Document-level Event Argument Extraction with Entity type-aware Graph Link Prediction

Tianqi Liu, Jie Yang and Hui Song (✉)

School of Computer Science and Technology, Donghua University, Shanghai, China
{2222672, 2212525}@mail.dhu.edu.cn, songhui@dhu.edu.cn

Abstract. Document-level event argument extraction faces challenges such as context modeling, cross-sentence correlations, and long-distance dependencies. Previous researches have introduced abstract meaning representation to capture the semantic structure of documents. However, there are still issues with incomplete argument spans and misclassified argument roles. To improve the performance of the model in argument identification and classification, we propose a novel model EBGE, which involves an entity type-aware bidirectional heterogeneous graph in. It updates node representations by means of relational graph attention network, and then predicts arguments through node representations and span entity type embeddings. Experimental results on public datasets, WikiEvents and RAMS, demonstrate that our model achieves improvements in F1 scores on both subtasks compared to previous state-of-the-art works.

Keywords: Event Argument Extraction, Abstract Meaning Representation, Entity Type.

1 Introduction

Event argument extraction (EAE) has always been a challenging problem in natural language processing and is crucial for various downstream applications [1, 2]. It aims to identify arguments (event participants) and predicts roles they played in specific event. In comparison to sentence-level EAE, document-level EAE is more practical as it extracts arguments across the entire document.

Currently, document-level EAE can be classified into discriminative methods and generative methods. Discriminative methods use attention mechanisms to encode contextual features and extract arguments [3]. On the other hand, generative methods use language models and event templates to generate arguments [4, 5]. However, these methods do not take full advantage of the complex event structural information, which leads to missing long-tail arguments in the documents. To capture long-distance dependencies between arguments and triggers, recent research attempts to introduce Abstract Meaning Representation (AMR) [6] into document-level EAE.

Xu et al. proposed TSAR [7], which added AMR as input features to enrich argument span representations. However, the implicit AMR prevents the model from directly utilizing discrete structure information. Instead, Yang et al. proposed TARA [8], which formulates EAE as a link prediction task. Their work prunes the AMR and uses Relational Graph Convolutional Networks (RGCN) [9] to identify arguments. Although AMR structures force the model focusing on the predicate-argument structure, it performs poorly in argument span identification and role classification. The main reasons are that span proposal only take context information, which leads to incomplete span, as well as undirected graph ignores the active-passive relationships information, which leads to opposite role.

To improve the performance of the model in argument identification and classification, this paper proposes a novel model EBGE (**E**ntity type-aware **B**idirectional Heterogeneous **G**raph-based **E**vent Argument Extraction). We construct a bidirectional heterogeneous graph based on AMR to establish relationships between arguments and triggers. For each graph node, the argument span representation is concatenated with entity type information, enabling model to accurately identify the start and end positions of argument spans. To facilitate the prediction of the types of relationships between nodes, we use attention mechanism to merge neighbor node information with different weights. Experimental results on public document-level EAE datasets, WikiEvents and RAMS, demonstrate that our method achieves state-of-the-art performance.

Our contributions are summarized as follows:

- We propose a novel document-level EAE model, which constructs an Entity type-aware Bidirectional Heterogeneous Graph (EBHG) through Graph Construction and Graph Augment modules, and extracts argument spans with their corresponding roles through Link Prediction module.
- We construct the EBHG, by which converting edges in vanilla AMR into bidirectional edges with active-passive types, and encode the candidate argument nodes by incorporating contextual information with entity type information.
- We propose a novel link prediction method, which performs message passing with Relational Graph Attention Networks (RGAT). During role classification, it takes consideration of the interaction between trigger and arguments as well as the entity type of the arguments.

2 Methodology

2.1 Task Formulation

Given an input document $D = \{w_1, w_2, \dots, w_N\}$, where the event trigger $t \in D$, event type e, and the set of role types corresponding to the event R_e, the goal of document-level EAE is to extract argument spans $s = (s_b, s_e) \in D$ and identify their roles $r \in R_e$. Here, s_b and s_e represent the start and end indices of the argument span.

Following Yang et al. [8], we formulate document-level EAE as a link prediction problem. By leveraging AMR, we represent D as a graph G. The nodes $n_u \in G$ align with text spans $s_u \in D$, and the edges between n_i and n_j correspond to relations between s_i and s_j. If model predicts an edge connecting n_u and trigger node n_t with type r, then s_u is considered an argument which playing the role r in the event e.

2.2 Model Architecture

We propose a model based on Entity type-aware Bidirectional Heterogeneous Graph for Event Argument Extraction, as illustrated in Fig. 1. Firstly, in Graph Construction module, we use AMR parser to convert input document into vanilla AMR and transform its edges into bidirectional edges with predefined types, resulting in Bidirectional Heterogeneous Graph (BHG). Next, in Graph Augment module, we use encoder to take the context representations of input document, combine them with entity type embedding to extract argument spans and enhance the BHG. Finally, in Link Prediction module, we use RGAT to update the node representations in the graph and then use classifier to predict roles for argument spans.

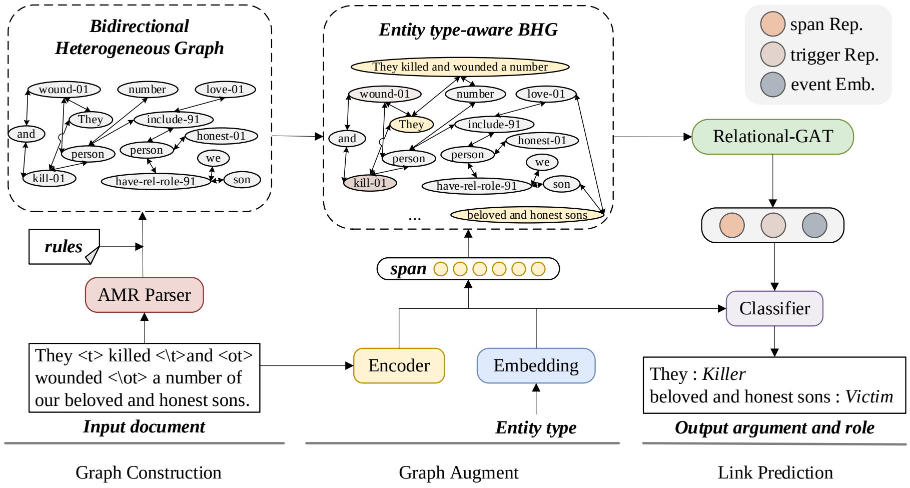


Fig. 1. The main architecture of EBGE.

Table 1. Edge types of Bidirectional Heterogeneous Graph.

Categories	AMR relation types	Edge label
Sub sentence	snt, NSENT	0
Spatial	location, location-of, destination, path	1
Temporal	year, time, duration, decade, weekday	2
Means	instrument, manner, topic, medium	3
Modifiers	mod, poss	4
Prepositions	prep-X	5
Operators	op-X	6
Core Roles	ARG0, ARG1, ARG2, ARG3, ARG4	7-11
Core Roles	ARG0-of, ARG1-of, ARG2-of, ARG3-of, ARG4-of	12-16
Others	Other AMR relation types	17

2.3 Graph Construction

We use pretrained model AMRBART as AMR parser to convert sentences into corresponding AMR. By connecting the root nodes of sentence-level AMR, we take the AMR for the whole document. To avoid the model being misled by irrelevant information, we compress the subgraph following [8]. We then cluster the fine-grained edge types into main categories shown on Table 1 to make them appropriate for the EAE task. Different from the previous EAE model that constructs undirected graph based on AMR, we consider the impact of the direction information and construct Bidirectional Heterogeneous Graph. For example, the edge types of ARG0 and ARG0-of are both prototype agents, but represent active and passive relationships respectively.

2.4 Graph Augment

Since AMR parser tends to build word-level nodes, we extract span-level nodes to enhance graph. Given an input document D, we use a pretrained encoder to capture contextual information in the text and obtain the context embedding h_k for each token w_k. In order to better predict argument span boundaries, we incorporate entity type embedding[1] $h_{i,j}^{entity}$, start and end representation h_i, h_j of the span $s_{i,j}$ into span representation $h_{i,j}$, as follows:

$$h_k = Encoder(w_k), k = 1,2, \dots, N \tag{1}$$

$$h_{i,j}^{entity} = Embedding\left(Type\left(s_{i,j}\right)\right) \tag{2}$$

$$h_{i,j} = W_0\left[\frac{1}{j-i+1}\sum_{k=i}^{j} h_k\, ; W_1 h_i; W_2 h_j; h_{i,j}^{entity}\right] \tag{3}$$

where N is the length of the document, $Type\left(s_{i,j}\right)$ represents the entity type corresponding to the argument span $s_{i,j}$, $Embedding$ represents an Embedding layer and W_0, W_1, W_2, W_3 are trainable parameters.

We set up a sliding window and extract the most possible spans as candidate arguments. The possible score is calculated based on span representation $h_{i,j}$. We also consider the span length information to calculate the argument identification score $S_{i,j}$, and use the binary cross-entropy loss function to calculate the loss of argument identification L_{span}, as follows:

$$S_{i,j} = FFNN(h_{i,j} + FFNN(Embedding(s_{width})) \tag{4}$$

$$L_{span} = -\left(ylog\left(S_{i,j}\right) + (1-y)log\left(1-S_{i,j}\right)\right) \tag{5}$$

[1] For dataset WikiEvents, we directly use the entity type provided by the dataset. For dataset RAMS, we use the tool spacy to extract entities existing in the document and record their types. Due to incompleteness of prediction, we manually add entities that serve as arguments.

where s_{width} represents the length of the span $s_{i,j}$ and $FFNN$ denotes a fully connected layer with $ReLU$ activation function.

We add the top-k extracted argument spans as new nodes in BHG and connect them to the original nodes using special type edges called $Context$. The nodes are categorized into four types in the final updated EBHG, as shown in Fig. 2. Therefore, the initial representation h_u of node n_u is the concatenation of the argument span representation $h_{i,j}$ and node type embedding, as follows:

$$h_u = h_{i,j} + Embedding\big(Type(n_u)\big) \tag{6}$$

where $Type(n_u)$ represents the type of node n_u.

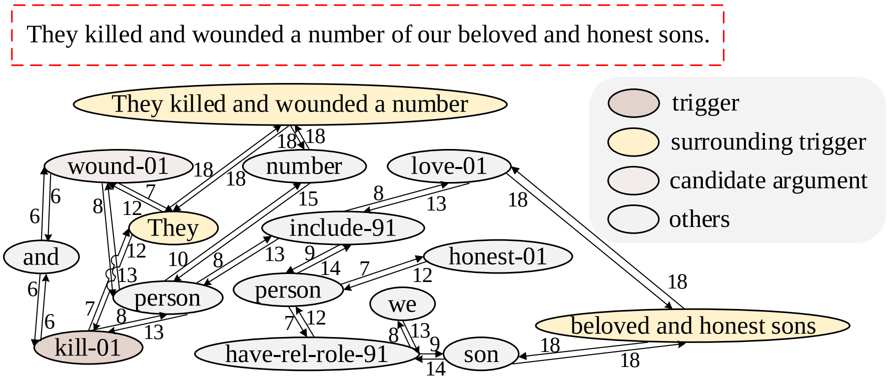


Fig. 2. Example of the EBHG with annotated node and edge types. Extracted argument spans are like "They killed and wounded a number" and "beloved and honest sons". The edge from "kill" to "they" is labelled as 7, indicating "ARG0". On the other hand, the edge from "they" to "kill" is labeled as 12, indicating "ARG0-of". They both mean that "they" is the agent of the "kill" action.

2.5 Link Prediction

To further efficiently extract dependency information for each argument, we propose a novel link prediction method which is based on the edge relationships R and node representations h_u in EBHG. We use RGAT [10] to build dependency relationships between nodes of the same edge type, regulating the flow and interaction of information, and updating node representations. Specifically, we use attention score to weight the neighbor node representations h_v of the same edge type and aggregate all the type-specific representations, as follows:

$$h'_u = ReLU\big(W_0 h_u + \textstyle\sum_{r\in R}\sum_{n_v\in N_u^r} Attention(h_v)\big) \tag{7}$$

where R is the set of edge types defined in Table 1, N_u^r represents the set of neighboring nodes connected to node n_u through edge relationships $r \in R$, $Attention$ represents a multi-Attention layer and W_0 is a trainable parameter.

Next, we use $FFNN$ as Classifier to compute the argument classification score S_u for node n_u. To accurately predict the role category corresponding to the argument, we calculate the score in two parts. In the first part, we concatenate the span node representation h'_u, the trigger node representation h_t, and the event type embedding h_{event}, and in the second part we take into account the entity type embedding for the argument span s_u corresponding to node n_u. The argument classification loss L_{role} is then calculated using the cross-entropy loss function, as follows:

$$h_{event} = Embedding\big(Type(e)\big) \tag{8}$$

$$S_u = FFNN(h'_u + h_t + h_{event}) + FFNN\left(Embedding\big(Type(n_u)\big)\right) \tag{9}$$

$$L_{role} = -\sum_{s_u} y_u \, logP(S_u = r_u) \tag{10}$$

where r_u represents the golden argument role for the span s_u, $y_u$represents the predicted argument role by the model. $Type(e)$ represents the type of the event e.

To jointly optimize the model, we calculate the total loss as the weighted sum of the argument span identification loss and the argument role classification loss, as follows:

$$L = L_{role} + \lambda L_{span} \tag{11}$$

3 Experiments

3.1 Datasets and Metrics

We evaluate the proposed model on two public document-level EAE datasets: WikiEvents [4] and RAMS [11]. The WikiEvents dataset contains 50 event types, 59 argument roles, 3,951 events, and 5,536 arguments. The RAMS dataset contains 139 event types, 65 argument roles, 9,124 events, and 21,237 arguments. We followed the official train/dev/test splits provided by the WikiEvents and RAMS datasets. The detailed data statistics are shown in Table 2.

Table 2. Detailed data statistics of WikiEvents and RAMS.

Dataset	Split	Doc	Sentence	Event	Argument	Event Types	Role Types
WikiEvents	train	206	5262	3241	4542	49	57
	dev	20	378	345	428	35	32
	test	20	492	365	566	34	44
RAMS	train	3194	7329	7329	17026	139	65
	dev	399	924	924	2188	131	62
	test	400	871	871	2023	-	-

We evaluate the two subtasks of EAE separately. Argument Identification: An argument span is considered to be correctly identified if the predicted argument span boundary matches the golden boundary. Argument Classification: An argument span is considered to be correctly classified if the role corresponding to the identified argument also matches the golden role. For WikiEvents, we follow Li et al. [4] to report metrics Head F1 and Coref F1 on both subtasks. For RAMS, we follow Yang et al. [8] to report metrics Span F1 and Head F1 on the classification task, as well as metric Span F1 on the identification task. Span F1 requires a complete match of the predicted argument span with the golden span, while Head F1 only focuses on whether the head word matches. Coref F1 considers two argument spans to match if they are coreferential.

3.2 Baselines and Experiment Setups

We compare the proposed model with several baseline models: (1) Generative models: BART-Gen [4], PAIE [12], EA2E [13]. (2) AMR-based models: TSAR [7], TARA [8].

We align the text with AMR using the state-of-the-art AMR parser AMRBART [14], use the pre-trained RoBERTa$_{\text{large}}$ [15] as the encoder architecture. During training, we always use a full sequence length of 512. Specifically, we set λ as 1.0, k as 50, L as 3 and train on a single NVIDIA-3090 GPU with an AdamW weight decay optimizer, learning rate warmup, linear decay of the learning rate and a batch size of 8.

3.3 Main Results

Table 3 presents the results of models on the WikiEvents test set. To ensure a fair comparison, we report results on large-scale pre-trained models, where generative models use BART$_{\text{large}}$ and AMR-based models use RoBERTa$_{\text{large}}$. As shown in the Table 3, compared to generative models, AMR-based models achieve higher Head F1 scores. This means that it is beneficial to capture rich semantic structure information from the text. Furthermore, EBGE outperforms previous methods in all metrics on both subtasks. In comparison to TARA, it improves the Head/Coref F1 on argument identification by 3.6/5.8 and the Head/Coref F1 on argument classification by 2.2/4.2. These results indicate that constructing entity type-aware bidirectional heterogeneous graphs and using entity type-aware link prediction method can improve the performance of EAE.

Table 3. Main results on the WikiEvents test set. "*" presents the results reproduced by us. "-" presents the results were not reported.

Model	Arg Identification		Arg Classification	
	Head F1	Coref F1	Head F1	Coref F1
BART-Gen	71.75	72.29	64.57	65.11
PAIE	-	-	68.40	-
EA2E	74.62	75.77	68.61	69.70
TSAR	76.62	75.52	69.70	68.79
TARA*	76.4151	75.0943	71.3208	70.1887
EBGE	**80.0000**	**80.9302**	**73.4884**	**74.4186**

Table 4 presents the results on the RAMS test set, which show similar conclusions to WikiEvents. Compared to TARA, EBGE achieves improvements of 4.4 and 3.6 in

Span F1 and Head F1 on identification and classification task as well as 5.5 in Span F1 on identification task. This further illustrates the improvement of the proposed EBGE model on EAE tasks.

Table 4. Main results on the RAMS test set.

Model	Arg Identification	Arg Classification	
	Span F1	Span F1	Head F1
BART-Gen	51.2*	48.64	57.32
PAIE	56.8	52.2	-
TSAR	-	51.18	58.53
TARA*	56.2098	51.5586	58.4364
EBGE	**62.3314**	**55.8014**	**62.8171**

3.4 Ablation Study

We conduct an ablation study to investigate the effects of different modules in the model. Table 5 presents the results on the WikiEvents test set after removing each module from the EBGE model.

Table 5. Ablation study results.

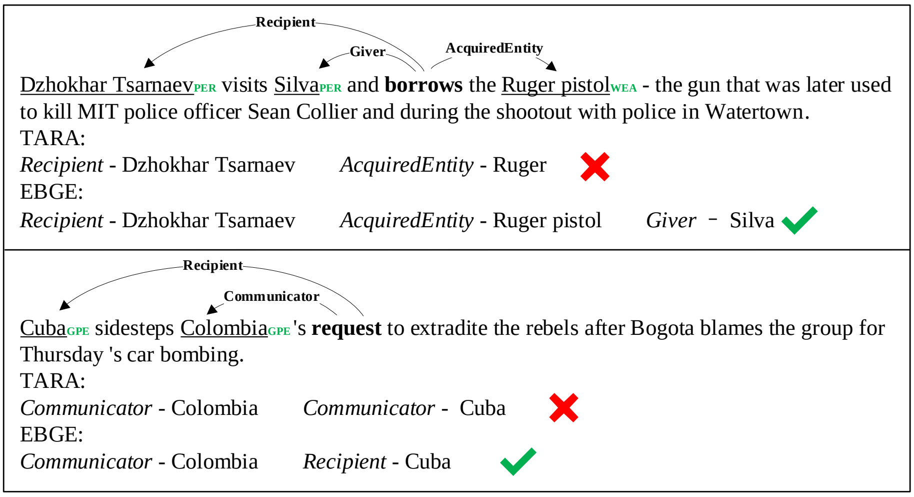
Model	Arg Identification		Arg Classification	
	Head F1	Coref F1	Head F1	Coref F1
TARA	76.4151	75.0943	71.3208	70.1887
EBGE	**80.0000**	**80.9302**	**73.4884**	**74.4186**
w/ single-direction	78.1481	79.2593	71.6667	72.9630
w/ undirected	79.2627	80.0000	73.2998	74.1440
w/o RGAT	78.0662	79.3196	71.6204	73.0528
w/o ET	76.6160	75.0951	71.8631	70.5323

To study the impact of graph structures on EAE, we experimented with single-direction heterogeneous graphs and undirected heterogeneous graphs. Single-direction heterogeneous graphs refer to keeping the direction of edges in the vanilla AMR, where there is only one directed edge between two nodes. Undirected heterogeneous graphs, similar to the TARA, extend the edges in the vanilla AMR to be bidirectional, but with the same edge types. We found that when we transform bidirectional heterogeneous graph into single-direction, all metrics show a decrease, especially in argument classification, where the Head/Coref F1 decreased by approximately 1.8/1.5 pt. This indicates that bidirectional edges contain more information compared to single-direction, which is beneficial for role classification. When we transform the bidirectional heterogeneous graph into an undirected graph, the Coref F1 for identification and classification tasks decreased by approximately 0.9 and 0.4pt, respectively. This demonstrates that the edge types have a significant impact on classification, thereby validating the rationality of our defined edge types.

To study the impact of message passing method on EAE, we replace the RGAT with RGCN. We find that the performance on argument classification decreased by approximately 1.0 and 0.7pt, respectively. This indicates that the introduced attention mechanism helps the model better utilize information from neighboring nodes, leading to more accurate role assignments for arguments.

To study the impact of entity type information on EAE, we remove entity type embeddings from both Graph Augment module and Link Prediction module. We find that all metrics show a decrease, with a reduction of approximately 5.8 and 3.9pt in identification and classification of Coref F1. This indicates that providing entity type annotations at the span level helps the model discover potential arguments and that the information of arguments themselves, as well as the relationships between arguments and triggers, jointly affect the performance of argument classification.

3.5 Case Study



Dzhokhar Tsarnaev$_{PER}$ visits Silva$_{PER}$ and **borrows** the Ruger pistol$_{WEA}$ - the gun that was later used to kill MIT police officer Sean Collier and during the shootout with police in Watertown.
TARA:
Recipient - Dzhokhar Tsarnaev *AcquiredEntity* - Ruger ✗
EBGE:
Recipient - Dzhokhar Tsarnaev *AcquiredEntity* - Ruger pistol *Giver* - Silva ✓

Cuba$_{GPE}$ sidesteps Colombia$_{GPE}$'s **request** to extradite the rebels after Bogota blames the group for Thursday 's car bombing.
TARA:
Communicator - Colombia *Communicator* - Cuba ✗
EBGE:
Communicator - Colombia *Recipient* - Cuba ✓

Fig. 3. Triggers are shown in bold, golden arguments are shown underlined with corresponding event types in green color. Here, the event type *WEA* represents weapons, *PER* represents individuals, *GPE* represents geographical entity.

We select examples from the WikiEvents dataset to illustrate, as shown in Fig. 3. TARA fails to predict the correct arguments for the event, while EBGE not only accurately predicts the boundaries of argument spans but also matches the correct argument roles. The above example shows that the entity type information helps the model to identify arguments as arguments are specific entities in events. The below example shows that the bidirectional heterogeneous graph helps the model to classify arguments as it can distinguish the subject and object of an action.

3.6 Error Analysis

To analyze the predictions of the model, we conduct a specific analysis of the error samples, focusing on span-level errors. We categorize the errors into four types: Missing Span represents the number of argument spans that the model fails to predict. Overpred Span represents the number of argument spans that the model redundantly predicts. Wrong Role represents the number of argument spans that the model correctly identifies but matches wrong argument role. Wrong Span represents the number of argument spans that the model predicts wrong start or end positions.

As shown in Table 6, compared to baseline TARA, EBGE can more accurately identify argument spans, effectively addressing the issue of unclear span boundaries, while also mitigating errors of Missing Span and Overpred Span. Additionally, we observe a slight increase in Wrong Role while correctly predicting argument spans. We found that the dataset has a class imbalance issue, with error samples related to Wrong Role accounting for less than 1%. We also analyze the specific effects of several proposed methods in this paper. Adding entity types to the model tends to predict a greater number of arguments, effectively mitigating Missing Span and Wrong Span errors. It indicates that entity type information helps the model to focus entities that can be candidate arguments. The addition of RGAT enables the model to more accurately identify argument roles, effectively reducing Overpred Span errors. It indicates that message passing method helps the model merge information of different types. The addition of BHG effectively mitigates Missing Span and Wrong Span errors. It indicates that bidirectional heterogeneous graph helps the model to take semantic structure information.

Table 6. Error Analysis on WikiEvents test set.

Model	Missing Span	Overpred Span	Wrong Role	Wrong Span
baseline	158	90	26	17
EBGE	133	82	35	0
baseline w/ ET	115	120	38	0
baseline w/ RGAT	161	85	24	16
baseline w/ BHG	127	118	36	4

4 Related Work

4.1 Sentence-level Event Argument Extraction

Early approaches to sentence-level event argument extraction primarily focused on discriminative methods which achieve good results (Shi and Lin [16]; Zhang et al. [3]). In recent years, with the development of generative models such as GPT, BART, and T5, more researchers have attempted to solve the EAE task using generative methods. These methods use predefined templates to construct informative prompts that guide the model to generate arguments. Dai et al. [17] use prompt learning to generate arguments by incorporating argument roles of context entities as prompts, in order to explore the impact of event argument interactions. Hsu et al. [18] propose the model AMPERE, which generates AMR-aware prefix prompts for each layer. To overcome potential noise introduced by the AMR graph, attention mechanisms are used to learn appropriate copying mechanisms. Other works have employed retrieval techniques to mine effective information as additional inputs. For example, Du et al. [19] retrieve the most similar QA pairs and extend them as prompts to the context of the current example.

4.2 Document-level Event Argument Extraction

Document-level event argument extraction poses significant challenges due to lengthy sequences and long-tailed distributions of arguments. Some models attempt to mine effective information from the document. SCPRG [20] employs an attention module to capture the semantic correlations between relevant context information and roles. It integrates non-argument cues and potential role information into candidate argument representations to improve argument prediction. Ren et al. [21] study various retrieval settings from the input and label distribution perspectives. They use pseudo-demonstrations sampled from event semantic regions for enhancing document-level EAE models. Zhou et al. [22] try to leverage redundant entity information to construct entity co-reference graphs and entity summary graphs to merge multiple extraction results. There are also some works that explore the correlation between triggers and arguments, as well as the correlation between arguments themselves. EA2E [13] employs alignment-enhanced training and iterative inference to self-augment the context by labeling the argument tags of adjacent events. This approach captures event-event relationships, thereby enhancing argument consistency. Du et al. [5] concatenates the retrieved most similar generated sequences with the corresponding templates of the target event as input, and then utilizes relationships between arguments obtained from world knowledge to guide the decoding process. Veyseh et al. [23] propose a sentence-level grammatical structure that simultaneously considers semantic and syntactic similarities, and then employ Optimal Transport (OT) to induce document structure.

5 Conclusion

We propose an entity type-aware bidirectional heterogeneous graph and introduce entity type information to aid in identifying candidate arguments. The bidirectional heterogeneous graph allows for more accurate representation of the relationships between arguments and triggers in the document. We also propose a novel link prediction method that utilizes relation graph attention network to update node information during message passing. During classification, we consider not only the representation of argument nodes and trigger node in the graph but also the entity type embedding of the corresponding argument text spans. Experimental results demonstrate that our proposed model effectively improves the accuracy of argument identification and argument classification, achieving state-of-the-art performance.

References

1. Li, M., Zareian, A., Lin, Y., Pan, X., Whitehead, S., Chen, B., Wu, B., Ji, H., Chang, S.-F., Voss, C., Napierski, D., Freedman, M.: GAIA: A Fine-grained Multimedia Knowledge Extraction System. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations, pp. 77–86. Association for Computational Linguistics, Online (2020)
2. Zhang, T., Chen, M., Bui, A.A.T.: Diagnostic Prediction with Sequence-of-sets Representation Learning for Clinical Events. In: Michalowski, M., Moskovitch, R. (Eds.), Artificial Intelligence in Medicine, Lecture Notes in Computer Science, pp. 348–358. Springer International Publishing, Cham (2020)
3. Zhang, Z., Kong, X., Liu, Z., Ma, X., Hovy, E.: A Two-Step Approach for Implicit Event Argument Detection. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pp. 7479–7485. Association for Computational Linguistics, Online (2020)
4. Li, S., Ji, H., Han, J.: Document-Level Event Argument Extraction by Conditional Generation. In: Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 894–908. Association for Computational Linguistics, Online (2021)
5. Du, X., Li, S., Ji, H.: Dynamic Global Memory for Document-level Argument Extraction. In: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 5264–5275. Association for Computational Linguistics, Dublin, Ireland (2022)
6. Banarescu, L., Bonial, C., Cai, S., Georgescu, M., Griffitt, K., Hermjakob, U., Knight, K., Koehn, P., Palmer, M., Schneider, N.: Abstract Meaning Representation for Sembanking . In: Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse, pp. 178–186. Association for Computational Linguistics, Sofia, Bulgaria (2013)
7. Xu, R., Wang, P., Liu, T., Zeng, S., Chang, B., Sui, Z.: A Two-Stream AMR-enhanced Model for Document-level Event Argument Extraction. In: Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 5025–5036. Association for Computational Linguistics, Seattle, United States (2022)
8. Yang, Y., Guo, Q., Hu, X., Zhang, Y., Qiu, X., Zhang, Z.: An AMR-based Link Prediction Approach for Document-level Event Argument Extraction. In: Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 12876 ‐ 12889. Association for Computational Linguistics, Toronto, Canada (2023)
9. Schlichtkrull, M., Kipf, T.N., Bloem, P., Van Den Berg, R., Titov, I., Welling, M.: Modeling Relational Data with Graph Convolutional Networks. arXiv preprint arXiv: 1703.06103 (2018)
10. Busbridge, D., Sherburn, D., Cavallo, P., Hammerla, N.Y.: Relational Graph Attention Networks. arXiv preprint arXiv: 1904.05811 (2019)
11. Ebner, S., Xia, P., Culkin, R., Rawlins, K., Van Durme, B.: Multi-Sentence Argument Linking. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pp. 8057 ‐ 8077. Association for Computational Linguistics, Online (2020)
12. Ma, Y., Wang, Z., Cao, Y., Li, M., Chen, M., Wang, K., Shao, J.: Prompt for Extraction? PAIE: Prompting Argument Interaction for Event Argument Extraction. In: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 6759 ‐ 6774. Association for Computational Linguistics, Dublin, Ireland (2022)

13. Zeng, Q., Zhan, Q., Ji, H.: EA2E: Improving Consistency with Event Awareness for Document-Level Argument Extraction. In: Findings of the Association for Computational Linguistics: NAACL 2022, pp. 2649 – 2655. Association for Computational Linguistics, Seattle, United States (2022)
14. Bai, X., Chen, Y., Zhang, Y.: Graph Pre-training for AMR Parsing and Generation. In: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 6001 – 6015. Association for Computational Linguistics, Dublin, Ireland (2022)
15. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., Stoyanov, V.: RoBERTa: A Robustly Optimized BERT Pretraining Approach. arXiv preprint arXiv:1907.11692 (2019)
16. Shi, P., Lin, J.: Simple BERT Models for Relation Extraction and Semantic Role Labeling. arXiv preprint arXiv:1904.05255 (2019)
17. Dai, L., Wang, B., Xiang, W., Mo, Y.: Bi-Directional Iterative Prompt-Tuning for Event Argument Extraction. In: Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, pp. 6251 – 6263. Association for Computational Linguistics, Abu Dhabi, United Arab Emirates (2022)
18. Hsu, I.-H., Xie, Z., Huang, K.-H., Natarajan, P., Peng, N.: AMPERE: AMR-Aware Prefix for Generation-Based Event Argument Extraction Model. In: Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 10976 – 10993. Association for Computational Linguistics, Toronto, Canada (2023)
19. Du, X., Ji, H.: Retrieval-Augmented Generative Question Answering for Event Argument Extraction. In: Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, pp. 4649 – 4666. Association for Computational Linguistics, Abu Dhabi, United Arab Emirates (2022)
20. Liu, W., Cheng, S., Zeng, D., Hong, Q.: Enhancing Document-level Event Argument Extraction with Contextual Clues and Role Relevance. In: Findings of the Association for Computational Linguistics: ACL 2023, pp. 12908 – 12922. Association for Computational Linguistics, Toronto, Canada (2023)
21. Ren, Y., Cao, Y., Guo, P., Fang, F., Ma, W., Lin, Z.: Retrieve-and-Sample: Document-level Event Argument Extraction via Hybrid Retrieval Augmentation. In: Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 293 – 306. Association for Computational Linguistics, Toronto, Canada (2023)
22. Zhou, H., Mao, K.: Document-Level Event Argument Extraction by Leveraging Redundant Information and Closed Boundary Loss. In: Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 3041 – 3052. Association for Computational Linguistics, Seattle, United States (2022)
23. Pouran Ben Veyseh, A., Nguyen, M.V., Dernoncourt, F., Min, B., Nguyen, T.: Document-Level Event Argument Extraction via Optimal Transport. In: Findings of the Association for Computational Linguistics: ACL 2022, pp. 1648 – 1658. Association for Computational Linguistics, Dublin, Ireland (2022)