Targeted Augmentation for Low-Resource Event Extraction

Sijia Wang, Lifu Huang
Virginia Tech
{sijiawang,lifuh}@vt.edu

Abstract

Addressing the challenge of low-resource information extraction remains an ongoing issue due to the inherent information scarcity within limited training examples. Existing data augmentation methods, considered potential solutions, struggle to strike a balance between weak augmentation (e.g., synonym augmentation) and drastic augmentation (e.g., conditional generation without proper guidance). This paper introduces a novel paradigm that employs targeted augmentation and back validation to produce augmented examples with enhanced *diversity*, *polarity*, *accuracy*, and *coherence*. Extensive experimental results demonstrate the effectiveness of the proposed paradigm. Furthermore, identified limitations are discussed, shedding light on areas for future improvement[1].

1 Introduction

Event extraction (**EE**) (Grishman, 1997; Chinchor and Marsh, 1998; Ahn, 2006) is the task of identifying and categorizing event mentions in natural language text. While supervised methods deliver impressive performance, they depend heavily on extensive manual annotations (Chen et al., 2020; Du and Cardie, 2020; Lin et al., 2020; Liu et al., 2020; Li et al., 2020a; Lyu et al., 2021). Generalizing these approaches to low-resource learning setting poses challenges (Pasupat and Liang, 2014; Huang et al., 2016; Huang and Ji, 2020; Lai et al., 2020b; Shen et al., 2021b; Lyu et al., 2021; Zhang et al., 2021b; Wang et al., 2023b).

Data augmentation is one direction for efficiently addressing the low-resource event extraction problem. However, it's remained unexplored what data augmentation strategies are the best for low-resource event extraction given its unique challenges. Previous studies show that weak augmentations, such as synonym augmentation (Wei and Zou, 2019) or through back translation (Edunov et al., 2018), contribute minimally to distribution enrichment, while drastic augmentations can leading to misguided acquisitions (Cao et al., 2015; Gao et al., 2022). Drastic augmentations usually undermine existing event structure, resulting in grammatical incorrectness, structure misalignment, or semantic drifting (Wang et al., 2023a).

In this work, we explore several dimensions for data augmentation, including *diversity*, *polarity*, *accuracy*, and *coherence*. Our focus revolves around enhancing *diversity* in the context of targeted augmentation for low-resource event extraction (**TALOR-EE**). This involves enriching event structures with entities drawn from a targeted subset (Gao et al., 2022). Simultaneously, we address the issue of *polarity* by not only generating positive event mentions based on actual occurrences but also incorporating negative event mentions, e.g., hypothetical event mentions (Linguistic Data Consortium, 2005). This approach is particularly valuable for overcoming limitations in generative event extraction models (Hsu et al., 2022; Liu et al., 2022). To ensure both *accuracy* and *coherence* in our generated content, we introduce a back-and-forth validation module **BACK-VALIDATION**. The rationale behind this module is that an accurate generation should align with the given event structure, while coherent generation should seamlessly integrate with the same structure.

Our research encompasses a series of comprehensive experiments conducted across various low-resource learning scenarios, including zero-shot and few-shot learning settings. These experiments span different event extraction models. The outcomes of these experiments consistently highlight the effectiveness of targeted augmentation in low-resource event extraction. Notably, among all the dimensions investigated, diversity emerges as the most crucial factor. Additionally, we meticulously scrutinize the quality of the generated sentences,

[1]The source code, model checkpoints, and data are publicly available at https://github.com/VT-NLP/TALOR-EE.

Findings of the Association for Computational Linguistics: NAACL 2024, pages 4414–4428

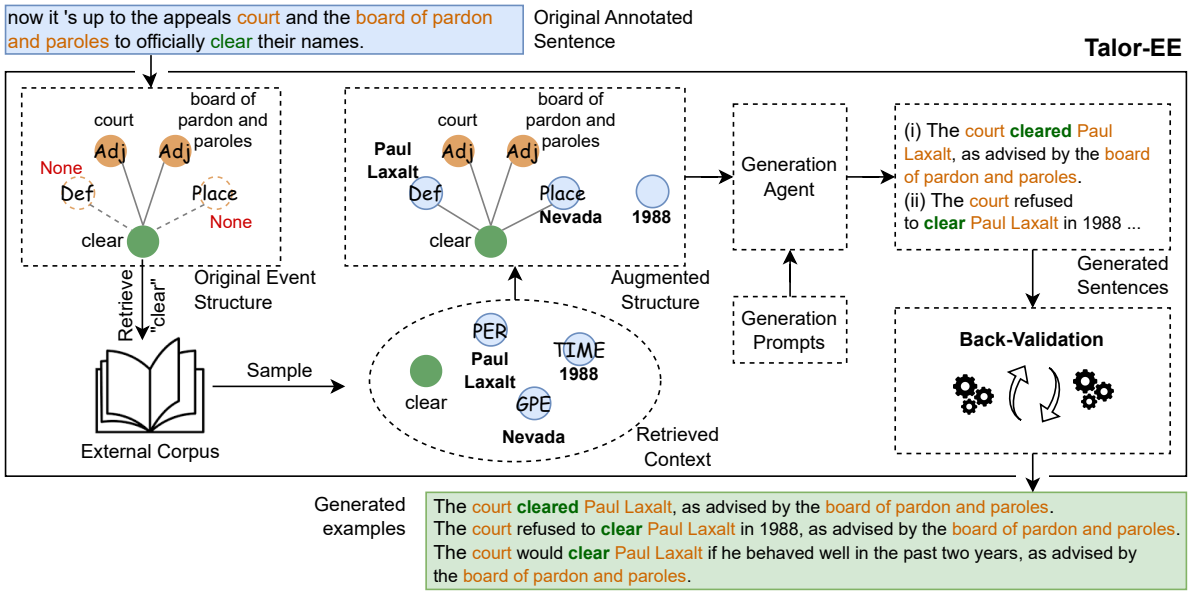

Figure 1: **TALOR-EE** framework overview.

shedding light on the limitations inherent in the proposed framework.

The contributions of this work are as follows:

- We explore the application of data augmentation techniques for low-resource event extraction.
- We develop a novel augmentation method that incorporates enriched event structures and contextual entities, retrieved from external corpus. The generated examples are validated through a back-validation module, ensuring accuracy and coherence.
- Comprehensive experiments are conducted to assess the effectiveness of the proposed paradigm across various models and datasets.

2 Related Work

Low-resource Event Extraction Although some studies have employed meta-learning (Kang et al., 2019; Li et al., 2021; Xiao and Marlet, 2020; Yan et al., 2019; Chowdhury et al., 2021), or metric learning (Sun et al., 2021; Wang et al., 2020a; Zhang et al., 2021a; Agarwal et al., 2021) to align candidate event semantics with a few examples of novel event types for few-shot event detection, their performance is inherently constrained by the limited examples provided (Lai et al., 2020a; Deng et al., 2020; Lai et al., 2020b; Cong et al., 2021; Chen et al., 2021; Shen et al., 2021b). Recent studies (Wei et al., 2023; Han et al., 2023; Li et al., 2023) have explored in-context learning by providing task instructions and a handful of in-context examples. Nevertheless, their experimental findings reveal a notable performance gap between in-context learning and approaches based on fine-tuning.

Data Augmentation creates synthetic data from the existing data. Traditional data augmentation approaches focus on expanding lexical diversity (Wei and Zou, 2019; Feng et al., 2020; Ng et al., 2020) or syntax variation (Kim et al., 2022; Loem et al., 2022; Hussein et al., 2022; Wang et al., 2023a). Post selection (Yang et al., 2020) or representative selection (Edwards et al., 2021) helps to prevent a waste of resources and time in generating new documents. Yet existing augmentation methods suffer from gradual drift problem (Hu et al., 2021a,b). The previous work (Ma et al., 2023) utilizes language models for training data synthesis but lacks assurance in the soundness and naturalness of event structures due to the random combination of sampled triggers and arguments. Additionally, it falls short by primarily relying on the self-reflection capability of language models, without fully leveraging annotations for existing event annotations. Thus, in addition to the lexical and syntactical diversity, we leverage the large-scale pre-trained autoregressive models to generate contextually diversified free texts.

Controlled Text Generation approaches (Ghosh et al., 2021) generate text with specific constraint.

Approaches that promote similarity (Guan et al., 2021) or coherence (Shen et al., 2021a; Wang et al., 2021a) towards the original sentences lack contextual diversity and might produce over-confident probability estimation (Wang et al., 2021a; Gowda and May, 2020). Rule-based constraint generation might generate meaningless tokens to meet constraints (Wang et al., 2021b), while template-based constraint generation (Cao and Wang, 2021) is difficult to generalize to new domains without human effort.

Learning with noisy labels Many works learn with noisy labels by detecting corrupted instances, e.g., (Han et al., 2018; Yu et al., 2019; Huang et al., 2019; Yao et al., 2020; Wei et al., 2020; Jiang et al., 2020; Zhang et al., 2021c), and their application to low-resource learning setting (Wang et al., 2020b; Li et al., 2020b; Cheng et al., 2021). However, joint training of the sample selection module and the target task model takes considerable iterations to converge. Traditional data-centric methods (Zhu et al., 2022) face limitations in low-resource settings due to biased neighbor information. This study demonstrates that training with relatively fair-quality labels can be effective.

3 Model

3.1 Problem Formulation

Given a sentence, the Event Extraction (EE) task aims to extract event mentions, represented by an event trigger and a set of event arguments. Formally, given a sentence $w = \{w_1, ..., w_n\}$, and a target event type e_i, if there is an event occurrence of e_i in w, a EE system aims to extract an event trigger t and its argument mentions $a = \{a_1, ..., a_g\}$. In this work, we focus on zero-shot and few-shot learning settings of EE. For few-shot EE (FSEE), training data contains two parts: (1) A large-scale data set $\mathcal{D}_{base} = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{M}$ that covers the seen event types (named *base types*), where M denotes the number of base event types; (2) a smaller data set $\mathcal{D}_{novel} = \{(\mathbf{x}_j, \mathbf{y}_j)\}_{j=1}^{N\times K}$ that covers N novel event types, with K examples each. Note that the base and novel event types are disjoint except for the `Other` class, indicating non-event type. In zero-shot event extraction (ZSEE), the training data set only contains a large-scale set $\mathcal{D}_{base} = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{M}$ for the base event types. The model f will be optimized on base event types and evaluated on the novel types. Following previous work, we set $N = 5, 10$ and $K = 0, 1, 5, 10$ in this work.

3.2 Targeted Augmentation [Diversity]

In contrast to previous data augmentation approaches (Wei and Zou, 2019; Feng et al., 2020; Ng et al., 2020; Kim et al., 2022; Loem et al., 2022; Hussein et al., 2022; Wang et al., 2023a), we have improved upon the conventional conditional generation method by transitioning from random sampling to a targeted selection strategy. The targeted augmentation module serves as a mechanism to ensure diversity. Theoretically, it can retrieve an infinite number of entities from the external corpus, seamlessly incorporating these entities into the given event structure. Consequently, the module can generate an infinite variety of new event structures. Thus, the targeted augmentation provides a theoretical framework for sampling and augmenting an extensive array of entities, particularly beneficial when working with a limited set of annotated event mentions.

Dependent Context Retrieval For a given event structure, we retrieve context candidates from the corpus that share tokens with the event structure. In our experiments, we gathered sentences containing the mention of the event trigger. To extract context information from the sampled sentences, we utilized the spaCy Named Entity Recognition (NER) parser[2] to identify entity mentions. Consequently, the extracted entity mentions from each sampled sentence serve as context candidates for the given event structure. The context corpus employed in this study is the NYT Annotated Corpus[3].

Targeted Generation Given an event structure $e_i = \{t_i, a_1, ..., a_p\}$ and a sampled context candidate $c = \{c_1, ..., c_q\}$, a generator is leveraged to generate a corresponding sentence. If the sampled context entities could potentially serve as argument roles in the original event structures, we employ an add-or-replace strategy, to further tailor the event structure. The feasibility of integrating an entity into the event structure depends on its entity type. If the argument role is vacant in the original structure, and the entity type of the sampled entity aligns with the argument role, we add the entity to the event

[2] https://spacy.io/usage/linguistic-features

[3] https://catalog.ldc.upenn.edu/LDC2008T19

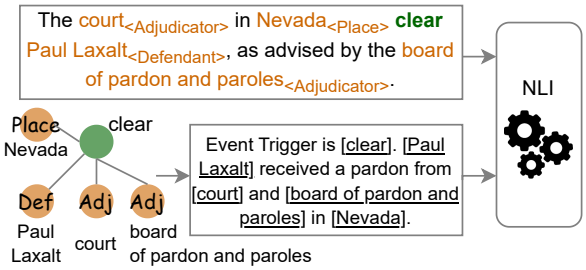

Figure 2: Event mention accuracy verification module.

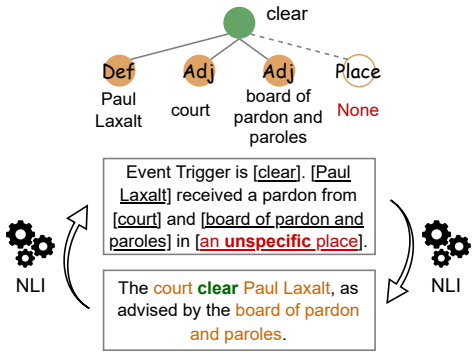

Figure 3: Event mention coherence verification module.

structure. If the argument role is already populated, we substitute it with the sampled entity.

For example, given an annotation on the sentence `"now it 's up to the appeals court and the board of pardon and paroles to officially clear their names."`, a *Justice:Pardon* event is represented by the event structure *{Trigger: clear, Adjudicator: court, Adjudicator: board of pardon and paroles}*. A complete *Justice:Pardon* structure may also include two argument roles, namely *Defendant* and *Place*. From the sampled context entities `[Paul Laxalt, 1988, Nevada]`, `Nevada` is added to the event structure as an *Place* role, and `Paul Laxalt` is added as a *Defendant* role. Note that `"Nevada"` is added because it is a GPE entity and a GPE entity is one of the possible entity types for a *Place* role. Similarly, `Paul Laxalt` is added as a *Defendant* because it is a PER entity. Here we present a generated sentence with the enriched event structure: `"The court in Nevada clear Paul Laxalt, as advised by the board of pardon and paroles."` The process is illustrated in Figure 1.

3.3 Negative Augmentation [Polarity]

Polarity is maintained through the negative augmentation design. This process generates not only positive event mentions but also negative mentions, including hypothetical mentions and believed event mentions. For event extraction, we focus on identifying event that occurs, and also negative mentions. For example, in the sentence "*John Hinkley **denied** his attempt to **assassinate** Ronald Reagan.*", a model, especially generative models, might overlook this Conflict:Attack mention triggered by the token ***assassinate***, because this is not an actual event that happens. More specifically, negative event mentions include (1) explicit negative mentions: expressed with a negative word such as *not* or *never*, or a negative lexical context such *deny*, *refuse* or *disobey*, (2) asserted mentions: including hypothetical events, believed events, or promised events, etc (Linguistic Data Consortium, 2005).

Thus in addition to augmenting high-quality positive training examples, particular attention is paid to augmenting negative training examples. In this work, we write negative/asserted expression prompts to guide their generation. Prompts and generated negative augmentation examples are listed in Table 6 and Table 7 in Appendix B.

3.4 Back-Validation

Given noisy training examples, previous research has utilized methods to detect and rectify corrupted data during training (Han et al., 2018; Yu et al., 2019; Huang et al., 2019; Yao et al., 2020; Wei et al., 2020; Jiang et al., 2020; Zhang et al., 2021c), but such approaches necessitate extensive training. In our context, where the generated data is considered of reasonable quality, we propose the incorporation of a back-and-forth validation module. This module aims to ensure the *accuracy* and *coherence* of the generated content, thereby enhancing the reliability of the augmented examples.

Event Mention Accuracy Verification [Accuracy] For each generated example, its accuracy can be verified through an entailment verification module. As shown in Figure 2, given the generated sentence and its source event structure, we first textualize the event structure into a passage to express the event structure, by a pre-defined template (Hsu et al., 2022). Then the two texts will be passed into an NLI entailment verification module. The intuition is that, for a valid generation, it should entail the template passage with the event structure.

Event Mention Coherence Verification [Coherence] In addition to ensuring generation accuracy, we aim for the generated sentence to exhibit strong coherence with the provided event structure. Specifically, there should be no extraneous or omitted arguments when compared to the given event structure. The intuition is that if the generated sentence aligns coherently with the provided event structure, a template passage incorporating the event structure should entail the generated sentence, and vice versa. A distinctive scenario arises when the event structure is incomplete. In such instances, we adapt the missing argument role in the template with the expression "an unspecific [argument role]." Illustrated in Figure 3, if the *Place* argument role is absent, we want to ensure that the generated event mention does not introduce an extraneous arbitrary Place argument role. Consequently, we substitute "[Place]" with "[an unspecific Place]." This modification ensures that the generated sentence fails the forward-and-backward entailment test in such scenarios.

3.5 Generative Event Extraction Model

DEGREE (Hsu et al., 2022) is a generative event extraction model that conceptualizes event extraction as a conditional generation problem. Given a sentence and a crafted prompt, DEGREE generates an output following a specified format. The predictions for event triggers and argument roles can be then parsed from the generated output using a deterministic algorithm. In contrast to earlier classification-based models, the generation framework offers a versatile approach to incorporate supplementary information and guidance. Through the creation of suitable prompts, DEGREE can better capture the dependencies between entities and, consequently diminish the requisite number of training examples.

The EE template defines the anticipated output format and is organized into two main parts. The initial segment is referred to as the trigger template, structured as "Event trigger is <Trigger>", with "<Trigger>" acting as a placeholder for event trigger in the original passage. The subsequent section is the argument template, and its composition varies based on the specific event type. For instance, the argument template for a Conflict:Attack event is "some people or some organization in somewhere was ordered by some adjudicator to pay a fine." Each underlined string, beginning with "some-," serves as a placeholder corresponding to an argument role for a Justice:Fine event. For example, "somewhere" corresponds to the Place where the event occurs. Note that every event type has its own argument template. Event extraction templates and the construction details can be found in (Hsu et al., 2022).

Algorithm 1 Robust Fine-tuning

Input: Base data set $\mathcal{D}_{base}$; few shot training set $\mathcal{D}_{novel}$; synthesized training set $\mathcal{D}_{gen}$.
Output: Model M, validator V

fine-tune V with back-validation data constructed from $\mathcal{D}_{train}$
pass $\mathcal{D}_{gen}$ into V, collect $\mathcal{D}'_{gen}$ that pass back-validation
for each epoch t **do**
  Sample meta batch D^t_{base} from $\mathcal{D}_{base}$
  Sample noisy batch D^t_{gen} from $\mathcal{D}'_{gen}$
  Update model M with D^t_{train}, $\mathcal{D}_{novel}$, and D^t_{gen}
  Discard corrupted data by semantic distance to the center instances
end for

Model	Time/Sentence(s)	Cost/Sentence($)
Vicuna-7B	2.7	0
LLaMA2-7B	8.7	0
GPT-3.5-turbo	2.4	∼0.0035

Table 1: Augmentation cost per sentence.

3.6 Robust Fine-tuning

Given the synthesized training samples $\mathcal{D}_{gen}$ that augment $\mathcal{D}_{train}$ for fine-tuning a classification M. The primary concern is the presence of label noise, where some generated samples may inaccurately align with their corresponding labels, potentially degrading model performance when using standard supervised learning. To address this challenge, we employ a noise-robust training procedure to enhance stability. We first fine-tune the back-validator V with the training data constructed from the base dataset. For negative examples, we construct two datasets: (1) sample unpaired event structures and sentences within the corpus and (2) replace argument roles in the template with "an unspecific [argument role]". Then we validate the augmented examples with the fine-tuned validator V, and validated examples are then used for fine-tuning the EE model M. Finally, we employ a random sample selection on the base data set $\mathcal{D}_{base}$ and the synthesized training set $\mathcal{D}_{gen}$, along with the entire few shot training set $\mathcal{D}_{novel}$ to update the EE model M. The algorithm is shown in Algorithm 1.

Method	K-shot	Common 5				Common 10			
		Tri-I	Tri-C	Arg-I	Arg-C	Tri-I	Tri-C	Arg-I	Arg-C
Matching Baseline	full	42.7	42.1	-	-	46.3	46.3	-	-
Lemmatization Baseline	full	51.5	50.2	-	-	56.0	56.0	-	-
OneIE	full	72.7	70.5	52.3	49.9	74.5	73.0	51.2	48.9
DEGREE	full	68.4	66.0	51.9	48.7	72.0	69.8	52.5	49.2
BERT_QA	1-shot	10.0	1.4	1.3	1.3	8.2	1.6	1.1	1.1
	5-shot	14.0	12.6	11.1	10.8	20.8	15.4	14.6	13.9
	10-shot	37.8	11.3	22.9	22.1	32.0	27.8	19.5	18.6
OneIE	1-shot	4.2	4.2	1.5	1.5	4.1	2.7	2.0	2.0
	5-shot	39.3	38.5	24.8	22.8	41.9	41.9	29.7	27.2
	10-shot	54.8	53.3	36.0	34.9	61.5	57.8	41.4	39.2
DEGREE	0-shot	53.3	46.8	29.6	25.1	60.9	54.5	42.0	31.4
	1-shot	60.1	53.3	38.8	31.6	61.2	60.9	41.1	34.7
	5-shot	57.8	55.5	40.6	36.1	65.8	64.8	45.3	42.7
	10-shot	63.8	61.2	46.0	42.0	72.1	68.8	**52.5**	48.4
TALOR-EE (Vicuna)	0-shot	66.1	62.3	38.7	32.9	71.6	68.7	40.7	35.9
	1-shot	63.5	55.7	37.5	32.0	69.2	64.5	**47.8**	**43.2**
	5-shot	67.0	**65.2**	**46.6**	**43.1**	72.7	70.0	50.1	44.9
	10-shot	**70.4**	66.2	**46.4**	42.7	73.9	**71.7**	49.2	44.9
TALOR-EE (LLaMA)	0-shot	65.0	62.5	41.0	36.5	65.6	64.8	**47.5**	**43.8**
	1-shot	66.5	61.0	**42.3**	34.4	71.5	66.7	45.4	42.4
	5-shot	**70.2**	63.9	46.3	42.4	71.7	70.1	**50.5**	**46.7**
	10-shot	70.0	**67.6**	46.2	**43.3**	70.5	70.2	51.2	**49.5**
TALOR-EE (GPT)	0shot	**67.9**	**66.1**	**46.1**	**40.0**	**72.5**	**70.3**	46.9	42.8
	1-shot	**68.5**	**64.8**	42.1	**35.6**	**72.5**	**68.1**	46.5	42.8
	5-shot	67.9	64.2	44.6	42.6	**73.6**	**70.6**	48.5	44.7
	10-shot	70.2	67.4	43.0	41.4	**74.2**	70.5	48.3	47.7

Table 2: Low-resource EE results on ACE05-E. Bold represents the highest score for the current setting.

4 Experiments

We perform experiments on three public benchmark datasets, including ACE05-E (Automatic Content Extraction)[4] and ERE (Entity Relation Event) (Song et al., 2015). To showcase the effectiveness of the proposed method under low resource settings, experiments are conducted under Nway-Kshot learning setting, where $N \in \{5, 10\}$, and $K \in \{0, 1, 5, 10\}$.

Compared baselines We consider the following baselines: (1) Matching baseline[5], a proposed baseline that makes trigger predictions by performing string matching between the input passage and the event keywords. (2) Lemmatization baseline, another proposed baseline that performs string matching on lemmatized input passage and the event keywords. (3) BERT_QA(Du and Cardie, 2020), (4) OneIE (Lin et al., 2020), (5) DEGREE (Hsu et al., 2022) and (6) QueryExtract (Wang et al., 2022). The implementation details can be found in Appendix A.

Generation Agents Three generation agents are experimented in this work, including `vicuna-7b-v1.3` (Vicuna), `Llama-2-7b` (LLaMA), and `gpt-3.5-turbo` (GPT). For each agent, we list the augmentation cost in Table 1, where two factors are listed including generation time and cost per sentence.

4.1 Main results

The experimental results for low-resource Event Extraction (EE) are presented in Table 2 and Figure 4 for ACE05-E, and Table 3 and Figure 5 for ERE, respectively. From the experiment results, several conclusions can be drawn: (1) With the augmented examples, the performance of low-resource EE generally exhibits improvement, evident in both zero-shot learning and few-shot learning settings. This improvement is consistent across different generation agents (Vicuna, LLaMA, and GPT) and backbone EE models. Table 8 displays experimental results on ACE05-E with QueryExtract as the

[4] https://catalog.ldc.upenn.edu/LDC2006T06

[5] (1) and (2) are baselines for event detection tasks, thus only trigger detection results are reported.

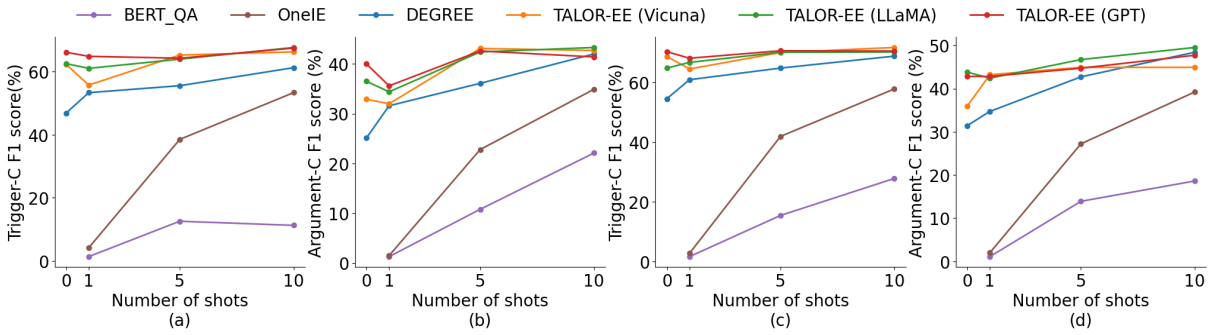

Figure 4: Experimental results on ACE05-E. (a-b) are visualizations for Common 5, and (c-d) for Common 10.

Method	**K-shot**	Common 5				Common 10			
		Tri-I	Tri-C	Arg-I	Arg-C	Tri-I	Tri-C	Arg-I	Arg-C
DEGREE	full	54.7	53.1	45.4	44.7	58.8	58.2	51.3	50.8
DEGREE	0-shot	32.2	26.8	16.1	15.5	47.7	45.4	28.7	28.0
	1-shot	34.4	33.8	28.0	26.2	39.4	39.4	30.7	29.9
	5-shot	44.8	39.2	28.9	28.7	56.3	55.5	44.5	42.7
	10-shot	48.4	45.8	39.3	38.8	59.3	57.8	48.4	47.8
TALOR-EE (Vicuna)	0-shot	41.9	40.2	31.0	28.9	50.6	49.0	37.9	36.6
	1-shot	48.5	38.7	31.3	30.4	47.8	41.6	35.9	34.8
	5-shot	45.8	43.0	35.8	33.4	56.2	53.7	42.5	41.0
	10-shot	55.7	52.0	40.6	37.6	58.2	56.7	47.8	44.9
TALOR-EE (LLaMA)	0-shot	40.8	34.7	26.2	23.8	51.6	45.4	37.8	36.4
	1-shot	47.4	39.1	33.4	**33.2**	47.3	**44.4**	**46.2**	**44.6**
	5-shot	48.9	44.5	37.7	34.8	55.3	54.6	48.5	**47.8**
	10-shot	**58.1**	**55.7**	**45.5**	42.5	58.2	57.5	52.2	48.4
TALOR-EE (GPT)	0-shot	**49.3**	**41.9**	**34.0**	**32.4**	**57.1**	**55.8**	**43.1**	**40.8**
	1-shot	**50.3**	**42.0**	**34.5**	32.1	**51.6**	44.3	43.7	42.1
	5-shot	**52.9**	**48.2**	**39.1**	**37.3**	**57.5**	**56.0**	**49.4**	45.5
	10-shot	56.9	54.6	43.5	**43.0**	**62.4**	**61.7**	**53.4**	**49.6**

Table 3: Low-resource EE results on ERE. Bold represents the highest score for the current setting.

backbone model, highlighting the effectiveness of augmented training examples across various EE models. (2) The observed improvement is more pronounced in extremely low-resource scenarios, particularly in zero-shot, 1-shot, and 5-shot scenarios. The impact is less significant when more clean training examples are available, such as in the 10-shot setting. (3) We observe that the performance of zero-shot augmented training can surpass that of 1-shot training with clean examples. This discrepancy arises because some sampled clean training examples may not straightforwardly express event information. For instance, the token "open" could trigger a *Start-Organization* event, introducing confusion in the semantics of the *Start-Organization* event type. (4) Augmented examples generated by different generation agents consistently enhance low-resource EE performance. Notably, greater performance gains are achieved with examples generated by LLaMA and GPT.

Additionally, we have evaluated the generation quality and the effectiveness of the proposed modules. Notably, for diversity, there is a substantial increase in unique argument roles compared to the few-shot examples. For example, in the common 10 and 5-shot settings, the count of unique argument roles surged from 142 to 1184, marking a remarkable increase of 2502 percentage points, on average across the generation models. Regarding polarity, among the 30 sampled augmentations verified through human evaluation, the generated event mention expressions consistently align with the targeted negative expression types. In terms of back-validation, the evaluation involved two annotators who each assessed 200 randomly sampled generations (100 for with back-validation generations and 100 for generations without back-validation). On average, seven generations were deemed not fluent

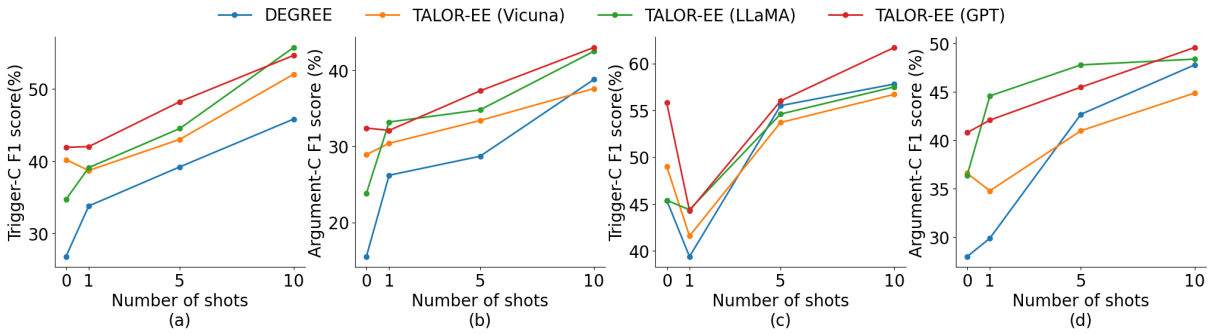

Figure 5: Experimental results on ERE. (a-b) are visualizations for Common 5, and (c-d) for Common 10.

Method	K-shot	Common 5				Common 10			
		Tri-I	Tri-C	Arg-I	Arg-C	Tri-I	Tri-C	Arg-I	Arg-C
TALOR-EE (LLaMA)	1-shot	66.5	61.0	42.3	34.4	71.5	66.7	45.4	42.4
	5-shot	70.2	63.9	46.3	42.4	71.7	70.1	50.5	46.7
	10-shot	70.0	67.6	46.2	43.3	70.5	70.2	51.2	49.5
- enriched context	1-shot	61.2	52.1	35.9	28.3	72.9	64.6	46.2	40.6
	5-shot	68.5	64.2	43.5	41.1	73.2	70.0	45.7	44.6
	10-shot	67.0	63.4	43.1	39.5	74.7	71.7	46.4	43.2
- negative augmentations	1-shot	70.5	65.1	41.8	34.4	74.1	67.4	44.4	38.8
	5-shot	69.3	62.6	41.8	39.3	77.4	73.4	48.4	42.8
	10-shot	69.1	61.3	40.8	39.6	74.1	70.5	46.6	44.3
- back-validation	1-shot	61.2	52.1	35.9	28.3	72.7	66.0	47.3	42.2
	5-shot	68.0	62.8	43.1	38.6	76.1	74.6	48.6	44.4
	10-shot	67.2	65.2	42.1	40.2	75.3	71.2	47.3	46.7

Table 4: Ablation study on ACE05-E.

when utilizing the back-validation module, while 19 generations were identified as not fluent without the back-validation module.

4.2 Ablation Studies

An ablation study was conducted to assess the effectiveness of each proposed module, and the experimental results are presented in Table 4. (Omitting the enriched context in the setting entails bypassing the Dependent Context Retrieval module, resulting in the absence of newly generated event structures.) On average, across all settings, the performance of trigger classification decreased by 2.5% and 1.9%, and argument classification decreased by 8.3% and 7.1%, in the absence of enriched context or back-validation, respectively. Without negative augmentations, the argument classification decreases by 7.5%, while trigger classification performance is on par with **TALOR-EE** (LLaMA). This highlights that the designed modules have a more pronounced impact on argumentation classification than on trigger detection. The absence of enriched context led to the most significant decrease in argument classification performance, emphasizing the crucial role of augmentation diversity in mitigating low-resource argument extraction.

4.3 Error Analysis

Table 5 illustrates several challenging examples. For event trigger detection, most of the errors are from the insufficient understanding of the trigger phrase. For example in example (a) in Table 5, linking the phrase "crumbling" to the *End-Org(anization)* event is challenging given the limited trigger training examples from either clean data or augmented data. Example (b) is challenging because the token "combination" entails closer semantic relation to the *Merge-Org* event. Example (c) illustrates a case where the current data augmentation model falls short in generating intricate event expressions. Example (d) illustrates a scenario in which the use of augmented data could potentially cause confusion. In this case, the actual event pertains to a film release rather than a judicial release. Despite inadequate context information, there is a likelihood that the augmented data might have generated a false prediction with increased confidence. One potential solution to this challenge is the abil-

ID	Text	GTH	Predictions
(a)	Hoon said Saddam 's regime was **crumbling** under the pressure of a huge air assault .	crumbling; End-Org; regime: Org;	None
(b)	The **combination** of the banking operations of Barclays Spain and Zaragozano will bring together two complementary businesses.	combination; Transfer-Ownership; Barclays Spain: Buyer; Zaragozano: Artifact;	combination, Merge-Org; businesses, Org
(c)	**Married** for the second time , Hariri has five children.	Married, Marry; Hariri: Person;	None
(d)	However the firm announced on Friday that it had reached a deal with the British arm of French distributors Pathe to show four **releases**.	None	releases; Release-Parole; firm: Entity;

Table 5: Case study for challenging examples

ity to distinguish between multiple meanings of the same word.

In contrast to event trigger detection, argument extraction presents greater challenges, as improvements in argument extraction prove less pronounced than those in trigger detection. Our conclusion stems from a meticulous analysis of the generated outputs and prediction results, revealing two primary reasons. The first reason is the lack of clear and comprehensive explanations for certain argument roles, for example, the argument role "agent" in the *Start-Org* event type. According to the definition (Linguistic Data Consortium, 2005), an "agent" in a *Start-Org* event is a "PER", "ORG", or "GPE" entity responsible for the "START-ORG" Event. However, it requires tremendous expert knowledge to write precise instructions for argument roles like this. The second reason pertains to the lack of clear distinctions among argument roles in generation prompts. We recognize that elucidating the purpose and differentiation of each argument role can be intricate. For instance, we observed minimal or even adverse effects of augmented data on the event type "Transfer-Ownership". This complexity arises from the potential confusion surrounding three specific argument roles: "Beneficiary", "Buyer", and "Seller", particularly when the trigger involves terms like "sell" or "acquire". Notably, altering the trigger from "sell" to "acquire" induces a substantial change in the sentence's entire syntactic structure.

5 Conclusion

In conclusion, this study proposes a new paradigm for tackling low-resource event extraction tasks. Generation agents are employed to create a diverse training dataset for event structures enriched with domain-invariant entities. The generated examples undergo a thorough back-and-forth validation process to assess accuracy and coherence. Our research encompasses extensive experiments in diverse low-resource learning scenarios, such as zero-shot and few-shot learning settings, across various event extraction models. The outcomes of these experiments highlight the effectiveness of the proposed framework. Furthermore, our proposed methodology can inspire researchers from diverse domains to embrace a comparable paradigm or delve into the investigation of data augmentation methods as a means of enriching their training datasets.

Limitations

TALOR-EE establishes a powerful starting point for advancing few-shot learning research, offering a flexible framework for framing new tasks through our proposed augmentation method. It encourages a systematic exploration of general and resilient enhancements for low-resource event extraction systems. However, augmenting non-event examples takes appropriate attention, as the proposed system may tend to predict additional event mentions. The absence of a clear distinction between an actual event and a non-event mention, due to the lack of a precise definition, underscores the need for appropriate action. We extend a warm invitation to future low-resource research endeavors and augmentation methods to delve into the structural aspects of event generation within a contrastive setting.

Acknowledgements

This research is supported by the award No. 2238940 from the Faculty Early Career Development Program (CAREER) of the National Science Foundation (NSF). The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright annotation therein.

References

Ashutosh Agarwal, Anay Majee, Anbumani Subramanian, and Chetan Arora. 2021. Attention guided cosine margin for overcoming class-imbalance in few-shot road object detection.

David Ahn. 2006. The stages of event extraction. In *Proceedings of the Workshop on Annotating and Reasoning about Time and Events*, pages 1–8.

Kai Cao, Xiang Li, Miao Fan, and Ralph Grishman. 2015. Improving event detection with active learning. In *Proceedings of the International Conference Recent Advances in Natural Language Processing*, pages 72–77, Hissar, Bulgaria. INCOMA Ltd. Shoumen, BULGARIA.

Shuyang Cao and Lu Wang. 2021. Controllable open-ended question generation with a new question type ontology.

Jiawei Chen, Hongyu Lin, Xianpei Han, and Le Sun. 2021. Honey or poison? solving the trigger curse in few-shot event detection via causal intervention.

Yunmo Chen, Tongfei Chen, Seth Ebner, Aaron Steven White, and Benjamin Van Durme. 2020. Reading the manual: Event extraction as definition comprehension. In *Proceedings of the Fourth Workshop on Structured Prediction for NLP*, Online. Association for Computational Linguistics.

Hao Cheng, Zhaowei Zhu, Xingyu Li, Yifei Gong, Xing Sun, and Yang Liu. 2021. Learning with instance-dependent label noise: A sample sieve approach. In *International Conference on Learning Representations*.

Nancy Chinchor and Elaine Marsh. 1998. Muc-7 information extraction task definition. In *Proceeding of the seventh message understanding conference (MUC-7), Appendices*, pages 359–367.

Arkabandhu Chowdhury, Mingchao Jiang, and Chris Jermaine. 2021. Few-shot image classification: Just use a library of pre-trained feature extractors and a simple classifier.

Xin Cong, Shiyao Cui, Bowen Yu, Tingwen Liu, Yubin Wang, and Bin Wang. 2021. Few-shot event detection with prototypical amortized conditional random field. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP*.

Shumin Deng, Ningyu Zhang, Jiaojian Kang, Yichi Zhang, Wei Zhang, and Huajun Chen. 2020. Meta-learning with dynamic-memory-based prototypical network for few-shot event detection. *Proceedings of the 13th International Conference on Web Search and Data Mining*.

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, Online. Association for Computational Linguistics.

Sergey Edunov, Myle Ott, Michael Auli, and David Grangier. 2018. Understanding back-translation at scale. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 489–500, Brussels, Belgium. Association for Computational Linguistics.

Aleksandra Edwards, Asahi Ushio, Jose Camacho-Collados, Hélène de Ribaupierre, and Alun Preece. 2021. Guiding generative language models for data augmentation in few-shot text classification.

Steven Y. Feng, Varun Gangal, Dongyeop Kang, Teruko Mitamura, and Eduard Hovy. 2020. GenAug: Data augmentation for finetuning text generators. In *Proceedings of Deep Learning Inside Out (DeeLIO): The First Workshop on Knowledge Extraction and Integration for Deep Learning Architectures*, pages 29–42, Online. Association for Computational Linguistics.

Irena Gao, Shiori Sagawa, Pang Wei Koh, Tatsunori Hashimoto, and Percy Liang. 2022. Out-of-distribution robustness via targeted augmentations. In *NeurIPS 2022 Workshop on Distribution Shifts: Connecting Methods and Applications*.

Sayan Ghosh, Zheng Qi, Snigdha Chaturvedi, and Shashank Srivastava. 2021. How helpful is inverse reinforcement learning for table-to-text generation? In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 71–79, Online. Association for Computational Linguistics.

Thamme Gowda and Jonathan May. 2020. Finding the optimal vocabulary size for neural machine translation. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3955–3964, Online. Association for Computational Linguistics.

Ralph Grishman. 1997. Information extraction: Techniques and challenges. In *International summer school on information extraction*, pages 10–27. Springer.

Jian Guan, Xiao-Xi Mao, Changjie Fan, Zitao Liu, Wenbiao Ding, and Minlie Huang. 2021. Long text generation by modeling sentence-level and discourse-level coherence. In *ACL*.

Bo Han, Quanming Yao, Xingrui Yu, Gang Niu, Miao Xu, Weihua Hu, Ivor W. Tsang, and Masashi Sugiyama. 2018. Co-teaching: Robust training of deep neural networks with extremely noisy labels. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS'18, page 8536–8546, Red Hook, NY, USA. Curran Associates Inc.

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.

I-Hung Hsu, Kuan-Hao Huang, Elizabeth Boschee, Scott Miller, Prem Natarajan, Kai-Wei Chang, and Nanyun Peng. 2022. Degree: A data-efficient generative event extraction model. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*.

Xuming Hu, Chenwei Zhang, Fukun Ma, Chenyao Liu, Lijie Wen, and Philip S. Yu. 2021a. Semi-supervised relation extraction via incremental meta self-training. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Xuming Hu, Chenwei Zhang, Yawen Yang, Xiaohe Li, Li Lin, Lijie Wen, and Philip S. Yu. 2021b. Gradient imitation reinforcement learning for low resource relation extraction. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Jinchi Huang, Lie Qu, Rongfei Jia, and Binqiang Zhao. 2019. O2u-net: A simple noisy label detection approach for deep neural networks. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3325–3333.

Lifu Huang, Taylor Cassidy, Xiaocheng Feng, Heng Ji, Clare Voss, Jiawei Han, and Avirup Sil. 2016. Liberal event extraction and event schema induction. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 258–268.

Lifu Huang and Heng Ji. 2020. Semi-supervised new event type induction and event detection. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 718–724.

Amir Hussein, Shammur Absar Chowdhury, Ahmed Abdelali, Najim Dehak, and Ahmed Ali. 2022. Code-switching text augmentation for multilingual speech processing.

Lu Jiang, Di Huang, Mason Liu, and Weilong Yang. 2020. Beyond synthetic noise: Deep learning on controlled noisy labels. In *ICML*.

Bingyi Kang, Zhuang Liu, Xin Wang, Fisher Yu, Jiashi Feng, and Trevor Darrell. 2019. Few-shot object detection via feature reweighting. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 8419–8428.

Hazel Kim, Daecheol Woo, Seong Joon Oh, Jeong-Won Cha, and Yo-Sub Han. 2022. Alp: Data augmentation using lexicalized pcfgs for few-shot text classification. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36:10894–10902.

Viet Dac Lai, Franck Dernoncourt, and Thien Huu Nguyen. 2020a. Exploiting the matching information in the support set for few shot event classification. *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, page 233–245.

Viet Dac Lai, Thien Huu Nguyen, and Franck Dernoncourt. 2020b. Extensively matching for few-shot learning event detection. In *Proceedings of the First Joint Workshop on Narrative Understanding, Storylines, and Events*, pages 38–45, Online. Association for Computational Linguistics.

Bo Li, Gexiang Fang, Yang Yang, Quansen Wang, Wei Ye, Wen Zhao, and Shikun Zhang. 2023. Evaluating chatgpt's information extraction capabilities: An assessment of performance, explainability, calibration, and faithfulness.

Bohao Li, Boyu Yang, Chang Liu, Feng Liu, Rongrong Ji, and Qixiang Ye. 2021. Beyond max-margin: Class margin equilibrium for few-shot object detection. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7359–7368.

Fayuan Li, Weihua Peng, Yuguang Chen, Quan Wang, Lu Pan, Yajuan Lyu, and Yong Zhu. 2020a. Event extraction as multi-turn question answering. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 829–838, Online. Association for Computational Linguistics.

Junnan Li, Richard Socher, and Steven C.H. Hoi. 2020b. Dividemix: Learning with noisy labels as semi-supervised learning. In *International Conference on Learning Representations*.

Ying Lin, Heng Ji, Fei Huang, and Lingfei Wu. 2020. A joint neural model for information extraction with global features. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7999–8009, Online. Association for Computational Linguistics.

Linguistic Data Consortium. 2005. English annotation guidelines for events. https://www.ldc.upenn.edu/sites/www.ldc.upenn.edu/files/english-events-guidelines-v5.4.3.pdf.

Jian Liu, Yubo Chen, Kang Liu, Wei Bi, and Xiaojiang Liu. 2020. Event extraction as machine reading comprehension. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1641–1651, Online. Association for Computational Linguistics.

Xiao Liu, Heyan Huang, Ge Shi, and Bo Wang. 2022. Dynamic prefix-tuning for generative template-based event extraction. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5216–5228, Dublin, Ireland. Association for Computational Linguistics.

Mengsay Loem, Sho Takase, Masahiro Kaneko, and Naoaki Okazaki. 2022. Extraphrase: Efficient data augmentation for abstractive summarization.

Qing Lyu, Hongming Zhang, Elior Sulem, and Dan Roth. 2021. Zero-shot Event Extraction via Transfer Learning: Challenges and Insights. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 322–332, Online. Association for Computational Linguistics.

Mingyu Derek Ma, Xiaoxuan Wang, Po-Nien Kung, P. Jeffrey Brantingham, Nanyun Peng, and Wei Wang. 2023. Star: Improving low-resource information extraction by structure-to-text data generation with large language models.

Nathan Ng, Kyunghyun Cho, and Marzyeh Ghassemi. 2020. SSMBA: Self-supervised manifold based data augmentation for improving out-of-domain robustness. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1268–1283, Online. Association for Computational Linguistics.

Panupong Pasupat and Percy Liang. 2014. Zero-shot entity extraction from web pages. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 391–401.

Lei Shen, Fandong Meng, Jinchao Zhang, Yang Feng, and Jie Zhou. 2021a. GTM: A generative triple-wise model for conversational question generation. 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 3495–3506, Online. Association for Computational Linguistics.

Shirong Shen, Tongtong Wu, Guilin Qi, Yuan-Fang Li, Gholamreza Haffari, and Sheng Bi. 2021b. Adaptive knowledge-enhanced bayesian meta-learning for few-shot event detection. In *Findings of the Association for Computational Linguistics*, page 2417–2429. Association for Computational Linguistics (ACL).

Zhiyi Song, Ann Bies, Stephanie Strassel, Tom Riese, Justin Mott, Joe Ellis, Jonathan Wright, Seth Kulick, Neville Ryant, and Xiaoyi Ma. 2015. From light to rich ere: annotation of entities, relations, and events. In *Proceedings of the the 3rd Workshop on EVENTS: Definition, Detection, Coreference, and Representation*, pages 89–98.

Bo Sun, Banghuai Li, Shengcai Cai, Ye Yuan, and Chi Zhang. 2021. Fsce: Few-shot object detection via contrastive proposal encoding. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7348–7358.

Bo Wang, Heyan Huang, Xiaochi Wei, Ge Shi, Xiao Liu, Chong Feng, Tong Zhou, Shuaiqiang Wang, and Dawei Yin. 2023a. Boosting event extraction with denoised structure-to-text augmentation. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 11267–11281, Toronto, Canada. Association for Computational Linguistics.

Sijia Wang, Mo Yu, Shiyu Chang, Lichao Sun, and Lifu Huang. 2022. Query and extract: Refining event extraction as type-oriented binary decoding. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 169–182, Dublin, Ireland. Association for Computational Linguistics.

Sijia Wang, Mo Yu, and Lifu Huang. 2023b. The art of prompting: Event detection based on type specific prompts. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 1286–1299, Toronto, Canada. Association for Computational Linguistics.

Xin Wang, Thomas E. Huang, Trevor Darrell, Joseph E Gonzalez, and Fisher Yu. 2020a. Frustratingly simple few-shot object detection.

Yida Wang, Yinhe Zheng, Yong Jiang, and Minlie Huang. 2021a. Diversifying dialog generation via adaptive label smoothing. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics*.

Yufei Wang, Ian D. Wood, Stephen Wan, Mark Dras, and Mark Johnson. 2021b. Mention flags (mf): constraining transformer-based text generators. 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)*, ACL-IJCNLP 2021 - 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, Proceedings of the Conference, pages 103–113. Association for Computational Linguistics (ACL).

Zhuowei Wang, Jing Jiang, Bo Han, Lei Feng, Bo An, Gang Niu, and Guodong Long. 2020b. Seminll: A framework of noisy-label learning by semi-supervised learning. *Transactions on Machine Learning Research*, 2022.

Hongxin Wei, Lei Feng, Xiangyu Chen, and Bo An. 2020. Combating noisy labels by agreement: A joint training method with co-regularization. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13723–13732.

Jason Wei and Kai Zou. 2019. EDA: Easy data augmentation techniques for boosting performance on text classification tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6383–6389, Hong Kong, China. Association for Computational Linguistics.

Xiang Wei, Xingyu Cui, Ning Cheng, Xiaobin Wang, Xin Zhang, Shen Huang, Pengjun Xie, Jinan Xu, Yufeng Chen, Meishan Zhang, Yong Jiang, and Wenjuan Han. 2023. Zero-shot information extraction via chatting with chatgpt.

Yang Xiao and Renaud Marlet. 2020. Few-shot object detection and viewpoint estimation for objects in the wild. In *ECCV*.

Xiaopeng Yan, Ziliang Chen, Anni Xu, Xiaoxi Wang, Xiaodan Liang, and Liang Lin. 2019. Meta r-cnn: Towards general solver for instance-level low-shot learning. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9576–9585.

Yiben Yang, Chaitanya Malaviya, Jared Fernandez, Swabha Swayamdipta, Ronan Le Bras, Ji-Ping Wang, Chandra Bhagavatula, Yejin Choi, and Doug Downey. 2020. Generative data augmentation for commonsense reasoning. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1008–1025, Online. Association for Computational Linguistics.

Quanming Yao, Hansi Yang, Bo Han, Gang Niu, and James Tin-Yau Kwok. 2020. Searching to exploit memorization effect in learning with noisy labels. In *International Conference on Machine Learning*.

Xingrui Yu, Bo Han, Jiangchao Yao, Gang Niu, Ivor Tsang, and Masashi Sugiyama. 2019. How does disagreement help generalization against label corruption? In *International Conference on Machine Learning*, pages 7164–7173.

Gongjie Zhang, Kaiwen Cui, Rongliang Wu, Shijian Lu, and Yonghong Tian. 2021a. Pnpdet: Efficient few-shot detection without forgetting via plug-and-play sub-networks. *2021 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 3822–3831.

Hongming Zhang, Haoyu Wang, and Dan Roth. 2021b. Zero-shot Label-aware Event Trigger and Argument Classification. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1331–1340, Online. Association for Computational Linguistics.

Mingyuan Zhang, Jane Lee, and Shivani Agarwal. 2021c. Learning from noisy labels with no change to the training process. In *International Conference on Machine Learning*.

Zhaowei Zhu, Zihao Dong, and Yang Liu. 2022. Detecting corrupted labels without training a model to predict. In *International Conference on Machine Learning*.

A Implementation

For a fair comparison with baseline approaches, we use the pre-trained `bert-large-uncased` model for fine-tuning and optimizing our model with BertAdam. We optimize the parameters with grid search: training epoch 10, learning rate $\in$ [3e-6, 1e-4], training batch size $\in \{8, 12, 16, 24, 32\}$, dropout rate $\in \{0.4, 0.5, 0.6\}$. Our experiments run on one Quadro RTX 8000. For trigger detection, the average runtime is 3.0 hours. For argument detection, the average runtime is 1.3 hours. We use Spacy to generate POS tags. We use three random seed 0, 39, 42 for all experiments, and report the mean scores.

Sampling Strategy Note that in the context of few-shot learning with an Nway-Kshot setting, the variable K denotes the number of event mentions rather than training examples. The original corpus contains numerous instances where a single sentence includes multiple event mentions, presenting a challenge for the few-shot example sampling process. Without regularization, the sampled examples may probably exceed the specified K event mentions.

To address this issue and ensure that, for every setting, the sampled examples with novel event types do not surpass K, we employ a sorting mechanism based on the frequency of event types in decreasing order. This involves sorting the event types and then sampling in the sorted order. For instance, consider the examples with "Justice:Acquit" mentions, one of which also includes a "Justice:Convict" mention. If we were to first sample examples for "Justice:Convict" and this particular example is omitted, we would miss the opportunity to include this crucial instance for "Justice:Acquit." This becomes especially significant in settings such as 5-shot or 10-shot, where "Justice:Acquit" has a total of four examples. Without this sampling approach, the mentioned example may be excluded from the training procedure, impacting the model's performance.

Generation Instruction The following instruction are used to prompt generations given the event structure: "You are a helpful assistant in generating fluent and reasonable sentences with event mentions. An Event is a specific occurrence involving participants. An Event is something that happens. An Event can frequently be described as a change of state. Please be sure the given event information is in the generated sentence. However, the given context information is optional in generation. Generate a sentence with {event_type_name} event, with optional context information: {list_of_context_entitites}. {event_template}." The {event_template} refers to the textual representation given the event structure, as presented in (Hsu et al., 2022).

B Negative Event Mentions Prompts

Table 6 list generation instructions of negative event mentions for generation agents. Table 7 shows negative augmentation examples.

C Experimental Results with QE

Table 8 shows Experimental results for ACE05-E with QueryExtract (QE) as the baseline model.

D Features Contributed by Augmented Data

The features that are better captured by the proposed approach include (1) The mapping between candidate triggers and event types. The presence of a greater variety of event mention expressions within diverse contexts enhances the robustness and comprehensiveness of the mapping between candidate triggers and event types. (2) The mapping between negative expressions and event types. Due to the limited occurrence of negative events in the training data, their availability as few-shot examples is restricted. With the integration of the negative augmentation module, the mapping between negative expressions and event types becomes clearer. (3) The relation between candidate triggers and arguments. The generated sentences exhibit a comparatively higher prevalence of straightforward event expressions than those present in annotated data, such as ACE2005. These less complex expressions contribute to a good fit for features related to the relation between candidate triggers and arguments, in the low-resource settings.

Event Expression Type	Instruction Prompt
Negative Events	An Event is NEGATIVE when it is explicitly indicated that the Event did not occur. Negative example 1: His wife was sitting in the backseat and was 'not hurt'. Negative example 2: Yeltsin ordered Skuratov's suspension, but parliament repeatedly 'refused to sack' him. Given the generated sentence, "[SENT]", change it into a negative expression that the Event did not occur.
Believed Events	Believed Events are event mentions that some people or organizations think or believe would happen but are not necessarily real or true event occurrences. Example 1: Rumors of 'arrests' circulated in Vancouver. Example 2: The charity was suspected of 'giving' money to al Qaeda. Given the generated sentence you provide, '[SENT]', change it into a believed event sentence:
Hypothetical Events	Hypothetical events are event mentions that are supposed to happen but are not necessarily real or true event occurrences. Example 1: Should he not 'pay' the money, they would 'kill' him. Example 2: A demonstration of how he would behave if he were to 'become' President. Given the generated sentence you provide, '[SENT]', change it into a hypothetical event sentence:
Promised Events	Promised Events are event mentions that are promised to happen but are not necessarily real or true event occurrences. Example 1: He said he would 'leave' town. Example 2: Promises of 'aid' made by Arab and European countries. Given the generated sentence you provide, '[SENT]', change it into a promised event sentence:
Desired Event	Desired events are event mentions that are desired to happen but not necessarily real or true event occurrences. Example: They wanted to 'acquire' the company last year. Given the generated sentence you provide, "[SENT]", change it into a Desired event sentence:

Table 6: Negative/asserted expression generation template. "[SENT]" is a placeholder for the generated sentence with a positive event expression. The instruction is adapted from (Linguistic Data Consortium, 2005).

id	**Note**	**Content**
1	Event Structure	Trigger: bankruptcy. Org: Hazelhurst & Associates Inc.
	Context	10 percent, yesterday, $22.5 million
	Positive mention	Hazelhurst & Associates Inc. declared bankruptcy yesterday, with $22.5 million in debts.
	Negative mention	Hazelhurst & Associates Inc. did not declare bankruptcy yesterday, with $22.5 million in debts.
	Asserted mention:	It is believed that Hazelhurst & Associates Inc. will declare bankruptcy tomorrow, with $30 million in debts.
2	Event Structure	Trigger: pardon, Place: Jordan, Adjudicator: Abdullah II, Defendant: Rich
	Context	Republicans, today, his darkest hours
	Positive mention	Rich received a pardon from Abdullah II during his darkest hours , as Republicans gathered today to offer their support .
	Negative mention	Rich's pardon from Abdullah II was canceled during his darkest hours, as Republicans did not gather.
	Asserted mention	Rich desired to receive a pardon from Abdullah II during his darkest hours, as Republicans gathered last year to offer their support.

Table 7: Negative Augmentation Example

Method	**K-shot**	Common 5				Common 10			
		Tri-I	Tri-C	Arg-I	Arg-C	Tri-I	Tri-C	Arg-I	Arg-C
QE	1-shot	58.6	48.7	33.1	29.3	58.6	51.2	37.5	30.1
	5-shot	61.9	57.1	37.6	33.1	66.7	61.1	41.7	36.5
	10-shot	64.1	62.2	40.3	38.6	72.0	67.2	45.6	45.2
TOLAR-QE (Vicuna)	1-shot	60.6	58.0	41.8	34.2	60.4	58.0	41.4	35.0
	5-shot	65.4	62.1	44.3	35.8	70.8	68.8	47.2	41.6
	10-shot	65.7	64.0	43.4	39.6	69.5	68.1	50.8	43.7
TOLAR-QE (LLaMa)	1-shot	64.7	57.6	39.3	28.3	57.8	54.9	43.5	33.9
	5-shot	61.6	59.4	42.3	37.1	71.2	65.1	46.2	40.9
	10-shot	66.0	64.9	44.1	39.8	68.2	67.4	49.4	44.9
TOLAR-QE (GPT)	1-shot	64.8	58.7	38.4	31.3	62.8	61.2	43.8	36.1
	5-shot	67.5	59.6	41.4	36.5	66.1	66.1	47.5	43.6
	10-shot	67.4	65.2	42.7	39.1	71.1	70.4	49.2	46.5

Table 8: Few-shot Event Extraction results with data augmentation on ACE05-E with QueryExtract (QE).