SNaRe 🥁: Domain-aware Data Generation for Low-Resource Event Detection

Tanmay Parekh Yuxuan Dong Lucas Bandarkar Artin Kim
I-Hung Hsu[†] **Kai-Wei Chang Nanyun Peng**
Computer Science Department, University of California, Los Angeles [†]Google
{tparekh, kwchang, violetpeng}@cs.ucla.edu

Abstract

Event Detection (ED) – the task of identifying event mentions from natural language text – is critical for enabling reasoning in highly specialized domains such as biomedicine, law, and epidemiology. Data generation has proven to be effective in broadening its utility to wider applications without requiring expensive expert annotations. However, when existing generation approaches are applied to specialized domains, they struggle with label noise, where annotations are incorrect, and domain drift, characterized by a distributional mismatch between generated sentences and the target domain. To address these issues, we introduce SNaRe, a domain-aware synthetic data generation framework composed of three components: Scout, Narrator, and Refiner. Scout extracts triggers from unlabeled target domain data and curates a high-quality domain-specific trigger list using corpus-level statistics to mitigate domain drift. Narrator, conditioned on these triggers, generates high-quality domain-aligned sentences, and Refiner identifies additional event mentions, ensuring high annotation quality. Experimentation on three diverse domain ED datasets reveals how SNaRe outperforms the best baseline, achieving average F1 gains of 3-7% in the zero-shot/few-shot settings and 4-20% F1 improvement for multilingual generation. Analyzing the generated trigger hit rate and human evaluation substantiates SNaRe's stronger annotation quality and reduced domain drift. We will release our code at `https://github.com/PlusLabNLP/SNaRe`.

1 Introduction

Event Detection (ED) (Sundheim, 1992; Doddington et al., 2004) involves identifying and categorizing significant events from natural language text based on a pre-defined ontology. It has widespread applications in domains such as biomedicine (Pyysalo et al., 2012), epidemiology (Parekh et al., 2024b,c), law (Francesconi et al., 2010). Due to the high cost of expert-annotated data, synthetic data generation (Xu et al., 2023) (i.e., generating sentence and event annotations) has emerged as a promising alternative, particularly for practical use-cases in specialized domains.

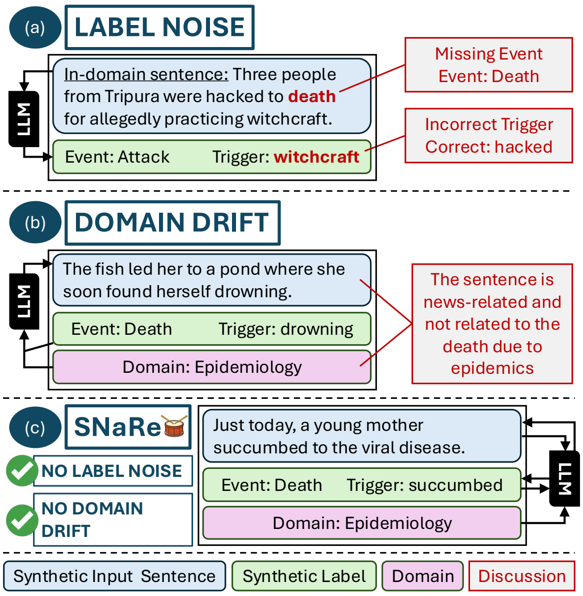

(a)
LABEL NOISE
LLM
In-domain sentence: Three people from Tripura were hacked to death for allegedly practicing witchcraft.
Event: Attack Trigger: witchcraft
Missing Event Event: Death
Incorrect Trigger Correct: hacked
(b)
DOMAIN DRIFT
LLM
The fish led her to a pond where she soon found herself drowning.
Event: Death Trigger: drowning
Domain: Epidemiology
The sentence is news-related and not related to the death due to epidemics
(c)
SNaRe
NO LABEL NOISE
NO DOMAIN DRIFT
Just today, a young mother succumbed to the viral disease.
Event: Death Trigger: succumbed
Domain: Epidemiology
LLM
Synthetic Input Sentence
Synthetic Label
Domain
Discussion

Figure 1: Highlighting the errors of existing data generation approaches. (a) Using LLMs to generate the labels from in-domain sentences leads to *label noise* owing to poor LLM reasoning. (b) Utilizing LLMs to generate sentences conditioned on event and domain causes *domain drift*, wherein the synthetic sentence is not aligned with the target domain. Finally, in (c), we illustrate how SNaRe minimizes both errors to generate higher quality synthetic data.

However, existing generation approaches often focus on general-domain settings and fail to address the distinct challenges of specialized domains (Song et al., 2025). Weak supervision methods (He et al., 2021; Chia et al., 2022) that use LLMs to generate labels for unlabeled sentences frequently introduce *label noise* (Figure 1(a)), where incorrect or incomplete labels arise due to weak LLM reasoning (Huang et al., 2024) or limited domain

Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pages 20583–20604
November 4-9, 2025 ©2025 Association for Computational Linguistics

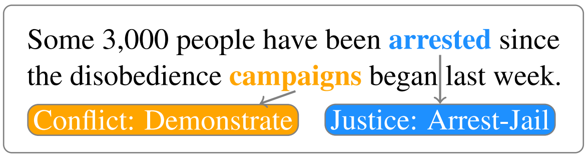
knowledge (Song et al., 2025). Downstream training on such incorrect labels can cause spurious bias propagation. Conversely, recent generation approaches (Josifoski et al., 2023; Ma et al., 2024) that utilize LLMs' self-knowledge to jointly generate labels and sentences struggle with *domain drift* (Figure 1(b)), often synthesizing sentences that are misaligned with the target domain. This can be attributed to the lack of utilization of target domain information for generation and can drastically hamper model training as lexical and structural cues are highly influential for ED (Tong et al., 2022). Overall, *label noise* and *domain drift* reduce synthetic data quality, eventually leading to subpar supervised downstream model performance.

To this end, we propose SNaRe 🥁, a novel domain-aware, three-stage data synthesis LLM pipeline comprising the **S**cout, **Na**rrator, and **Re**finer modules. Scout surveys unlabeled target domain data to identify salient triggers via prompt-based trigger extraction. Using corpus-level statistics for automated aggregation and filtering, Scout curates a list of high-quality domain-specific triggers per event type. Next, Narrator samples from these domain-specific triggers and utilizes LLMs to synthesize diverse sentences for each event type. Utilizing specialized and diverse domain information for conditional generation aids Narrator in generating more domain-aligned sentences, eventually reducing domain drift in the synthesized sentences. Since Narrator sentences could mention additional events apart from the input set, we design the Refiner to utilize LLM inference to annotate such missing events in these sentences. Narrator's conditional text generation and Refiner's missing label annotation aid in reducing the label noise and ensuring high data quality. We provide an illustration of SNaRe's generation in Figure 1(c).

We benchmark SNaRe on ED datasets from three domains: ACE (Doddington et al., 2004) (news), SPEED (Parekh et al., 2024c) (epidemiology), and GENIA2011 (Kim et al., 2011) (biomedical). For evaluation, we report the ED performance of DEGREE (Hsu et al., 2022) trained on the synthesized data. Across the zero-shot and few-shot settings, SNaRe performs the best, outperforming the previous state-of-the-art baselines (Ding et al., 2023; Ma et al., 2024) by an average of 3-7% F1 points. Under multilingual generation for Arabic and Chinese, SNaRe outshines even more, with improvements of 4-20% F1 over the best baseline. Our analysis reveals how SNaRe's synthesized triggers overlap 4-11% more (relative to baselines) with the gold trigger set, demonstrating the reduction in domain drift. Finally, human evaluation provides qualitative evidence for SNaRe's superior data annotation quality and domain alignment.[1]

2 Problem Definition

We focus on the task of Event Detection (Sundheim, 1992; Doddington et al., 2004) for this work. ED aims to extract mentions of any events of interest from natural language text. Following ACE 2005 (Doddington et al., 2004), we define an *event* as something that happens or describes a change of state and is labeled by a specific *event type*. The word/phrase that most distinctly highlights the occurrence of the event is defined as the *event trigger*, and the trigger-event type pair is known as the *event mention*. *Event Detection* requires extracting the event *triggers* from the sentence and classifying them into one of the pre-defined event types. We provide an illustration of this task below, where *arrested* and *campaigns* trigger the events of *Justice: Arrest-Jail* and *Conflict: Demonstrate*, respectively.

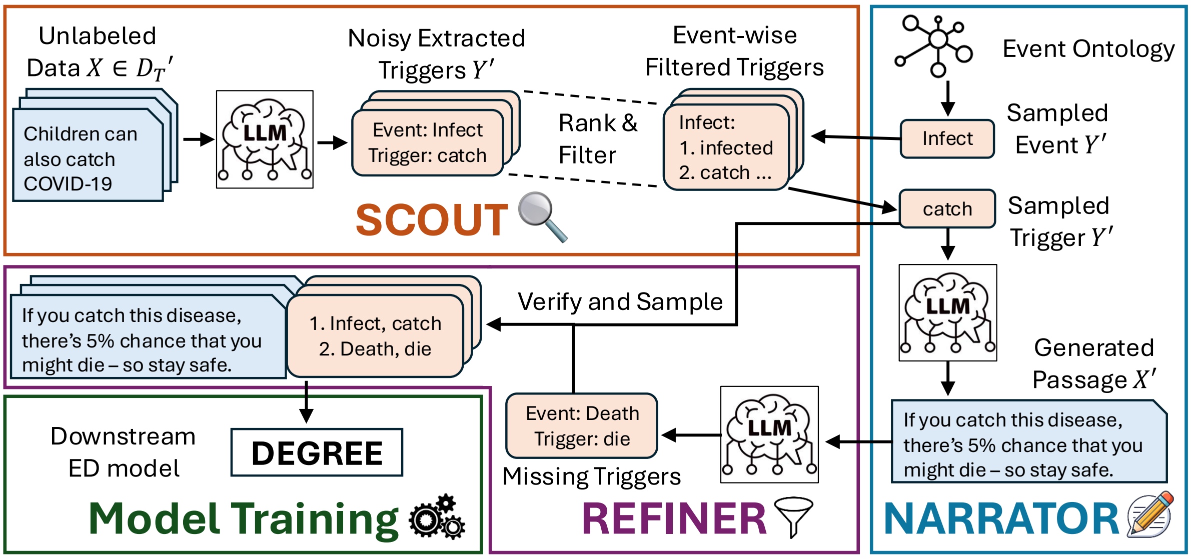

Some 3,000 people have been arrested since
the disobedience campaigns began last week.
Conflict: Demonstrate
Justice: Arrest-Jail

In our work, we specifically focus on ED in diverse and specialized domains (e.g., biomedical), where procuring a training dataset D_T of annotated data points is expensive, but unlabeled data D'_T is available. We focus on two realistic low-resource data setups - **zero-shot** (zero labeled data) and **few-shot** (k labeled datapoints per event type) settings. Unlike domain transfer, we do not consider any labeled data for the source domain, and directly optimize model performance for the target domain.

3 Related Works

Data Generation for Information Extraction LLM-powered synthetic data generation has been successful for various NLP tasks (Li et al., 2023b; Wang et al., 2023c; Wu et al., 2024; Shao et al., 2025). For information extraction, works have explored knowledge retrieval (Chen and Feng, 2023; Amalvy et al., 2023), translation (Parekh et al., 2024a; Le et al., 2024), data re-editing (Lee et al., 2021; Hu et al., 2023), and label extension (Zhang

[1] We will release our code and data upon acceptance.



Children can also catch COVID-19

If you catch this disease, there's 5% chance that you might die – so stay safe.

Figure 2: Model Architecture Diagram highlighting the various components of SNARE. First, Scout extracts and filters domain-specific triggers, then Narrator generates passages conditioned on these triggers. Finally, Refiner adds any missing annotations and sample N data points per event for downstream training.

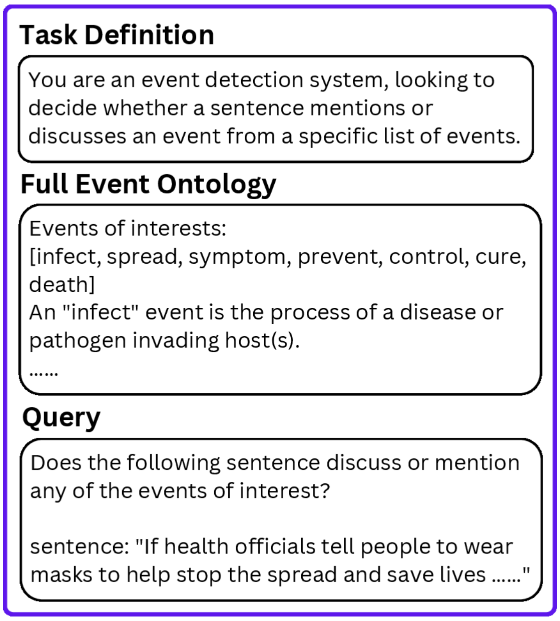
et al., 2024). Recent works utilize LLMs to generate labels for sentences (Chia et al., 2022; Ye et al., 2022; Wang et al., 2023a; Tang et al., 2023), while some other works explore the generation of sentences from labels (Josifoski et al., 2023; Ma et al., 2024). Our work introduces SNARE focused on infusing better domain-specific information for data generation.

Low-resource Event Detection Event Detection (ED) has been studied extensively (Sundheim, 1992; Grishman and Sundheim, 1996), leading to diverse datasets in news (Doddington et al., 2004; Song et al., 2015; Ellis et al., 2015), Wikipedia (Li et al., 2021; Pouran Ben Veyseh et al., 2022), and general domains (Wang et al., 2020; Parekh et al., 2023), as well as niche areas like biomedical (Pyysalo et al., 2012; Kim et al., 2011, 2013), multimedia (Li et al., 2020), cybersecurity (Satyapanich et al., 2020), epidemiology (Parekh et al., 2024b,c), and pharmacovigilance (Sun et al., 2022). To address the growing need for event detection across expanding domains, prior works have explored transfer learning via Abstract Meaning Representation (Huang et al., 2018), Semantic Role Labeling (Zhang et al., 2021), and Question Answering (Lyu et al., 2021). Reformulating ED as a conditional generation task has also aided low-resource training (Hsu et al., 2022, 2023b; Huang et al., 2022). Recently, LLM-based reasoning (Li et al., 2023a; Gao et al., 2023; Wang et al., 2023b; Parekh et al., 2025a) and transfer-learning (Cai et al., 2024) has been explored, but their performance remains inferior to supervised models (Huang et al., 2024). This motivates efforts in LLM-powered synthetic data generation for low-resource ED. Although our work only demonstrates results on ED, we believe our work can be extended to other tasks outside information extraction as well, like question-answering (Rajpurkar et al., 2016; Parekh et al., 2025b) and long-form generation (Suvarna et al., 2024; Chen et al., 2020; Parekh et al., 2020).

4 Methodology - SNARE

In this work, we focus on domain-aware synthetic data generation using LLMs to alleviate the need for expert-annotated training data. By generating a large dataset $D_s = \{(X, Y)\}$, we can train domain-specific downstream ED models using minimal supervised data.

Existing weak-supervision approaches (Mintz et al., 2009; Wang et al., 2021) utilizing automatic methods to assign labels to unlabeled sentences often generate incorrect labels (*label noise*). This can be attributed to the domain-specific context understanding and deep reasoning requirement of ED, leading to poor automatic label quality, even when using recent LLMs (Huang et al., 2024). Another route of approaches that utilize LLMs to generate sentences conditioned on labels (Schick and Schütze, 2021; Josifoski et al., 2023), i.e., synthesize X for a designated Y, often curate sentences that are distributionally divergent from the target



Task Definition

You are an event detection system, looking to decide whether a sentence mentions or discusses an event from a specific list of events.

Full Event Ontology

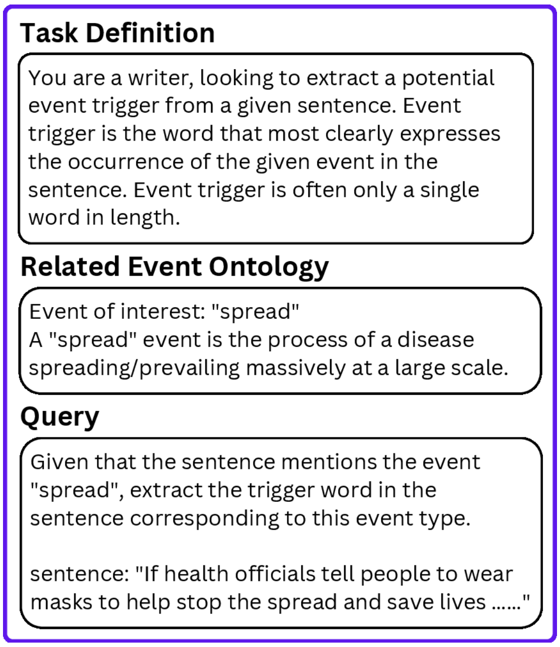
Events of interests:
[infect, spread, symptom, prevent, control, cure, death]
An "infect" event is the process of a disease or pathogen invading host(s).
……

Query

Does the following sentence discuss or mention any of the events of interest?

sentence: "If health officials tell people to wear masks to help stop the spread and save lives ……"

Figure 3: Prompt for stage 1 of Scout.



Task Definition

You are a writer, looking to extract a potential event trigger from a given sentence. Event trigger is the word that most clearly expresses the occurrence of the given event in the sentence. Event trigger is often only a single word in length.

Related Event Ontology

Event of interest: "spread"
A "spread" event is the process of a disease spreading/prevailing massively at a large scale.

Query

Given that the sentence mentions the event "spread", extract the trigger word in the sentence corresponding to this event type.

sentence: "If health officials tell people to wear masks to help stop the spread and save lives ……"

Figure 4: Prompt for stage 2 of Scout.

domain (*domain drift*). This is mainly since these approaches focus on the general domain and fail to utilize any target domain signals in their generation. Both label noise and domain drift hurt the synthetic data quality, in turn, diminishing the downstream supervised model performance.

To mitigate these issues, we propose SNARE 🥁, a domain-aware data synthesizer, that generate and verifies LLM generations to ensure high-quality data (Hsu et al., 2024), comprising three components: **S**cout, **Na**rrator, and **Re**finer: Scout studies unlabeled target domain data D'_T to curate domain-specific triggers, in turn, reducing domain drift. Narrator generates domain-specific sentences conditioned on Scout's curated triggers, while Refiner adds additional annotations to ensure high-quality labels. Overall, SNARE is a training-free LLM inference pipeline and easily deployable and scalable across domains. We provide our architectural diagram in Figure 2 and explain each component of our pipeline below.

4.1 Scout

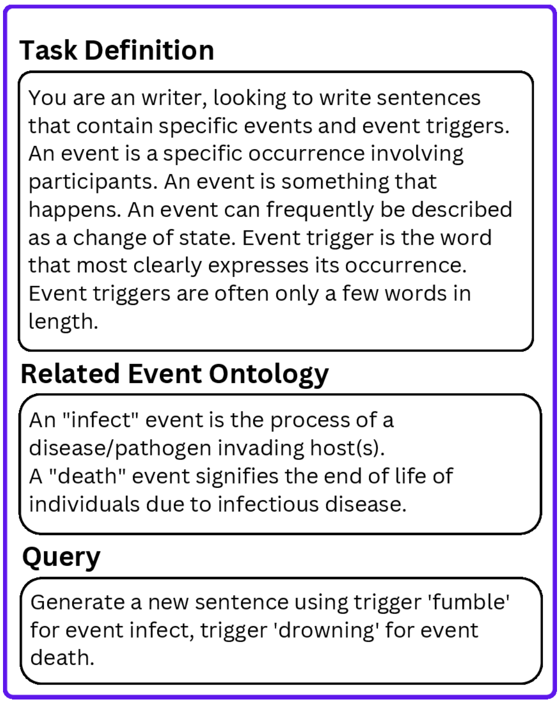
Scout is tasked with the curation of domain-specific triggers that are later utilized for sentence generation. Events can assume a wide range of triggers depending on the domain and context. For example, an "Attack" event can be triggered by *war*, *killed* in news, or *breach*, *phish* in cybersecurity, or *infect*, *transmit* in epidemiology domains.

Thus, unlike past works (Ma et al., 2024) that utilize only LLMs' internal knowledge for trigger generation, we develop **Scout**, which extracts high-precision domain-specific triggers using unlabeled target domain data D'_T. Specifically, trigger extraction involves a two-stage prompt setup: (1) The first stage is tasked with identifying and filtering possible event types mentioned in the target domain sentence, and (2) The second stage aims to find the most appropriate trigger word from the unlabeled sentence for each filtered event type. We provide the prompts for Scout in Figure 3 and 4.

To ensure high-precision of the triggers, we develop an aggregation and filtering mechanism by incorporating corpus-level statistics. Specifically, for each event type, we aggregate the counts of the triggers at the corpus level and filter out the top $t = 10$ triggers as the curated list of high-quality domain-specific event-indexed triggers. These triggers carry important target domain signals that help in generating domain-specific sentences (§ 4.2), in turn reducing domain drift of our synthetic data.

4.2 Narrator

Narrator is tasked with the synthesis of domain-specific sentences for our synthetic dataset. Existing works (Josifoski et al., 2023; Ma et al., 2024) do not utilize any target domain information, which causes domain drift in their synthesized sentences. Instead, in our work, we condition the Narrator to utilize the rich and diverse domain-specific triggers from Scout to synthesize domain-specific sentences, which, in turn, reduce domain drift.



Task Definition

You are an writer, looking to write sentences that contain specific events and event triggers. 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. Event trigger is the word that most clearly expresses its occurrence. Event triggers are often only a few words in length.

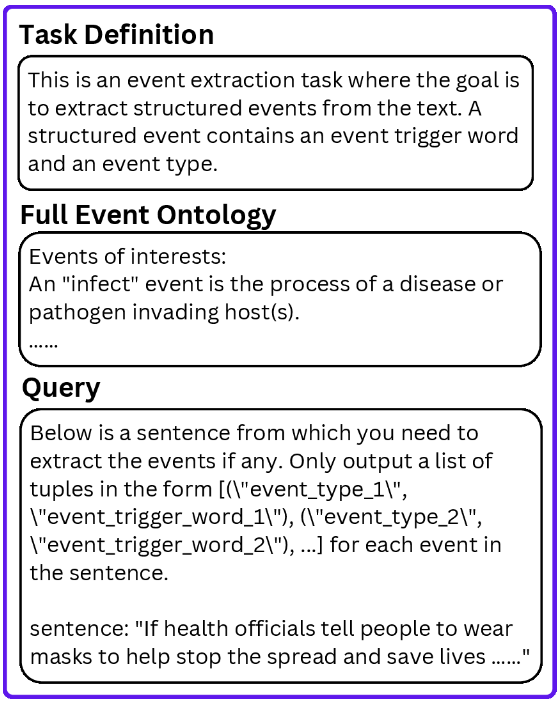
Related Event Ontology

An "infect" event is the process of a disease/pathogen invading host(s).
A "death" event signifies the end of life of individuals due to infectious disease.

Query

Generate a new sentence using trigger 'fumble' for event infect, trigger 'drowning' for event death.

Figure 5: Prompt for Narrator.



Task Definition

This is an event extraction task where the goal is to extract structured events from the text. A structured event contains an event trigger word and an event type.

Full Event Ontology

Events of interests:
An "infect" event is the process of a disease or pathogen invading host(s).
......

Query

Below is a sentence from which you need to extract the events if any. Only output a list of tuples in the form [(\"event_type_1\", \"event_trigger_word_1\"), (\"event_type_2\", \"event_trigger_word_2\"), ...] for each event in the sentence.

sentence: "If health officials tell people to wear masks to help stop the spread and save lives ......"

Figure 6: Prompt for Refiner.

Specifically, Narrator samples 1-2 event types per synthetic data instance and corresponding domain-specific triggers from Scout's curated trigger list – constituting the label Y. Next, it prompts the LLM with the task instructions and the event definitions, and asks it to generate a passage X' that mentions the sampled events using the sampled triggers (Y). We illustrate this prompt in Figure 5. The generated sentences are naturally more aligned with the target domain owing to the conditioning on the domain-specific triggers (qualitative examples shown in § 7.6). Further domain-specific adaptations are possible by fine-tuning the LLM on unlabeled domain-specific data (analysis in § 7.5).

4.3 Refiner

While the Narrator ensures the sampled triggers are mentioned in the passage, it could potentially introduce new unknown events in the passage, leading to under-annotated (X', Y) data instances. We illustrate this in Figure 7 where the sampled trigger *positive* for *infect* event is mentioned, but the sentence also mentions the missing *symptom* event triggered by *got*.

To account for such missing annotations, we introduce **Refiner** - tasked to annotate missing events in the generated sentence. Here, we simply prompt the LLM to find all event mentions in the generated sentence (illustrated in Figure 6) and append them to the original sampled Y. Since these new refined labels can be noisy, we avoid updating them for the already present events from the Scout, and only add them for newly discovered events. To further improve data quality, we apply an automated rule to remove passages that do not mention the target trigger. Additionally, we standardize trigger annotations by correcting variations in trigger word forms. Such normalization and conservative filtering further aid in ensuring high data quality. Finally, we apply a greedy sampling algorithm to sample $N = 50$ instances (X', Y) per event type to create our final synthetic dataset D_s.

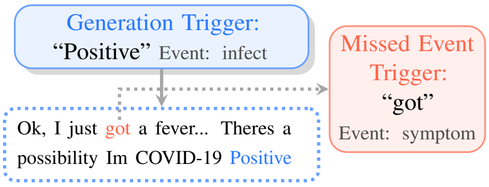


Figure 7: Illustration of how inverse generation can produce unannotated event mentions. Blue box = target event mention, red box = unannotated event mention.

Downstream Model Training: The final component utilizes the generated synthetic data D_s to train downstream ED models in a supervised manner. The trained ED models are then used to infer on the test set and for eventual evaluation. Since we use small BART-based language models for inference, our inference time computation is negligible

Base LLM	Method	Unlabeled Data Source	ACE		SPEED		GENIA		Average	
			Eve-I	Tri-C	Eve-I	Tri-C	Eve-I	Tri-C	Eve-I	Tri-C
Llama3-8B	Inference	-	30.2	23.8	39.8	25.4	21.9	17.2	30.6	22.1
	STAR	-	44.9	35.0	21.0	10.1	25.9	19.0	30.6	21.4
	Weak Sup	train	41.7	37.8	45.6	31.5	26.9	21.4	38.1	30.2
	SNaRe (ours)	train	57.4	50.2	44.6	31.5	**35.2**	**28.9**	45.7	**36.9**
	SNaRe (ours)	external	**57.7**	**52.6**	**47.8**	**32.9**	33.6	24.6	**46.4**	36.7
Llama3-70B	Inference	-	46.9	41.3	46.9	35.6	34.2	28.2	42.7	35.0
	STAR	-	50.0	42.3	18.3	13.8	23.3	16.9	30.5	24.3
	Weak Sup	train	53.2	48.0	**52.8**	**39.6**	36.2	29.1	47.4	38.9
	SNaRe (ours)	train	58.1	53.8	49.9	38.7	38.0	29.7	48.7	40.7
	SNaRe (ours)	external	**59.7**	**55.6**	50.1	39.2	**39.2**	**31.5**	**49.7**	**42.1**
GPT-3.5	Inference	-	33.0	26.2	44.2	32.9	31.2	24.7	36.1	27.9
	STAR	-	45.0	36.6	21.3	14.6	21.8	14.3	29.4	21.8
	Weak Sup	train	49.7	44.6	**50.7**	**37.5**	37.7	30.1	46.1	37.4
	SNaRe (ours)	train	**54.8**	48.3	50.3	36.8	**39.3**	**31.1**	**48.1**	**38.7**
	SNaRe (ours)	external	54.0	**48.5**	50.1	36.1	38.7	29.4	47.6	38.0
(Upper Bound)	Gold Data	-	64.6	61.6	64.0	53.5	51.3	44.0	60.0	53.0

Table 1: Zero-shot results comparing SNaRe with other baselines across three datasets and three base LLMs. Except for Inference, all other evaluations are performances of downstream DEGREE (Hsu et al., 2022) model trained on data generated by each technique (50 datapoints per event type).

compared to LLM inference methods.

5 Experimental Setup

Datasets: We consider three ED datasets from diverse domains for our experiments: (1) ACE (Doddington et al., 2004), in the news domain, (2) SPEED (Parekh et al., 2024c), in the social media domain, and (3) GENIA (Kim et al., 2011), in the biomedical domain. We simplify GENIA by converting the original document-level annotations to sentence-level annotations. We consider the Arabic and Chinese versions of ACE for cross-lingual experiments. For the few-shot setting, we sample k few-shot examples from the training data.

For our unlabeled data, we consider two sources: (1) **Train** - annotation-free training splits (i.e., only the text) of each dataset and (2) **External** - unlabeled data from other external sources. For this external data source, for ACE, we utilize News Category Dataset (Misra, 2022) comprising Huffpost news articles from 2012-2022. We filter articles corresponding to political, financial, and business articles. For SPEED, we utilize COVIDKB (Zong et al., 2022), comprising tweets from the Twitter COVID-19 Endpoint. Finally, we utilize GENIA2013 (Kim et al., 2013). We provide statistics about these datasets in Table 10.

Baseline methods: We consider three LLM-based techniques for low-resource ED as the baselines for our work. (1) Inference (Gao et al., 2023): LLMs are used to directly infer on the target test data using their reasoning capability. (2) STAR (Ma et al., 2024): The state-of-the-art generation model for ED utilizing LLMs for trigger and passage generation without using any unlabeled data, (3) Weak Supervision (Weak Sup) (Ding et al., 2023): LLMs are utilized to synthesize labels for unlabeled data. For an upper bound reference, we also include a Gold Data generation baseline wherein we sample from the gold training data of each dataset to train the downstream ED model.

Base models: For our base LLMs, we consider three instruction-tuned LLMs of varying sizes, namely Llama3-8B-Instruct (8B model), Llama3-70B-Instruct (70B model) (Dubey et al., 2024), and GPT-3.5 (175B model) (Brown et al., 2020). For our downstream ED model, we consider a specialized low-resource model DEGREE (Hsu et al., 2022), a generative model prompted to fill event templates powered by a BART-large pre-trained language model (Lewis et al., 2020).

Evaluation: Our primary evaluation metric is supervised model performance trained on the synthesized data. We consider two low-resource settings - zero-shot (no labeled data) and few-shot (k datapoints per event type are used). For Inference baseline, the LLM is directly run on the test set to procure model predictions. We report the F1 scores for two metrics (Ahn, 2006): (1) Event Identification (Eve-I) - correct identification of events, and (2) Trigger Classification (Tri-C) - correct identifi-

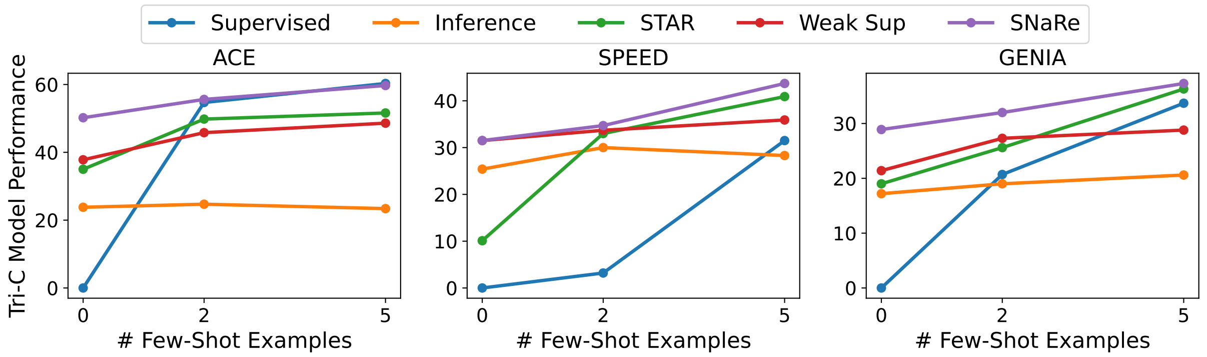


Figure 8: Few-shot results comparing SNARE with other baselines across three datasets using Llama3-8B-Instruct as the base LLM. Except for Inference, all other evaluations are performances of the downstream DEGREE (Hsu et al., 2022) model trained on data generated by each technique. Tri-C: Trigger Classification F1, #: Number of.

cation of trigger-event pairs.

Implementation Details: We follow STAR for the implementation of the baseline models and most hyperparameter settings. For SNARE's passage generation, we select the top $t = 10$ triggers (except $t = 8$ for GENIA) for passage generation. We generate $N = 50$ datapoints per event type for each generation strategy. All our experimental results are reported over an average of three runs. Additional details are provided in Appendix B.

6 Results

We present the results for our zero-shot, few-shot settings, and cross-lingual experiment below.

6.1 Zero-shot Results

We present the main zero-shot results comparing the baselines across different LLMs and datasets in Table 1 and discuss our findings below.

SNARE performs the best: On average, SNARE outperforms STAR by 17.3% Eve-I F1 and 16.3% Tri-C F1 points – demonstrating how domain-specific cues from unlabeled data aid in tackling domain drift. Compared to Weak Sup, SNARE provides average gains of 3.6% Eve-I F1 and 3.3% Tri-C F1, suggesting how cleaner label quality can help improve model performance.

External data source is effective: Assuming access to the training data as the unlabeled data source can be a strong assumption and bias for SNARE. To verify the robustness of our approach, we also evaluate SNARE with external data sources. Surprisingly, as seen in Table 1, SNARE with external data provides similar gains of 4% Eve-I F1 and 3.4% Tri-C F1 over the best baseline. We posit that the higher volume of external data leads to the extraction of cleaner domain-specific triggers, which eventually aids in better downstream performance.

LLM	Method	Arabic		Chinese	
		EI	TC	EI	TC
Llama3-8B	Inference	21.5	13.4	15.0	11.8
	STAR	11.5	10.5	19.7	16.0
	Weak Sup	21.5	16.2	26.3	19.1
	SNARE (ours)	**40.1**	**33.6**	**35.9**	**31.1**
Llama3-70B	Inference	37.5	27.7	32.0	29.1
	STAR	37.1	30.9	26.0	22.0
	Weak Sup	30.0	20.4	28.1	26.3
	SNARE (ours)	**47.5**	**44.0**	**40.4**	**33.1**

Table 2: Comparing SNARE with zero-shot inference for other powerful LLMs for the ACE dataset.

6.2 Few-shot Results

We also study the various methods in the presence of small annotated data as part of our few-shot experiments. Specifically, we study the $k = 2$ and $k = 5$ few-shot settings, where k annotated examples per event type are utilized. We utilize the k-shots as in-context examples in the LLM prompts and append these few-shot examples to the synthesized training data as well. Additionally, we consider another baseline (Supervised) of downstream models trained only on the k-shot examples. We present the Tri-C results for all the datasets for the Llama3-8B model in Figure 8. Similar to zero-shot results, we observe that SNARE consistently outperforms all other baseline models. On average, SNARE outperforms STAR and Weak Sup by 5.4% Tri-C F1 and 7% Tri-C F1 respectively.

Method	ACE	SPEED	GENIA
SNaRe	50.2	31.5	28.9
– Scout	43.2	27.8	28.2
– Narrator	37.8	31.5	21.4
– Refiner	47.4	23.3	22.0

Table 3: Ablation study for SNaRe's Scout, Narrator, and Refiner measured as Tri-C F1 performance across the three datasets.

LLM + Method	Eve-I	Tri-C
Llama3-70B + Inference	46.9	41.3
Llama3.3-70B + Inference	48.9	43.0
Qwen2.5-72B + Inference	40.9	34.2
GPT4o-mini + Inference	34.5	28.8
GPT4o + Inference	51.4	47.7
QwQ-32B + Inference	49.7	43.5
Deepseek-R1-L3-70B + Inference	41.8	36.6
Llama3-70B + SNaRe (train)	**58.1**	**53.8**
Llama3-70B + SNaRe (external)	**59.7**	**55.6**

Table 4: Comparing SNaRe with zero-shot inference for other powerful LLMs for the ACE dataset.

6.3 Zero-shot Multilingual Results

To highlight the utility of our work, we apply our work across languages, specifically Arabic (ar) and Chinese (zh). We used multilingual ACE data (Doddington et al., 2004) for this experiment and utilized TagPrime (Hsu et al., 2023a), powered by XLM-Roberta-large (Conneau et al., 2019), as the downstream ED model. We present our results for Llama3-8B-Instruct and Llama3-70B-Instruct LLMs in Table 2. Surprisingly, SNaRe performs the best out-of-the-box, with improvements ranging 10-20% F1 for Arabic and 4-12% F1 for Chinese, highlighting the broader impact of our work.

7 Analysis

In this section, we study the superior performance of SNaRe through various analyses. Unless specified, we use Llama3-8B-Instruct as the base LLM.

7.1 Ablation study

Table 3 shows the ablation study for Scout and Refiner. For ablating Scout, we replace it by prompting LLM to directly generate triggers. Ablating Narrator is similar to the Weak Supbaseline, while adding additional refiner and dataset statistics to remove noisy datapoints. We observe how all the components are critical, with average performance reductions of 3.8% F1, 6.6% F1, and 6% F1 upon removing the respective components of Scout, Narrator, and Refiner.

Method	ACE		SPEED		GENIA	
	EI	TC	EI	TC	EI	TC
SNaRe	**57.4**	**50.2**	**44.6**	**31.5**	**35.2**	**28.9**
Weak Sup + STAR	46.9	38.9	44.5	29.5	30.2	24.3

Table 5: Comparing SNaRe with data-mixing of synthesized data from previous works.

Method	ACE	SPEED	GENIA
STAR	9.6%	15.3%	15.1%
Weak Sup	19.1%	38.4%	44.8%
SNaRe	**23.2%**	**49.4%**	**52.5%**

Table 6: Reporting the hit rate of synthesized data triggers relative to gold test triggers.

7.2 Comparison with Powerful LLMs

To further highlight the efficacy of SNaRe, we compare it with zero-shot inference using more powerful recent LLMs for the ACE dataset in Table 4. As seen, Llama3-70B with SNaRe significantly outperforms stronger LLMs like GPT4o and thinking-based models like QwQ-32B by 8-10% Eve-I F1 and 8-12% Tri-C F1 scores, respectively. This highlights the strong efficacy of synthetic generation SNaRe over zero-shot inference even when using more powerful LLMs.

7.3 Comparison with Data mixing

Data-mixing (Hoffmann et al., 2022; Xie et al., 2023) is a widely used technique to leverage complementary information across datasets to promote robust downstream model training. We mix data from our two baselines of Weak Sup and STAR as a hybrid baseline to compare with SNaRe. To keep the comparisons fair, we consider $N/2 = 25$ data instances per event type from each dataset. Results from Table 5 demonstrate how SNaRe outperforms the data-mixing based hybrid model by 5-6% F1, underlining the significance of our three-stage model design over simpler data-mixing.

7.4 Analyzing domain drift and label quality

ED models have a strong tendency to over-rely on lexical relations between triggers and events (Tong et al., 2022). Thus, we compare the synthetic data triggers with the gold test triggers as a raw study of the domain drift of triggers in the synthesized data. Specifically, we extract triggers per event type in the synthetic datasets and measure the hit rate/overlap of the synthesized triggers with the gold set of triggers, as reported in Table 6. STAR's

Dataset	Event	Method	Trigger	Sentence
ACE	Attack	STAR	raid	As the rebels embarked on a daring trek across the desert, they launched a surprise **raid** on the heavily guarded fortress, catching the enemy off guard.
		SNARE	shooting	As the rival businessman signed the contract, a sudden **shooting** erupted outside, causing chaos in the midst of the transaction.
SPEED	Death	STAR	asphyxiation	The hiker's life was tragically cut short as **asphyxiation** occurred after she became stuck in the narrow cave crevice.
		SNARE	killed	The patient's feverish state was triggered when they tested positive for the virus, which ultimately led to their being **killed** by the rapidly spreading infection.
GENIA	Binding	STAR	merge	The regulatory protein's ability to activate a specific region of the DNA triggers the **merge** of two proteins, leading to the modification of gene expression.
		SNARE	bound	During the phosphorylation of the enzyme, it **bound** to the DNA sequence, initiating the transcription process.

Table 7: Qualitative examples demonstrating STAR and SNARE's trigger and sentence generation quality.

Method	Naturalness	Event Relevance	Annotation Quality
STAR	3.1	3.4	3.1
Weak Sup	**4.2**	-	2.9
SNARE	3.6	**4.0**	**3.6**

Table 8: Human evaluation for sentence naturalness, relevance of event in generated sentence, and the annotation quality. 1 = worst, 5 = best.

Method	ACE		SPEED		GENIA	
	EI	TC	EI	TC	EI	TC
SNARE	**57.4**	50.2	44.6	31.5	35.2	28.9
+ SFT LLM	55.2	**51.7**	**46.9**	**35.8**	**36.7**	**29.1**

Table 9: Measuring performance improvement by fine-tuning an LLM on unlabeled train data for SNARE.

low hit rate indicates the poor overlap with the gold triggers, which is a primary reason for its domain drift. Furthermore, the consistently stronger coverage of SNARE explains its lower domain drift.

To further study the quality and relevance of the label, we performed a human evaluation. Specifically, a human expert in ED is tasked with scoring generations (between 1-5) on the naturalness of the sentence specific to the target domain, the relevance of the event to the semantic actions described in the generated sentence, and the annotation quality evaluating the selection of triggers (details in § C.6). We provide the averaged scores across the three datasets for 90 samples in Table 8.[2] Weak Sup has high sentence quality but poor label annotations; STAR suffers from poor event relevance indicating domain drift. Overall, SNARE performs the best with high annotation quality and event relevance.

7.5 Domain-adapted LLM Fine-tuning

We fine-tune the base LLM for Narrator on the unlabeled target domain train data D'_T to better align the generated passages. Naturally, this can be applied only for smaller LLMs owing to fine-tuning costs. We present the results of fine-tuning Llama3-8B-Instruct on the unlabeled train data in Table 9. On average, we observe that target data fine-tuning additionally improves SNARE by 0.5-2% F1. Qualitative studies indicate that the generated passages are distributionally closer to the target domain, further reducing domain drift.

7.6 Qualitative analysis of generated data

We provide qualitative evidence for SNARE's reduction in domain drift by Scout's domain-specific triggers in Table 7 (more examples in Table 23). We compare with STAR, which uses LLM's internal knowledge to generate the triggers. Lack of domain grounding often results in STAR's triggers and sentences being misaligned (e.g. *asphyxiation* for *death* event related to pandemics) relative to the target domain. In contrast, SNARE's triggers are better aligned to the target domain corpus, resulting in better quality data and reduced domain drift.

8 Conclusion and Future Work

We introduce SNARE, a domain-aware synthetic data generation approach composed of Scout, Narrator, and Refiner. Utilizing Scout's domain-specific triggers for synthesizing sentences, along with Narrator's conditional generation and Refiner's annotations, helps reduce domain drift and label noise. Experiments on three diverse datasets in zero-shot, few-shot, and multilingual settings demonstrate the efficacy of SNARE, establishing SNARE as a strong data generation framework.

[2] Since Weak Sup annotates the unlabeled target domain sentences, event relevance is analogous to annotation quality and we do not explicitly evaluate it.

Acknowledgments

We express our gratitude to Anh Mac, Po-Nien Kung, and Christina Chance for their valuable time, reviews of our work, and constructive feedback. We thank the anonymous reviewers, area chairs, and the program committee for their reviews and feedback. This work was partially supported by the National Science Foundation CAREER award #2339766 and the Amazon AGI Research Award awarded to Nanyun Peng. This work was partially supported by NSF 2200274, 2106859, 2312501, DARPA HR00112290103/HR0011260656, NIH U54HG012517, U24DK097771. We also thank Optum Labs for their support. Nanyun Peng and Kai-Wei Chang are also supported in part by a grant from DARPA to the Simons Institute for the Theory of Computing.

Limitations

We consider only Event Detection (ED) as the main task for data generation, but our method can be extended to other structured prediction tasks as well. We leave this exploration for future works. We consider three specialized domains of news, social media, and biomedical to provide a proof-of-concept of our work. There are other specialized domains for ED as well which can be explored as part of future work. Finally, our proposed method SNARE makes a practical assumption of access to unlabeled data to procure target domain cues to guide the data generation. However, for specific super-specialized domains or if data has privacy concerns, this may not be possible and our method may not be applicable here. We assume such cases to be super rare and beyond the scope of our work.

Ethical Considerations

The theme of our work is to generate high-quality domain-specific data using Large Language Models (LLMs). The inherent LLMs can have certain biases, which can lead to potentially harmful or biased generations. Furthermore, the LLM can introduce potential hallucinations in the annotations, which can hurt the model's performance. We do not check or consider any bias/hallucination detection method as part of our work, as it is beyond the scope. Future works should take due consideration of this vulnerability.

Our proposed method SNARE utilizes unlabeled data as a basis to procure domain-specific cues. If there are any biases in this data, it can propagate to the downstream model as well. We provide a proof-of-concept about our method in this work, but do not detect or rectify such biases.

SNARE's Narrator utilizes LLMs to generate sentences/passages. However, as noticed in past work, LLMs can potentially copy these sentences from the pre-training data on which it has been trained. This can potentially lead to copyright infringements, and we do not consider any such violations under consideration for our method. Users should consider this vulnerability before usage in commercial applications.

We would also like to mention and acknowledge that we have utilized AI chatbots to help with the writing of the work.

References

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

Arthur Amalvy, Vincent Labatut, and Richard Dufour. 2023. Learning to rank context for named entity recognition using a synthetic dataset. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 10372–10382, Singapore. Association for Computational Linguistics.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. *CoRR*, abs/2005.14165.

Zefan Cai, Po-Nien Kung, Ashima Suvarna, Mingyu Ma, Hritik Bansal, Baobao Chang, P. Jeffrey Brantingham, Wei Wang, and Nanyun Peng. 2024. Improving event definition following for zero-shot event detection. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2842–2863, Bangkok, Thailand. Association for Computational Linguistics.

Fanglin Chen, Ta-Chung Chi, Shiyang Lyu, Jianchen Gong, Tanmay Parekh, Rishabh Joshi, Anant Kaushik, and Alexander Rudnicky. 2020. Tartan: A two-tiered dialog framework for multi-domain social chitchat. *Alexa prize proceedings*.

Feng Chen and Yujian Feng. 2023. Chain-of-thought prompt distillation for multimodal named entity and multimodal relation extraction. *arXiv preprint arXiv:2306.14122*.

Yew Ken Chia, Lidong Bing, Soujanya Poria, and Luo Si. 2022. RelationPrompt: Leveraging prompts to generate synthetic data for zero-shot relation triplet extraction. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 45–57, Dublin, Ireland. Association for Computational Linguistics.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale. *CoRR*, abs/1911.02116.

Bosheng Ding, Chengwei Qin, Linlin Liu, Yew Ken Chia, Boyang Li, Shafiq Joty, and Lidong Bing. 2023. Is GPT-3 a good data annotator? In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11173–11195, Toronto, Canada. Association for Computational Linguistics.

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

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurélien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Rozière, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Graeme Nail, Grégoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel M. Kloumann, Ishan Misra, Ivan Evtimov, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, and et al. 2024. The llama 3 herd of models. *CoRR*, abs/2407.21783.

Joe Ellis, Jeremy Getman, Dana Fore, Neil Kuster, Zhiyi Song, Ann Bies, and Stephanie M. Strassel. 2015. Overview of linguistic resources for the TAC KBP 2015 evaluations: Methodologies and results. In *Proceedings of the 2015 Text Analysis Conference, TAC 2015, Gaithersburg, Maryland, USA, November 16-17, 2015, 2015*. NIST.

Enrico Francesconi, Simonetta Montemagni, Wim Peters, and Daniela Tiscornia, editors. 2010. *Semantic Processing of Legal Texts: Where the Language of Law Meets the Law of Language*, volume 6036 of *Lecture Notes in Computer Science*. Springer.

Jun Gao, Huan Zhao, Changlong Yu, and Ruifeng Xu. 2023. Exploring the feasibility of chatgpt for event extraction. *CoRR*, abs/2303.03836.

Ralph Grishman and Beth Sundheim. 1996. Message Understanding Conference- 6: A brief history. In *COLING 1996 Volume 1: The 16th International Conference on Computational Linguistics*.

Xuanli He, Islam Nassar, Jamie Kiros, Gholamreza Haffari, and Mohammad Norouzi. 2021. Generate, annotate, and learn: Generative models advance self-training and knowledge distillation. *CoRR*, abs/2106.06168.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katherine Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Oriol Vinyals, Jack W. Rae, and Laurent Sifre. 2022. An empirical analysis of compute-optimal large language model training. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

I-Hung Hsu, Kuan-Hao Huang, Elizabeth Boschee, Scott Miller, Prem Natarajan, Kai-Wei Chang, and Nanyun Peng. 2022. DEGREE: A data-efficient generation-based event extraction model. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1890–1908, Seattle, United States. Association for Computational Linguistics.

I-Hung Hsu, Kuan-Hao Huang, Shuning Zhang, Wenxin Cheng, Prem Natarajan, Kai-Wei Chang, and Nanyun Peng. 2023a. TAGPRIME: A unified framework for relational structure extraction. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12917–12932, Toronto, Canada. Association for Computational Linguistics.

I-Hung Hsu, Zifeng Wang, Long Le, Lesly Miculicich, Nanyun Peng, Chen-Yu Lee, and Tomas Pfister. 2024. CaLM: Contrasting large and small language models to verify grounded generation. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 12782–12803, Bangkok, Thailand. Association for Computational Linguistics.

I-Hung Hsu, Zhiyu Xie, Kuan-Hao Huang, Prem Natarajan, and Nanyun Peng. 2023b. AMPERE: AMR-aware prefix for generation-based event argument extraction model. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10976–10993, Toronto, Canada. Association for Computational Linguistics.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *CoRR*, abs/2106.09685.

Xuming Hu, Yong Jiang, Aiwei Liu, Zhongqiang Huang, Pengjun Xie, Fei Huang, Lijie Wen, and Philip S. Yu. 2023. Entity-to-text based data augmentation for various named entity recognition tasks. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 9072–9087, Toronto, Canada. Association for Computational Linguistics.

Kuan-Hao Huang, I-Hung Hsu, Prem Natarajan, Kai-Wei Chang, and Nanyun Peng. 2022. Multilingual generative language models for zero-shot cross-lingual event argument extraction. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4633–4646, Dublin, Ireland. Association for Computational Linguistics.

Kuan-Hao Huang, I-Hung Hsu, Tanmay Parekh, Zhiyu Xie, Zixuan Zhang, Prem Natarajan, Kai-Wei Chang, Nanyun Peng, and Heng Ji. 2024. TextEE: Benchmark, reevaluation, reflections, and future challenges in event extraction. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 12804–12825, Bangkok, Thailand. Association for Computational Linguistics.

Lifu Huang, Heng Ji, Kyunghyun Cho, Ido Dagan, Sebastian Riedel, and Clare Voss. 2018. Zero-shot transfer learning for event extraction. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2160–2170, Melbourne, Australia. Association for Computational Linguistics.

Martin Josifoski, Marija Sakota, Maxime Peyrard, and Robert West. 2023. Exploiting asymmetry for synthetic training data generation: SynthIE and the case of information extraction. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 1555–1574, Singapore. Association for Computational Linguistics.

Jin-Dong Kim, Yue Wang, Toshihisa Takagi, and Akinori Yonezawa. 2011. Overview of Genia event task in BioNLP shared task 2011. In *Proceedings of BioNLP Shared Task 2011 Workshop*, pages 7–15, Portland, Oregon, USA. Association for Computational Linguistics.

Jin-Dong Kim, Yue Wang, and Yamamoto Yasunori. 2013. The Genia event extraction shared task, 2013 edition - overview. In *Proceedings of the BioNLP Shared Task 2013 Workshop*, pages 8–15, Sofia, Bulgaria. Association for Computational Linguistics.

Duong Minh Le, Yang Chen, Alan Ritter, and Wei Xu. 2024. Constrained decoding for cross-lingual label projection. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Kenton Lee, Kelvin Guu, Luheng He, Tim Dozat, and Hyung Won Chung. 2021. Neural data augmentation via example extrapolation. *CoRR*, abs/2102.01335.

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

Bo Li, Gexiang Fang, Yang Yang, Quansen Wang, Wei Ye, Wen Zhao, and Shikun Zhang. 2023a. Evaluating chatgpt's information extraction capabilities: An assessment of performance, explainability, calibration, and faithfulness. *CoRR*, abs/2304.11633.

Manling Li, Alireza Zareian, Qi Zeng, Spencer Whitehead, Di Lu, Heng Ji, and Shih-Fu Chang. 2020. Cross-media structured common space for multimedia event extraction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2557–2568, Online. Association for Computational Linguistics.

Sha Li, Heng Ji, and Jiawei Han. 2021. Document-level event argument extraction by conditional generation. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 894–908, Online. Association for Computational Linguistics.

Zhuoyan Li, Hangxiao Zhu, Zhuoran Lu, and Ming Yin. 2023b. Synthetic data generation with large language models for text classification: Potential and limitations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 10443–10461, Singapore. Association for Computational Linguistics.

Rensis Likert. 1932. A technique for the measurement of attitudes. *Archives of Psychology*.

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. 2024. STAR: boosting low-resource information extraction by structure-to-text data generation with large language models. In *Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2024, Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence, IAAI 2024, Fourteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2014, February 20-27, 2024, Vancouver, Canada*, pages 18751–18759. AAAI Press.

Sourab Mangrulkar, Sylvain Gugger, Lysandre Debut, Younes Belkada, Sayak Paul, and Benjamin Bossan. 2022. Peft: State-of-the-art parameter-efficient fine-tuning methods. `https://github.com/huggingface/peft`.

Ryan* Marten, Trung* Vu, Charlie Cheng-Jie Ji, Kartik Sharma, Shreyas Pimpalgaonkar, Alex Dimakis, and Maheswaran Sathiamoorthy. 2025. Curator: A Tool for Synthetic Data Creation. `https://github.com/bespokelabsai/curator`.

Mike Mintz, Steven Bills, Rion Snow, and Daniel Jurafsky. 2009. Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 1003–1011, Suntec, Singapore. Association for Computational Linguistics.

Rishabh Misra. 2022. News category dataset. *CoRR*, abs/2209.11429.

Tanmay Parekh, Emily Ahn, Yulia Tsvetkov, and Alan W Black. 2020. Understanding linguistic accommodation in code-switched human-machine dialogues. In *Proceedings of the 24th Conference on Computational Natural Language Learning*, pages 565–577, Online. Association for Computational Linguistics.

Tanmay Parekh, I-Hung Hsu, Kuan-Hao Huang, Kai-Wei Chang, and Nanyun Peng. 2023. GENEVA: Benchmarking generalizability for event argument extraction with hundreds of event types and argument roles. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3664–3686, Toronto, Canada. Association for Computational Linguistics.

Tanmay Parekh, I-Hung Hsu, Kuan-Hao Huang, Kai-Wei Chang, and Nanyun Peng. 2024a. Contextual label projection for cross-lingual structured prediction. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 5738–5757, Mexico City, Mexico. Association for Computational Linguistics.

Tanmay Parekh, Jeffrey Kwan, Jiarui Yu, Sparsh Johri, Hyosang Ahn, Sreya Muppalla, Kai-Wei Chang, Wei Wang, and Nanyun Peng. 2024b. SPEED++: A multilingual event extraction framework for epidemic prediction and preparedness. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 12936–12965, Miami, Florida, USA. Association for Computational Linguistics.

Tanmay Parekh, Anh Mac, Jiarui Yu, Yuxuan Dong, Syed Shahriar, Bonnie Liu, Eric Yang, Kuan-Hao Huang, Wei Wang, Nanyun Peng, and Kai-Wei Chang. 2024c. Event detection from social media for epidemic prediction. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 5758–5783, Mexico City, Mexico. Association for Computational Linguistics.

Tanmay Parekh, Kartik Mehta, Ninareh Mehrabi, Kai-Wei Chang, and Nanyun Peng. 2025a. Dicore: Enhancing zero-shot event detection via divergent-convergent LLM reasoning. *CoRR*, abs/2506.05128.

Tanmay Parekh, Pradyot Prakash, Alexander Radovic, Akshay Shekher, and Denis Savenkov. 2025b. Dynamic strategy planning for efficient question answering with large language models. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 6038–6059, Albuquerque, New Mexico. Association for Computational Linguistics.

Amir Pouran Ben Veyseh, Javid Ebrahimi, Franck Dernoncourt, and Thien Nguyen. 2022. MEE: A novel multilingual event extraction dataset. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9603–9613, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Sampo Pyysalo, Tomoko Ohta, Makoto Miwa, Han-Cheol Cho, Junichi Tsujii, and Sophia Ananiadou. 2012. Event extraction across multiple levels of biological organization. *Bioinform.*, 28(18):575–581.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Taneeya Satyapanich, Francis Ferraro, and Tim Finin. 2020. CASIE: extracting cybersecurity event information from text. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial*

Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020, pages 8749–8757. AAAI Press.

Timo Schick and Hinrich Schütze. 2021. Generating datasets with pretrained language models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6943–6951, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Yunfan Shao, Linyang Li, Yichuan Ma, Peiji Li, Demin Song, Qinyuan Cheng, Shimin Li, Xiaonan Li, Pengyu Wang, Qipeng Guo, Hang Yan, Xipeng Qiu, Xuanjing Huang, and Dahua Lin. 2025. Case2Code: Scalable synthetic data for code generation. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 11056–11069, Abu Dhabi, UAE. Association for Computational Linguistics.

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 3rd Workshop on EVENTS: Definition, Detection, Coreference, and Representation*, pages 89–98, Denver, Colorado. Association for Computational Linguistics.

Zirui Song, Bin Yan, Yuhan Liu, Miao Fang, Mingzhe Li, Rui Yan, and Xiuying Chen. 2025. Injecting domain-specific knowledge into large language models: A comprehensive survey. *CoRR*, abs/2502.10708.

Zhaoyue Sun, Jiazheng Li, Gabriele Pergola, Byron Wallace, Bino John, Nigel Greene, Joseph Kim, and Yulan He. 2022. PHEE: A dataset for pharmacovigilance event extraction from text. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5571–5587, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Beth M. Sundheim. 1992. Overview of the fourth Message Understanding Evaluation and Conference. In *Fourth Message Understanding Conference (MUC-4): Proceedings of a Conference Held in McLean, Virginia, June 16-18, 1992*.

Ashima Suvarna, Xiao Liu, Tanmay Parekh, Kai-Wei Chang, and Nanyun Peng. 2024. QUDSELECT: Selective decoding for questions under discussion parsing. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 1288–1299, Miami, Florida, USA. Association for Computational Linguistics.

Ruixiang Tang, Xiaotian Han, Xiaoqian Jiang, and Xia Hu. 2023. Does synthetic data generation of llms help clinical text mining? *CoRR*, abs/2303.04360.

MeiHan Tong, Bin Xu, Shuai Wang, Meihuan Han, Yixin Cao, Jiangqi Zhu, Siyu Chen, Lei Hou, and Juanzi Li. 2022. DocEE: A large-scale and fine-grained benchmark for document-level event extraction. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3970–3982, Seattle, United States. Association for Computational Linguistics.

Leandro von Werra, Younes Belkada, Lewis Tunstall, Edward Beeching, Tristan Thrush, Nathan Lambert, Shengyi Huang, Kashif Rasul, and Quentin Gallouédec. 2020. Trl: Transformer reinforcement learning. `https://github.com/huggingface/trl`.

Qing Wang, Kang Zhou, Qiao Qiao, Yuepei Li, and Qi Li. 2023a. Improving unsupervised relation extraction by augmenting diverse sentence pairs. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12136–12147, Singapore. Association for Computational Linguistics.

Xiaozhi Wang, Ziqi Wang, Xu Han, Wangyi Jiang, Rong Han, Zhiyuan Liu, Juanzi Li, Peng Li, Yankai Lin, and Jie Zhou. 2020. MAVEN: A Massive General Domain Event Detection Dataset. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1652–1671, Online. Association for Computational Linguistics.

Xingyao Wang, Sha Li, and Heng Ji. 2023b. Code4Struct: Code generation for few-shot event structure prediction. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3640–3663, Toronto, Canada. Association for Computational Linguistics.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2023c. Self-instruct: Aligning language models with self-generated instructions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13484–13508, Toronto, Canada. Association for Computational Linguistics.

Zirui Wang, Adams Wei Yu, Orhan Firat, and Yuan Cao. 2021. Towards zero-label language learning. *CoRR*, abs/2109.09193.

Likang Wu, Zhi Zheng, Zhaopeng Qiu, Hao Wang, Hongchao Gu, Tingjia Shen, Chuan Qin, Chen Zhu, Hengshu Zhu, Qi Liu, Hui Xiong, and Enhong Chen. 2024. A survey on large language models for recommendation. *World Wide Web (WWW)*, 27(5):60.

Sang Michael Xie, Shibani Santurkar, Tengyu Ma, and Percy Liang. 2023. Data selection for language models via importance resampling. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Derong Xu, Wei Chen, Wenjun Peng, Chao Zhang, Tong Xu, Xiangyu Zhao, Xian Wu, Yefeng Zheng, and Enhong Chen. 2023. Large language models for generative information extraction: A survey. *CoRR*, abs/2312.17617.

Jiacheng Ye, Jiahui Gao, Qintong Li, Hang Xu, Jiangtao Feng, Zhiyong Wu, Tao Yu, and Lingpeng Kong. 2022. ZeroGen: Efficient zero-shot learning via dataset generation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 11653–11669, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Hongming Zhang, Haoyu Wang, and Dan Roth. 2021. 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.

Weiyan Zhang, Wanpeng Lu, Jiacheng Wang, Yating Wang, Lihan Chen, Haiyun Jiang, Jingping Liu, and Tong Ruan. 2024. Unexpected phenomenon: LLMs' spurious associations in information extraction. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 9176–9190, Bangkok, Thailand. Association for Computational Linguistics.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, Zheyan Luo, Zhangchi Feng, and Yongqiang Ma. 2024. Llamafactory: Unified efficient fine-tuning of 100+ language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, Bangkok, Thailand. Association for Computational Linguistics.

Shi Zong, Ashutosh Baheti, Wei Xu, and Alan Ritter. 2022. Extracting a knowledge base of COVID-19 events from social media. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 3810–3823, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

A Data Statistics

We discuss details about our dataset in § 5. Our test target domain data includes the test data splits of (1) ACE (Doddington et al., 2004) in the news domain, (2) SPEED (Parekh et al., 2024c) in the social media domain, and (3) GENIA (Kim et al., 2011) in the biomedical domain For unlabeled data, we utilize the training data of each dataset as one data source. For the other data source, we utilize data from external sources, specifically: (1) News Category Dataset (HuffPost) (Misra, 2022) comprising Huffpost news articles from 2012-2022 for ACE. We filter articles corresponding to political, financial, and business articles, (2) COVIDKB (Zong et al., 2022) mining tweets from the Twitter COVID-19 Endpoint released in April 2020 as the external data source, (3) GENIA2013 dataset (Kim et al., 2013) as the external data for GENIA. Finally, we also provide some statistics about the multilingual splits of the ACE dataset utilized for the Arabic and Chinese zero-shot experiments.[3] We provide statistics about this data in Table 10.

Data Source	# Sents	# Event Mentions	Average Length
Test Data			
ACE - test	832	403	22.9
SPEED - test	586	672	28.1
GENIA - test	2,151	1,805	29.7
Unlabeled Train Data			
ACE - train	17,172	-	15.6
SPEED - train	1,601	-	33.5
GENIA - train	6,431	-	30.1
Unlabeled External Data			
HuffPost	43,350	-	17.4
COVIDKB	7,311	-	30.6
GENIA2013	6,542	-	17.4
Multilingual Test Data			
ACE - Arabic	313	198	24.6
ACE - Chinese	486	211	44.2
Unlabeled Multlingual Train Data			
ACE - Arabic	3,218	-	26.1
ACE - Chinese	6,301	-	45.5

Table 10: Data Statistics for the various test and unlabeled datasets used in our work. # = Number of.

B Implementation Details

Here, we provide detailed implementation details for each component and the models used in our work. We run most of our experiments on NVIDIA RTX A6000/A100 machines with support for 8 GPUs, while for GPT3.5, we make API calls through OpenAI using Curator (Marten et al., 2025).

B.1 LLM-based Generation

We provide details on the various hyperparameters for using LLMs in all the components of STAR and SNARE. For Llama3-8B-Instruct and Llama3-70B-Instruct, we present the hyperparameters in Table 11; while Table 12 presents the hyperparameters for GPT3.5.

Batch Size	32
Temperature	0.6
Top-p	0.9
Max Generation Length	250

Table 11: Hyperparameters for decoding using Llama3-8B/70B model.

Base LLM	gpt-3.5-turbo-0125
Temperature	1.0
Top-p	1.0
Max Generation Length	500

Table 12: Hyperparameters for decoding using GPT3.5 model.

B.2 Few-shot Implementation Details

For the few-shot setting, we can access additional k datapoints per event type to aid better performance. For LLM-based prompting, we simply add these examples in the prompt as in-context examples to help the model do better reasoning/generation. For STARand SNARE, we do not add the k triggers to the trigger list, as it led to a drop in model performance. This can be attributed to the presence of duplicate information, as the trigger generation/extraction already accounts for these gold triggers. We also append the k datapoints to the synthetically generated data to provide signals from the gold data.

B.3 Downstream Model Training

We choose DEGREE (Hsu et al., 2022) as our downstream model for evaluation, a generation-based prompting model that utilizes natural language templates. We implemented the DEGREE model under the TextEE framework (Huang et al., 2024). Table 13 presents the primary hyperparameters for this model.

[3] For Chinese, the average length indicates the average number of characters.

Pre-trained LM	BART-Large
Training Epochs	25
Warmup Epochs	5
Training Batch Size	32
Eval Batch Size	32
Learning Rate	0.00001
Weight Decay	0.00001
Gradient Clipping	5
Beam Size	1
Negative Samples	15
Max Sequence Length	250
Max Output Length	20

Table 13: Hyperparameters for DEGREE model.

B.4 LLM Fine-tuning

We discuss domain-adapted passage generation through LLM fine-tuning in § 7.5. Specifically, we conduct a low-rank finetuning (LoRA) (Hu et al., 2021) to reduce computational overhead to fine-tune Llama3-8B-Instruct. We implement LoRA using the `peft` and `trl` packages (Mangrulkar et al., 2022; von Werra et al., 2020). We choose the task of causal language modeling (i.e., continual pre-training) to perform domain adaptation on unlabeled in-domain sentences. We utilize cross-entropy loss on the dev split of the unlabeled data to select the best model. We provide additional details about the hyperparameters for this fine-tuning for each dataset in Table 14 below.

ACE	
Lora Rank	32
Lora Alpha	16
Lora Dropout	0.1
Learning Rate	0.0001
Weight Decay	0.05
Training Batch Size	32
Training Epochs	3
Eval Steps	20
SPEED	
Lora Rank	32
Lora Alpha	16
Lora Dropout	0.1
Learning Rate	0.00008
Weight Decay	0.05
Training Batch Size	32
Training Epochs	10
Eval Steps	20
GENIA	
Lora Rank	32
Lora Alpha	16
Lora Dropout	0.1
Learning Rate	0.00008
Weight Decay	0.05
Training Batch Size	32
Training Epochs	6
Eval Steps	20

Table 14: Hyperparameters for LoRA fine-tuning Llama3-8B-Instruct.

C Additional analyses

In this section, we provide additional analyses to support our main experiments.

C.1 STAR with domain-specific prompt

A simple way to infuse domain-specific information in past works like STAR would be to add domain-related information in the prompts to the LLM. We experiment with two such methods: (1) domain-mention, where we provide the target domain information in the prompt and ask the model to generate accordingly, and (2) domain-reference, where we use some examples from the unlabeled data in the prompt as reference sentences to better guide the passage generation. We provide results for these explorations using the Llama3-8B-Insturct model in Table 16. As observed, the results are generally poor, with an average drop of 0.1-0.6% F1 for domain-mention and 3.1-3.8% F1 for domain-reference. This is mainly because LLMs over-compensate, producing longer and more stereotypical information in their generations, which hurts the naturalness of the sentence and causes further domain drift. Furthermore, the LLM makes more errors in mentioning the event as a part of its reasoning, which is utilized to make the generation in the domain style. We provide some qualitative examples for such generations in Table 18. In some ways, it also puts into light and amplifies the gains obtained by doing target domain SFT for SNARE as discussed in § 7.5.

Method	ACE		SPEED		GENIA	
	EI	TC	EI	TC	EI	TC
STAR	**44.9**	**35.0**	**21.0**	10.1	25.9	19.0
+ mention	44.1	32.9	17.1	**10.3**	**28.7**	**20.4**
+ references	35.5	27.3	19.0	9.2	25.8	18.1

Table 15: Measuring model performance improvement providing domain-specific cues in the form of domain-mention (mention) or domain sentence references (references) to the LLM for STAR. EI: Event Identification F1, TC: Trigger Classification F1.

C.2 Impact of different number of training samples

We perform a small analysis to study the impact of changing the number of generated samples on the downstream model performance for SNARE. We present the results for Llama3-8B-Instruct in

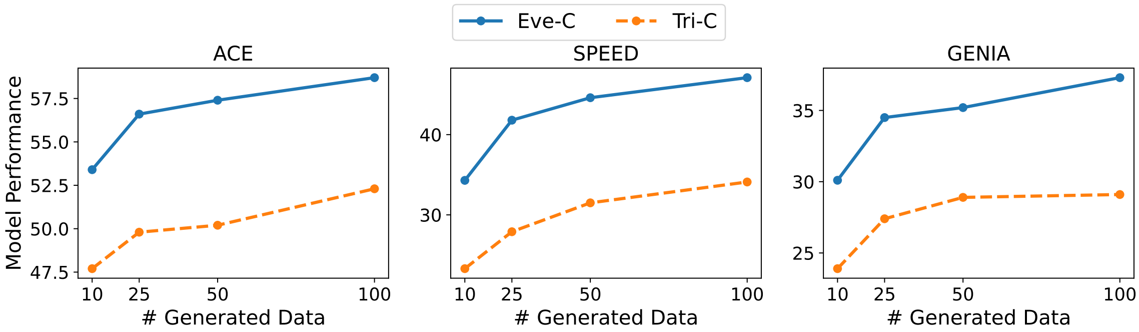


Figure 9: Model performance for SNARE as keep change the number of generated datapoints N using Llama3-8B-Instruct for the three datasets.

Method	**Original ACE**		**Synthetic ACE**	
	EI	**TC**	**EI**	**TC**
LLM Training	71.5	67.4	54.3	46.8
Small Model Training	72.2	68.9	57.4	50.2

Table 16: Measuring model performance improvement providing domain-specific cues in the form of domain-mention (mention) or domain sentence references (references) to the LLM for STAR. EI: Event Identification F1, TC: Trigger Classification F1.

Number of Triggers	**ACE**		**SPEED**		**GENIA**	
	EI	**TC**	**EI**	**TC**	**EI**	**TC**
$t = 5$	55.9	48.9	43.2	29.9	32.2	26.5
$t = 8$	**57.9**	49.7	**45.1**	30.8	**35.2**	**28.9**
$t = 10$	57.4	**50.2**	44.6	**31.5**	34.8	28.1
$t = 12$	56.1	49.3	44.3	31.2	33.9	27.4

Table 17: Ablating the impact of the number of triggers for the Scout component on downstream model performance across the three datasets. EI: Event Identification F1, TC: Trigger Classification F1.

Figure 9. As observed, performance continues to increase as we increase the data from $N = 10$ to $N = 100$ datapoints per event type. This promises that data generation will provide continued improvements by practicing greater and better control over the data distribution.

C.3 Training LLMs with synthetic data

Since we generate synthetic data using LLMs, it would be natural to fine-tune LLMs to be better at Event Detection (ED). We conduct a small experiment to test and compare the model training for LLMs and small models. Specifically, we train a Llama3-8B-Instruct model using Low Rank Adaptation (LoRA) (Hu et al., 2021) and compare it with the best performing small models like DEGREE (Hsu et al., 2022) and TagPrime (Hsu et al., 2023a). We conduct sets of training: (1) Original ACE training data, and (2) Synthetic data generated by SNARE for the ACE domain. We summarize our results and findings in Table **??**. Overall, we note how our synthetic data can be used to train LLMs and improve their performance iteratively as well. However, we would like to note how small language model still learn better and provide better scores than LLMs on this task. We would also like to note that our LLM training can be suboptimal, as we tried using the original hyperparameters and didn't tune the hyperparameters much. We utilized Llama-Factory (Zheng et al., 2024) for our LLM fine-tuning.

C.4 Ablating the design choice for number of triggers

We utilize $t = 10$ for ACE and SPEED and $t = 8$ for GENIA. Here, we provide additional experiments to justify this design choice. Specifically, we provide the performance of training smaller models using different number of triggers (t) for the Scout using Llama3-8B-Instruct LLM in Table 17. Overall, we observe that changing the number of triggers affects the performance slightly, and can be optimally chosen to improve performance. However, we note that even with different t, SNARE still outperforms the other baselines.

C.5 Ablating the design choice for trigger selection

We utilize frequency-based trigger selection in the Scout, but this can lead to missing out on low-frequency, long-tail, rare triggers. Our major motive is to avoid the noisy triggers which would also have low frequency. In order to justify this design choice, we conduct experiments with various other trigger selection designs, specifically: (1) Sam-

ACE
A 35-year-old cyclist was hit by a speeding car while riding to work, leaving her with severe injuries, while in a separate incident, a local retail giant filed a petition to restructure its debt, sparking concerns about its financial stability.
As the war on terror raged on, the Mujahideen Advisory Council distributed a statement inviting Arab and foreign media reporters to enter Fallujah and cover the battles, while simultaneously, the ownership of the ancient artifacts was transferred to the museum, with the landlord demanding rent on the premises.
SPEED
As the influencer's viral challenge went viral, her followers were suddenly struck with a mysterious illness after the splash of a contaminated drink, leading to a shocking explosion of fatalities on social media.
As the community struggled to come to terms with the devastating accident that had claimed the lives of several residents, the authorities swiftly implemented a strict quarantine to prevent the spread of the infectious disease, hoping to mitigate the tragedy.
GENIA
The specific transcription factor was elevated by the presence of the hormone, thereby increasing the expression of the target gene, while the inhibitory protein curbed the activity of a competing transcription factor, preventing the expression of a repressor gene.
The binding of PEBP2/CBF to the promoter region boosts the expression of the gene, which turns on the production of a crucial cytokine in response to the immune response.

Table 18: Example passages of overly long and more stereotypical sentences generated when the domain is mentioned or references are added to the LLM prompt for STAR.

Trigger Selection	**EI**	**TC**
Frequency Ranking (current)	**57.4**	**50.2**
Sampling	54.7	47.3
Weighted Sampling	55.2	48.1
Reranking	49.6	37.9
Minimum Count Filtering	52.4	44.8
Clustering	48.0	40.7

Table 19: Ablating downstream model performance on the trigger selection strategy for the Scout component on the ACE dataset. EI: Event Identification F1, TC: Trigger Classification F1.

Sentence	**Score**
The sudden crash of the ambulance sent shockwaves through the hospital as medical staff rushed to the scene to monitor the patient's life signs, but it was too late, as the patient succumbed to the infectious disease.	SN: 2 ER: 1 AQ: 1
The wealthy entrepreneur transferred ownership of the struggling tech company to her trusted business partner, relinquishing control and financial responsibility	SN: 5 ER: 5 AQ: 5
Taken together, these data suggest that Id1 could be a possible target gene for mediating the effects of BMP-6 in human B cells, whereas Id2 and Id3 not seem to be involved.	SN: 4 ER: 3 AQ: 2

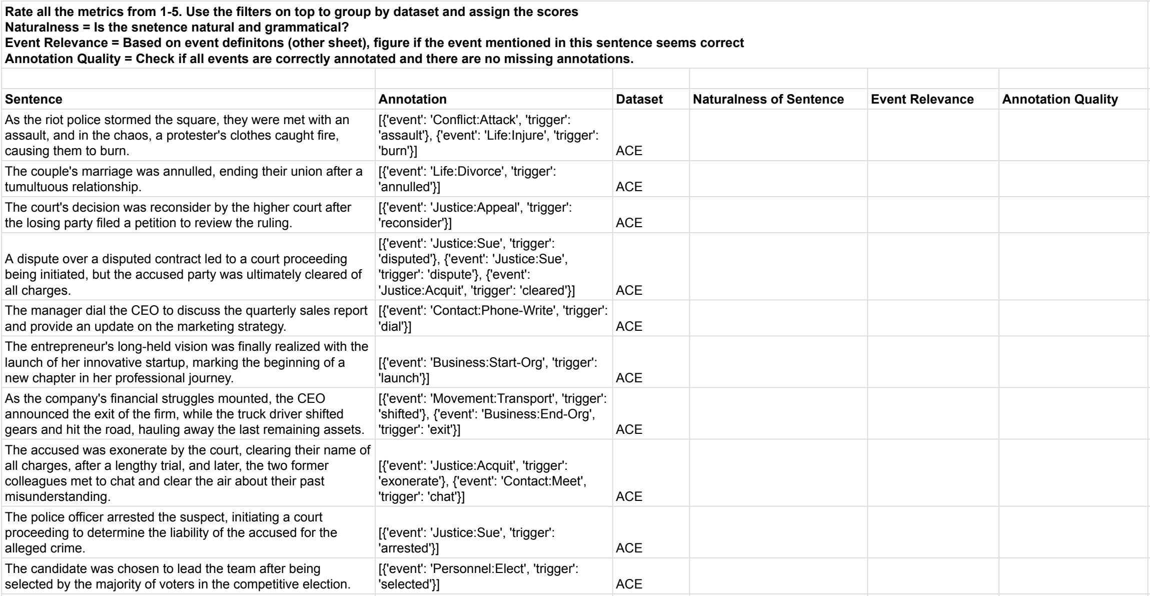
Table 20: Illustration examples for the human evaluation metrics. SN: sentence naturalness, ER: event relevance, AQ: annotation quality.

pling: Instead of filtering and selecting, we sample uniformly from the trigger list, (2) Weighted Sampling: We sample using the extraction frequency of the triggers as the weights, (3) Reranking: Since some triggers might not be extracted correctly, we rerank the triggers based on the number of occurrences in holdout set (basically avoiding the bias of extracting them using the LLM), (4) Minimum Count Filtering: Instead of filtering based on ranks, we simply remove all triggers with a set minimum count (as this long tail might be noisy) and sample from the remaining triggers, (5) Clustering: We use k-means clustering to form clusters of similar triggers and sample one trigger from each - thus, providing enhanced diversity. We compare these various methods for trigger selection with the Scout using Llama3-8B-Instruct LLM on the ACE dataset in Table 19. Overall, this study shows how the different methods, which introduce more noise and provide different ranges of the precision-recall balance, are worse in comparison to our existing frequency ranking method.

C.6 Human Evaluation Details

We conduct a small human evaluation to judge the quality of the synthetic data in § 7.4. Here, we provide additional details about the human study and evaluation. Since the evaluation is conducted on three diverse and niche domains, we only utilize a single human annotator who is an ED expert and has previously worked on all three datasets as the primary annotator.

We majorly evaluate on three dimensions: (1) Sentence naturalness (SN): This metric judges whether the sentence seems grammatical, natural, and fits the domain of the target data. (2) Event Relevance (ER): This metric is computed only for generation methods that generate sentences from labels. This evaluation judges whether the sampled event and trigger are appropriately used to generate a sensible alignment with the target domain. Furthermore, it is verified if the right event defi-



Rate all the metrics from 1-5. Use the filters on top to group by dataset and assign the scores
Naturalness = Is the snetence natural and grammatical?
Event Relevance = Based on event definitons (other sheet), figure if the event mentioned in this sentence seems correct
Annotation Quality = Check if all events are correctly annotated and there are no missing annotations.

Sentence	Annotation	Dataset	Naturalness of Sentence	Event Relevance	Annotation Quality
As the riot police stormed the square, they were met with an assault, and in the chaos, a protester's clothes caught fire, causing them to burn.	[{'event': 'Conflict:Attack', 'trigger': 'assault'}, {'event': 'Life:Injure', 'trigger': 'burn'}]	ACE			
The couple's marriage was annulled, ending their union after a tumultuous relationship.	[{'event': 'Life:Divorce', 'trigger': 'annulled'}]	ACE			
The court's decision was reconsider by the higher court after the losing party filed a petition to review the ruling.	[{'event': 'Justice:Appeal', 'trigger': 'reconsider'}]	ACE			
A dispute over a disputed contract led to a court proceeding being initiated, but the accused party was ultimately cleared of all charges.	[{'event': 'Justice:Sue', 'trigger': 'disputed'}, {'event': 'Justice:Sue', 'trigger': 'dispute'}, {'event': 'Justice:Acquit', 'trigger': 'cleared'}]	ACE			
The manager dial the CEO to discuss the quarterly sales report and provide an update on the marketing strategy.	[{'event': 'Contact:Phone-Write', 'trigger': 'dial'}]	ACE			
The entrepreneur's long-held vision was finally realized with the launch of her innovative startup, marking the beginning of a new chapter in her professional journey.	[{'event': 'Business:Start-Org', 'trigger': 'launch'}]	ACE			
As the company's financial struggles mounted, the CEO announced the exit of the firm, while the truck driver shifted gears and hit the road, hauling away the last remaining assets.	[{'event': 'Movement:Transport', 'trigger': 'shifted'}, {'event': 'Business:End-Org', 'trigger': 'exit'}]	ACE			
The accused was exonerate by the court, clearing their name of all charges, after a lengthy trial, and later, the two former colleagues met to chat and clear the air about their past misunderstanding.	[{'event': 'Justice:Acquit', 'trigger': 'exonerate'}, {'event': 'Contact:Meet', 'trigger': 'chat'}]	ACE			
The police officer arrested the suspect, initiating a court proceeding to determine the liability of the accused for the alleged crime.	[{'event': 'Justice:Sue', 'trigger': 'arrested'}]	ACE			
The candidate was chosen to lead the team after being selected by the majority of voters in the competitive election.	[{'event': 'Personnel:Elect', 'trigger': 'selected'}]	ACE			

Figure 10: Illustration of the interface for the human evaluation of the synthetically generated data. Short instructions are provided at the top. Each query comprises the sentence, annotation, and dataset. The human annotator is expected to score 1-5 for each of the three metrics on the right.

Method	# Unlabeled Data	Eve-I	Tri-C
Inference	-	46.9	41.3
STAR	-	50.0	42.3
Weak Sup	100% train	53.2	48.0
Weak Sup	20% train	51.4	46.1
Weak Sup	5% train	50.1	44.9
SNARE	100% train	58.1	53.8
SNARE	20% train	55.9	52.1
SNARE	5% train	**55.1**	**51.2**

Table 21: Ablating the amount of unlabeled data utilized for the ACE dataset by the different data generation methods using Llama3-70B-Instruct and its impact on downstream model performance.

nition is used. (3) Annotation Quality (AQ): This metric judges if the right trigger is used for each event mentioned in the synthetic output. If there are any missing events, then this score is penalized. For each metric, a score is given on a Likert scale (Likert, 1932) from 1 (worst) to 5 (best). We also provide event definitions for each event in each dataset as a reference for better judgment. We illustrate the annotation interface in Figure 10 and provide some sample examples in Table 20.

C.7 Analysis in low-resource unlabeled data settings

In rare cases of hyper-specialized domains or domains with privacy concerns, there is a possibility of having super-low availability of unlabeled data. We believe that our current experimentation already simulates this setting, where we use only 773 documents for GENIA and 1.6k tweets for SPEED. Secondly, by providing strong efficacy of external data relative to in-distribution data, we demonstrate how even mildly related unlabeled data can also work well for SNARE. Finally, we provide a deeper study of varying the amount of unlabeled data for the ACE dataset for Llama3-70B-Instruct as an ablative study to further substantiate the efficacy of our method in Table 21. Lowering the number of unlabeled data reduces the model performance relative to using higher number of data samples. However, even with 5% unlabeled training data (∼850 samples), SNARE still outperforms the other baselines, showing the efficacy of our method.

Method	Arabic		Chinese	
	EI	TC	EI	TC
Inference	33.0	18.7	25.0	21.0
STAR	28.1	22.0	32.3	25.9
Weak Sup	29.4	20.2	30.2	28.4
SNARE (ours)	**38.4**	**31.9**	**39.6**	**29.0**

Table 22: Comparing SNARE with zero-shot inference for Qwen3-8B for the ACE dataset.

C.8 Additional zero-shot multilingual experiments

In § 6.3, we demonstrated the utility of our framework for zero-shot multilingual setting using the

Llama3-8B and Llama3-70B models. Here, we additionally show results using the Qwen3-8B LLM in Table 22. We note that SNARE outperforms the other baselines for both languages and metrics, even with the Qwen3-8B LLM.

C.9 Additional Qualitative Examples

In § 7.6, we discussed how SNARE improves domain drift qualitatively relative to STARand provided some examples. Here, we provide more examples to further support that study in Table 23. This table further demonstrates how STAR can have a domain drift owing to a lack of domain-specific cues, while SNARE is better grounded in the target domain.

Dataset	**Event**	**Method**	**Trigger**	**Sentence**
ACE	Attack	STAR	raid	As the rebels embarked on a daring trek across the desert, they launched a surprise **raid** on the heavily guarded fortress, catching the enemy off guard.
			siege	The government issued fines to the rebel group, while the city was under **siege**, with gunfire erupting throughout the streets.
			slay	The wealthy businessman filed a complaint against his former partner, leading to a court proceeding, just as the rival gang leader **slays** his arch-nemesis in a brutal battle.
		SNARE	shooting	As the rival businessman signed the contract, a sudden **shooting** erupted outside, causing chaos in the midst of the transaction.
			bombing	As the city was rocked by a devastating **bombing**, thousands of protesters took to the streets to demand justice and peace.
			fight	The rival gangs engaged in a brutal **fight**, sparking a wave of gunfire that left several people injured and buildings damaged.
SPEED	Death	STAR	asphyxiation	The hiker's life was tragically cut short as **asphyxiation** occurred after she became stuck in the narrow cave crevice.
			overdose	A young musician's **overdose** became the tragic finale of a life cut short in the midst of chaos.
			drowning	As she struggled to stay afloat, her heart stopped beating, and she succumbed to **drowning** in the icy waters.
		SNARE	killed	The patient's feverish state was triggered when they tested positive for the virus, which ultimately led to their being **killed** by the rapidly spreading infection.
			lost	As the pandemic spread rapidly across the globe, thousands of people **lost** their lives due to the deadly virus.
			died	The elderly man, who had been suffering from a severe case of tuberculosis, **died** in his sleep.
GENIA	Binding	STAR	merge	The regulatory protein's ability to activate a specific region of the DNA molecule triggers the **merge** of two proteins, leading to the modification of gene expression.
			fuse	When the proteins **fuse** together, the activity of the transcription factor is inhibited, preventing the gene expression from proceeding.
			snap	When the two proteins **snap** together, the binding of the complex inhibits the expression of the target gene by deactivating a specific region of the DNA molecule.
		SNARE	bound	During the phosphorylation of the enzyme, it **bound** to the DNA sequence, initiating the transcription process.
			translocation	The protein **translocation** to the nucleus triggers the induction of gene expression.
			binds	When the enzyme **binds** to the substrate, it activates the addition of a phosphate group to the target molecule, marking a crucial change in its function.

Table 23: Comparison of generated triggers and sentences from STAR and SNARE methods