Hyperspherical Dynamic Multi-Prototype with Arguments Dependencies and Role Consistency for Event Argument Extraction

Anonymous ACL submission

Abstract

Event Argument Extraction (EAE) aims to identify arguments and assign them to predefined roles within a document. Existing methods face challenges in modeling intra-class variance and inter-class ambiguity, hindering accurate role assignment. Inspired by how humans dynamically adjust classification criteria while maintaining category consistency (e.g., distinguishing "victim" and "attacker" roles based on contextual relationships), We propose HDMAR (***H**yperspherical **D**ynamic **M**ulti-Prototype with **A**rguments Dependencies and **R**ole Consistency*), where three innovations tackle these challenges: (1) Hyperspherical dynamic multi-prototype learning is used to capture intra-role diversity and enforce inter-role separation via hyperspherical optimization and optimal transport, (2) cross-event role consistency is used to align role representations across events, and (3) an arguments dependencies-guided encoding module enhances contextual understanding of intra-event and inter-event dependencies. Experiments on RAMS and WikiEvents demonstrate gains in accuracy, with further analysis validating the contributions of each module.

1 Introduction

Event Argument Extraction (EAE) is a pivotal task in information extraction (Xia et al., 2022), and aims to identify event-related arguments and their corresponding roles within natural language texts (Doddington et al., 2004). As a foundational component of event understanding, EAE underpins numerous downstream applications, including question answering (Souza Costa et al., 2020), recommendation systems (Han et al., 2025), and dialogue systems (Zhang et al., 2020a). Despite substantial advancements in EAE research, existing methodologies encounter significant challenges when addressing the complexities inherent in document-level documents, particularly in terms of ineffectiveness in cross-event reasoning and difficulties in modeling role diversity.

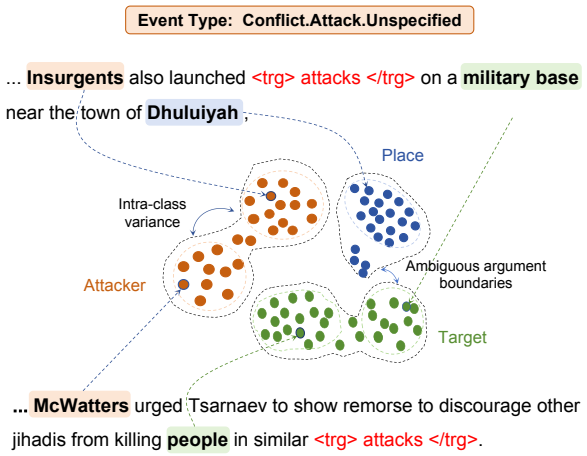

Figure 1: A document from WikiEvents (Li et al., 2021) for document-level EAE. The trigger word is included in special tokens <trg>and </trg>with red color. We demonstrate two kinds of inductive biases found in EAE. (1) Intra-class variance, due to semantic variations, arguments sharing the same role might be assigned to distinct sub-clusters, and (2) Ambiguous role arguments boundaries, the large margin separations are disregarded, resulting in unclear distinctions between arguments of different roles.

Mainstream EAE works (Liu et al., 2023; Ren et al., 2023; Mettes et al., 2019) typically process a single event at a time or assign only one prototype per category, overlooking the semantic variations that exist within the same category. A significant challenge in EAE pertains to role-based inductive biases. Specifically, two key phenomena complicate the extraction process. **Intra-class variance**, where arguments assigned to the same role may cluster into distinct subspaces due to semantic differences. For instance (Figure 1), in a "*Conflict.Attack.Unspecified*" event, both "Insurgents" and "McWatters" can fulfill the "Attacker" role;

however, the former represents an organization, while the latter denotes an individual. Similarly, "military base" and "people" may both serve as "Target", yet they belong to different semantic categories (facility vs. social group). **Ambiguous role arguments boundaries**, where semantically similar arguments (e.g., "military base" vs. "Dhuluiyah") blur the distinctions necessary for accurate classification. These issues complicate the representation of arguments in the embedding space and hinder precise role assignment. Although DEEIA (Liu et al., 2024) employs a multi-event prompt mechanism and HMPEAE (Zhang et al., 2024) utilizes hyperspherical multi-prototype to address these problems, the accuracy remains suboptimal.

To address these limitations, this paper introduces HDMAR (***H**yperspherical **D**ynamic **M**ulti-Prototype with **A**rguments Dependencies and **R**ole Consistency*), a novel model designed to handle the intricacies of multi-event documents. Building upon the TableEAE (He et al., 2023) architecture, HDMAR integrates key advancements to mitigate the aforementioned challenges: **(1) Hyperspherical Dynamic Multi-Prototype Learning**: This component captures intra-role diversity by assigning multiple dynamically learned prototypes to each role, thereby accommodating the varied semantic nuances within the same role. Concurrently, hyperspherical optimization and optimal transport techniques are employed to maintain inter-role distinctions. **(2) Cross-Event Role Consistency**: HDMAR models document-level event correlations by propagating and aligning role representations across events within a document, ensuring coherent extractions for recurring roles.

Furthermore, HDMAR leverages arguments dependencies-guided context encoding, enhancing the TableEAE framework with a specialized attention bias that incorporates both intra- and inter-event dependencies. This mechanism enables the model to better understand the relationships between event triggers, roles, and arguments, facilitating efficient and contextually aware processing of complex multi-event scenarios. The contributions of this paper can be summarized as follows:

- Introduce HDMAR, an EAE method that leverages argument dependencies across different events, thereby improving the performance of the EAE task.
- To address the challenges of EAE, we propose a hyperspherical dynamic multi-prototype learning module and a cross-event role consistency module, which capture intra-class variance and model inter-event correlations, respectively.
- Extensive experiments demonstrate that HDMAR outperforms major benchmarks in performance.

2 Related Work

2.1 Span-Based and Generation-Based Methods

Span-based methods represent a traditional line of research in EAE, where candidate spans are identified and then classified into roles (Zhang et al., 2020b; Yang et al., 2023). These methods are widely used due to their intuitive structure and reasonable performance but often struggle with modeling long-distance dependencies and semantic correlations across arguments. To address these limitations, generation-based methods (Li et al., 2021; Du et al., 2021; Wei et al., 2021) leverage generative pre-trained language models (PLMs), such as BART (Lewis et al., 2020) and T5 (Raffel et al., 2020), to sequentially generate arguments for events. While effective in capturing complex dependencies, these generation-based approaches often suffer from high computational costs and limited scalability, especially in multi-event scenarios.

2.2 Prompt-Based Methods

Prompt-based methods have recently gained prominence due to their flexibility and generalizability in handling diverse EAE scenarios. Approaches like (Ma et al., 2022) and (Nguyen et al., 2023) employ slotted prompts for argument extraction, utilizing generative slot-filling techniques to enhance efficiency and performance. TableEAE (He et al., 2023) further explores the multi-event paradigm by training models to process events in a tabular format, enabling direct modeling of event co-occurrence (Zeng et al., 2022). Despite their advancements, these methods require separate processing of prompts for individual events, making them computationally expensive and less efficient for multi-event documents.

2.3 Multi-Event Argument Extraction and Role Diversity

While most traditional EAE approaches process events in isolation (Single-EAE), recent research highlights the importance of modeling correlations

between events in multi-event documents (Liu et al., 2023; Xu et al., 2024). Multi-event argument extraction methods, such as DEEIA (Liu et al., 2024), attempt to concurrently extract arguments for all events in a document, significantly improving efficiency. However, these methods often neglect the challenges posed by intra-role diversity and inter-role ambiguity. For instance, arguments assigned to the same role may exhibit significant semantic differences (e.g., organization vs. individual for the "Attacker" role), while semantically similar arguments may blur the boundaries between roles (e.g., "military base" vs. "Dhuluiyah").

3 Methodology

In this section, we present HDMAR (Hyperspherical Dynamic Multi-Prototype with Arguments Dependencies and Role Consistency for Event Argument Extraction), a novel framework developed to address the challenges of multi-event argument extraction. Building upon the TableEAE framework, HDMAR introduces significant advancements: (1) **Dynamic Multi-Prototype Learning**, which captures intra-role diversity and inter-role relationships, and (2) **Cross-Event Role Consistency**, which ensures coherence across roles within multi-event documents. They helps to effectively model the intricacies of document-level multi-event extraction tasks.

3.1 Overview

Given a document D containing a set of triggers $T = \{t_1, t_2, \ldots, t_m\}$ representing events, and a predefined set of argument roles $R = \{r_1, r_2, \ldots, r_k\}$, the objective of HDMAR is to extract arguments $A = \{a_1, a_2, \ldots, a_n\}$ corresponding to specific roles in R for each trigger t. Unlike previous approaches that treat events independently or rely solely on fixed role representations, HDMAR processes entire documents holistically by modeling table-structured embeddings for triggers and roles, while integrating dynamic prototypes and cross-event constraints.

3.2 Arguments Dependencies-guided Context Encoding

To address the differences and similarities between various arguments in multi-event documents, we employ an **Arguments Dependencies-guided Context Encoding (ADCE)** module, which extends the TableEAE (He et al., 2023) framework through the incorporation of dependency-guided attention mechanisms. This module generates table-structured representations for events and roles, simultaneously capturing both intra-event and inter-event dependencies.

3.2.1 Table-Based Contextual Representation

Following the structured embedding approach of TableEAE, the document D is encoded into a **table-based representation** that aligns triggers, roles, and contextual information. Specifically, for a document D, we construct a table H_D, where each row corresponds to an event trigger t_i, each column corresponds to a role r_j, and each cell $H_D[i, j]$ captures the joint representation of t_i and r_j within the document context. The table embeddings are generated using a hierarchical transformer encoder:

$$H_D = \mathit{Encoder}_{\text{Table}}(D, T, R) \tag{1}$$

where $H_D \in \mathbb{R}^{m \times k \times d}$, with m triggers, k roles, and d representing the embedding dimension. This representation ensures structured and role-specific embeddings for all triggers and roles within the document.

3.2.2 Arguments Dependencies Types

To further enhance the table-based representations, we incorporate explicit dependency constraints that model relationships between triggers, roles, and arguments. Inspired by DEEIA (Liu et al., 2024), we define two types of dependencies:

Intra-Event Dependencies: These dependencies model the connections within an event, ensuring that argument roles are contextually aligned with their respective events.

Inter-Event Dependencies: These dependencies capture relationships between different events, where one event may overlap with or influence the same role in another event.

3.2.3 Dependency-Guided Attention

To integrate these dependencies into the encoding process, we extend the transformer's self-attention mechanism with a dependency-guided attention bias. For a pair of tokens (x_i, x_j), the attention score is adjusted as follows:

$$a_{ij} = \frac{\exp(e_i \cdot e_j + b_{ij})}{\sum_{k=1}^{n} \exp(e_i \cdot e_k + b_{ik})} \tag{2}$$

where e_i and e_j are the embeddings of tokens x_i and x_j, and b_{ij} is a learnable bias encoding the dependency relation between x_i and x_j.

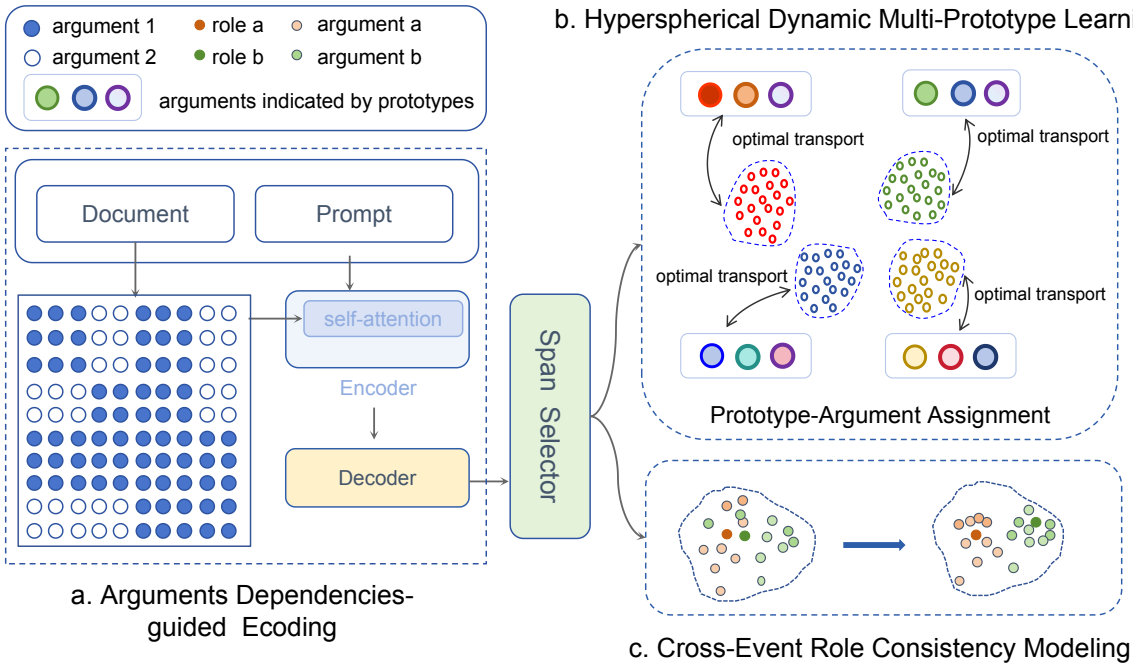

Figure 2: An overview of our proposed HDMAR.

The bias b_{ij} is computed as:

$$b_{ij} = W_{\text{dep}} \cdot \phi(x_i, x_j) \tag{3}$$

where $\phi(x_i, x_j)$ encodes the type (intra-event or inter-event) and strength of the dependency, and W_{dep} is a learnable weight vector.

Utilizing the refined attention scores, the DCE module generates role-trigger-specific representations $h_{t_i}^{r_j}$ for each (t, r) pair:

$$h_{t_i}^{r_j} = \text{Attention}(H_D, H_D[i][j]) \tag{4}$$

where $H_D[i][j]$ corresponds to the cell embedding for trigger t_i and role r_j in the table.

3.3 Hyperspherical Dynamic Multi-Prototype Learning

Traditional methods assume that each role r_j can be represented by a single static prototype. However, in real-world scenarios, the same role (e.g., Attacker) may exhibit diverse semantic behaviors depending on the event context, while inter-role boundaries (e.g., military base vs. Dhuluiyah) may overlap. To address these issues, we propose a hyperspherical dynamic multi-prototype learning mechanism that assigns multiple dynamic prototypes to each role and adapts them to contextual variations.

3.3.1 Multi-Prototype Representation

For each role r, we define M prototypes $P_{r_j} = \{p_{r_j}^1, p_{r_j}^2, \ldots, p_{r_j}^M\}$, where each prototype $p_{r_j}^k \in \mathbb{R}^d$ is a vector in a hyperspherical space. To ensure diversity among prototypes and avoid redundancy, we initialize the prototypes with maximal inter-prototype distances:

$$\|p_i - p_j\| \geq \delta, \quad \forall i \neq j \tag{5}$$

where δ is a margin controlling the distance between prototypes.

Given the role-specific representation $h_{t_i}^{r_j}$, the assignment of an argument a to a prototype is modeled as a soft probability distribution:

$$\pi(a, p_{r_j}^k) = \frac{\exp\left(- \parallel h_{t_i}^{r_j} - p_{r_j}^k \parallel^2\right)}{\sum_{l=1}^{M} \exp\left(- \parallel h_{t_i}^{r_j} - p_{r_j}^l \parallel^2\right)} \tag{6}$$

3.3.2 Optimal Transport for Prototype Assignment

The prototype-argument assignment process is formulated as an optimal transport (OT) problem, minimizing the overall transport cost of assigning arguments to prototypes:

$$L_{\text{OT}} = \min_{\pi} \sum_{a \in A} \sum_{k=1}^{M} \pi(a, p_{r_j}^k) \cdot \parallel h_{t_i}^{r_j} - p_{r_j}^k \parallel^2 \tag{7}$$

with the constraint $\sum_{k=1}^{M} \pi(a, p_{r_j}^k) = 1$ for all arguments a. To ensure prototypes are well-separated, we add a **separation regularization** term:

$$L_{\text{Proto-Sep}} = \sum_{i \neq k} \max(0, \delta - \parallel p^i_{r_j} - p^j_k \parallel) \quad (8)$$

The total prototype optimization objective is:

$$L_{\text{Proto}} = L_{\text{OT}} + \lambda_{\text{Sep}} \cdot L_{\text{Proto-Sep}} \quad (9)$$

where λ_{Sep} balances the contributions of the OT loss and separation regularization.

3.4 Cross-Event Role Consistency Modeling

Multi-event documents often include recurring roles (e.g., the same "Agent" across multiple events), which require consistency in their representation. Existing methods process events independently, leading to fragmented role representations. To address this, we introduce a **Cross-Event Role Consistency (CERC)** mechanism, which propagates role semantics across events within a document.

3.4.1 Graph-Based Role Propagation

We represent the document as a graph $G = (V, E)$, where nodes V are role representations $h^{r_j}_{t_i}$, and edges E capture semantic relationships between roles across events. A Graph Neural Network (GNN) is used to propagate information across nodes:

$$h^{r_j}_{t_i} \leftarrow \text{GNN}(h^{r_j}_{t_i}, \{h^{r_l}_{t_k} : (r_j, r_l) \in E\}) \quad (10)$$

where the GNN aggregates role representations $h^{r_l}_{t_k}$ from neighboring nodes. This ensures that recurring roles across events share consistent representations while maintaining inter-role distinctions.

3.4.2 Contrastive Role Alignment

To further enforce cross-event consistency, we adopt a contrastive loss to align representations of the same role across events while separating different roles:

$$L_{\text{CERC}} = -\sum_{(r_j, r'_j)} \log \frac{\exp(\text{sim}(h_{r_j}, h'_{r_j}))}{\sum_{r_k \neq r_j} \exp(\text{sim}(h_i, h_k))} \quad (11)$$

where $\text{sim}(,)$ is cosine similarity, and (r_j, r'_i) are instances of the same role across events.

3.4.3 Training Objective

The overall training objective integrates span selection, prototype optimization, and cross-event consistency:

$$L = L_{\text{Span}} + \lambda_{\text{Proto}} \cdot L_{\text{Proto}} + \lambda_{\text{CERC}} \cdot L_{\text{CERC}} \quad (12)$$

where L_{Span} is the span-based argument extraction loss, and λ_{Proto}, λ_{CERC} controls the contributions of the prototype and consistency losses respectively. By jointly optimizing for dynamic prototypes, cross-event consistency, and argument extraction accuracy, HDMAR achieves superior performance in multi-event scenarios.

4 Experiments

4.1 Experiment Setup

Datasets We evaluate our model on two widely-used event argument extraction benchmarks: RAMS (Ebner et al., 2020) and WikiEvents (Li et al., 2021). These datasets provide comprehensive coverage of event argument structures and roles, facilitating robust assessment of our approach.

RAMS is a document-level EAE corpus with 3,993 English annotated documents totaling 9,124 examples, 139 event types, and 65 argument roles. Each example contains five sentences, one event, and some arguments. We follow the original dataset split.

WikiEvents is another document-level EAE corpus with 246 English annotated documents, 50 event types, and 59 argument roles. These documents are obtained from English Wikipedia articles that describe real-world occurrences and then follow the reference links to crawl related news articles.

Evaluation Metric Following previous works (Ma et al., 2022; He et al., 2023), we evaluate performance using two metrics: (1) Argument Identification F_1 (Arg-I), where a predicted argument is considered correct if its span matches that of any golden argument for the event. (2) Argument Classification F_1 (Arg-C), where a predicted argument is considered correct if both its span and role type are accurate.

Baselines We compare HDMAR against state-of-the-art methods in EAE, including: (1) Classification-based methods, EEQA (Du and Cardie, 2020) and TSAR (Xu et al., 2022); (2) Generation-based methods, BART-Gen (Li et al.,

Model	PLM	RAMS		WikiEvents	
		Arg-I	Arg-C	Arg-I	Arg-C
EEQA⋆ (2020)	BERT	48.7	46.7	56.9	54.5
EEQA⋆ (2020)	RoBERTa	51.9	47.5	60.4	57.2
BART-Gen⋆ (2021)	BART	51.2	47.1	66.8	62.4
TSAR⋆ (2022)	RoBERTa	57.0	52.1	71.1	65.8
PAIE⋆ (2022)	BART	57.1	52.6	70.2	65.1
TabEAE* (2023)	RoBERTa	57.0	52.5	70.8	65.4
DEEIA* (2024)	RoBERTa	58.0	53.4	71.8	<u>67.0</u>
HMPEAE* (2024)	RoBERTa	<u>58.6</u>	<u>53.7</u>	<u>72.1</u>	66.6
HDMAR (Ours)	RoBERTa	**58.7**	**54.6**	**72.4**	**67.4**

Table 1: Overall results. We highlight the best result in **bold** and underline the second-best result. * indicates that we have rerun the relevant code. The symbol ⋆ indicates results from He et al. (2023). All pre-trained models (PLMs) are of large-scale.

2021), PAIE (Ma et al., 2022), TabEAE (He et al., 2023), HMPEAE (Zhang et al., 2024) and DEEIA (Liu et al., 2024).

Implementations. Each experiment is conducted on a single NVIDIA GeForce RTX 3090 24 GB. Due to the GPU memory limitation, we use different batch sizes for diverse models and corpora. For the RAMS and WikiEvents, the batch size for the base and large models are 8 and 4, respectively. The learning rate is set to 2e-5 for the AdamW optimizer with Linear scheduler, and the warmup ratio is 0.1. The epoch is set to 50, and the early stop is set to 8, denoting the training will stop if the F1 score does not increase during 8 epochs on the development set. Furthermore, the max decoder sequence length of the EAE template is set to 50 and 80 for RAMS and WikiEvents, respectively. Moreover, the input in the document-level dataset sometimes exceeds the constraint of the max encoder sequence length; thus we add a window centering on the trigger words and only encode the words within the window. Following Ma et al. (2022), the window size is 250. Considering that a word will be tokenized into multiple sub-words, we average the representation of sub-words as the representation of the original word.

4.2 Main Results

Table 1 summarizes the performance comparison between HDMAR and baseline models on the RAMS and WikiEvents datasets. Our model demonstrates comprehensive improvements across both datasets, with notable improvements in Arg-C. The following observations can be made from the results: (1) HDMAR achieves the highest scores for both Arg-I and Arg-C on RAMS and WikiEvents. Specifically, on the Arg-C metric, which measures the correctness of both boundaries and role types, HDMAR outperforms the second-best model by 0.9 on RAMS and 0.4 on WikiEvents. This indicates that HDMAR significantly enhances classification accuracy by addressing role ambiguity and improving role-specific representation. (2) While the improvement in Arg-I is more modest, with HDMAR surpassing HDMAR by 0.1 on RAMS and 0.3 on WikiEvents, the Arg-C improvement is significantly larger. This suggests that the innovations in HDMAR, such as Dynamic Multi-Prototype Learning and Cross-Event Role Consistency, not only improve argument identification but also refine the model's ability to classify arguments into their correct roles, especially for complex multi-event scenarios. (3) Compared to other prompt-based methods, such as DEEIA and TabEAE, HDMAR achieves higher scores on all metrics. Even when directly compared with the hyperspherical HDMAR model, which shares a similar design philosophy, HDMAR demonstrates its superiority by effectively modeling intra-role diversity and inter-role distinctions through its dynamic prototype approach.

4.3 Ablation Study

To evaluate the contribution of each component in HDMAR, we conduct ablation studies on the

Model	RAMS		Wikievents	
	Arg-I	Arg-C	Arg-I	Arg-C
w/o DMP	56.8	51.7	69.8	62.9
w/o CERC	57.2	52.4	70.1	65.1
w/o ADCE	56.8	51.3	69.5	63.6
w/o CL	57.5	52.7	69.6	64.9
w/o Hypersphere	57.3	52.1	69.1	63.5
w/o EMA	55.4	50.7	70.4	65.3
HDMP	**58.7**	**54.6**	**72.4**	**67.4**

Table 2: Ablation experiments on both datasets.The score would decrease without any kind of module.

RAMS and WikiEvents datasets (Table 2).

(1) **w/o Dynamic Multi-Prototypes (DMP)**. We drop dynamic update mechanism and just set multiple prototypes for each role.

(2) **w/o Cross-Event Role Consistency (CERC)**. In the structure of the model, we removed the cross-event role consistency mechanism.

(3) **w/o Arguments Dependencies-guided Context Encoding (ADCE)**. We replace the arguments dependencies-guided encoding module with a vanilla transformer encoder.

(4) **w/o Compactness Loss (CL)**. In training the EAE model, we remove the compactness loss.

(5) **w/o Hypersphere**. We remove the hypersphere setting, and just simply randomly generate multiple prototypes for each role.

(6) **w/o EMA**. During training, we freeze the prototypes and do not optimize them.

The results from the ablation study (Table 2) confirm the effectiveness of each individual component in the HDMAR framework. Removing any of the modules leads to a decline in performance, highlighting the complementary nature of the proposed innovations. Specifically, dynamic multi-prototype learning and cross-event role consistency are critical for addressing role diversity and capturing inter-event correlations. The dependency-guided encoding and compactness loss further enhance the model's ability to maintain context and regularize the embedding space, while the hyperspherical constraint and EMA ensure stable and effective prototype learning. Together, these components contribute significantly to the superior performance of HDMAR on both the RAMS and WikiEvents datasets.

4.4 Experiments on Different Number of Prototypes

M	RAMS		WikiEvents	
	Arg-I	Arg-C	Arg-I	Arg-C
1	57.1	52.3	69.3	63.1
2	**58.6**	**53.7**	70.2	65.1
3	57.1	52.4	**72.1**	**66.6**
4	57.3	52.6	69.9	65.0

Table 3: Experiments with the different numbers of prototypes. M denotes the number of prototypes for each role.

We analyze the impact of the number of prototypes by increasing the number of role prototypes to find the optimal setup for each dataset. As shown in Table 3, setting two prototypes for each role achieves the best performance on RAMS. Setting three prototypes for each role achieves the best performance on WikiEvents. We did not conduct prototype experiments with more settings because additional prototypes would incur higher computational costs. And setting too large will affect the performance because there may not be enough argument features to learn representative prototypes, which leads to underfitting.

4.5 Comparing with Large Language Models

ChatGPT has stimulated the research boom in the field of large language models (LLMs). To investigate the effect of LLMs on EAE, we follow (Wadden et al., 2019) and (Lin et al., 2020) to preprocess, resulting in two variants: ACE05-E and ACE05-E$^+$. Both contain 33 event types and 22 argument roles.

From Table 4, HDMAR demonstrates competitive performance with DEGREE and AMPERE. In contrast to PAIE, HDMAR exhibits a significant 2.1% improvement in ACE05-E. When considering ACE05-E, ChatGPT could achieve 33.95% and 42.79% performance of our model under zero-shot and 5-shot ICL setting, respectively. Similarly, comparable results could be seen in ACE05-E$^+$. Notably, ChatGPT consistently exhibits superior performance under the 5-shot ICL setting than the zero-shot scenario, highlighting the impact of task-specific information in enhancing model performance. Nevertheless, there is still a huge performance gap between ChatGPT and HDMAR. For EAE, it is evident that substantial progress is re-

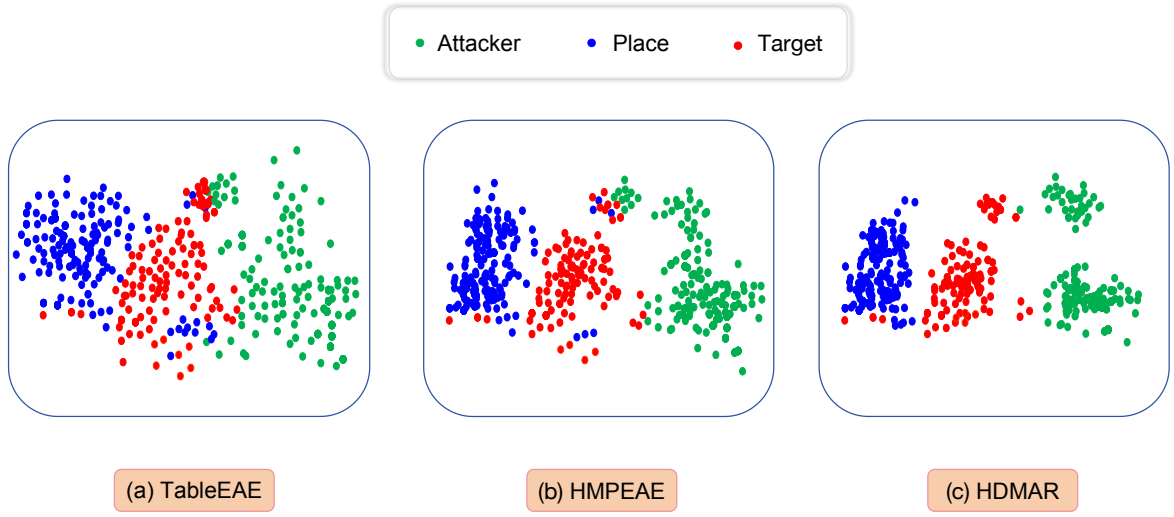

Figure 3: The t-SNE visualization above demonstrates the feature distributions of argument roles extracted from the "Conflict.Attack.Unspecified" event type.

Method	ACE05-E	ACE05-E$^+$
BERT (Devlin et al., 2019)	65.3	64
RoBERTa (Liu et al., 2019)	68	66.5
PAIE (Ma et al., 2022)	72.7	-
DEGREE (Hsu et al., 2022)	73.5	73
AMPERE (Hsu et al., 2023)	74.2	-
Zero-shot ∗	25.09	25.80
5-shot ICL ∗	31.62	32.02
HDMAR	**74.8**	**73.9**

Table 4: Argument classification F1-scores for EAE on ACE05-E and ACE05-E$^+$. ∗ Following Han et al. (2023), we evaluate the performance of ChatGPT under 2 settings: zero-shot prompts and 5-shot in-context learning (ICL) prompts.

quired for LLMs. Presently, our model demonstrates a notable capacity for achieving superior results. Looking ahead to future research, it is apparent that large models hold promise as a valuable auxiliary resource for more complex extraction tasks.

4.6 Visual Analysis

We extract argument features of the event type "Conflict.Attack.Unspecified" from the best checkpoint on Wikievents and transform them into 2D features using t-SNE. As shown in Figure 3, firstly, arguments playing the role of "Attacker" form two sub-clusters in the feature space, which suggests intra-class variation. HDMAR can capture such intra-class variance by setting multiple prototypes for each role, resulting in more compact clusters for arguments of the same type than TableEAE and HMPEAE. Moreover, during the encoding phase, we can introduce a bias term to the attention layers of the encoder based on the dependency relationships between different arguments, which helps to make the boundaries between the arguments more distinct.

Second, compared to TableEAE and HMPEAE, there is a clearer separation between the argument types of "Place" and "Target" in HDMAR. Additionally, we observe that arguments of "Place" do not partition into multiple sub-clusters within the feature space in HDMAR, while this is more pronounced in TableEAE and HMPEAE. This suggests that not all roles exhibit significant semantic differences, and HDMAR better consolidates semantically similar arguments, improving the overall distinction between argument types.

5 Conclusion

In this paper, we presented HDMAR, a novel approach for document-level Event Argument Extraction (EAE) that effectively addresses the challenges of intra-class variance and ambiguous role argument boundaries. By introducing Hyperspherical Dynamic Multi-Prototype Learning, Cross-Event Role Consistency, and an Arguments Dependencies-guided Encoding modules, HDMAR offers a comprehensive solution to improve both the accuracy and efficiency of multi-event argument extraction. Our method captures intra-role diversity, enforces inter-role separation, and ensures coherent role assignment across events, while simultaneously considering the contextual dependencies between arguments and roles.

6 Limitations

Our approach exhibits two primary limitations in prototype learning and input processing. First, the uniform allocation of prototypes across categories may artificially inflate inter-class variance for classes with inherently low intra-class variation, while simultaneously failing to sufficiently model the complex substructures of categories containing multiple latent subclusters in the embedding space. Second, the sequence concatenation strategy encounters length constraints that necessitate suboptimal sliding window processing with averaged overlapping embeddings, potentially compromising the integrity of contextual representations for lengthy text inputs.

References

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.

George Doddington, Alexis Mitchell, Mark Przybocki, Lance Ramshaw, Stephanie Strassel, and Ralph Weischedel. 2004. The automatic content extraction (ACE) program – tasks, data, and evaluation. In *Proceedings of the Fourth International Conference on Language Resources and Evaluation (LREC'04)*, pages 837–840.

Xinya Du and Claire Cardie. 2020. Event extraction by answering (almost) natural questions. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 671–683.

Xinya Du, Alexander M Rush, and Claire Cardie. 2021. Template filling with generative transformers. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 909–914.

Seth Ebner, Patrick Xia, Ryan Culkin, Kyle Rawlins, and Benjamin Van Durme. 2020. Multi-sentence argument linking. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8057–8077.

Ridong Han, Tao Peng, Chaohao Yang, Benyou Wang, Lu Liu, and Xiang Wan. 2023. Is information extraction solved by chatgpt? an analysis of performance, evaluation criteria, robustness and errors. *arXiv preprint arXiv:2305.14450*.

Xiaofeng Han, Xiangwu Meng, and Yujie Zhang. 2025. Exploiting multiple influence pattern of event organizer for event recommendation. *Information Processing Management*, page 103966.

Yuxin He, Jingyue Hu, and Buzhou Tang. 2023. Revisiting event argument extraction: Can EAE models learn better when being aware of event co-occurrences? In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12542–12556.

I-Hung Hsu, Kuan-Hao Huang, Elizabeth Boschee, Scott Miller, Prem Natarajan, Kai-Wei Chang, and Nanyun Peng. 2022. DEGREE: A data-efficient generation-based event extraction model. In *Proceedings of the 2022 NAACL*, pages 1890–1908.

I-Hung Hsu, Zhiyu Xie, Kuan-Hao Huang, Prem Natarajan, and Nanyun Peng. 2023. AMPERE: AMR-aware prefix for generation-based event argument extraction model. In *Proceedings of the 2023 ACL*, pages 10976–10993.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880.

Sha Li, Heng Ji, and Jiawei Han. 2021. 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*, pages 894–908.

Ying Lin, Heng Ji, Fei Huang, and Lingfei Wu. 2020. A joint neural model for information extraction with global features. In *Proceedings of the 2020 ACL*, pages 7999–8009.

Wanlong Liu, Shaohuan Cheng, Dingyi Zeng, and Qu Hong. 2023. Enhancing document-level event argument extraction with contextual clues and role relevance. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 12908–12922.

Wanlong Liu, Li Zhou, DingYi Zeng, Yichen Xiao, Shaohuan Cheng, Chen Zhang, Grandee Lee, Malu Zhang, and Wenyu Chen. 2024. Beyond single-event extraction: Towards efficient document-level multi-event argument extraction. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 9470–9487.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Yubo Ma, Zehao Wang, Yixin Cao, Mukai Li, Meiqi Chen, Kun Wang, and Jing Shao. 2022. 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)*, pages 6759–6774.

Pascal Mettes, Elise van der Pol, and Cees Snoek. 2019. Hyperspherical prototype networks. In *Advances in Neural Information Processing Systems*.

Chien Nguyen, Hieu Man, and Thien Nguyen. 2023. Contextualized soft prompts for extraction of event arguments. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 4352–4361.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, pages 1–67.

Yubing Ren, Yanan Cao, Ping Guo, Fang Fang, Wei Ma, and Zheng Lin. 2023. 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)*, pages 293–306.

Tarcísio Souza Costa, Simon Gottschalk, and Elena Demidova. 2020. Event-qa: A dataset for event-centric question answering over knowledge graphs. In *Proceedings of the 29th CIKM*, page 3157–3164.

David Wadden, Ulme Wennberg, Yi Luan, and Hannaneh Hajishirzi. 2019. Entity, relation, and event extraction with contextualized span representations. In *Proceedings of the 2019 EMNLP and the 9th IJCNLP*, pages 5784–5789.

Kaiwen Wei, Xian Sun, Zequn Zhang, Jingyuan Zhang, Guo Zhi, and Li Jin. 2021. Trigger is not sufficient: Exploiting frame-aware knowledge for implicit event argument extraction. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4672–4682.

Yuwei Xia, Mengqi Zhang, Qiang Liu, Shu Wu, and Xiao-Yu Zhang. 2022. MetaTKG: Learning evolutionary meta-knowledge for temporal knowledge graph reasoning. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 7230–7240.

Runxin Xu, Peiyi Wang, Tianyu Liu, Shuang Zeng, Baobao Chang, and Zhifang Sui. 2022. 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*, pages 5025–5036.

Zijie Xu, Peng Wang, Wenjun Ke, Guozheng Li, Jiajun Liu, Ke Ji, Xiye Chen, and Chenxiao Wu. 2024. Incorporating schema-aware description into document-level event extraction. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI-24*, pages 6597–6605.

Yuqing Yang, Qipeng Guo, Xiangkun Hu, Yue Zhang, Xipeng Qiu, and Zheng Zhang. 2023. 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)*, pages 12876–12889.

Qi Zeng, Qiusi Zhan, and Heng Ji. 2022. EA^{2}E: Improving consistency with event awareness for document-level argument extraction. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 2649–2655.

Guangjun Zhang, Hu Zhang, YuJie Wang, Ru Li, Hongye Tan, and Jiye Liang. 2024. Hyperspherical multi-prototype with optimal transport for event argument extraction. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9271–9284.

Tianran Zhang, Muhao Chen, and Alex A. T. Bui. 2020a. Diagnostic prediction with sequence-of-sets representation learning for clinical events. In *Artificial Intelligence in Medicine*, pages 348–358.

Zhisong Zhang, Xiang Kong, Zhengzhong Liu, Xuezhe Ma, and Eduard Hovy. 2020b. A two-step approach for implicit event argument detection. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7479–7485.