Learning to Generate and Extract: A Multi-Agent Collaboration Framework For Zero-shot Document-level Event Arguments Extraction

Guangjun Zhang[1], Hu Zhang[1,2*], Yazhou Han[1], Yue Fan[1], Yuhang Shao[1], Hongye Tan[1,2], Ru Li[1,2]

[1]School of Computer and Information Technology, Shanxi University, Taiyuan, China
[2]Key Laboratory of Computational Intelligence and Chinese Information Processing of Ministry of Education, Shanxi University, Taiyuan, China
zgj2866@gmail.com, {zhanghu, hanyazhou, 202322408041, tanhongye, liru}@sxu.edu.cn, yuefan24@163.com

Abstract

Document-level event argument extraction (DEAE) is essential for knowledge acquisition, aiming to extract participants of events from documents. In the zero-shot setting, existing methods employ LLMs to generate synthetic data to address the challenge posed by the scarcity of annotated data. However, relying solely on *Event-type-only prompts* makes it difficult for the generated content to accurately capture the contextual and structural relationships of unseen events. Moreover, ensuring the reliability and usability of synthetic data remains a significant challenge due to the absence of quality evaluation mechanisms. To this end, we introduce a multi-agent collaboration framework for zero-shot document-level event argument extraction (ZS-DEAE), which simulates the human collaborative cognitive process of "Propose–Evaluate–Revise." Specifically, the framework comprises a generation agent and an evaluation agent. The generation agent synthesizes data for unseen events by leveraging knowledge from seen events, while the evaluation agent extracts arguments from the synthetic data and assesses their semantic consistency with the context. The evaluation results are subsequently converted into reward signals, with event structure constraints incorporated into the reward design to enable iterative optimization of both agents via reinforcement learning. In three zero-shot scenarios constructed from the RAMS and WikiEvents datasets, our method achieves improvements both in data generation quality and argument extraction performance, while the generated data also effectively enhances the zero-shot performance of other DEAE models.

Code — https://github.com/GJZhang2866/GenExtract

Introduction

Document-level event argument extraction (DEAE) is one of the essential tasks in information extraction. With the rapid development of large language models (LLMs), DEAE has shown new potential and value in enhancing information retrieval quality, expanding the coverage of knowledge graphs, and enabling trustworthy model editing. Zero-shot document-level event argument extraction (ZS-DEAE) targets the extraction of arguments for unseen event types by utilizing training data associated with seen events. The sets of event types used during training and testing are disjoint. However, they are allowed to share overlapping event roles to maintain event structure definition (Huang et al. 2018). Hence, ZS-DEAE aligns with the generalized zero-shot learning paradigm (Pourpanah et al. 2023).

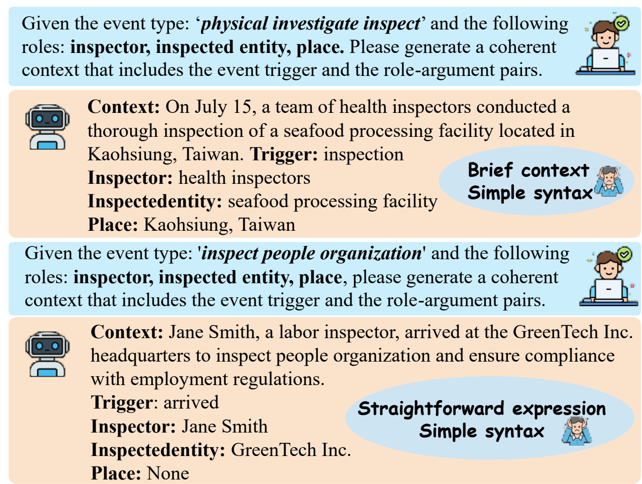


Figure 1: Examples of DEAE data generated by GPT-4o.

Most zero-shot event extraction studies focus on sentence-level settings and frame the task as knowledge transfer. Models are typically fine-tuned on seen events using shared semantic spaces or prototypical networks (Huang et al. 2018), enabling limited generalization to unseen types. Another line of work reformulates argument extraction as question answering (Du and Cardie 2020; Lyu et al. 2021), crafting role-specific queries and using pre-trained QA models to identify arguments directly from text. However, in document-level zero-shot scenarios, these approaches still lag far behind fully supervised models. Empirical evidence shows that DEAE performance improves notably with more annotated data (Ma et al. 2022; Zhang et al. 2024a), highlighting the continued dependence on high-quality supervision. Although LLMs offer strong reasoning and extensive parametric knowledge (Yan et al. 2025), their effectiveness in ZS-DEAE remains weak (Sharif et al. 2025). This raises a key question: Can LLMs be leveraged to generate

[*]corresponding author.

structured synthetic data for unseen events to address data scarcity? Recent efforts explore LLM-generated data for low-resource information extraction (Zhou et al. 2024b; Xu et al. 2024; Wang and Huang 2024b), yet producing high-quality DEAE data is still highly challenging. Document-level events often involve numerous roles, cross-sentence arguments, and complex semantic relations. This demands that LLMs not only deeply understand event and role semantics, but also generate event instances that are linguistically natural and structurally coherent. Moreover, evaluating the quality of generated data is inherently difficult. Without effective quality control, noisy samples can be introduced, which may degrade downstream extraction performance. As illustrated in Fig.1, the event types "*inspect people organization*" and "*physical investigate inspect*" both involve the action "inspect". The former emphasizes examining people or organizations, while the latter focuses on physical observation via sensory perception or instruments. However, LLM-generated examples often fail to clearly distinguish between these event types. Additionally, the generated contexts are often linguistically straightforward and arguments that are concentrated, lacking the contextual richness and structural complexity essential for the DEAE.

To address these challenges, we propose a multi-agent collaboration framework that simulates the human cooperative process of "proposal–evaluate–rectify". The framework comprises two agents: a generation agent that produces document-level contexts with event triggers and structured role–argument pairs based on given event types and roles, and an evaluation agent that extracts arguments and assesses their semantic consistency with the context. The log-likelihood computed by the evaluation agent serves as a quality indicator and guides subsequent training. During the generation process, we observe that the generation agent tends to produce instances with multiple empty arguments (roles without corresponding arguments). Although such samples lack valid arguments, the evaluation agent tends to assign them high log-likelihood scores for correctly predicting `None`, biasing the generation agent toward structurally incomplete events and creating a feedback loop of cumulative bias. To mitigate this, we introduce event structural constraints and combine them with log-likelihood into a unified reward signal. We employ reinforcement learning to iteratively optimize multi-agent systems, thereby achieving improvements in both data generation quality and argument extraction performance. Our contributions are as follows:

- We propose a multi-agent collaboration framework for ZS-DEAE to tackle the key challenge of insufficient annotated data for unseen events.
- Evaluated on three zero-shot settings derived from RAMS and WikiEvents, our method achieves simultaneous improvements in synthetic data quality and extraction performance.
- Our method is generalizable, enhance the zero-shot capabilities of other models, providing a novel solution for ZS-DEAE.

Related Work

Event argument extraction. Event argument extraction (EAE) has received growing attention, with datasets such as RAMS (Ebner et al. 2020) and WikiEvents (Li, Ji, and Han 2021) driving progress in DEAE. Existing methods include classification-based methods (Xu et al. 2022; Liu et al. 2023), template-based frameworks (Ma et al. 2022; He, Hu, and Tang 2023; Zhang et al. 2024a; Liu et al. 2024), and recent LLM-driven models (Zhang et al. 2024b; Zhou et al. 2024a; Shuang et al. 2024). Despite strong performance, these methods remain heavily dependent on supervised data and show limited generalization to unseen event types. In zero-shot EAE, early work (Huang et al. 2018) maps events into shared neural spaces. QA-based methods were later adapted for unseen argument extraction (Du and Cardie 2020; Lyu et al. 2021). Lu et al. (2024) further improves generalization by disentangling arguments, roles, and triggers, while other studies enhance zero-shot argument classification using candidate spans (Zhang, Wang, and Roth 2021; Sainz et al. 2022; Lin, Zhang, and Song 2023).

Synthetic Data for IE. Synthetic data generation has become a promising strategy for low-resource information extraction. In relation extraction, Zhou et al. (2024c) use LLMs to create seed instances for fine-tuning smaller models and expand pattern diversity with iterative feedback. Xu et al. (2024) introduce a dual-agent framework to iteratively refine synthetic samples, while Li, Dai, and Li (2025) integrate in-context learning with direct preference optimization to increase data diversity. For event extraction, Wang and Huang (2024c) leverage external corpora to enrich lexical variation in generated samples.

Multi-agent for IE. Multi-agent collaboration has recently emerged as an influential paradigm in IE, spanning cooperative, competitive, and debate-based modes. Wang et al. (2025) propose a cooperative multi-agent system for zero-shot NER that separates entity recognition from type feature extraction and incorporates self-reflection to assess demonstration utility. Zhao et al. (2024, 2025) develop agent-guided mechanisms to improve explanation generation. Wang and Huang (2024a) design a debate-style optimization strategy for few-shot event extraction, and Hou et al. (2024) present a more scalable multi-agent framework.

Method

Task Definition

Document-level Event Argument Extraction (DEAE) aims to extract textual spans that serve as arguments for specific roles within a given event type, based on the entire input document. Formally, given a document d, an event type e, a trigger t, and a predefined role set $R_e = \{r_e^i\}_{i=1}^{N_{R_e}}$, the goal of DEAE is to extract a text span corresponding to each role r_e^i from d, serving as the argument for that role in the event.

Zero-shot DEAE (ZS-DEAE) targets the same task without access to labeled data for unseen event types. Specifically, given a set of seen event types E_s and their training dataset $D_s = \{(d_{e_s}, e_s, t, R_{e_s})_i\}_{i=1}^{|D_s|}$, the goal is to perform argument extraction from documents D_u under unseen event

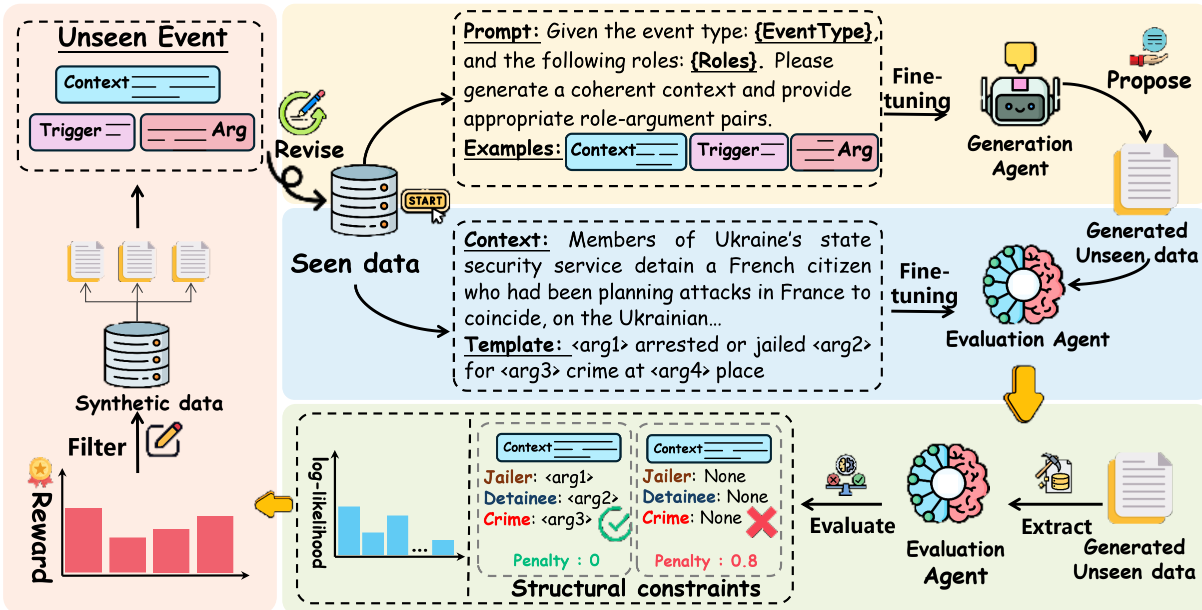


Figure 2: The overall workflow of multi-agent collaboration for ZS-DEAE

types E_u, where $E_s \cap E_u = \emptyset$. Although the event types differ between training and testing, role can be shared between them to ensure the integrity of the event schema.

Definition of Agents

Generation Agent. This agent is designed to generate a document-level context d, an event trigger t, and a set of role-argument pairs $A_e = \{(r_e^i, a_{r_e^i}) | r_e^i \in R_e, a_{r_e^i} \in d, i \in [0, N_{R_e})\}$, conditioned on a given event type e and its corresponding role set $R_e = \{r_e^i\}_{i=1}^{N_{R_e}}$, where $a_{r_e^i}$ denotes the argument corresponding to role r_e^i.

To fully leverage the generation capabilities of LLMs, we formalize the task in a prompt-based input-output format. Specifically, the input prompt is defined as: $E_{\text{in}}(e, R_e)$= " Given the event type: e and the following roles: R_e, please generate a coherent context that includes the event trigger and the role-argument pairs." The output is formatted as: $E_{\text{out}}(d, t, A_e)$ =" Context:d, Trigger:t,Role-Arguments: A_e." For roles that do not appear in the generated context, their corresponding arguments are marked as `None`. The agent is optimized on the training set D_s using an autoregressive objective:

$$\mathcal{L}_g = -\log P(E_{\text{out}} \mid E_{\text{in}}). \tag{1}$$

Evaluation Agent. This agent aims to extract arguments for a given event type e from a document-level context d, and to evaluate the structural completeness and semantic consistency of the generated content. In this work, we adopt Bart-Gen(Li, Ji, and Han 2021) as the evaluation agent, which is based on a conditional generation framework. The model takes the context d and an unfilled template sentence containing `<arg>` placeholders as input. The full input is formatted as:

$$c = \texttt{<s>}\ \text{template}\ \texttt{<s>}\ \texttt{</s>}\ d\ \texttt{</s>}, \tag{2}$$

and the output is the filled template x, where each `<arg>` placeholder is replaced with an argument extracted from the d. The generation process is modeled autoregressively as generating the target text x token by token, conditioned on both the input c and previously generated tokens:

$$p(x \mid c) = \prod_{i=1}^{|x|} p(x_i \mid x_{<i}, c). \tag{3}$$

To reduce hallucination, Bart-Gen restricts its output vocabulary to V_d, containing only tokens present in d. During training, the loss function minimizes the negative log-likelihood over the training set D_s:

$$\mathcal{L}_c = -\log P(x \mid c). \tag{4}$$

Multi-Agent Collaboration

To address the challenge of limited annotated data in the zero-shot setting, we propose a multi-agent collaborative framework in which two agents work in tandem to simultaneously improve the quality of synthetic data and the performance of event argument extraction. This framework emulates the human collaborative cognitive process of "Propose–Evaluate–Revise". Specifically, it consists of the following three core stages: (1) Propose: The generation agent attempts to express its understanding of unseen events by generating a document-level context, trigger, and a set of role-argument pairs for a given unseen event type; (2) Evaluate: the evaluation agent performs argument extraction on the generated data and assesses its semantic plausibility and structural completeness; and (3) Revise: the evaluation results are transformed into feedback signals to optimize both agents. Through this iterative and cooperative process, the two agents engage in a dynamic interaction that enables them to improve together over time. The overall workflow is described in Fig. 2.

Propose. Given an unseen event type e_{u_i} and its associated roles $R_{e_{u_i}}$, the generation agent expresses its understanding by generating document-level contexts, an event trigger, and role-argument pairs. Specifically, the event and roles are first mapped into a natural language prompt $E_{\text{in}}(e_{u_i}, R_{e_{u_i}})$, and the generation agent produces K candidate sequences based on this prompt. During generation, some contexts may omit the specified trigger or certain arguments. Therefore, samples missing the trigger in the generated context are discarded, while missing arguments are assigned the value `None`. Each valid generated sequence is decoded into a synthetic data instance represented as the tuple $(d, t, A_{e_{u_i}})$. Collecting all K samples yields synthetic dataset for e_{u_i} denoted as $D_{\text{syn}}^{e_{u_i}} = \{(d_j, t_j, A_{e_{u_i}}^j)\}_{j=1}^K$. The complete synthetic dataset for all unseen event types is formed by the union $D_{\text{syn}} = D_{\text{syn}}^{e_{u_1}} \cup \cdots \cup D_{\text{syn}}^{e_{u_m}}$.

Evaluate. After the generation agent expresses its understanding of the event, we use the evaluation agent to assess whether this understanding is accurate. The generation probability $P(x \mid c)$ of the evaluation agent can be interpreted as the likelihood of generating a specific argument-filled sentence x given the input c, or as the semantic match between c and x, which can then be used to measure the quality of the synthetic data $(d, t, A_{e_{u_i}}^j) \in D_{\text{syn}}^{e_{u_i}}$. Specifically, for each synthetic sample, we compute the log-likelihood of its corresponding filled template:

$$\ell_i = \log P(x_i \mid c_i). \tag{5}$$

High-quality samples are expected to yield higher log-likelihoods, while lower-quality samples should have lower scores, thereby enabling the differentiation of synthetic data quality. To further standardize this scoring, we normalized the log-likelihood:

$$\alpha_i = \frac{\ell_i - \mu}{\delta}, \tag{6}$$

where μ and δ denote the mean and standard deviation of log-likelihood values over the entire synthetic dataset D_{syn}.

During evaluation, we observed that many synthetic samples contain multiple empty arguments, meaning arguments that are generated as `None`. Such samples often receive high likelihood scores from the evaluation agent as it correctly predicts these `None`. This tendency may further induce the generation agent to prefer producing structurally incomplete events, thereby introducing and amplifying bias in the feedback loop. To mitigate this issue, we introduce the structural completeness constraint by penalizing the proportion of empty arguments in each synthetic sample. Specifically, for each sample, we compute the proportion of empty arguments ρ_i. We aim to keep this proportion close to that observed in the training data, whose expected value and standard deviation are denoted by τ and ε, respectively. The penalty term is defined as:

$$\mathrm{p}_i = \begin{cases} 0 & \text{if } \tau - \varepsilon \leq \rho_i \leq \tau + \varepsilon \\ |\rho_i - \tau| & \text{otherwise} \end{cases} \tag{7}$$

Finally, we integrate this penalty into the normalized log-likelihood score to obtain the final quality score for each synthetic sample:

$$\alpha_i = \frac{\ell_i - \mu}{\delta} - \mathrm{p}_i. \tag{8}$$

Revise. Reinforcement learning (RL) has been shown to be effective in learning preferences among data instances. To this end, we adopt an RL-based approach to refine both agents. By assigning higher rewards to higher-quality synthetic samples, the two agents are encouraged to better understand unseen events through data augmentation, thereby achieving joint improvement in both synthetic data quality and argument extraction performance. Specifically, we treat the normalized score α_i as the reward and optimize the generation agent and evaluation agent via policy gradient methods based on the expected reward over the synthetic dataset D_{syn}. The parameter updates for the two agents are defined as:

$$\mathcal{G}_i = \mathcal{G}_{i+1} + \gamma_1 \nabla_{\mathcal{G}} \mathbb{E}[\alpha], \tag{9}$$

$$\mathcal{E}_i = \mathcal{E}_{i+1} + \gamma_2 \nabla_{\mathcal{E}} \mathbb{E}[\alpha], \tag{10}$$

where γ_1 and γ_2 are the learning rates for the generation agent and the evaluation agent, respectively. The gradients are computed as:

$$\nabla_G \mathbb{E}[\alpha] = \mathbb{E}[\alpha_i \nabla_{\mathcal{G}} \log P(E_{\text{out}} \mid E_{\text{in}})], \tag{11}$$

$$\nabla_C \mathbb{E}[\alpha] = \mathbb{E}[\alpha_i \nabla_{\mathcal{E}} \log P(x \mid c)]. \tag{12}$$

Through continuous cycles of propose-evaluate-revise, the two agents progressively enhance their understanding of unseen events, effectively distinguish subtle semantic differences between similar event types, and iteratively improve their respective capabilities.

Experiments

Experimental Settings

Datasets. We construct three zero-shot settings based on the two most widely used DEAE datasets, RAMS and WikiEvents. These settings are `RAMS2RAMS`, `RAMS2Wiki`, and `Wiki2Wiki`. In the `RAMS2RAMS`, the RAMS dataset is split into seen and unseen event types. For `RAMS2Wiki`, we remove overlapping event types in WikiEvents that appear in RAMS, ensuring the test set contains only disjoint event types. The `Wiki2Wiki` setting follows the same protocol as `RAMS2RAMS`, where WikiEvents is split into seen and unseen event types.

Experimental Configuration and Metrics We select LLaMA3.1-8B (Dubey et al. 2024) and Qwen2.5-7B (Yang et al. 2024) as generation agent, fine-tuned via LoRA (Hu et al. 2022) for parameter-efficient adaptation. LoRA modules are inserted into the attention layers (query and value projections), with rank 8, scaling factor 32, and dropout rate 0.05. For the evaluation agent, we use Bart-large (Lewis

Model	RAMS2RAMS			RAMS2Wiki			Wiki2Wiki		
	Seen R.	Unseen R.	Overall	Seen R.	Unseen R.	Overall	Seen R.	Unseen R.	Overall
DEAE Models									
PAIE	32.52	28.87	30.80	19.57	31.72	20.15	23.58	23.57	24.42
TabEAE	37.16	35.26	36.22	16.94	**35.05**	26.74	37.19	28.84	30.97
DEEIA	36.57	39.49	37.95	1.50	7.17	5.12	34.11	19.48	22.51
HMPEAE	35.18	37.74	36.44	16.89	32.74	25.61	38.43	27.48	30.20
TSAR	38.10	21.56	30.90	15.77	13.37	11.71	14.40	13.86	13.95
SCPRG	38.93	26.97	33.58	10.80	10.00	9.40	45.80	11.89	21.90
Zero-shot Models									
EEQA	26.75	26.99	26.86	14.74	21.39	18.24	23.05	14.00	21.50
ZSTL	-	-	12.24	-	-	9.40	-	-	9.97
Bart-Gen	39.89	37.09	38.53	24.66	33.45	28.52	48.11	32.68	40.82
Distar	18.26	19.68	18.98	13.11	3.20	10.60	2.72	20.18	17.51
Large Language Models									
Phi-4	-	-	10.10	-	-	4.12	-	-	7.30
+CoT	-	-	12.89	-	-	2.78	-	-	5.78
Gemma-1.1	-	-	13.39	-	-	5.91	-	-	6.31
+CoT	-	-	5.94	-	-	3.19	-	-	4.78
Mixtral	-	-	19.35	-	-	4.70	-	-	6.34
+CoT	-	-	15.96	-	-	4.81	-	-	6.23
LLaMA-3.1	-	-	20.56	-	-	8.37	-	-	10.55
+CoT	-	-	17.05	-	-	6.78	-	-	8.87
GPT-4o	-	-	19.64	-	-	10.17	-	-	11.36
+CoT	-	-	18.36	-	-	7.98	-	-	9.65
DS-V3	-	-	22.00	-	-	10.89	-	-	11.60
+CoT	-	-	15.24	-	-	7.63	-	-	8.62
DS-R1	-	-	24.41	-	-	12.27	-	-	12.84
+CoT	-	-	23.47	-	-	9.24	-	-	10.40
Ours (LLaMA)	**46.46**	45.06	**45.77**	**30.81**	34.43	**32.38**	47.83	46.19	46.96
Ours (Qwen)	44.06	**45.11**	44.59	31.74	30.47	31.18	**47.39**	**47.82**	**47.62**

Table 1: Main Results on three zero-shot settings. Seen R. refers to roles shared between seen and unseen event schemas, while Unseen R. denotes roles unique to unseen schemas. Overall indicates the F1 score calculated over all roles. We use **bold** text to indicate the best performance and underline to indicate the second best. For ZSTL and LLM-based methods, since they are not trained on D_s but instead make predictions directly via queries or prompts, they do not distinguish between seen and unseen roles—thus, the corresponding entries are left blank.

et al. 2020) as the pre-trained model for Bart-Gen (Li, Ji, and Han 2021). We run the five-round agent interaction optimization with three different random seeds, and report the average result of the best-performing round from each run, to reduce the impact of randomness. For evaluation metrics, we adopt Span-F1 as the primary metric, where a prediction is correct only if the span exactly matches the gold span.

Baselines. We compare our method with three categories of baselines. The first includes **DEAE models**, such as PAIE (Ma et al. 2022), TabEAE (He, Hu, and Tang 2023), DEEIA (Liu et al. 2024), HMPEAE (Zhang et al. 2024a), TSAR (Xu et al. 2022), and SCPRG(Liu et al. 2023). Since TSAR and SCPRG are not designed for zero-shot settings, we adapt them into prototype-based classifiers for transfer to unseen event types. The second includes **sentence-level zero-shot models**, including EEQA (Du and Cardie 2020), ZSTL(Lyu et al. 2021), Bart-gen (Li, Ji, and Han 2021), and Distar(Lu et al. 2024). We adapt EEQA and ZSTL to our task format by redesigning their prompt templates. The third includes **LLMs**, such as Phi-4 (14B) (Abdin et al. 2024), Gemma-1.1 (7B) (Mesnard et al. 2024), Mixtral (8x22B) (Jiang et al. 2024), LLaMA3.1 (70B) (Dubey et al. 2024), GPT-4o(OpenAI 2023), DeepSeek V3 (DS-V3) (DeepSeek-AI 2024) and DeepSeek R1 (DS-R1) (DeepSeek-AI 2025). We evaluate them using both zero-shot and Chain-of-Thought (CoT) prompting following Sharif et al. (2025).

Main results

The main results are shown in Table 1. **Comparison with DEAE Models.** Our method, based on LLaMA and Qwen, consistently outperforms existing DEAE models across all three zero-shot settings. Ours (LLaMA) exceeds the strongest baseline DEEIA by 6.57, 5.57, and 7.82 F1 on seen roles, unseen roles, and overall in RAMS2RAMS, and achieves the highest overall F1 scores of 32.38 and 46.96 in RAMS2Wiki and Wiki2Wiki. These results reveal the limitations of prior DEAE models in zero-shot scenarios, as their dependence on manually annotated data reduces generalization. Our method also achieves more balanced performance between seen and unseen roles. **Compared with**

Model	R2R	R2W	W2W
Ours((LLaMA)	**45.28**	**32.38**	**46.96**
-reward	42.46	24.93	38.39
-constraint	44.72	29.03	40.98
Ours(QWEN)	**44.59**	**31.18**	**47.62**
-reward	40.52	27.19	43.30
-constraint	40.87	27.87	38.70

Table 2: Ablation study on the effects of reward and constraint across three datasets. **R2R**: `RAMS2RAMS`. **R2W**: `RAMS2Wiki`, **W2W**: `Wiki2Wiki`

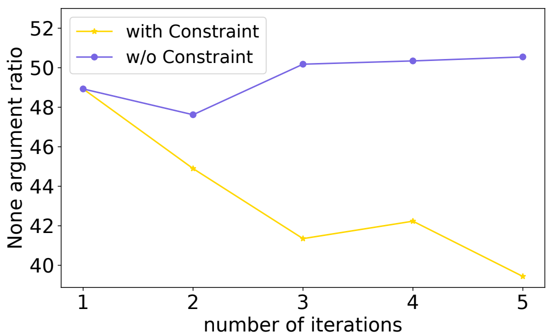
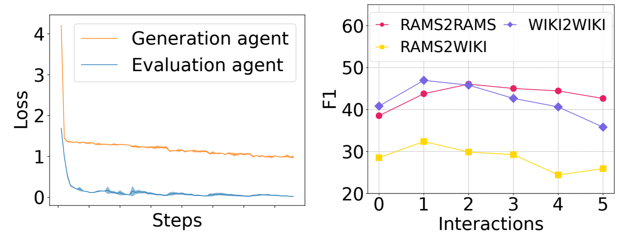


Figure 3: Proportion of empty arguments in synthetic data.

zero-shot models. Our method shows substantial gains over Bart-Gen, with overall F1 improvements of 7.53, 3.86, and 6.14 across datasets, benefiting from collaborative interaction between generation and evaluation agents. In contrast, EEQA, ZSTL, and Dister perform poorly, indicating limited capacity in document-level settings. **Comparison with LLMs.** We evaluate several mainstream LLMs using both Zero-shot and CoT prompts. The overall results show that LLMs perform significantly worse in Span-F1, with generally low performance levels. This aligns with observations by Sharif et al. (2025). A major factor is the strict boundary matching required by Span-F1. Although LLMs demonstrate strong language understanding, they often struggle to accurately identify precise argument spans. More flexible evaluation metrics may better reflect their potential.

Ablation Study

Our framework consists of the RL feedback mechanism and the event structure constraint. To evaluate their contributions, we conduct ablation studies by removing each component, as shown in Table 2. Removing either the RL reward or the structure constraint reduces performance, which shows that both components are effective. In most settings, removing the structure constraint produces a smaller performance drop, which suggests that multi-agent collaboration is the main factor. However, Fig.3 shows that the absence of this constraint increases the proportion of empty arguments and harms both data quality and extraction accuracy. In contrast, applying the constraint reduces empty arguments and improves the completeness of arguments as well as the quality of synthetic data.



(a) (b)

Figure 4: (a): Loss of the two agent during reinforcement learning on `RAMS2RAMS`. (b): F1 trend across three datasets.

Analysis

Impact of Interaction Rounds

Fig. 4a shows the loss curves of the generation and evaluation agents, assessing RL process stability. Both losses decline steadily during training, indicating that the optimization method supports cooperative convergence and ensures stable training. Fig. 4b presents F1 trends across three dataset settings. The model reaches peak F1 scores within one to two interaction rounds, but performance gradually declines with more interactions. As analyzed later, this may result from reduced diversity in generated samples, which undermines the model's generalization.

Analysis of Generation Agent

To evaluate the quality of synthetic data generated by the Generation Agent, we augment the training sets of baseline models and compare them with data generated directly by LLaMA under the same settings. Performance improvements from each source serve as an indirect measure of data quality.We conduct experiments on two representative DEAE models: TabEAE, which shows balanced performance (Table 1), and Bart-Gen, a stronger zero-shot model. As shown in Table 3, our method consistently achieves stable and significant gains across different settings, outperforming LLaMA-generated data. These results indicate that our synthetic data is higher in quality and generalizability, effectively enhancing ZS-DEAE.

Analysis of Evaluation Agent

To assess the evaluation agent's ability to distinguish different data quality, we construct low-quality variants of the test set by randomly replacing or removing arguments.Each instance is scored using log-likelihood, with results shown in Fig.5. consistently assigns higher log-likelihoods to high-quality data and lower scores to degraded examples. This indicates that the agent possesses a certain degree of sensitivity to data quality and can effectively differentiate between well-formed and degraded examples.

Diversity Analysis of Synthetic Data

We evaluate synthetic data diversity following Kirk et al. (2024), which separates per-input and across-input diversity, and adopt the four-dimensional analysis of Guo, Shang,

Model	Data	RAMS2RAMS	RAMS2Wiki	Wiki2Wiki
TabEAE	Seen data	36.22	26.74	30.97
	+ Synthetic data ((LLaMA 3.1)	37.37	26.17	12.48
	+ Synthetic data (Ours)	**44.43**	**28.42**	**33.36**
Bart-Gen	Seen data	38.53	28.52	40.82
	+ Synthetic data (LLaMA 3.1)	41.04	28.95	44.17
	+ Synthetic data (Ours)	**46.06**	**32.38**	**46.96**

Table 3: Performance comparison on three transfer settings using unseen and synthetic data.

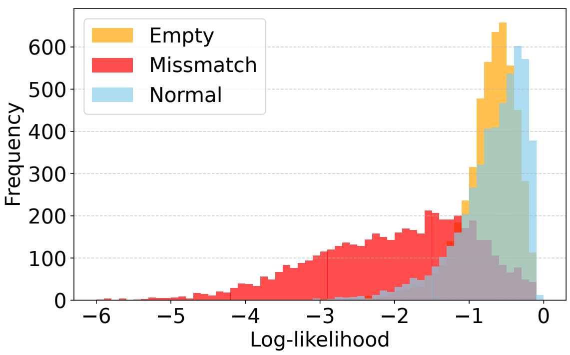
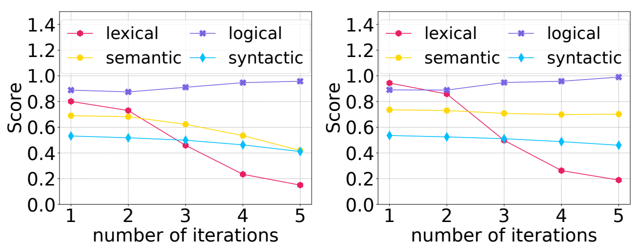


Figure 5: Log-likelihood scores assigned by the evaluation agent to different types of synthetic data. Empty: randomly removing arguments. Mismatch: randomly swapping argument-role mappings. Normal: correctly generated data without perturbation.



(a) Per-input diversity (b) Across-input diversity

Figure 6: Diversity of Synthetic Data.

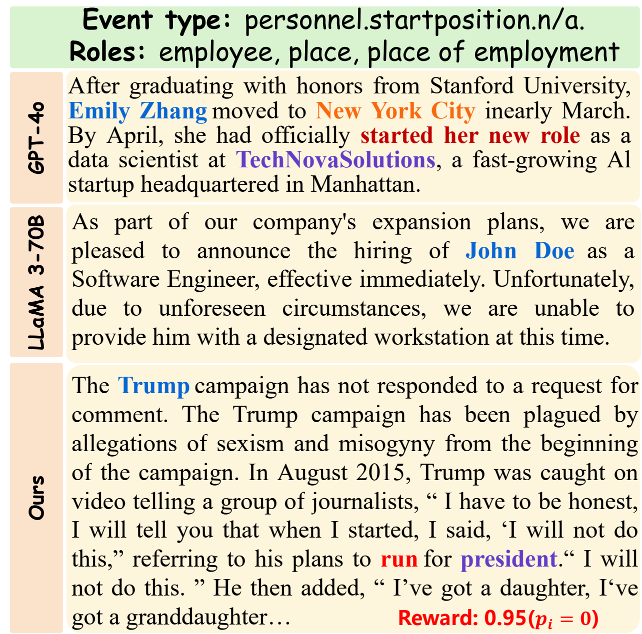


Figure 7: Case studies of synthetic data.

and Clavel (2024): lexical, semantic, logical, and syntactic. Results appear in Fig.6. As interaction rounds increase, diversity decreases in the lexical, semantic and syntactic dimensions, indicating that the model shifts toward stable generation strategies that favor patterns with higher log-likelihood. This convergence may improve consistency but reduces generalization and contributes to performance decline in later rounds. Logical-level diversity remains relatively stable, possibly because lexical and syntactic convergence introduces inconsistencies that sustain variation in logical content. Future work should enhance diversity, especially in the lexical, semantic and syntactic.

Case Study

We present a case study in Fig.7, analyzing synthetic data generated by GPT-4o, LLaMA3.1 (70B), and our method using LLaMA3.1 (8B). The outputs from GPT-4o and LLaMA3.1 (70B) tend to be short, syntactically simple, and contains densely packed arguments, making them less suitable for DEAE. Notably, LLaMA3.1 (70B) often omits the trigger, as well as key arguments of *place* and *placeofemployment*. In contrast, our method generates a longer document with more widely dispersed arguments. Moreover, in the fourth case, a mismatch between the context and event type results in a low reward, highlighting the effectiveness of evaluation agent.

Conclusion

This paper presents a multi-agent collaborative framework for ZS-DEAE that mitigates the shortage of annotated data. The framework models a "Propose–Evaluate–Revise" cycle to iteratively improve synthetic data and enhance extraction accuracy. Experiments show clear improvements in data quality and argument extraction, surpassing several mainstream LLMs. Additional analyses indicate that the generated data can also strengthen other models in zero-shot settings, providing a promising solution for ZS-DEAE. Future work will extend this framework to broader information extraction tasks and other low-resource scenarios.

Acknowledgments

We thank all the anonymous reviewers for their constructive comments and suggestions. This work is supported by the National Natural Science Foundation of China (62176145, 62476161), and the Interdisciplinary Research Fund of Shanxi University.

References

Abdin, M. I.; Aneja, J.; Behl, H. S.; and et al. 2024. Phi-4 Technical Report. *arXiv preprint*.

DeepSeek-AI. 2024. DeepSeek-V3 Technical Report. *arXiv preprint*, abs/2412.19437.

DeepSeek-AI. 2025. DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning. *arXiv preprint*, abs/2501.12948.

Du, X.; and Cardie, C. 2020. Event Extraction by Answering (Almost) Natural Questions. In Webber, B.; Cohn, T.; He, Y.; and Liu, Y., eds., *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, 671–683. Association for Computational Linguistics.

Dubey, A.; Jauhri, A.; Pandey, A.; and et al. 2024. The Llama 3 Herd of Models. *CoRR*, abs/2407.21783.

Ebner, S.; Xia, P.; Culkin, R.; Rawlins, K.; and Van Durme, B. 2020. Multi-Sentence Argument Linking. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 8057–8077.

Guo, Y.; Shang, G.; and Clavel, C. 2024. Benchmarking Linguistic Diversity of Large Language Models. *CoRR*, abs/2412.10271.

He, Y.; Hu, J.; and Tang, B. 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)*, 12542–12556.

Hou, W.; Jia, N.; Liu, X.; Zhao, W.; and Wang, Z. 2024. A Multiagent-Based Document-Level Relation Extraction System With Entity Pair Awareness and Sentence Significance. *IEEE Syst. J.*, 18(4): 1905–1916.

Hu, E. J.; Shen, Y.; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; Wang, L.; and Chen, W. 2022. LoRA: Low-Rank Adaptation of Large Language Models. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*.

Huang, L.; Ji, H.; Cho, K.; Dagan, I.; Riedel, S.; and Voss, C. R. 2018. Zero-Shot Transfer Learning for Event Extraction. In Gurevych, I.; and Miyao, Y., eds., *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, 2160–2170. Association for Computational Linguistics.

Jiang, A. Q.; Sablayrolles, A.; Roux, A.; and et al. 2024. Mixtral of Experts. *arXiv preprint*, abs/2401.04088.

Kirk, R.; Mediratta, I.; Nalmpantis, C.; Luketina, J.; Hambro, E.; Grefenstette, E.; and Raileanu, R. 2024. Understanding the Effects of RLHF on LLM Generalisation and Diversity. In *The Twelfth International Conference on Learning Representations, Vienna, Austria, May 7-11, 2024*.

Lewis, M.; Liu, Y.; Goyal, N.; Ghazvininejad, M.; Mohamed, A.; Levy, O.; Stoyanov, V.; and Zettlemoyer, L. 2020. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. In Jurafsky, D.; Chai, J.; Schluter, N.; and Tetreault, J. R., eds., *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 7871–7880. Association for Computational Linguistics.

Li, S.; Ji, H.; and Han, J. 2021. Document-Level Event Argument Extraction by Conditional Generation. In Toutanova, K.; Rumshisky, A.; Zettlemoyer, L.; Hakkani-Tur, D.; Beltagy, I.; Bethard, S.; Cotterell, R.; Chakraborty, T.; and Zhou, Y., eds., *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 894–908. Association for Computational Linguistics.

Li, Z.; Dai, H.; and Li, P. 2025. Generating Diverse Training Samples for Relation Extraction with Large Language Models. In Che, W.; Nabende, J.; Shutova, E.; and Pilehvar, M. T., eds., *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 713–726. Vienna, Austria: Association for Computational Linguistics. ISBN 979-8-89176-251-0.

Lin, Z.; Zhang, H.; and Song, Y. 2023. Global Constraints with Prompting for Zero-Shot Event Argument Classification. In *Findings of the Association for Computational Linguistics: EACL 2023*, 2527–2538.

Liu, W.; Cheng, S.; Zeng, D.; and Qu, H. 2023. Enhancing Document-level Event Argument Extraction with Contextual Clues and Role Relevance. In *Findings of the Association for Computational Linguistics*, 12908–12922.

Liu, W.; Zhou, L.; Zeng, D.; Xiao, Y.; Cheng, S.; Zhang, C.; Lee, G.; Zhang, M.; and Chen, W. 2024. Beyond Single-Event Extraction: Towards Efficient Document-Level Multi-Event Argument Extraction. In *Findings of the Association for Computational Linguistics, ACL 2024,*, 9470–9487.

Lu, Z.; Zeng, Z.; Wang, J.; et al. 2024. Zero-shot event argument extraction by disentangling trigger from argument and role. *International Journal of Machine Learning and Cybernetics*, 16: 3233–3251.

Lyu, Q.; Zhang, H.; Sulem, E.; and Roth, D. 2021. Zero-shot Event Extraction via Transfer Learning: Challenges and Insights. In Zong, C.; Xia, F.; Li, W.; and Navigli, R., eds., *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, 322–332. Online: Association for Computational Linguistics.

Ma, Y.; Wang, Z.; Cao, Y.; and et al. 2022. Prompt for Extraction? PAIE: Prompting Argument Interaction for Event Argument Extraction. In Muresan, S.; Nakov, P.; and Villavicencio, A., eds., *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1:*

Long Papers), 6759–6774. Dublin, Ireland: Association for Computational Linguistics.

Mesnard, T.; Hardin, C.; Dadashi, R.; and et al. 2024. Gemma: Open Models Based on Gemini Research and Technology. *arXiv preprint*, abs/2403.08295.

OpenAI. 2023. GPT-4 Technical Report. *arXiv preprint*, abs/2303.08774.

Pourpanah, F.; Abdar, M.; Luo, Y.; Zhou, X.; Wang, R.; Lim, C. P.; Wang, X.; and Wu, Q. M. J. 2023. A Review of Generalized Zero-Shot Learning Methods. *IEEE Trans. Pattern Anal. Mach. Intell.*, 45(4): 4051–4070.

Sainz, O.; Gonzalez-Dios, I.; Lopez de Lacalle, O.; Min, B.; and Agirre, E. 2022. Textual Entailment for Event Argument Extraction: Zero- and Few-Shot with Multi-Source Learning. In *Findings of the Association for Computational Linguistics: NAACL 2022*, 2439–2455.

Sharif, O.; Gatto, J.; Basak, M.; and Preum, S. M. 2025. REGen: A Reliable Evaluation Framework for Generative Event Argument Extraction. *arXiv preprint arXiv:2502.16838*.

Shuang, K.; Zhouji, Z.; Qiwei, W.; and Guo, J. 2024. Thinking about how to extract: Energizing LLMs' emergence capabilities for document-level event argument extraction. In *Findings of the Association for Computational Linguistics: ACL 2024*, 5520–5532.

Wang, S.; and Huang, L. 2024a. Debate as Optimization: Adaptive Conformal Prediction and Diverse Retrieval for Event Extraction. In Al-Onaizan, Y.; Bansal, M.; and Chen, Y.-N., eds., *Findings of the Association for Computational Linguistics: EMNLP 2024*, 16422–16435. Miami, Florida, USA: Association for Computational Linguistics.

Wang, S.; and Huang, L. 2024b. Targeted Augmentation for Low-Resource Event Extraction. In Duh, K.; Gómez-Adorno, H.; and Bethard, S., eds., *Findings of the Association for Computational Linguistics: NAACL 2024, Mexico City, Mexico, June 16-21, 2024*, 4414–4428. Association for Computational Linguistics.

Wang, S.; and Huang, L. 2024c. Targeted Augmentation for Low-Resource Event Extraction. In Duh, K.; Gomez, H.; and Bethard, S., eds., *Findings of the Association for Computational Linguistics: NAACL 2024*, 4414–4428. Mexico City, Mexico: Association for Computational Linguistics.

Wang, Z.; Zhao, Z.; Lyu, Y.; Chen, Z.; de Rijke, M.; and Ren, Z. 2025. A Cooperative Multi-Agent Framework for Zero-Shot Named Entity Recognition. In Long, G.; Blumestein, M.; Chang, Y.; Lewin-Eytan, L.; Huang, Z. H.; and Yom-Tov, E., eds., *Proceedings of the ACM on Web Conference 2025, WWW 2025, Sydney, NSW, Australia, 28 April 2025-2 May 2025*, 4183–4195. ACM.

Xu, R.; Wang, P.; Liu, T.; and et al. 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*, 5025–5036. Association for Computational Linguistics.

Xu, T.; Yang, H.; Zhao, F.; Wu, Z.; and Dai, X. 2024. A Two-Agent Game for Zero-shot Relation Triplet Extraction. In Ku, L.; Martins, A.; and Srikumar, V., eds., *Findings of the Association for Computational Linguistics, ACL 2024, Bangkok, Thailand and virtual meeting, August 11-16, 2024*, 7510–7527. Association for Computational Linguistics.

Yan, Z.; Wang, J.; Chen, J.; Li, X.; Liang, J.; Li, R.; and Pan, J. Z. 2025. Atomic fact decomposition helps attributed question answering. *IEEE Transactions on Knowledge and Data Engineering*.

Yang, A.; Yang, B.; Zhang, B.; and et al. 2024. Qwen2.5 Technical Report. *CoRR*, abs/2412.15115.

Zhang, G.; Zhang, H.; Wang, Y.; Li, R.; Tan, H.; and Liang, J. 2024a. Hyperspherical Multi-Prototype with Optimal Transport for Event Argument Extraction. In Ku, L.-W.; Martins, A.; and Srikumar, V., eds., *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 9271–9284. Bangkok, Thailand: Association for Computational Linguistics.

Zhang, H.; Wang, H.; and Roth, D. 2021. Zero-shot Label-Aware Event Trigger and Argument Classification. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, 1331–1340.

Zhang, X. F.; Blum, C.; Choji, T.; Shah, S.; and Vempala, A. 2024b. ULTRA: Unleash LLMs' Potential for Event Argument Extraction through Hierarchical Modeling and Pairwise Self-Refinement. In *Findings of the Association for Computational Linguistics: ACL 2024*, 8172–8185.

Zhao, Y.; Wang, Z.; Li, X.; Liang, J.; and Li, R. 2024. AGR: Reinforced Causal Agent-Guided Self-explaining Rationalization. In Ku, L.-W.; Martins, A.; and Srikumar, V., eds., *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*, 510–518. Bangkok, Thailand: Association for Computational Linguistics.

Zhao, Y.; Wang, Z.; Yu, X.; Li, X.; Liang, J.; and Li, R. 2025. Learnable Game-theoretic Policy Optimization for Data-centric Self-explanation Rationalization. *arXiv preprint arXiv:2510.13393*.

Zhou, H.; Qian, J.; Feng, Z.; Hui, L.; Zhu, Z.; and Mao, K. 2024a. LLMs Learn Task Heuristics from Demonstrations: A Heuristic-Driven Prompting Strategy for Document-Level Event Argument Extraction. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 11972–11990.

Zhou, S.; Meng, Y.; Jin, B.; and Han, J. 2024b. Grasping the Essentials: Tailoring Large Language Models for Zero-Shot Relation Extraction. In Al-Onaizan, Y.; Bansal, M.; and Chen, Y., eds., *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, EMNLP 2024, Miami, FL, USA, November 12-16, 2024*, 13462–13486. Association for Computational Linguistics.

Zhou, S.; Meng, Y.; Jin, B.; and Han, J. 2024c. Grasping the Essentials: Tailoring Large Language Models for Zero-Shot Relation Extraction. In Al-Onaizan, Y.; Bansal, M.; and Chen, Y.-N., eds., *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, 13462–13486. Miami, Florida, USA: Association for Computational Linguistics.