rank-sum test on F1-scores.

**Table 3.** Comparison with the baseline model on the thyroid and mammary gland dataset

| | | Miwa M. and Bansal M. [5] | | | Our model | | |
|---|---|---|---|---|---|---|---|
| Dataset | Report | P | R | F1 | P | R | F1 |
| Thyroid | Ultrasonic | 72.0 | 73.7 | 72.8 | 83.4 | 85.7 | 84.5 |
| | X-ray/CT | 71.4 | 74.9 | 73.1 | 80.0 | 89.7 | 84.6 |
| | Puncture | 79.6 | 77.6 | 78.6 | 83.8 | 82.9 | 83.3 |
| | Pathology | 69.3 | 76.0 | 72.5 | 77.5 | 83.3 | 80.3 |
| Mammary gland | Ultrasonic | 76.5 | 72.7 | 74.6 | 80.8 | 83.3 | 82.1 |
| | X-ray/CT | 78.2 | 78.9 | 78.6 | 83.6 | 88.9 | 86.2 |
| | Puncture | 78.5 | 78.1 | 78.3 | 84.4 | 83.8 | 84.1 |
| | Pathology | 68.9 | 74.6 | 71.6 | 82.0 | 86.4 | 84.1 |

**Entity-Level Attention.**

To measure the contribution and effect of entity-level attention, we conduct the tests of composition change to compare three types of dependency layers with different selection of weighted nodes on four kinds of reports from the thyroid dataset. Shortest Path

Tree(SP-Tree) [5] only consists of the nodes on the shortest path in dependency parsing tree between the target entity pairs, SubTree selects the nodes in the subtree under the lowest common ancestor of the target entity pair and FullTree take all the nodes into the entity-level attention. Results show the performance of F1 is significantly upgraded with assigned weights of other word nodes($p<0.1$). [2] We find the F1 performance no worse than no attention mechanism but the improvement is slight (about 1-2%) in puncture reports because the syntax information puncture reports are commonly less complicated than others. SP-Tree, SubTree and FullTree respectively improve F1 on the whole thyroid dataset by 8.4%, 7.2%, 6.7%, and especially boost the recall score, compared with no entity-attention, but we do not observe significant difference between them. SP-Tree and FullTree achieve similar scores and SubTree slightly improves F1, which is slightly different from traditional information-sparse texts. However, SP-Tree is the most time-saving method, SubTree is the second and FullTree takes the longest amount of time.

**Table 4.** Comparison of different entity-level attention on the whole thyroid dataset

| Type | Method | P | R | F1 |
|---|---|---|---|---|
| No attention | SP-Tree | 76.5 | 77.0 | 76.7 |
| | SubTree | 79.4 | 76.9 | 78.2 |
| | FullTree | 78.1 | 78.0 | 78.0 |
| Attention | SP-Tree | 82.2 | 84.2 | 83.2 |
| | SubTree | 82.1 | 85.6 | 83.8 |
| | FullTree | 82.5 | 84.0 | 83.3 |

**Sub Sentence-Level Attention.**

To assess the sub sentence-level attention mechanism, we also conduct the tests of composition change to compare no attention and two types attention mechanism on four kinds of reports from the thyroid dataset. “No attention” directly outputs the prediction through the softmax layer, “Simple Attention” [7] simply uses a weighted sum of all the sub-sentences including a waited classification of relation in one sentences and not distinguish the target pairs from other context sub-sentences. “Context Attention” uses the method as we mentioned in Section 3, fully exploits the interaction between the target entity pairs and its context information.

Compared with “Context Attention”, F1 is significantly hurt without attention($p<0.1$).[2] No sub sentence-level attention degrades at least 4.2% in F1 on every kind of report and achieves a 8.6% reduction of F1 on the whole Thyroid dataset. “Context Attention” performs better than “Simple Attention”, though the difference in prediction is not big.

[2] The statistical significance is computed by the Wilcoxon rank-sum test on F1-scores of every report from the thyroid dataset.

**Table 5.** Comparison of different sub sentence-level attention on the whole thyroid dataset

| Method | P | R | F1 |
|---|---|---|---|
| No Attention | 76.4 | 76.9 | 76.7 |
| Simple Attention [7] | 79.5 | 82.1 | 80.8 |
| Context Attention | 82.5 | 84.0 | 83.3 |

## 5 Conclusion

In this paper, we propose a context-aware end-to-end neural model with two level attention to exploit the interaction between these two sub tasks and considering latent syntactic information within entities themselves and multiple relations in a single sentence. Our experiments give the following findings: (1) In comparison to the baseline model, taking entity-level as well as sub-sentence level attention into account is beneficial to clinical texts. (2) Compared with no attention models, entity-level attention based on parsing paths between the entity pairs, has been demonstrated to be able to acquire underlying syntactic information within entities. Entity-level attention using SP-Tree is a good and effective choice when saving time is more prior based on the tests of composition change. (3) By incorporating attention at sub sentence-level, we are allowed to capture the interaction between a sub-sentence, including the target entity pair, and its context sub-sentences, including other entity pairs, in the same sentence. Context attention achieves significantly greater performance than no attention and shows a slight improvement compared with simple sentence-level attention.

**Acknowledgments.**

This work was supported by the Shanghai Innovation Action Project of Science and Technology (15511106900), the Science and Technology Development Foundation of Shanghai (16JC1400802), and the Shanghai Specific Fund Project for Information Development (XX-XXFZ-01-14-6349).

## 6 References

1. Roth D., Yih W.: Global inference for entity and relation identification via a linear programming formulation[J]. Introduction to statistical relational learning, 553-580(2007).
2. Kate R. J., Mooney R. J.: Joint entity and relation extraction using card-pyramid parsing[C]. Fourteenth Conference on Computational Natural Language Learning. Association for Computational Linguistics, 203-212 (2010).
3. Li Q., Ji H.: Incremental Joint Extraction of Entity Mentions and Relations[C]. Meeting of the Association for Computational Linguistics. 402-412 (2014).
4. Miwa M., Sasaki Y.: Modeling Joint Entity and Relation Extraction with Table Representation[C]. Conference on Empirical Methods in Natural Language Processing. 944-948 (2014).
5. Miwa M., Bansal M.: End-to-End Relation Extraction using LSTMs on Sequences and Tree Structures[C]. Meeting of the Association for Computational Linguistics. 1105-1116 (2016).

6. Fei L., Zhang M., Fu G., Ji, D.: A neural joint model for entity and relation extraction from biomedical text[J]. Bmc Bioinformatics 18(1), 198 (2017).
7. Zhou P., Shi W., Tian J., Qi Z., Li B., Hao H., et al: Attention-Based Bidirectional Long Short-Term Memory Networks for Relation Classification[C]. Meeting of the Association for Computational Linguistics. 207-212 (2016).
8. Sorokin D, Gurevych I.: Context-Aware Representations for Knowledge Base Relation Extraction[C]. Conference on Empirical Methods in Natural Language Processing. (1784-1789) 2017.
9. Tai K. S., Socher R., Manning C. D.: Improved Semantic Representations From Tree-Structured Long Short-Term Memory Networks[J]. Computer Science 5(1), 36 (2015).
10. Passos A., Kumar V., Mccallum A.: Lexicon Infused Phrase Embeddings for Named Entity Resolution[J]. Computer Science, 2014.
11. Luo G., Huang X., Lin C. Y., Nie Z.: Joint Entity Recognition and Disambiguation[C]. Conference on Empirical Methods in Natural Language Processing. (879-888) 2016.
12. Huang Z., Xu W., Yu K.: Bidirectional LSTM-CRF Models for Sequence Tagging[J]. Computer Science, 2015.
13. Lample G., Ballesteros M., Subramanian S., Kawakami K., Dyer C.: Neural Architectures for Named Entity Recognition[J]. 260-270 (2016).
14. Sak H., Senior A., Beaufays F.: Long short-term memory recurrent neural network architectures for large scale acoustic modeling[J]. Computer Science, (338-342) 2014.
15. Chung J., Gulcehre C., Cho K. H., Bengio Y.: Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling[J]. Eprint Arxiv, 2014.
16. Zelenko D., Aone C., Richardella A.: Kernel Methods for Relation Extraction.[J]. Journal of Machine Learning Research 3(3), (1083-1106) 2003.
17. Bunescu R. C., Mooney R. J.: A shortest path dependency kernel for relation extraction[C]. Conference on Human Language Technology and Empirical Methods in Natural Language Processing. Association for Computational Linguistics, (724-731) 2005.
18. Santos C. N. D., Xiang B., Zhou B.: Classifying Relations by Ranking with Convolutional Neural Networks[J]. Computer Science 86(86), (132-137) 2015.
19. Xu K., Feng Y., Huang S., Zhao D.: Semantic Relation Classification via Convolutional Neural Networks with Simple Negative Sampling[J]. Computer Science 71(7), (941-9) 2015.
20. Yan X., Mou L., Li G., Chen Y., Peng H., Jin Z.: Classifying Relations via Long Short Term Memory Networks along Shortest Dependency Path[J]. Computer Science 42(1), 56-61(2015).
21. Chiu J. P. C., Nichols E.: Named Entity Recognition with Bidirectional LSTM-CNNs[J]. Computer Science, 2015.
22. HanLP Tool, https://github.com/hankcs/HanLP.