Abstract

Document-level relation extraction (RE) poses new challenges compared to its sentence-level RE counterpart. One document commonly contains multiple entity pairs, and one entity pair occurs multiple times in the document associated with multiple possible relations. In this paper, we propose two novel techniques, adaptive thresholding and localized context pooling, to solve the multi-label and multi-entity problems. The adaptive thresholding replaces the global threshold for multi-label classification in the prior work by a learnable entities-dependent threshold. The localized context pooling directly transfers attention from pre-trained language models to locate relevant context that is useful to decide the relation. We experiment on three document-level RE benchmark datasets: DocRED, a recently released large-scale RE dataset, and two datasets CDR and GDA in the biomedical domain. Our ATLOP model achieves an F1 score of 63.4; and also significantly outperforms existing models on both CDR and GDA.

1 Introduction

Relation extraction (RE) aims to identify the relationship between two entities in a given text and plays an important role in information extraction. Existing work mainly focuses on sentence-level relation extraction, i.e., predicting the relationship between entities in a single sentence (Zeng et al., 2014; Miwa and Bansal, 2016; Zhang et al., 2018). However, large amounts of relationships, such as relational facts from Wikipedia articles and biomedical literature, are expressed by multiple sentences in real-world applications (Verga et al., 2018; Yao et al., 2019). This problem, commonly referred to as document-level relation extraction, necessitates models that can capture complex interactions among entities in the whole document.

Compared to sentence-level RE, document-level RE poses unique challenges. For sentence-level RE datasets such as TACRED (Zhang et al., 2017) and SemEval 2010 Task 8 (Hendrickx et al., 2009), a sentence only contains one entity pair to classify. On the other hand, for document-level RE, one document contains multiple entity pairs, and we need to classify the relations of them all at once. It requires the RE model to identity and focus on the part of the document with relevant context for a particular entity pair. In addition, one entity pair can occur many times in the document associated with distinct relations for document-level RE, in contrast to one relation per entity pair for sentence-level RE. This multi-entity (multiple entity pairs to classify in a document) and multi-label (multiple
relation types for a particular entity pair) properties of document-level relation extraction make it harder than its sentence-level counterpart. Figure 1 shows an example from the DocRED dataset (Yao et al., 2019). The task is to classify the relation types of pairs of entities (highlighted in color). For a particular entity pair (John Stanistreet, Bendigo), it expresses two relations place of birth and place of death by the first two sentences and the last sentence. Other sentences contain irrelevant information to this entity pair.

To tackle the multi-entity problem, most current approaches construct a document graph with dependency structures, heuristics, or structured attention (Peng et al., 2017; Liu and Lapata, 2018; Christopoulou et al., 2019; Nan et al., 2020), and then perform inference with graph neural models (Liang et al., 2016; Guo et al., 2019). The constructed graphs bridge entities that spread far apart in the document and thus alleviate the deficiency of RNN-based encoders (Hochreiter and Schmidhuber, 1997; Chung et al., 2014) in capturing long-distance information (Khandelwal et al., 2018). However, as transformer-based models (Vaswani et al., 2017) can implicitly model long-distance dependencies (Clark et al., 2019; Tenney et al., 2019), it is unclear whether graph structures still help on top of pre-trained language models such as BERT (Devlin et al., 2019). There have also been approaches to directly apply pre-trained language models without introducing graph structures (Wang et al., 2019b; Tang et al., 2020a). They simply average the embedding of entity tokens to obtain the entity embeddings and feed them into the classifier to get relation labels. However, each entity has the same representation in different entity pairs, which can bring noise from irrelevant context.

In this paper, instead of introducing graph structures, we propose a localized context pooling technique. This technique solves the problem of using the same entity embedding for all entity pairs. It enhances the entity embedding with additional context that is relevant to the current entity pair. Instead of training a new context attention layer from scratch, we directly transfer the attention heads from pre-trained language models to get entity-level attention. Then, for two entities in a pair, we merge their attentions by multiplication to find the context that is important to both of them.

For the multi-label problem, existing approaches reduce it to a binary classification problem. After training, a global threshold is applied to the class probabilities to get relation labels. This method involves heuristic threshold tuning and introduces decision errors when the tuned threshold from development data may not be optimal for all instances.

In this paper, we propose the adaptive thresholding technique, which replaces the global threshold with a learnable threshold class. The threshold class is learned with our adaptive-threshold loss, which is a rank-based loss that pushes the logits of positive classes above the threshold and pulls the logits of negative classes below in model training. At the test time, we return classes that have higher logits than the threshold class as the predicted labels or return NA if such class does not exist. This technique eliminates the need for threshold tuning, and also makes the threshold adjustable to different entity pairs, which leads to much better results.

By combining the proposed two techniques, we propose a simple yet effective relation extraction model, named ATLOP (Adaptive Thresholding and Localized cOntext Pooling), to fully utilize the power of pre-trained language models (Devlin et al., 2019; Liu et al., 2019). Experiments on three document-level relation extraction datasets, DocRED (Yao et al., 2019), CDR (Li et al., 2016), and GDA (Wu et al., 2019b), demonstrate that our ATLOP model significantly outperforms the state-of-the-art methods. The contributions of our work are summarized as follows:

- We propose adaptive-thresholding loss, which enables the learning of an adaptive threshold that is dependent on entity pairs and reduces the decision errors caused by using a global threshold.
- We propose localized context pooling, which transfers pre-trained attention to grab related context for entity pairs to get better entity representations.
- We conduct experiments on three public document-level relation extraction datasets. Experimental results demonstrate the effectiveness of our ATLOP model that achieves the new state-of-the-art performance on the three benchmark datasets.

2 Problem Formulation

Given a document $d$ and a set of entities $\{e_i\}_{i=1}^n$, the task of document-level relation extraction is to
predict a subset of relations from $\mathcal{R} \cup \{ \text{NA} \}$ between the entity pairs $(e_s, e_o)_{s,o=1\ldots n; s \neq o}$, where $\mathcal{R}$ is a pre-defined set of relations of interest, $e_s, e_o$ are identified as subject and object entities, respectively. An entity $e_i$ can occur multiple times in the document by bilinear function and sigmoid activation. This process is formulated as:

$$z_s = \tanh (W_s h_{e_i})$$

$$z_o = \tanh (W_o h_{e_o})$$

$$P(r | e_s, e_o) = \sigma (z_s^T W_r z_o + b_r),$$

where $W_s \in \mathbb{R}^{d \times d}$, $W_o \in \mathbb{R}^{d \times d}$, $W_r \in \mathbb{R}^{d \times d/k}$ are model parameters. The representation of one entity is the same among different entity pairs. To reduce the number of parameters in the bilinear classifier, we use the group bilinear (Zheng et al., 2019), which splits the embedding dimensions into $k$ equal-sized groups (Tang et al., 2020b) and applies bilinear within the groups:

$$z_s = \left[ z_s^1; \ldots; z_s^k \right],$$

$$z_o = \left[ z_o^1; \ldots; z_o^k \right],$$

$$P(r | e_s, e_o) = \sigma \left( \sum_{i=1}^{k} z_s^i W_r^i z_o^i + b_r \right),$$

where $W_r^i \in \mathbb{R}^{d/k \times d/k}$ for $i = 1\ldots k$ are model parameters, $P(r | e_s, e_o)$ is the probability that relation $r$ is associated with the entity pair $(e_s, e_o)$. In this way, we can reduce the number of parameters from $d^2$ to $d^2/k$. We use the binary cross entropy loss for training. During inference, we tune a global threshold $\theta$ that maximizes evaluation metrics ($F_1$ score for RE) on the development set and return $r$ as an associated relation if $P(r | e_s, e_o) > \theta$ or return NA if no relation exists.

Our enhanced base model achieves near state-of-the-art performance in our experiments, significantly outperforms existing BERT baselines.

### 4 Adaptive Thresholding

The RE classifier outputs the probability $P(r | e_s, e_o)$ within the range $[0, 1]$, which needs thresholding to be converted to relation labels. As the threshold neither has a closed-form solution nor is differentiable, a common practice for deciding threshold is enumerating several values in the range $(0, 1)$ and picking the one that maximizes the evaluation metrics ($F_1$ score for RE). However, the model may have different confidence for different entity pairs or classes in which one global threshold does not suffice. The number of relations varies (multi-label problem) and the models may not be globally calibrated so that the same probability does not mean the same for all
This threshold class learns an entities-dependent positive classes would have higher probabilities than \( TH \) and negative classes would have lower probabilities than \( TH \) to separate positive classes and negative classes: positive labels or return \( TH \) classes with higher logits than the \( TH \) classes (see Eq. (5)). At the test time, we return an adaptive-thresholding loss based on the standard function that considers the threshold and thus eliminates the need for tuning threshold value. It is a substitute for the global threshold with a learnable, adaptive one, \( TH \). Here we introduce a threshold class which has already learned token-level dependencies by multi-head self-attention (Vaswani et al.,

If an entity pair is classified correctly, the logits of positive labels should be higher than the threshold while those of negative labels should be lower. Here we introduce a threshold class \( TH \), which is automatically learned in the same way as other classes (see Eq.(5)). At the test time, we return classes with higher logits than the \( TH \) class as positive labels or return NA if such classes do not exist. This threshold class learns an entities-dependent threshold value. It is a substitute for the global threshold and thus eliminates the need for tuning threshold on the development set.

To learn the new model, we need a special loss function that considers the \( TH \) class. We design our adaptive-thresholding loss based on the standard categorical cross entropy loss. The loss function is broken down to two parts as shown below:

\[
L_1 = - \sum_{r \in P_T} \log \left( \frac{\exp (\logit_r)}{\sum_{r' \in P_T \cup \{TH\}} \exp (\logit_{r'})} \right), \\
L_2 = - \log \left( \frac{\exp (\logit_{TH})}{\sum_{r' \in N_T \cup \{TH\}} \exp (\logit_{r'})} \right), \\
L = L_1 + L_2.
\]

The proposed adaptive-thresholding loss is illustrated in Figure 2. It obtains a large performance gain to the global threshold in our experiments.

### 5 Localized Context Pooling

The logsumexp pooling (see Eq. (2)) accumulates the embedding of all mentions for an entity across the whole document and generates one embedding for this entity. The entity embedding is then used in the classification of all entity pairs. However, since some context may express relations unrelated to the entity pair, it is better to have a localized representation that only attends to the relevant context in the document that is useful to decide to relation(s) for the entity pair.

Therefore we propose the localized context pooling, where we enhance the embedding of an entity pair with an additional context embedding that is related to both entities. In this work, since we use pre-trained transformer-based models as the encoder, which has already learned token-level dependencies by multi-head self-attention (Vaswani et al.,...
we consider directly using their attention heads for localized context pooling. This method transfers the well-learned dependencies from the pre-trained language model without learning new attention layers from scratch.

Specifically, we use the token-level attention heads $A$ from the last transformer layer in the pre-trained language model, where $A_{ij} \in \mathbb{R}^{1 \times l}$ represents the importance of token $i$ for token $j$ in the $i^{th}$ attention head. For entity mention that spans from the $j'$ token ("*" symbol), we take $A_{ij}=j'$ as the mention-level attention, then average the attention over mentions of the same entity to obtain entity-level attentions $\{A^E_i\}_{i=1}^m$, where each attention $A^E_i \in \mathbb{R}^{H \times l}$ denotes the importance of context tokens to the $i^{th}$ entity in $H$ attention heads. Then for entity pair $(e_s, e_o)$, we obtain the context tokens that are important to both entities by multiplying their entity-level attentions followed by normalization:

$$A^{(s,o)} = A^E_s \cdot A^E_o,$$

$$q^{(s,o)} = \sum_{i=1}^H A_i^{(s,o)},$$

$$a^{(s,o)} = q^{(s,o)}/1^\top q^{(s,o)},$$

$$c^{(s,o)} = H^\top a^{(s,o)},$$

where $c^{(s,o)}$ is the localized contextual embedding for $(e_s, e_o)$. The contextual embedding is fused into the pooled entity embedding to obtain entity representations that are different for different entity pairs, by modifying the original linear layer in Eq. (3) and Eq. (4) as follows:

$$z^{(s,o)}_s = \tanh \left( W_s h_{e_s} + W_{c_1} c^{(s,o)} \right),$$

$$z^{(s,o)}_o = \tanh \left( W_o h_{e_o} + W_{c_2} c^{(s,o)} \right),$$

where $W_{c_1}, W_{c_2} \in \mathbb{R}^{d \times d}$ are model parameters. The proposed localized context pooling is illustrated in Figure 3.

### 6 Experiments

#### 6.1 Datasets

We evaluate our ATLOP model on three public document-level relation extraction datasets. The dataset statistics are shown in Table 1.

- **DocRED** (Yao et al., 2019) is a large-scale general-purpose dataset for document-level RE constructed from Wikipedia articles. It consists of 3053 human-annotated documents for training. For entity pairs that express relation(s), about 7% of them have more than one relation label.

- **CDR** (Li et al., 2016) is a human-annotated dataset in the biomedical domain. It consists of 500 documents for training. The task is to predict the binary interactions between Chemical and Disease concepts.

- **GDA** (Wu et al., 2019b) is a large-scale dataset in the biomedical domain. It consists of 29192 articles for training. The task is to predict the binary interactions between Gene and Disease concepts.

#### 6.2 Experiment Settings

Our model is implemented based on Pytorch\(^3\) and Huggingface’s Transformers\(^3\). We use cased BERT-base (Devlin et al., 2019) or RoBERTa-large (Liu et al., 2019) as the encoder on DocRED, and cased SciBERT-base (Beltagy et al., 2019) on CDR and GDA. We use mixed precision training (Micikevicius et al., 2018) based on the Apex library\(^4\). Our model is optimized with AdamW (Loshchilov and Hutter, 2019) using learning rates $\in \{2e^{-5}, 3e^{-5}, 5e^{-5}, 1e^{-4}\}$, with a lin-

---

\(^3\)https://pytorch.org/  
\(^3\)https://github.com/huggingface/transformers  
\(^4\)https://github.com/NVIDIA/apex

---

| Statistics / Dataset | DocRED | CDR | GDA |
|----------------------|--------|-----|-----|
| # Train              | 3053   | 500 | 23353 |
| # Dev                | 1000   | 500 | 5839 |
| # Test               | 3053   | 500 | 23353 |
| # Relations          | 97     | 2   | 2   |
| Avg.# entities per Doc. | 19.5 | 7.6 | 5.4 |

Table 1: Statistics of the datasets in experiments.

| Hyperparam / Dataset | DocRED | CDR | GDA |
|----------------------|--------|-----|-----|
| Batch size           | 4      | 4   | 16  |
| # Epoch              | 30     | 30  | 10  |
| Learning rate for encoder | 5e-5 | 2e-5 | 2e-5 |
| Learning rate for classifier | 1e-4 | 1e-4 | 1e-4 |
| Group size           | 64     | 64  | 64  |
| Dropout              | 0.1    | 0.1 | 0.1 |
| Gradient clipping    | 1.0    | 1.0 | 1.0 |

Table 2: Hyper-parameters of ATLOP.
Table 3: Main results (%) on the development and test set of DocRED. We report the mean and standard deviation of $F_1$ on the development set by conducting 5 runs of training using different random seeds.

| Model                        | Dev | Test |
|------------------------------|-----|------|
| **Sequence-based Models**    |     |      |
| CNN (Yao et al., 2019)       | 41.58 | 43.45  | 40.33 | 42.26 |
| BiLSTM (Yao et al., 2019)    | 48.87 | 50.94 | 48.78 | 51.06 |
| **Graph-based Models**       |     |      |
| BiLSTM-AGGCN (Guo et al., 2019) | 46.29 | 52.47 | 48.89 | 51.45 |
| BiLSTM-LSR (Nan et al., 2020) | 48.82 | 55.17 | 52.15 | 54.18 |
| BERT-LSR (Nan et al., 2020)  | 52.43 | 59.00 | 56.97 | 59.05 |
| **Transformer-based Models** |     |      |
| BERT$_{BASE}$ (Wang et al., 2019b) | - | 54.16 | - | - |
| BERT$_{BASE}$-TS (Wang et al., 2019b) | - | 54.42 | - | - |
| HIN-BERT$_{BASE}$ (Tang et al., 2020a) | 54.29 | 56.31 | 53.70 | 55.60 |
| CorefBERT$_{BASE}$ (Ye et al., 2020) | 55.32 | 57.51 | 54.54 | 56.96 |
| CorefRoBERTa$_{LARGE}$ (Ye et al., 2020) | 57.84 | 59.93 | 57.68 | 59.91 |
| **Our Methods**              |     |      |
| BERT$_{BASE}$ (our implementation) | 54.27±0.28 | 56.39±0.18 | - | - |
| BERT$_{BASE}$-E | 56.51±0.16 | 58.52±0.19 | - | - |
| BERT-ATLOP$_{BASE}$ | 59.22±0.15 | 61.09±0.16 | 59.31 | 61.30 |
| RoBERTa-ATLOP$_{LARGE}$ | 61.32±0.14 | 63.18±0.19 | 61.39 | 63.40 |

Table 4: Test $F_1$ score (in %) on CDR and GDA dataset. Our ATLOP model with the SciBERT encoder outperforms the current SOTA results.

| Model                        | CDR | GDA  |
|------------------------------|-----|------|
| BRAN (Verga et al., 2018)    | 62.1 | -    |
| CNN (Nguyen and Verspoor, 2018) | 62.3 | -    |
| EoG (Christopoulou et al., 2019) | 63.6 | 81.5 |
| LSR (Nan et al., 2020)       | 64.8 | 82.2 |
| SciBERT$_{BASE}$ (our implementation) | 65.1±0.6 | 82.5±0.3 |
| SciBERT-E$_{BASE}$           | 65.9±0.5 | 83.3±0.3 |
| SciBERT-ATLOP$_{BASE}$       | 69.4±1.1 | 83.9±0.2 |

6.3 Main Results

We compare ATLOP with sequence-based models, graph-based models, and transformer-based models on the DocRED dataset. The experiment results are shown in Table 3. Following Yao et al. (2019), we use $F_1$ and Ign $F_1$ in evaluation. The Ign $F_1$ denotes the $F_1$ score excluding the relational facts that are shared by the training and dev/test sets.

Sequence-based Models. These models use neural architectures such as CNN (LeCun et al., 2015) and bidirectional LSTM (Schuster and Paliwal, 1997) to encode the entire document, then obtain entity embeddings and predict relations for each entity pair with bilinear function.

Graph-based Models. These models construct document graphs by learning latent graph structures of the document and perform inference with graph neural network (Kipf and Welling, 2017). We include two state-of-the-art graph-based models, AGGCN (Guo et al., 2019) and LSR (Nan et al., 2020), for comparison. The result of AGGCN is from the re-implementation by Nan et al. (2020).

Transformer-based Models. These models adapt pre-trained language models to document-level RE without using graph structures. They can be further divided into pipeline models (BERT-TS (Wang et al., 2019b)), hierarchical models (HIN-BERT (Tang et al., 2020a)), and pre-training methods (CorefBERT and CorefRoBERTa (Ye et al., 2020)). We also include BERT baseline (Wang et al., 2019b) in our comparison.

We find that our re-implemented BERT baseline gets significantly better results than Wang et al. (2019b), and outperforms the state-of-the-art RNN-based model BiLSTM-LSR by 1.2%. It demonstrates that pre-trained language models can capture long-distance dependencies among entities without explicitly using graph structures. After integrating other techniques, our enhanced baseline BERT-E$_{BASE}$ achieves an F1 score of 58.52%,
which is close to the current state-of-the-art model BERT-LSR<sub>BASE</sub>. Our BERT-ATLOP<sub>BASE</sub> model further improves the performance of BERT-E<sub>BASE</sub> by 2.6%, demonstrating the efficacy of the proposed novel techniques. Using RoBERTa-large as the encoder, our ATLOP model achieves an F1 score of 63.40%, which is a new state-of-the-art result on DocRED. We held the first position on Colab leaderboard<sup>5</sup> as of September 1st, 2020.

### 6.5 Ablation Study

To show the efficacy of our proposed techniques, we conduct two sets of ablation studies on ATLOP and enhanced baseline, by turning off one component at a time. We observe that all components contribute to model performance. The adaptive thresholding and localized context pooling are equally important to model performance, leading to a drop of 0.89% and 0.97% in dev F1 score respectively when removed from ATLOP. Note that the adaptive thresholding only works when the model is optimized with the adaptive-thresholding loss. Applying adaptive thresholding to models trained with binary cross entropy results in dev F1 of 41.74%.

For our enhanced baseline model BERT-E<sub>BASE</sub>, both group bilinear and logsumexp pooling lead to about 1% increase in dev F1 score. We find the improvement from entity markers is minor (0.24% in dev F1) but still use the technique in the model as it makes the derivation of mention embedding and mention-level attention easier.

### 6.6 Analysis of Thresholding

Global thresholding does not consider the variations of model confidence in different classes or instances, and thus yields suboptimal performance. One interesting problem is whether we can improve global thresholding by tuning different thresholds for different classes. Thus we experiment on tuning class-dependent thresholds to maximize the F1 score on the development set of DocRED using the cyclic optimization algorithm (Fan and Lin, 2007). Results are shown in Table 6. We find that using per-class thresholding significantly improves the dev F1 score to 61.73%, which is even higher than the result of adaptive thresholding. However, this gain does not transfer to the test set. The result of per-class thresholding is even worse than that of global thresholding. While our adaptive thresholding technique uses a learnable threshold that can automatically generalize to the test set.

---

<sup>5</sup>https://competitions.codalab.org/competitions/20717

---

Table 5: Ablation study of ATLOP on DocRED. We turn off different components of the model one at a time. These ablation results show that both adaptive thresholding and localized context pooling are effective. Logsumexp pooling and group bilinear both bring noticeable gain to the baseline.

| Model               | Ign F1 | F1  |
|---------------------|--------|-----|
| BERT-ATLOP<sub>BASE</sub> | 59.22  | 61.09 |
| - Adaptive Thresholding | 58.32  | 60.20 |
| - Localized Context Pooling | 58.19  | 60.12 |
| - Adaptive-Thresholding Loss | 39.52  | 41.74 |
| BERT-E<sub>BASE</sub>  | 56.51  | 58.52 |
| - Entity Marker      | 56.22  | 58.28 |
| - Group Bilinear     | 55.51  | 57.54 |
| - Logsumexp Pooling | 55.35  | 57.40 |

Table 6: Result of different thresholding strategies on DocRED. Our adaptive thresholding consistently outperforms other strategies on the test set.

| Strategy                  | Dev F1 | Test F1 |
|---------------------------|--------|---------|
| Global Thresholding       | 60.14  | 60.62   |
| Per-class Thresholding    | 61.73  | 60.35   |
| Adaptive Thresholding     | 61.27  | 61.30   |

6.4 Results on Biomedical Datasets

Experiment results on two biomedical datasets are shown in Table 4. Verga et al. (2018) and Nguyen and Verspoor (2018) are both sequence-based models that use self attention network and CNN as the encoders, respectively. Christopoulou et al. (2019) and Nan et al. (2020) use graph-based models that construct document graphs by heuristics or structured attention, and perform inference with graph neural network. To our best knowledge, transformer-based pre-trained language models have not been applied to document-level RE datasets in the biomedical domain. In experiments, we replace the encoder with SciBERT<sub>BASE</sub>, which is pre-trained on multi-domain corpora of scientific publications. The SciBERT<sub>BASE</sub> baseline already outperforms all existing methods. Our SciBERT-ATLOP<sub>BASE</sub> model further improves the F1 score by 4.3% and 1.4% on CDR and GDA, respectively, and yields the new state-of-the-art results on these two datasets.
6.7 Analysis of Context Pooling

To show that our localized context pooling (LOP) technique mitigates the multi-entity issue, we divide the documents in the development set of DocRED into different groups by the number of entities, and evaluate models trained with or without localized context pooling on each group. Experiment results are shown in Figure 4. We observe that for both models, their performance gets worse when the number of entities contains more entities. The model w/ LOP consistently outperforms the model w/o LOP except when the document contains very few entities (1 to 5), and the improvement gets larger when the number of entities increases. However, the number of documents that only contain 1 to 5 entities is very small (4 in the dev set), and the documents in DocRED contain 19 entities on average. Therefore our localized context pooling still improves the overall F1 score significantly. This indicates that the localized context pooling technique can capture related context for entity pairs and thus alleviates the multi-entity problem.

We also visualize the context weights of the example in Figure 1. As shown in Figure 5, our localized context pooling gives high weights to born and died, which are most relevant to both entities (John Stanistreet, Bendigo). These two tokens are also evidence for the two ground truth relationships place of birth and place of death, respectively. Tokens like elected and politician get much smaller weights because they are only related to the subject entity John Stanistreet. The visualization demonstrates that the localized context can locate the context that is related to both entities.

7 Related Work

Early research efforts on relation extraction concentrate on predicting the relationship between two entities within a sentence. Various approaches including sequence-based methods (Zeng et al., 2014; Wang et al., 2016; Zhang et al., 2017), graph-based methods (Miwa and Bansal, 2016; Zhang et al., 2018; Guo et al., 2019; Wu et al., 2019a), transformer-based methods (Alt et al., 2019; Shi and Lin, 2019), and pre-training methods (Zhang et al., 2019; Soares et al., 2019) have been shown effective in tackling this problem.

However, as large amounts of relationships are expressed by multiple sentences (Verga et al., 2018; Yao et al., 2019), recent work starts to explore document-level relation extraction. Most approaches on document-level RE are based on document graphs, which were introduced by Quirk and Poon (2017). Specifically, they use words as nodes and inner and inter-sentential dependencies (dependency structures, coreferences, etc.) as edges. This document graph provides a unified way of extracting the features for entity pairs. Later work extends the idea by improving neural architectures (Peng et al., 2017; Verga et al., 2018; Song et al., 2018; Jia et al., 2019; Gupta et al., 2019) or adding more types of edges (Christopoulou et al., 2019; Nan et al., 2020). In particular, Christopoulou et al. (2019) constructs nodes of different granularities (sentence, mention, entity), connects them with heuristically generated edges, and infers the relations with an edge-oriented model (Christopoulou et al., 2018). Nan et al. (2020) treats the document graph as a latent variable and induces it by structured attention (Liu and Lapata, 2018). Their LSR model achieved state-of-the-art performance on document-level RE.

There have also been models that directly apply pre-trained language models without introducing document graphs, since edges such as dependency
structures and coreferences can be automatically learned by pre-trained language models (Clark et al., 2019; Tenney et al., 2019; Vig and Belinkov, 2019; Hewitt and Manning, 2019). In particular, Wang et al. (2019b) proposes a pipeline model that first predicts whether a relationship exists in an entity pair and then predicts the specific relation types. Tang et al. (2020a) proposes a hierarchical model that aggregates entity information from the entity level, sentence level, and document level. Ye et al. (2020) introduces a copy-based training objective to the pre-training stage of language models. However, none of the models focus on the multi-entity and multi-label problems, which are among the key differences of document-level RE to its sentence-level RE counterpart. Our ATLOP model deals with the problems by two techniques: adaptive thresholding and localized context pooling, and significantly outperforms existing models.

8 Conclusion
In this work, we propose the ATLOP model for document-level relation extraction, which features two novel techniques: adaptive thresholding and localized context pooling. The adaptive thresholding technique replaces the global threshold in multi-label classification with a learnable threshold class that can decide the best threshold for each entity pair. The localized context pooling utilizes pre-trained attention heads to locate relevant context for entity pairs and thus helps in alleviating the multi-entity problem. Experiments on three public document-level relation extraction datasets demonstrate that our ATLOP model significantly outperforms existing models and yields the new state-of-the-art results on all datasets.

References
Christoph Alt, Marc Hübner, and Leonhard Hennig. 2019. Improving relation extraction by pre-trained language representations. ArXiv, abs/1906.03088.
Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. Scibert: A pretrained language model for scientific text. In EMNLP/IJCNLP.
Fenia Christopoulou, Makoto Miwa, and Sophia Ananiadou. 2018. A walk-based model on entity graphs for relation extraction. ArXiv, abs/1902.07023.
Fenia Christopoulou, Makoto Miwa, and Sophia Ananiadou. 2019. Connecting the dots: Document-level neural relation extraction with edge-oriented graphs. In EMNLP/IJCNLP.
J. Li, Yueping Sun, Robin J. Johnson, Daniela Sciaky, Chih-Hsuan Wei, Robert Leaman, A. P. Davis, C. Mattingly, Thomas C. Wiegers, and Zhiyong Lu. 2016. Biocreative v cdr task corpus: a resource for chemical disease relation extraction. Database: The Journal of Biological Databases and Curation, 2016.

Xiaodan Liang, X. Shen, Jiashi Feng, L. Lin, and S. Yan. 2016. Semantic object parsing with graph lstm. ArXiv, abs/1603.07063.

Y. Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, M. Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. ArXiv, abs/1907.11692.

Yang Liu and Mirella Lapata. 2018. Learning structured text representations. Transactions of the Association for Computational Linguistics, 6:63–75.

I. Loshchilov and F. Hutter. 2019. Decoupled weight decay regularization. In ICLR.

A. Menon, A. Rawat, S. Reddi, and S. Kumar. 2019. Multilabel reductions: what is my loss optimising? In NeurIPS.

P. Micikevicius, Sharan Narang, J. Alben, G. Diamos, E. Elsen, D. Garcia, B. Ginsburg, Michael Houston, O. Kuchaiev, G. Venkatesh, and H. Wu. 2018. Mixed precision training. ArXiv, abs/1710.03740.

Makoto Miwa and Mohit Bansal. 2016. End-to-end relation extraction using lstms on sequences and tree structures. ArXiv, abs/1601.00770.

G. Nan, Zhijiang Guo, Ivan Sekulic, and W. Lu. 2020. Reasoning with latent structure refinement for document-level relation extraction. ArXiv, abs/2005.06312.

Dat Quoc Nguyen and Karin M. Verspoor. 2018. Convolutional neural networks for chemical-disease relation extraction are improved with character-based word embeddings. ArXiv, abs/1805.10586.

Nanyun Peng, Hoifung Poon, Chris Quirk, Kristina Toutanova, and Wen tau Yih. 2017. Cross-sentence n-ary relation extraction with graph lstms. Transactions of the Association for Computational Linguistics, 5:101–115.

Chris Quirk and Hoifung Poon. 2017. Distant supervision for relation extraction beyond the sentence boundary. ArXiv, abs/1609.04873.

S. Reddi, S. Kale, F. Yu, Daniel N. Holtmann-Rice, Jiecao Chen, and S. Kumar. 2019. Stochastic negative mining for learning with large output spaces. ArXiv, abs/1810.07076.

Mike Schuster and K. Paliwal. 1997. Bidirectional recurrent neural networks. IEEE Trans. Signal Process., 45:2673–2681.

Peng Shi and Jimmy Lin. 2019. Simple bert models for relation extraction and semantic role labeling. ArXiv, abs/1904.05255.

Livio Baldini Soares, N. FitzGerald, Jeffrey Ling, and T. Kwiatkowski. 2019. Matching the blanks: Distributional similarity for relation learning. ArXiv, abs/1906.03158.

Linfeng Song, Yue Zhang, Z. Wang, and Daniel Gildea. 2018. N-ary relation extraction using graph state lstm. In EMNLP.

Hengzhu Tang, Yanan Cao, Zhenyu Zhang, Jiangxia Cao, Fang Fang, Shigang Wang, and Pengfei Yin. 2020a. Hin: Hierarchical inference network for document-level relation extraction. Advances in Knowledge Discovery and Data Mining, 12084:197 – 209.

Yun Tang, Jing Huang, Guangtao Wang, Xiaodong He, and Bowen Zhou. 2020b. Orthogonal relation transforms with graph context modeling for knowledge graph embedding. In ACL.

Ian Tenney, Dipanjan Das, and Ellie Pavlick. 2019. Bert rediscovers the classical nlp pipeline. In ACL.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, L. Kaiser, and Illia Polosukhin. 2017. Attention is all you need. ArXiv, abs/1706.03762.

Pat Verga, Emma Strubell, and Andrew McCallum. 2018. Simultaneously self-attending to all mentions for full-abstract biological relation extraction. In NAACL-HLT.

J. Vig and Yonatan Belinkov. 2019. Analyzing the structure of attention in a transformer language model. ArXiv, abs/1906.04284.

Haoyu Wang, M. Tan, Mo Yu, S. Chang, Dakuo Wang, Kun Xu, Xiaoxiao Guo, and Saloni Potdar. 2019a. Extracting multiple-relations in one-pass with pretrained transformers. ArXiv, abs/1902.01030.

Hong Wang, Christfried Focke, Rob Sylvester, Nilesh Mishra, and William W. J. Wang. 2019b. Finite-tune bert for docred with two-step process. ArXiv, abs/1909.11898.

L. Wang, Z. Cao, G. Melo, and Zhiyuan Liu. 2016. Relation classification via multi-level attention cnns. In ACL.

Felix Wu, Tianyi Zhang, A. Souza, Christopher Fifty, Tao Yu, and Kilian Q. Weinberger. 2019a. Simplifying graph convolutional networks. In ICML.

Y. Wu, Ruibang Luo, H. Leung, H. Ting, and T. Lam. 2019b. Renet: A deep learning approach for extracting gene-disease associations from literature. In RECOMB.
Yuan Yao, D. Ye, Peng Li, Xu Han, Yankai Lin, Zheng-hao Liu, Z. Liu, Lixin Huang, Jie Zhou, and M. Sun. 2019. Docred: A large-scale document-level relation extraction dataset. In ACL.

Deming Ye, Yankai Lin, Jiaju Du, Zhenghao Liu, Maosong Sun, and Zhiyuan Liu. 2020. Coreferential reasoning learning for language representation. ArXiv, abs/2004.06870.

Daojian Zeng, Kang Liu, Siwei Lai, G. Zhou, and Jun Zhao. 2014. Relation classification via convolutional deep neural network. In COLING.

Yuhao Zhang, Peng Qi, and Christopher D. Manning. 2018. Graph convolution over pruned dependency trees improves relation extraction. In EMNLP.

Yuhao Zhang, Victor Zhong, Danqi Chen, Gabor Angeli, and Christopher D. Manning. 2017. Position-aware attention and supervised data improve slot filling. In EMNLP.

Zhengyan Zhang, Xu Han, Z. Liu, Xin Jiang, M. Sun, and Qun Liu. 2019. Ernie: Enhanced language representation with informative entities. ArXiv, abs/1905.07129.

Heliang Zheng, J. Fu, Z. Zha, and Jiebo Luo. 2019. Learning deep bilinear transformation for fine-grained image representation. ArXiv, abs/1911.03621.