GADePo: Graph-Assisted Declarative Pooling Transformers for Document-Level Relation Extraction

Anonymous ACL submission

Abstract

Document-level relation extraction typically relies on text-based encoders and hand-coded pooling heuristics to aggregate information learned by the encoder. In this paper, we leverage the intrinsic graph processing capabilities of the Transformer model and propose replacing hand-coded pooling methods with new tokens in the input, which are designed to aggregate information via explicit graph relations in the computation of attention weights. We introduce a joint text-graph Transformer model and a graph-assisted declarative pooling (GADePo) specification of the input, which provides explicit and high-level instructions for information aggregation. GADePo allows the pooling process to be guided by domain-specific knowledge or desired outcomes but still learned by the Transformer, leading to more flexible and customisable pooling strategies. We evaluate our method across diverse datasets and models and show that our approach yields promising results that are consistently better than those achieved by the hand-coded pooling functions.

1 Introduction

Document-level relation extraction is an important task in natural language processing, which involves identifying and categorising meaningful relationships between entities within a document, as exemplified in Figure 1. This task is foundational to many applications, including knowledge base population and completion (Banko et al., 2007; Ji et al., 2020), information retrieval and extraction (Manning et al., 2008; Theodoropoulos et al., 2021), question answering (Chen et al., 2017; Feng et al., 2022) and sentiment analysis (Pang and Lee, 2008), to name a few.

Standard methods that approach this challenge generally employ pretrained text-based encoders (Devlin et al., 2019; Beltagy et al., 2019; Zhuang et al., 2021; Cui et al., 2021), which are responsible for capturing the nuances of information contained in the entity mentions and their contextual surroundings. Previous successful methods often then use hand-coded pooling heuristics to aggregate the information learned by the encoder, with some aimed at creating entity representations, while others directly exploiting the pattern of attention weights to capture context aware relations between entity mentions (Zhou et al., 2021; Xiao et al., 2022; Tan et al., 2022a; Ma et al., 2023). These pooling heuristics can be very effective at leveraging the information in a pretrained encoder. However, as shown in Conneau et al. (2017); Jia et al. (2019); Reimers and Gurevych (2019); Choi et al. (2021), the selection of an appropriate pooling function can be model-dependent, task-specific, resource-intensive and time-consuming to determine, thereby limiting flexibility.

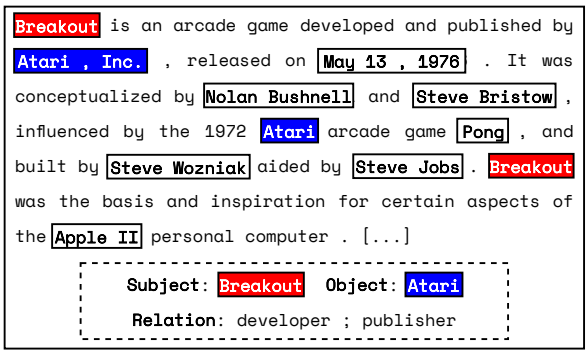


Figure 1: Document from the Re-DocRED (Tan et al., 2022b) dataset involving multiple entities and labels. Subject entity Breakout (red) and object entity Atari (blue) express relations "developer" and "publisher". Other entities are indicated as Mention (white).

In this paper, we address these issues with a new approach where we leverage the intrinsic graph processing capabilities of the Transformer model (Vaswani et al., 2017), leveraging insights from the work of Mohammadshahi and Henderson (2020); Henderson (2020); Mohammadshahi and Henderson (2021); Henderson et al. (2023). They argue that attention weights and graph relations are functionally equivalent and show how to incorporate

structural dependencies between input elements by simply adding relation features to the attention functions. Transformers easily learn to integrate these relation features into their pretrained attention functions, resulting in very successful graph-conditioned models (Mohammadshahi and Henderson, 2021; Miculicich and Henderson, 2022; Mohammadshahi and Henderson, 2023). Given this effective method for integrating explicit graphs with pretrained attention functions, we propose to use the attention function itself for aggregation. We replace the rigid pooling methods with new tokens which act as aggregation nodes, plus explicit graph relations which steer the aggregation.

We introduce a joint text-graph Transformer model and a graph-assisted declarative pooling (GADePo) method[1] that leverages these special tokens and graph relations, to provide an explicit high-level declarative specification for the information aggregation process. By integrating these graphs in the attention functions of a pretrained model, GADePo exploits the pretrained embeddings and attention patterns but still has the flexibility of being trained on data. This enables the pooling to be guided by domain-specific knowledge or desired outcomes but still learned by the Transformer, opening up a more customisable but still data-driven relation extraction process.

We evaluate our method across diverse datasets and models commonly employed in document-level relation extraction tasks, and show that our approach yields promising results that are consistently better than those achieved by the hand-coded pooling functions.

Contributions We propose a new method for exploiting pretrained Transformer models which replaces hand-coded aggregation functions with explicit graph relations and aggregation nodes. We introduce a novel form of joint text-graph Transformer model. We evaluate our approach across various datasets and models, showing that it yields promising results that are consistently better than those achieved by hand-coded pooling functions.

2 Related Work

In recent studies, the scope of relation extraction has been expanded to include not only individual sentences but entire documents. This extension, known as document-level relation extraction, presents a more realistic and challenging scenario as it seeks to extract relations both within sentences and across multiple sentences (Yao et al., 2019). Transformer-based (Vaswani et al., 2017) models have shown great potential in addressing this task.

Wang et al. (2019) and Tang et al. (2020) show that the BiLSTM-based (Hochreiter and Schmidhuber, 1997) baselines lack the capacity to model complex interactions between multiple entities. They propose a more robust approach, which consists of using the pretrained BERT (Devlin et al., 2019) model and a two-step prediction process, i.e., first identifying if a link between two entities exists, followed by predicting the specific relation type.

GAIN (Zeng et al., 2020) leverages BERT as a text encoder and GCNs (Kipf and Welling, 2017) to process two types of graphs, one at mention level and another at entity level, showing notable performance in inter-sentence and inferential scenarios.

Mohammadshahi and Henderson (2020, 2021) propose the G2GT model and show how to leverage the intrinsic graph processing capabilities of the Transformer model by incorporating structural dependencies between input elements as features input to the self-attention weight computations.

SSAN (Xu et al., 2021) leverages this idea and considers the structure of entities. It employs a transformation module that creates attentive biases from this structure to regulate the attention flow during the encoding phase.

DocuNet (Zhang et al., 2021) reformulates the task as a semantic segmentation problem. It employs a U-shaped segmentation module and an encoder module to capture global interdependencies and contextual information of entities, respectively.

PL-Marker (Ye et al., 2022) introduces a method that takes into account the interplay between spans via a neighbourhood-oriented and subject-oriented packing approach, highlighting the importance of capturing the interrelation among span pairs in relation extraction tasks.

SAIS (Xiao et al., 2022) explicitly models key information sources such as relevant contexts and entity types. It improves extraction quality and interpretability, while also boosting performance through evidence-based data augmentation and ensemble inference.

KD-DocRE (Tan et al., 2022a) proposes a semi-supervised framework with three key components. Firstly, an axial attention module enhances performance in handling two-hop relations by capturing the interdependence of entity pairs. Secondly,

[1] Code will be made available upon publication.

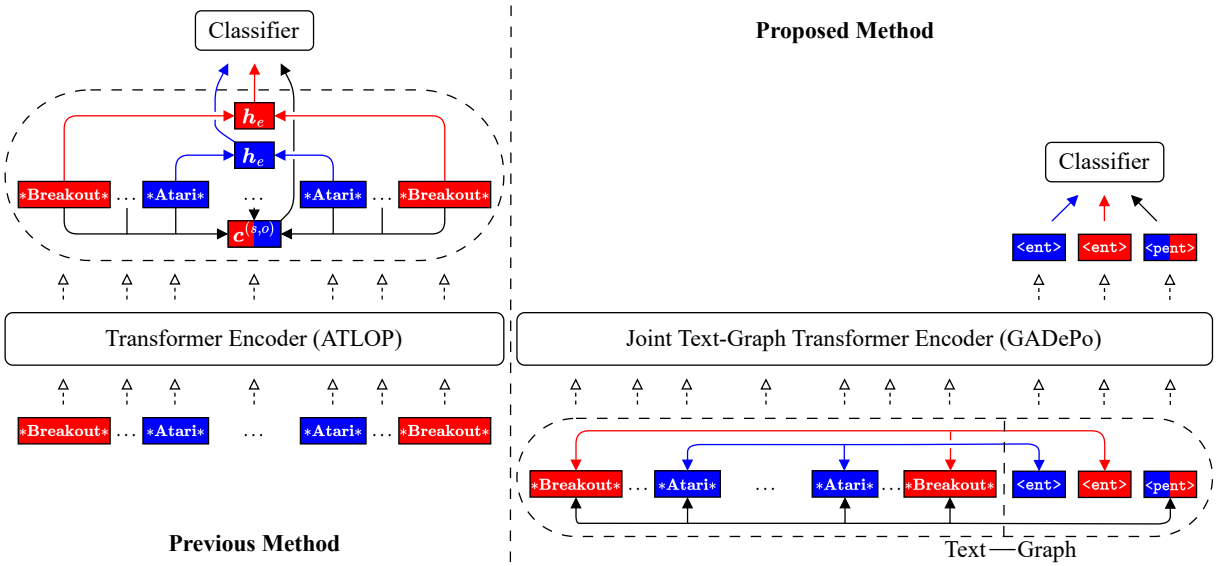


Figure 2: Comparison between the previous method ATLOP (left) and the proposed method GADePo (right), illustrating the document in Figure 1 containing two entities (red and blue), each with two mentions. In ATLOP, the mentions' encoder outputs are aggregated into entity representations $\boldsymbol{h}_e$, and the encoder's attention weights are used to identify which outputs to aggregate for entity-pair representations $\boldsymbol{c}^{(s,o)}$. In GADePo, the textual input is extended to include the graph special tokens `<ent>` for entity representations and `<pent>` for entity-pair representations, and explicit directional graph relations specify their associated mentions. A joint text-graph Transformer model is then used to encode this declarative pooling specification graph and compute the relevant aggregations.

an adaptive focal loss solution addresses the class imbalance issue. Lastly, the framework employs knowledge distillation to improve robustness and overall effectiveness by bridging the gap between human-annotated and distantly supervised data.

DREEAM (Ma et al., 2023) is a method designed to enhance document-level relation extraction by addressing memory efficiency and annotation limitations in evidence retrieval. It employs evidence as a supervisory signal to guide attention and introduces a self-training strategy to learn evidence retrieval without requiring evidence annotations.

SAIS (Xiao et al., 2022), KD-DocRE (Tan et al., 2022a), and DREEAM (Ma et al., 2023) have been built upon the foundations of ATLOP (Zhou et al., 2021). ATLOP introduces two innovative techniques, adaptive thresholding, and localised context pooling, to address challenges in multi-label and multi-entity problems. Adaptive thresholding employs a learnable entities-dependent threshold, replacing the global threshold used in previous approaches for multi-label classification (Peng et al., 2017; Christopoulou et al., 2019; Nan et al., 2020; Wang et al., 2020). Localised context pooling leverages the attention patterns of a pretrained language model to identify and extract relevant context crucial for determining the relation between entities, using specific hand-coded pooling functions.

3 Background

The foundational work of ATLOP (Zhou et al., 2021) has been the basis of many State-of-the-Art (SotA) models (Xiao et al., 2022; Tan et al., 2022a; Ma et al., 2023). Given the problems with hand-coded pooling functions, discussed in Section 1, we aim to provide a new baseline that can serve as the foundation for future SotA models. For this reason, we evaluate our proposed models by comparing them to this established baseline. Our goal is to demonstrate that our method not only achieves results comparable to or better than ATLOP, but also offers a novel approach which addresses its limitations. To provide a better understanding of ATLOP and its components, we present a detailed breakdown in the left portion of Figure 2, which we elaborate on in this section.

3.1 Problem Formulation

The document-level relation extraction task involves analysing a document D that contains a set of entities $\mathcal{E}_D=\{e_i\}_{i=1}^{|\mathcal{E}_D|}$. The main objective is to determine the presence or absence of various relation types between all entity pairs $(e_s, e_o)_{s,o\in\mathcal{E}_D,s\neq o}$, where the subject and object entities are denoted as e_s and e_o, respectively. A key aspect to consider is that an entity can appear multiple times in the document, resulting in a cluster of

multiple mentions $\mathcal{M}_e=\{m_i\}_{i=1}^{|\mathcal{M}_e|}$ for each entity e. The set of relations is defined as $\mathcal{R} \cup \emptyset$, where $\emptyset$ represents the absence of a relation, often referred to as "no-relation". Given the clusters of mentions $\mathcal{M}_{e_s}$ and $\mathcal{M}_{e_o}$, the task consists of a multi-label classification problem where there can be multiple relations between entities e_s and e_o.

3.2 Previous Method: ATLOP

Text Encoding A special token $*$ is added at the start and end of every mention. Tokens $\mathcal{T}_D=\{t_i\}_{i=1}^{|\mathcal{T}_D|}$ are encoded via a Pretrained Language Model (PLM) as follows:

$$\boldsymbol{H}, \boldsymbol{A} = PLM(\mathcal{T}_D), \tag{1}$$

where $\boldsymbol{H} \in \mathbb{R}^{|\mathcal{T}_D| \times d}$ and $\boldsymbol{A} \in \mathbb{R}^{|\mathcal{T}_D| \times |\mathcal{T}_D|}$ represent the token embeddings and the average attention weights of all attention heads, respectively, extracted from the last layer of the PLM.

Entity Embedding (EE) For each individual entity e with mentions $\mathcal{M}_e=\{m_i\}_{i=1}^{|\mathcal{M}_e|}$, an entity embedding $\boldsymbol{h}_e \in \mathbb{R}^d$ is computed as follows:

$$\boldsymbol{h}_e = log \sum_{i=1}^{|\mathcal{M}_e|} exp(\boldsymbol{H}_{m_i}), \tag{2}$$

where $\boldsymbol{H}_{m_i} \in \mathbb{R}^d$ is the embedding of the special token $*$ at the starting position of mention m_i. The choice of the $logsumexp$ pooling function is based on the research conducted by Jia et al. (2019). Their study offers empirical evidence that supports the use of this pooling function over others, as it facilitates accumulating weak signals from individual mentions, thanks to its smoother characteristics.

Localised Context Embedding (LCE) ATLOP introduces the concept of localised context embedding to accommodate the variations in relevant mentions and context for different entity pairs (e_s, e_o). Since the attention mechanism in the PLM captures the importance of each token within the context, it can be used to determine the context relevant for both entities. The importance of each token can be computed from the cross-token dependencies matrix $\boldsymbol{A}$ obtained in Equation 1. When evaluating entity e_s, the importance of individual tokens is determined by examining the cross-token dependencies across all mentions associated with e_s, denoted as $\mathcal{M}_{e_s}$. Initially, ATLOP collects and averages the attention $\boldsymbol{A}_{m_i} \in \mathbb{R}^{|\mathcal{T}_D|}$ at the special token $*$ preceding each mention $m_i \in \mathcal{M}_{e_s}$. This process results in $\boldsymbol{a}_s \in \mathbb{R}^{|\mathcal{T}_D|}$, which represents the importance of each token concerning entity e_s (and analogously $\boldsymbol{a}_o$ for e_o). Subsequently, the importance of each token for a given entity pair (e_s, e_o), denoted as $\boldsymbol{q}^{(s,o)} \in \mathbb{R}^{|\mathcal{T}_D|}$, is computed using $\boldsymbol{a}_s$ and $\boldsymbol{a}_o$ as follows:

$$\boldsymbol{q}^{(s,o)} = \frac{\boldsymbol{a}_s \circ \boldsymbol{a}_o}{\boldsymbol{a}_s^\top \boldsymbol{a}_o}, \tag{3}$$

where $\circ$ represents the Hadamard product. Consequently, $\boldsymbol{q}^{(s,o)}$ represents a distribution that indicates the importance of each token for both tokens in (e_s, e_o). Finally, the localised context embedding is computed as follows:

$$\boldsymbol{c}^{(s,o)} = \boldsymbol{H}^\top \boldsymbol{q}^{(s,o)}, \tag{4}$$

So $\boldsymbol{c}^{(s,o)} \in \mathbb{R}^d$ corresponds to a weighted average over all token embeddings that are important for both e_s and e_o.

Relation Classification and Loss Function The representations $\boldsymbol{h}_{e_s}$, $\boldsymbol{h}_{e_o}$ and $\boldsymbol{c}^{(s,o)}$ are input to a relation classifier, and the full model is fine-tuned to predict the relation labels for (e_s, e_o). The relation classifier and its loss function are detailed in Appendix Subsection A.1.

4 Proposed Method: GADePo

We propose to avoid the reliance on the EE (i.e., $\boldsymbol{h}_e$) and LCE (i.e., $\boldsymbol{c}^{(s,o)}$) heuristic aggregation functions by leveraging Transformers' attention functions to do aggregation. Given the observation of Henderson (2020); Mohammadshahi and Henderson (2020, 2021); Henderson et al. (2023) that attention weights and graph relations are functionally equivalent, we introduce the inductive biases of EE and LCE directly into the model's input as graph relations.

Our proposed graph-assisted declarative pooling (GADePo) method replaces the hand-coded aggregation functions EE and LCE with a declarative graph specification. By using the intrinsic graph processing capabilities of the Transformer model, the specified graph serves as an explicit high-level directive for the information aggregation process of the Transformer. By inputting the graph relations to the Transformer's self-attention layers, GADePo enables the aggregation to be steered by domain-specific knowledge or desired outcomes, while still allowing it to be learned by the Transformer, opening up the possibility for a more tailored and customised yet data-driven relation extraction.

Our GADePo model is illustrated in the right portion of Figure 2. We address both EE and LCE with the introduction of two special tokens, <ent> (i.e., entity) and <pent> (i.e., pair entity), and two explicit graph relations of types <ent> $\longleftrightarrow *$ and <pent> $\longleftrightarrow *$ in both directions, where $*$ represents the special token at the starting position of a specific mention. The set of relations is specified as $\boldsymbol{c}_{ij} \in \mathcal{C}$ which each identify the relation label from i to j. Each of these relation labels is associated with an embedding vector of dimension d, as are the special token inputs <ent> and <pent>. These two special tokens are added to the PLM's vocabulary of input tokens, while relation label embeddings are input to the self-attention functions for every pair of related tokens. These new embeddings represent learnable parameters that are trained during the PLM fine-tuning on the downstream tasks. As reported in Appendix Subsection A.2, GADePo adds a negligible number of extra parameters, namely only the special token inputs and the graph directional relation inputs.

Special Token <ent> To tackle the EE pooling function, we add to the input tokens $\mathcal{T}_D$ as many <ent> special tokens as entities in the document. This way each entity e has a corresponding entity token <ent> in the input. We connect each <ent> token with its corresponding cluster of mentions $\mathcal{M}_e{=}\{m_i\}_{i=1}^{|\mathcal{M}_e|}$, and vice-versa. The two graph relations we use are thus <ent> $\longrightarrow *$ and $*$ $\longrightarrow$ <ent>, where $*$ represents the special token at the starting position of mention m_i. Each <ent> token receives the same <ent> embedding, with no positional encoding, since each one collectively represents a set of mentions from different positions in the input graph. These identical inputs are only disambiguated through the connections to and from mentions expressed as the <ent> $\longrightarrow *$ and $*$ $\longrightarrow$ <ent> graph relations. These relations tell the self-attention mechanism to use the <ent> token to aggregate information from the associated mentions, and thus the <ent> tokens have a direct correspondence to the computed $\boldsymbol{h}_e$ in Equation 2.

Special Token <pent> ATLOP performs information filtering by calculating via Equation 4 a localised context embedding (LCE) $\boldsymbol{c}^{(s,o)}$ that is dependent on the cross-token attention matrix $\boldsymbol{A}$ output by the PLM. The intuition behind it is that the dependencies between different tokens are encoded as attention weights. We propose a straightforward adjustment of the input graph used for the EE pooling to effectively model and capture these dependencies. To address the LCE pooling function, we add to the input tokens $\mathcal{T}_D$ as many <pent> special tokens as the number of all possible pairs of entities. Each special token <pent> thus refers to a pair of entities (e_s, e_o). We connect each <pent> token with each mention in the two clusters of mentions $\mathcal{M}_{e_s}{=}\{m_i\}_{i=1}^{|\mathcal{M}_{e_s}|}$ and $\mathcal{M}_{e_o}{=}\{m_i\}_{i=1}^{|\mathcal{M}_{e_o}|}$ and vice-versa. Since the attention weights used in LCE are computed from these mention embeddings, we expect that they are sufficient for the Transformer to learn to find the relevant contexts. The two graph relations we use are thus <pent> $\longrightarrow *$ and $*$ $\longrightarrow$ <pent>. Analogously to the <ent> tokens, the <pent> tokens all receive the same <pent> embedding, with no positional embeddings, and thus are only disambiguated by their different <pent> $\longrightarrow *$ and $*$ $\longrightarrow$ <pent> graph relations. These relations tell the <pent> token to pay attention to its associated mentions, which in turn allows it to find the relevant context shared by these mentions. Thus, each <pent> token can be seen as having a direct correspondence to the computed $\boldsymbol{c}^{(s,o)}$ in Equation 4.

All equations relative to the relation classification and the corresponding loss function reported in Appendix Subsection A.1 remain valid as we merely substitute the hand-coded computations of $\boldsymbol{h}_e$ and $\boldsymbol{c}^{(s,o)}$ with the embeddings of <ent> and <pent>, respectively.

Text-Graph Encoding We follow Mohammadshahi and Henderson (2020, 2021); Henderson et al. (2023) in leveraging the intrinsic graph processing capabilities of the Transformer model by incorporating graph relations as relation embeddings input to the self-attention function. For every pair of input tokens ij, the pre-softmax attention weight $e_{ij} \in \mathbb{R}$ is computed from both the respective token embeddings $\boldsymbol{x}_i, \boldsymbol{x}_j \in \mathbb{R}^d$, and an embeddings of the graph relation $\boldsymbol{c}_{ij}$ between the i-th and j-th tokens. However, we change the attention weight computation to:

$$e_{ij} = \frac{\boldsymbol{x}_i \boldsymbol{W}_Q \ \mathrm{diag}(\mathrm{LN}(\boldsymbol{c}_{ij}\boldsymbol{W}_C)) \ (\boldsymbol{x}_j \boldsymbol{W}_K)^\top}{\sqrt{d}}, \tag{5}$$

where $\boldsymbol{W}_Q, \boldsymbol{W}_K \in \mathbb{R}^{d\times d}$ represent the query and key matrices, respectively. $\boldsymbol{c}_{ij} \in \{0,1\}^{|\mathcal{C}|}$ represents a $0/1$ encoded label of the graph relation between the i-th and j-th input elements, and $\boldsymbol{W}_C \in \mathbb{R}^{|\mathcal{C}|\times d}$ represents the relations' embedding

		Re-DocRED		HacRED		
Model	**Aggregation**	Ign F_1	F_1	P	R	F_1
ATLOP*	$\boldsymbol{h}_e$	75.27	75.92	**76.27**	76.83	76.55
GADePo (ours)	<ent>	**75.55**	**76.38**	74.13	**79.46**	**76.70**
ATLOP$^{\bullet,\diamond}$	$\boldsymbol{h}_e$; $\boldsymbol{c}^{(s,o)}$	76.82	77.56	77.89	76.55	77.21
ATLOP*	$\boldsymbol{h}_e$; $\boldsymbol{c}^{(s,o)}$	77.62	78.38	76.36	78.86	77.59
GADePo (ours)	<ent> ; <pent>	**77.70**	**78.40**	**78.27**	**79.03**	**78.65**

Table 1: Comparative analysis between the previous method ATLOP and the proposed method GADePo on the test set. ATLOP* indicates our reimplementation of the previous method. For Re-DocRED and HacRED we report in percentage the results obtained by Tan et al. (2022b) (ATLOP$^{\bullet}$) and Cheng et al. (2021) (ATLOP$^{\diamond}$), respectively. The results are reported in terms of F_1 scores, Precision (P), and Recall (R), following the same metrics reported in prior research specific to each dataset. Ign F_1 denotes the F_1 score that excludes relational facts shared between the training and evaluation sets. We also comply with the standard practice where test scores are determined based on the best checkpoint from five training runs with distinct random seeds.

matrix, so $\boldsymbol{c}_{ij}\boldsymbol{W}_C$ is the embedding of the relation between i and j. Finally, LN stands for the $LayerNorm$ operation and diag returns a diagonal matrix.

Compared to the standard attention function, where $e_{ij} = \boldsymbol{x}_i\boldsymbol{W}_Q(\boldsymbol{x}_j\boldsymbol{W}_K)^\top/\sqrt{d}$, the relation embedding determines a weighting of the different dimensions. This is a novel way to condition on the relation embedding compared to the original formulation, which only models query-relation interactions (Mohammadshahi and Henderson, 2020). This change is motivated by our task requiring a more flexible formulation which models query-relation-key interactions via a multiplicative mechanism, without requiring a full $d \times d$ matrix of bilinear parameters. This way, a key will be relevant to a query only when both agree on the relation. In preliminary experiments, we explored various methods for biasing attention and found that the formulation presented in Equation 5 produced the best results.

5 Experiments

5.1 Datasets and Models

Re-DocRED (Tan et al., 2022b) is a revisited version of the DocRED (Yao et al., 2019) dataset. It is built from English Wikipedia and Wikidata and contains both distantly-supervised and human-annotated documents with named entities, coreference data, and intra- and inter-sentence relations, supported by evidence. It requires analysing multiple sentences to identify entities, establish their relationships, and integrate information from the entire document. We comply with the model used by the authors and employ the RoBERTa$_{\text{LARGE}}$ (Zhuang et al., 2021) model in our experiments.

HacRED (Cheng et al., 2021) is a large-scale, high-quality Chinese document-level relation extraction dataset, with a special focus on practical hard cases. As the authors did not provide specific information about the model used in their study, we conducted our experiments using the Chinese BERT$_{\text{BASE}}$ with whole word masking model (Cui et al., 2021).

Datasets statistics Re-DocRED and HacRED exhibit notable distinctions in their statistics, as summarised in Table 2. Re-DocRED comprises a larger number of facts, entities per document, and relations compared to HacRED. This indicates a potentially richer and more extensive dataset in terms of factual information and relationship types. However, HacRED contains more documents and may present a broader range of scenarios for relation extraction, including more challenging cases, as it has been specifically created with a focus on practical hard cases.

Statistic	**Re-DocRED**	**HacRED**
Facts	120,664	65,225
Relations	96	26
Documents	4,053	9,231
Average Entities	19.4	10.8

Table 2: Re-DocRED and HacRED human-annotated datasets statistics.

5.2 Results and Discussion

We follow the standard practice from prior research and report the results of our experiments on the Re-DocRED and HacRED datasets in Table 1 and Figure 4. For all datasets and models, we provide our reimplementation of the ATLOP baseline (indicated as ATLOP*), which achieves or surpasses pre-

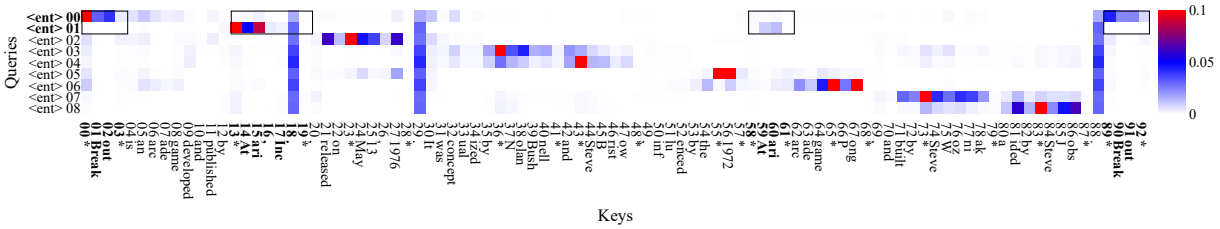


Figure 3: Attention weights $\boldsymbol{A}$ from GADePo via Equation 1 for the document in Figure 1. For clarity, only a subset of <ent> and document tokens are shown on the y-axis (queries) and x-axis (keys), respectively.

viously reported results for ATLOP, and compare the proposed GADePo model against this model. We evaluate all datasets using the F_1 metric. For Re-DocRED, Ign F_1 (or Ignored F_1) is also reported, and refers to the F_1 score that excludes relational facts that are shared between the training and development/test sets. This is done to avoid potential biases in the evaluation metrics due to overlap in content between the sets, which might not reflect the model's ability to generalise to truly unseen data. For HacRED, we adhere to the format introduced by Cheng et al. (2021) and report also the Precision (P) and Recall (R) metrics. We comply with previous research and report the test score achieved by the best checkpoint on the development set. In Appendix Subsection A.4, we additionally present the mean and standard deviation on the development set, calculated from five training runs with distinct random seeds. We also provide in Appendix Subsection A.4, the same set of experiments conducted on the original DocRED dataset. Training details and hyperparameters are outlined in Appendix Subsection A.3.

Re-DocRED Results We evaluate our proposed GADePo method against the previous ATLOP method in two stages, first comparing the use of <ent> tokens against the use of EE pooling ($\boldsymbol{h}_e$), and then comparing our full model against the full ATLOP model, including <pent> tokens and LCE pooling ($\boldsymbol{c}^{(s,o)}$), respectively.

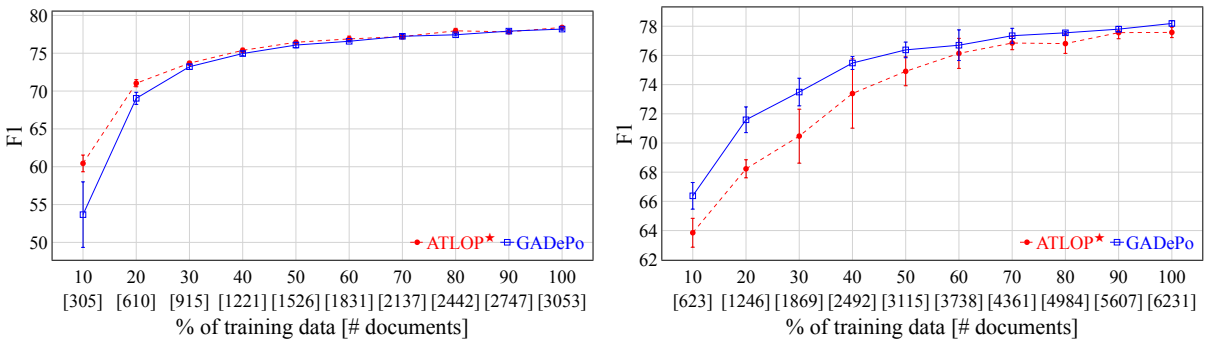
Table 1 highlights the effectiveness of our proposed method. When comparing $\boldsymbol{h}_e$ with <ent>, we observe a noticeable improvement in both Ign F_1 and F_1 scores, achieving 75.55% and 76.38% respectively, compared to 75.27% and 75.92% attained by ATLOP*. This demonstrates the practical utility of employing the special token <ent> for information aggregation. This is illustrated in the attention weights heatmap in Figure 3. Incorporating $\boldsymbol{c}^{(s,o)}$ and <pent> into the comparison, GADePo maintains performance parity with the significantly enhanced ATLOP*, which outperformed ATLOP$^\bullet$ from Tan et al. (2022b). The latter improvement suggests that a more refined hyperparameter search can lead to performance gains, as evidenced by the increase in F_1 score from 77.56% to 78.38%. GADePo achieves an F_1 score of 78.40%, affirming its competitive edge and the effectiveness of employing <pent> for aggregation.

Model	**Aggregation**	Ign F_1	F_1
ATLOP*	$\boldsymbol{h}_e$	76.39	76.97
GADePo (ours)	<ent>	**76.99**	**77.79**
ATLOP*	$\boldsymbol{h}_e$; $\boldsymbol{c}^{(s,o)}$	77.49	78.09
GADePo (ours)	<ent> ; <pent>	**77.50**	**78.15**

Table 3: Re-DocRED results on the test set following prior finetuning on the distantly supervised dataset.

Table 3 illustrates the results obtained with prior finetuning on the distantly supervised dataset, which contains approximately $100K$ documents (Yao et al., 2019). Interestingly, distant supervision appears to have a slightly negative impact on the results of both methods when incorporating $\boldsymbol{c}^{(s,o)}$ or <pent>. However, it proves to be highly beneficial when utilising solely $\boldsymbol{h}_e$ or <ent> for aggregation. This suggests that although distant supervision might introduce noise into the training process, it can also provide valuable information that improves model generalisation, particularly when leveraging simpler feature representations like $\boldsymbol{h}_e$ and <ent>, possibly due to their robustness in capturing essential information amidst noise.

HacRED Results We observe a similar pattern to Re-DocRED, with ATLOP* displaying a slight performance advantage over ATLOP$^\diamond$ from Cheng et al. (2021) (Table 1). On this dataset, GADePo shows a significantly improved performance, primarily driven by a substantial increase in Recall (R), indicating that the GADePo model is more effective at identifying relevant instances. As already reported for the Re-DocRED dataset, the performance boost after the inclusion of $\boldsymbol{c}^{(s,o)}$ and



(a) Re-DocRED (b) HacRED

Figure 4: Performance of ATLOP⋆ ($\boldsymbol{h}_e$; $\boldsymbol{c}^{(s,o)}$) and GADePo (<ent> ; <pent>) on the development set under varying data availability conditions on Re-DocRED (4a) and HacRED (4b). The x-axis represents the percentage and number of documents from the training dataset, while the y-axis displays the F_1 score in percentage. Each point on the graph represents the mean value, while error bars indicate the standard deviation derived from five distinct training runs with separate random seeds.

<pent> into ATLOP⋆ and GADePo, respectively, highlight the significant contributions of these features. GADePo outperforms ATLOP⋆ with an F_1 score of 78.65% compared to 77.59%. This larger improvement on HacRED suggests that GADePo is better at handling challenging cases, which is not surprising given its greater flexibility over the fixed pooling functions of ATLOP.

Data Ablation To evaluate the models' sensitivity to dataset size, the performance evaluation depicted in Figure 4 compares ATLOP⋆ ($\boldsymbol{h}_e$; $\boldsymbol{c}^{(s,o)}$) and GADePo (<ent> ; <pent>) on the development set, considering different levels of training data availability on the Re-DocRED and HacRED datasets. Accuracies generally converge as the dataset sizes increase, but on the challenging cases of HacRED, GADePo maintains a substantial advantage across the full range. On Re-DocRED, GADePo catches up with and slightly outperforms ATLOP⋆ as data size increases. This lower performance on smaller datasets is presumably because GADePo must learn how to exploit the graph relations to the special tokens <ent> and <pent> and pool information through them, whereas for ATLOP this pooling is hand-coded. On the Re-DocRED dataset, ATLOP⋆ appears to have relatively consistent variance, while GADePo exhibits higher variance in the smaller training sets, while on the HacRED dataset, GADePo is significantly more stable for smaller datasets.

The data ablation analysis shows that the performance of hand-coded pooling functions can be dataset-specific, which restricts their adaptability. In contrast, GADePo consistently outperforms its hand-coded counterparts on larger datasets, and matches them on all but some smaller datasets, presumably due to its flexibility. This pattern suggests that GADePo has a greater potential for optimisation, particularly on larger datasets. This is supported by GADePo's better performance on HacRED, which is both larger and designed to be more challenging than Re-DocRED.

6 Conclusion

In this paper we proposed a novel approach to document-level relation extraction, challenging the conventional reliance on hand-coded pooling functions for information aggregation. Our method leverages the power of Transformer models by incorporating explicit graph relations as instructions for information aggregation. By combining graph processing with text-based encoding, we introduced the graph-assisted declarative pooling (GADePo) specification, which allows for more flexible and customisable specification of pooling strategies which are still learned from data.

We conducted evaluations using diverse datasets and models commonly employed in document-level relation extraction tasks. The results of our experiments demonstrated that our approach achieves promising performance that is comparable to or better than that of hand-coded pooling functions. This suggests that our method can serve as a viable basis for other relation extraction methods, providing a more adaptable and tailored approach. In particular, recent methods have improved performance by exploiting information about evidence, which can naturally be incorporated in our graph-based approach.

Limitations

While the proposed GADePo model offers a promising and innovative approach to relation extraction, there are issues which the current study does not address. According to the data in Appendix Table 2, the average number of entities per document across datasets is approximately 15. This means that, on average, there will be an additional 15 <ent> tokens and 105 <pent> tokens. Given that the maximum allowable input length for the models is 512 tokens, the inclusion of these extra tokens results in roughly a 3% and 20% increase in the overall input length for <ent> and <pent>, respectively. It's evident that the majority of the increase in input length is due to the quadratic number of <pent> special tokens, but we believe that an appropriate pruning strategy could easily reduce this number to linear in the number of entities without degrading accuracy. One such pruning strategy could involve an <ent>-only model with a binary classifier which is trained to predict pairs of related entities. This model could then be used to prune the set of candidate entity pairs for the final relation classification, with <pent> tokens being instantiated only for these candidate pairs. We have chosen to leave this approach as a potential avenue for future work, opting instead to focus on demonstrating the promise of the current simpler formulation.

Ethics Statement

We do not anticipate any ethical concerns related to our work, as it primarily presents an alternative approach to a previously proposed method. Our main contribution lies in introducing a novel methodology for relation extraction. In our experiments, we use the same datasets and pretrained models as previous research, all of which are publicly available. However, it is important to acknowledge that these datasets and models may still require further examination for potential fairness issues and the knowledge they encapsulate.

References

Michele Banko, Michael J. Cafarella, Stephen Soderland, Matthew Broadhead, and Oren Etzioni. 2007. Open information extraction from the web. In *CACM*.

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. SciBERT: A pretrained language model for scientific text. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3615–3620, Hong Kong, China. Association for Computational Linguistics.

Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. 2017. Reading Wikipedia to answer open-domain questions. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1870–1879, Vancouver, Canada. Association for Computational Linguistics.

Qiao Cheng, Juntao Liu, Xiaoye Qu, Jin Zhao, Jiaqing Liang, Zhefeng Wang, Baoxing Huai, Nicholas Jing Yuan, and Yanghua Xiao. 2021. HacRED: A large-scale relation extraction dataset toward hard cases in practical applications. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 2819–2831, Online. Association for Computational Linguistics.

Hyunjin Choi, Judong Kim, Seongho Joe, and Youngjune Gwon. 2021. Evaluation of bert and albert sentence embedding performance on downstream nlp tasks. *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 5482–5487.

Fenia Christopoulou, Makoto Miwa, and Sophia Ananiadou. 2019. Connecting the dots: Document-level neural relation extraction with edge-oriented graphs. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4925–4936, Hong Kong, China. Association for Computational Linguistics.

Alexis Conneau, Douwe Kiela, Holger Schwenk, Loïc Barrault, and Antoine Bordes. 2017. Supervised learning of universal sentence representations from natural language inference data. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 670–680, Copenhagen, Denmark. Association for Computational Linguistics.

Yiming Cui, Wanxiang Che, Ting Liu, Bing Qin, and Ziqing Yang. 2021. Pre-training with whole word masking for chinese bert. *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, 29:3504–3514.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

William Falcon and The PyTorch Lightning team. 2019. PyTorch Lightning.

Yue Feng, Zhen Han, Mingming Sun, and Ping Li. 2022. Multi-hop open-domain question answering over structured and unstructured knowledge. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 151–156, Seattle, United States. Association for Computational Linguistics.

James Henderson. 2020. The unstoppable rise of computational linguistics in deep learning. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6294–6306, Online. Association for Computational Linguistics.

James Henderson, Alireza Mohammadshahi, Andrei Coman, and Lesly Miculicich. 2023. Transformers as graph-to-graph models. In *Proceedings of the Big Picture Workshop*, pages 93–107, Singapore. Association for Computational Linguistics.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Computation*, 9:1735–1780.

Shaoxiong Ji, Shirui Pan, E. Cambria, Pekka Marttinen, and Philip S. Yu. 2020. A survey on knowledge graphs: Representation, acquisition, and applications. *IEEE Transactions on Neural Networks and Learning Systems*, 33:494–514.

Robin Jia, Cliff Wong, and Hoifung Poon. 2019. Document-level n-ary relation extraction with multiscale representation learning. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3693–3704, Minneapolis, Minnesota. Association for Computational Linguistics.

Thomas N. Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations (ICLR)*.

Liyuan Liu, Haoming Jiang, Pengcheng He, Weizhu Chen, Xiaodong Liu, Jianfeng Gao, and Jiawei Han. 2020. On the variance of the adaptive learning rate and beyond. In *International Conference on Learning Representations*.

Youmi Ma, An Wang, and Naoaki Okazaki. 2023. DREEAM: Guiding attention with evidence for improving document-level relation extraction. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 1971–1983, Dubrovnik, Croatia. Association for Computational Linguistics.

Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. 2008. *Introduction to Information Retrieval*. Cambridge University Press.

Lesly Miculicich and James Henderson. 2022. Graph refinement for coreference resolution. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2732–2742, Dublin, Ireland. Association for Computational Linguistics.

Alireza Mohammadshahi and James Henderson. 2020. Graph-to-graph transformer for transition-based dependency parsing. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3278–3289, Online. Association for Computational Linguistics.

Alireza Mohammadshahi and James Henderson. 2021. Recursive non-autoregressive graph-to-graph transformer for dependency parsing with iterative refinement. *Transactions of the Association for Computational Linguistics*, 9:120–138.

Alireza Mohammadshahi and James Henderson. 2023. Syntax-aware graph-to-graph transformer for semantic role labelling. In *Proceedings of the 8th Workshop on Representation Learning for NLP (RepL4NLP 2023)*, pages 174–186, Toronto, Canada. Association for Computational Linguistics.

Guoshun Nan, Zhijiang Guo, Ivan Sekulic, and Wei Lu. 2020. Reasoning with latent structure refinement for document-level relation extraction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1546–1557, Online. Association for Computational Linguistics.

Bo Pang and Lillian Lee. 2008. Opinion mining and sentiment analysis. *Found. Trends Inf. Retr.*, 2:1–135.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.

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.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Qingyu Tan, Ruidan He, Lidong Bing, and Hwee Tou Ng. 2022a. Document-level relation extraction with adaptive focal loss and knowledge distillation. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1672–1681, Dublin, Ireland. Association for Computational Linguistics.

Qingyu Tan, Lu Xu, Lidong Bing, Hwee Tou Ng, and Sharifah Mahani Aljunied. 2022b. Revisiting DocRED - addressing the false negative problem in relation extraction. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 8472–8487, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Hengzhu Tang, Yanan Cao, Zhenyu Zhang, Jiangxia Cao, Fang Fang, Shi Wang, and Pengfei Yin. 2020. Hin: Hierarchical inference network for document-level relation extraction. In *Advances in Knowledge Discovery and Data Mining: 24th Pacific-Asia Conference, PAKDD 2020, Singapore, May 11–14, 2020, Proceedings, Part I 24*, pages 197–209. Springer.

Christos Theodoropoulos, James Henderson, Andrei Catalin Coman, and Marie-Francine Moens. 2021. Imposing relation structure in language-model embeddings using contrastive learning. In *Proceedings of the 25th Conference on Computational Natural Language Learning*, pages 337–348, Online. Association for Computational Linguistics.

Ashish Vaswani, Noam M. Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NIPS*.

Difeng Wang, Wei Hu, Ermei Cao, and Weijian Sun. 2020. Global-to-local neural networks for document-level relation extraction. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3711–3721, Online. Association for Computational Linguistics.

Hong Wang, Christfried Focke, Rob Sylvester, Nilesh Mishra, and William Yang Wang. 2019. Fine-tune bert for docred with two-step process. *ArXiv*, abs/1909.11898.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Yuxin Xiao, Zecheng Zhang, Yuning Mao, Carl Yang, and Jiawei Han. 2022. SAIS: Supervising and augmenting intermediate steps for document-level relation extraction. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2395–2409, Seattle, United States. Association for Computational Linguistics.

Benfeng Xu, Quan Wang, Yajuan Lyu, Yong Zhu, and Zhendong Mao. 2021. Entity structure within and throughout: Modeling mention dependencies for document-level relation extraction. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(16):14149–14157.

Yuan Yao, Deming Ye, Peng Li, Xu Han, Yankai Lin, Zhenghao Liu, Zhiyuan Liu, Lixin Huang, Jie Zhou, and Maosong Sun. 2019. DocRED: A large-scale document-level relation extraction dataset. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 764–777, Florence, Italy. Association for Computational Linguistics.

Deming Ye, Yankai Lin, Peng Li, and Maosong Sun. 2022. Packed levitated marker for entity and relation extraction. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4904–4917, Dublin, Ireland. Association for Computational Linguistics.

Shuang Zeng, Runxin Xu, Baobao Chang, and Lei Li. 2020. Double graph based reasoning for document-level relation extraction. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1630–1640, Online. Association for Computational Linguistics.

Ningyu Zhang, Xiang Chen, Xin Xie, Shumin Deng, Chuanqi Tan, Mosha Chen, Fei Huang, Luo Si, and Huajun Chen. 2021. Document-level relation extraction as semantic segmentation. In *International Joint Conference on Artificial Intelligence*.

Wenxuan Zhou, Kevin Huang, Tengyu Ma, and Jing Huang. 2021. Document-level relation extraction with adaptive thresholding and localized context pooling. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Liu Zhuang, Lin Wayne, Shi Ya, and Zhao Jun. 2021. A robustly optimized BERT pre-training approach with post-training. In *Proceedings of the 20th Chinese National Conference on Computational Linguistics*, pages 1218–1227, Huhhot, China. Chinese Information Processing Society of China.

A Appendix

A.1 ATLOP: Relation Classification and Loss Function

Relation Classification To predict the relation between the subject entity e_s and object entity e_o, ATLOP first generates context-aware subject and object representations as follows:

$$\boldsymbol{z}_s = tanh(\boldsymbol{W}_s[\boldsymbol{h}_{e_s}; \boldsymbol{c}^{(s,o)}] + \boldsymbol{b}_s) \quad (6)$$

$$\boldsymbol{z}_o = tanh(\boldsymbol{W}_o[\boldsymbol{h}_{e_o}; \boldsymbol{c}^{(s,o)}] + \boldsymbol{b}_o), \quad (7)$$

where $\boldsymbol{z}_s, \boldsymbol{z}_o \in \mathbb{R}^d$, $[\cdot;\cdot]$ represents the concatenation of two vectors, and $\boldsymbol{W}_s, \boldsymbol{W}_o \in \mathbb{R}^{d \times 2d}$ together

with $\boldsymbol{b}_s, \boldsymbol{b}_o \in \mathbb{R}^d$ are trainable parameters. Then, the entity pair representation is computed as:

$$\boldsymbol{x}^{(s,o)} = \boldsymbol{z}_s \otimes \boldsymbol{z}_o, \tag{8}$$

where $\boldsymbol{x}^{(s,o)} \in \mathbb{R}^{d^2}$ and $\otimes$ stands for the vectorised Kronecker product. Finally, relation scores are computed as:

$$\boldsymbol{y}^{(s,o)} = \boldsymbol{W}_r \boldsymbol{x}^{(s,o)} + \boldsymbol{b}_r, \tag{9}$$

where $\boldsymbol{y}^{(s,o)} \in \mathbb{R}^{|\mathcal{R}|}$, with $\boldsymbol{W}_r \in \mathbb{R}^{|\mathcal{R}| \times d^2}$ and $\boldsymbol{b}_r \in \mathbb{R}^{|\mathcal{R}|}$ representing learnable parameters. The probability of relation $r \in \mathcal{R}$ between the subject and object entities is computed as follows:

$$P(r|s,o) = \sigma(\boldsymbol{y}^{(s,o)}), \tag{10}$$

where σ is the sigmoid function. To reduce the number of parameters in the classifier, a grouped function is used, which splits the embedding dimensions into k equal-sized groups and applies the function within the groups as follows:

$$\boldsymbol{z}_s = [\boldsymbol{z}_s^1; \ldots; \boldsymbol{z}_s^k] \tag{11}$$

$$\boldsymbol{z}_o = [\boldsymbol{z}_o^1; \ldots; \boldsymbol{z}_o^k] \tag{12}$$

$$\boldsymbol{x}^{(s,o)} = [\boldsymbol{x}^{(s,o)^1}; \ldots; \boldsymbol{x}^{(s,o)^k}] \tag{13}$$

$$\boldsymbol{y}^{(s,o)} = \sum_{i=1}^{k} \boldsymbol{W}_r^i \boldsymbol{x}^{(s,o)^i} + \boldsymbol{b}_r, \tag{14}$$

where $\boldsymbol{z}_s^i, \boldsymbol{z}_o^i \in \mathbb{R}^{d/k}$, $\boldsymbol{x}^{(s,o)^i} \in \mathbb{R}^{d^2/k}$, and $\boldsymbol{W}_r^i \in \mathbb{R}^{|\mathcal{R}| \times d^2/k}$. This way, the number of parameters can be reduced from d^2 to d^2/k.

Loss Function ATLOP introduces the adaptive thresholding loss concept. This approach involves training a model to learn a hypothetical threshold class TH, which dynamically adjusts for each relation class $r \in \mathcal{R}$. During training, for each entity pair (e_s, e_o), the loss enforces the model to generate scores above TH for positive relation classes $\mathcal{R}_P$ and scores below TH for negative relation classes $\mathcal{R}_N$. The loss is computed as follows:

$$\mathcal{L} = -\sum_{s \neq o} \sum_{r \in \mathcal{R}_P} \frac{exp(y_r^{(s,o)})}{\sum_{r' \in \mathcal{R}_P \cup \{TH\}} exp(y_{r'}^{(s,o)})} - \frac{exp(y_{TH}^{(s,o)})}{\sum_{r' \in \mathcal{R}_N \cup \{TH\}} exp(y_{r'}^{(s,o)})} \tag{15}$$

A.2 GADePo's Extra Parameters

GADePo introduces few extra parameters to the PLM. The amount of parameters is reported in Table 4.

Parameter	Model	
	RoBERTa$_{\text{LARGE}}$	BERT$_{\text{BASE}}$
<ent>	1024	768
<pent>	1024	768
<ent> $\longrightarrow *$	24 $\times$ 1024	12 $\times$ 768
$* \longrightarrow$ <ent>	24 $\times$ 1024	12 $\times$ 768
<pent> $\longrightarrow *$	24 $\times$ 1024	12 $\times$ 768
$* \longrightarrow$ <pent>	24 $\times$ 1024	12 $\times$ 768
Total	100,352	38,400

Table 4: GADePo's extra parameters count.

A.3 Training Details

We generally comply with the hyperparameters of ATLOP and set the output dimension in Equation 6 and Equation 7 to 768. We also set the block size in Equation 11 and Equation 12 to 64, i.e., $k = 12$.

In all our experiments we perform early stopping on the development set based on the Ign $F_1 + F_1$ score for DocRED and Re-DocRED, and F_1 score for HacRED. The five different seeds we use are $\{73, 21, 37, 7, 3\}$.

We use RAdam (Liu et al., 2020) as our optimizer. On the RoBERTa$_{\text{LARGE}}$ based models we train for 8 epochs and set the learning rates to $3e^{-5}$ and $1e^{-4}$ for the PLM parameters and the new additional parameters, respectively. On the BERT$_{\text{BASE}}$ based models we train for 10 epochs and set the learning rates to $1e^{-5}$ and $1e^{-4}$ for the PLM parameters and the new additional parameters, respectively. We use a cosine learning rate decay throughout the training process.

In all our experiments the batch size is set to 4 for ATLOP and 2 for GADePo, with gradient accumulation set to 1 and 2, for ATLOP and GADePo, respectively. We clip the gradients to a max norm of 1.0. All models are trained with mixed precision.

We run our experiments on two types of GPUs, namely the NVIDIA V100 32GB for the RoBERTa$_{\text{LARGE}}$ based models and NVIDIA RTX 3090 24GB for the BERT$_{\text{BASE}}$ based models, respectively.

We use PyTorch (Paszke et al., 2019), Lightning (Falcon and The PyTorch Lightning team, 2019), and Hugging Face's Transformers (Wolf et al., 2020) libraries to develop our models.

		Dev		Test	
Model	**Aggregation**	Ign F_1	F_1	Ign F_1	F_1
ATLOP*	$\boldsymbol{h}_e$	75.46 ± 0.16	76.16 ± 0.16	75.27	75.92
GADePo (ours)	<ent>	75.46 ± 0.20	76.31 ± 0.24	**75.55**	**76.38**
ATLOP$^\bullet$	$\boldsymbol{h}_e$; $\boldsymbol{c}^{(s,o)}$	76.79	77.46	76.82	77.56
ATLOP*	$\boldsymbol{h}_e$; $\boldsymbol{c}^{(s,o)}$	77.75 ± 0.08	78.41 ± 0.10	77.62	78.38
GADePo (ours)	<ent> ; <pent>	77.48 ± 0.12	78.19 ± 0.14	**77.70**	**78.40**

Table 5: Results in percentage for the development and test sets of Re-DocRED. We report the results obtained by Tan et al. (2022b) (ATLOP$^\bullet$) on Re-DocRED. ATLOP* indicates our reimplementation of the previous method. We report the mean and standard deviation of Ign F_1 and F_1 on the development set, calculated from five training runs with distinct random seeds. We report the test score achieved by the best checkpoint on the development set. Ign F_1 refers to the F_1 score that excludes relational facts shared between the training and development/test sets.

		Dev			Test		
Model	**Aggregation**	P	R	F_1	P	R	F_1
ATLOP*	$\boldsymbol{h}_e$	77.37 ± 0.22	77.40 ± 0.31	77.39 ± 0.13	**76.27**	76.83	76.55
GADePo (ours)	<ent>	72.96 ± 0.96	79.22 ± 1.20	75.96 ± 0.99	74.13	**79.46**	**76.70**
ATLOP$^\diamond$	$\boldsymbol{h}_e$; $\boldsymbol{c}^{(s,o)}$	–	–	–	77.89	76.55	77.21
ATLOP*	$\boldsymbol{h}_e$; $\boldsymbol{c}^{(s,o)}$	77.18 ± 0.14	77.98 ± 0.66	77.58 ± 0.36	76.36	78.86	77.59
GADePo (ours)	<ent> ; <pent>	75.98 ± 0.94	80.54 ± 0.72	78.19 ± 0.19	**78.27**	**79.03**	**78.65**

Table 6: Results in percentage for the development and test sets of HacRED. We report the results obtained by Cheng et al. (2021) (ATLOP$^\diamond$) on HacRED. ATLOP* indicates our reimplementation of the previous method. We report the mean and standard deviation of Precision (P), Recall (R) and F_1 on the development set, calculated from five training runs with distinct random seeds. We report the test score achieved by the best checkpoint on the development set.

		Dev		Test	
Model	**Aggregation**	Ign F_1	F_1	Ign F_1	F_1
ATLOP*	$\boldsymbol{h}_e$	59.66 ± 0.20	61.60 ± 0.21	59.22	61.37
GADePo (ours)	<ent>	59.04 ± 0.52	61.18 ± 0.46	**59.30**	**61.63**
ATLOP$^\circ$	$\boldsymbol{h}_e$; $\boldsymbol{c}^{(s,o)}$	61.32 ± 0.14	63.18 ± 0.19	61.39	63.40
ATLOP*	$\boldsymbol{h}_e$; $\boldsymbol{c}^{(s,o)}$	61.41 ± 0.26	63.38 ± 0.28	**61.62**	63.72
GADePo (ours)	<ent> ; <pent>	61.19 ± 0.55	63.26 ± 0.48	61.52	**63.75**

Table 7: Results in percentage for the development and test sets of DocRED. We report the results obtained by Zhou et al. (2021) (ATLOP$^\circ$) on DocRED. ATLOP* indicates our reimplementation of the previous method. We report the mean and standard deviation of Ign F_1 and F_1 on the development set, calculated from five training runs with distinct random seeds. We report the test score achieved by the best checkpoint on the development set. Ign F_1 refers to the F_1 score that excludes relational facts shared between the training and development/test sets.

A.4 Additional Results

Re-DocRED and HacRED Table 5 and Table 6 present additional results for Re-DocRED and HacRED, respectively. In addition to the results outlined in Section 5, these tables include the mean and standard deviation on the development set, calculated from five training runs with distinct random seeds, as reported in Appendix Subsection A.3.

DocRED results The DocRED (Yao et al., 2019) dataset consists of $56,354$ facts, 96 relations, $5,053$ documents, and 26.2 average number of entities per document. In line with the approach taken for Re-DocRED and HacRED, Table 7 and Figure 5 illustrate the results for DocRED.

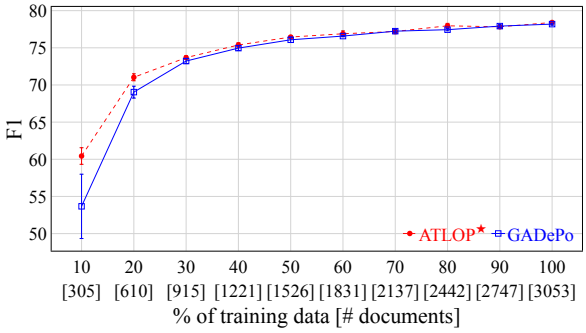


Figure 5: Performance of ATLOP* ($\boldsymbol{h}_e$; $\boldsymbol{c}^{(s,o)}$) and GADePo (<ent> ; <pent>) on the development set under varying data availability conditions on DocRED. The x-axis represents the percentage and number of documents from the training dataset, while the y-axis displays the F_1 score in percentage. Each point on the graph represents the mean value, while error bars indicate the standard deviation derived from five distinct training runs with separate random seeds.