Is a Large Language Model a Good Annotator for Event Extraction?

Ruirui Chen[1], Chengwei Qin[2], Weifeng Jiang[2], Dongkyu Choi[1]
[1]Institute of High Performance Computing (IHPC),
Agency for Science, Technology and Research (A*STAR)
1 Fusionopolis Way, #16-16 Connexis, Singapore 138632, Republic of Singapore
[2]School of Computer Science and Engineering, Nanyang Technological University
50 Nanyang Avenue, Singapore 639798, Republic of Singapore
chen_ruirui@ihpc.a-star.edu.sg, {chengwei003, s220077}@e.ntu.edu.sg, choi_dongkyu@ihpc.a-star.edu.sg

Abstract

Event extraction is an important task in natural language processing that focuses on mining event-related information from unstructured text. Despite considerable advancements, it is still challenging to achieve satisfactory performance in this task, and issues like data scarcity and imbalance obstruct progress. In this paper, we introduce an innovative approach where we employ large language models (LLMs) as expert annotators for event extraction. We strategically include sample data from the training dataset in the prompt as a reference, ensuring alignment between the data distribution of LLM-generated samples and that of the benchmark dataset. This enables us to craft an augmented dataset that complements existing benchmarks, alleviating the challenges of data imbalance and scarcity and thereby enhancing the performance of fine-tuned models. We conducted extensive experiments to validate the efficacy of our proposed method, and we believe that this approach holds great potential for propelling the development and application of more advanced and reliable event extraction systems in real-world scenarios.

Introduction

Event extraction is a long-standing and crucial task within the natural language processing (NLP) community, aimed at extracting specific events from sentences or documents. This task is generally divided into two subtasks: event detection (ED) and event argument extraction (EAE). The former involves identifying trigger words or phrases and determining the event types they belong to, while the latter aims to extract specific information about the events identified in the previous stage, such as the location or the involved individuals of the events. Despite extensive previous work, the latest performance of event extraction still falls short of expectations. While sentence-level ED has achieved an impressive 80% success rate (Yang et al. 2019), sentence-level EAE lags behind at 60% (Wang et al. 2021) even assuming that the entities are already given.

One reason for the unsatisfactory performance is the insufficient availability of labeled data, particularly for EAE. For instance, ACE 2005 (Walker et al. 2006), a widely used sentence-level event extraction dataset, provides annotations for 33 distinct event types, but it has an alarmingly low number of labeled samples for some event types like "Justice.Pardon", with fewer than five in certain cases. While the MAVEN dataset (Wang et al. 2020) was introduced to address this shortage, it does not include argument labeling, and the long-tail distribution of events persists. These issues complicate effective training of models on some events.

Recently, large language models (LLMs) achieved remarkable success across various NLP tasks. However, certain deficiencies persist, as evidenced in tasks like abstract reasoning (Gendron et al. 2023) and named entity recognition (Qin et al. 2023). Given that event extraction is an inherently more challenging task that requires the extraction of complex structured data, we expected LLMs to face difficulties in this task. To check this, we tested LLMs for event extraction and evaluated their performance in zero-shot and one-shot settings. Even when provided with an example of each event type in the prompt, the LLMs' capability to identify events within a given sentence did not meet our requirements. We ascribe this shortfall to several factors, including:

- **Variations in Annotation Understanding**: LLMs may interpret the annotation differently from the ground truth annotators of the specific dataset, leading to discrepancies in event type identification.
- **Semantic Limitations**: Some semantic nuances and contextual information may not be adequately captured by LLMs at the sentence level, hindering their accurate identification of event types.
- **Undefined Event Types or Augmented Arguments**: Even when limited to a predefined list of event types, LLMs might produce undefined event types that bear similar meanings. Furthermore, they might provide a more detailed text rather than a succinct one.
- **Deviation in Output Format**: The outputs from LLMs may not always conform to the specific format, resulting in the omission of certain accurate predictions.

In light of these limitations, we opted to pivot our approach away from exclusively relying on pre-trained LLMs for event extraction. We devised a novel strategy by utilizing LLMs to annotate additional data, effectively tackling the data scarcity issue. Our contributions encompass three key aspects:

- **Extensive Evaluation**: We conducted a comprehensive evaluation of popular LLMs on commonly used benchmark datasets, highlighting the strengths and weaknesses of LLMs for event extraction.

- **LLM-based Annotation Approach**: We propose a novel idea of leveraging LLMs to annotate more data, which we subsequently used for fine-tuning models, and demonstrated its effectiveness in improving event extraction performance.
- **Dataset Publication**: We mitigate the long-tail problem by releasing annotated samples expanded through our use of LLMs on widely adopted datasets. This facilitates training with a greater number of annotated samples.[1]

In the next few sections, we will first review some related work and describe our proposed method in detail. Then, we will explain the setup for our experiments and present a comprehensive analysis of our experimental results. Finally, we will have a summarizing discussion before we conclude.

Related Work

Large Language Models

In recent years, LLMs have captured significant attention across various domains. The emergence of ChatGPT, in particular, has amplified interest across diverse research realms encompassing natural language processing (Qin et al. 2023), computer vision (Zhao et al. 2023), and robotics (Ichter et al. 2022; Driess et al. 2023). These models showcase impressive capabilities in zero-shot and few-shot scenarios (Wei et al. 2022a), particularly when coupled with the chain-of-thought methodology (Wei et al. 2022b; Kojima et al. 2022; Zhang et al. 2023). Nevertheless, a discernible disparity appears to persist between the performance of LLMs and fine-tuned models in specific research domains, such as named entity recognition (Qin et al. 2023).

Event Extraction

Event extraction has long been a focal point in the field of information extraction, receiving substantial attention over decades. It spans both sentence-level and document-level extractions, tackling the complexity of unstructured text. Within its scope, this task inherently involves two main subtasks: ED and EAE (Wang et al. 2021). To date, there are generally four categories of event extraction methods (Peng et al. 2023): classification methods (Chen et al. 2015), sequence labeling methods (Wang et al. 2020), span prediction methods (Du and Cardie 2020), and conditional generation methods (Lu et al. 2021). However, the progress has been significantly hindered by the persistent challenge of data scarcity. This is particularly evident in popular benchmark datasets like ACE 2005 (Walker et al. 2006) where some event types have fewer than ten labeled instances. In an effort to address this, MAVEN (Wang et al. 2020) was introduced in 2020, providing a substantially larger dataset, though still struggling with long-tail data distribution and a lack of argument annotations.

Data Augmentation

The pursuit of improvement has extended beyond the creation of novel datasets. For example, the study by Liu et al. (2016) utilizes FrameNet (Baker, Fillmore, and Lowe 1998) to bolster ED performance. In a unique approach, PLMEE (Yang et al. 2019) generates labeled data through iterative edits to prototypes and then selects samples based on their quality. RCEE_ER (Liu et al. 2020) reframes event extraction as a machine reading comprehension (MRC) task, tapping into the capabilities of advanced MRC methods and extensive externally annotated MRC data. And CLEVE (Wang et al. 2021) introduces a contrastive pre-training framework for event extraction, aiming to more effectively extract event knowledge from expansive unsupervised data sets.

This research prompts the question of whether there might be a more direct way to gather labeled data to enhance data augmentation. Could a strategy be formulated to autonomously create labeled datasets for event types with scarce annotations? Remarkably, the answer is affirmative, facilitated by the advanced capabilities of modern LLMs. As outlined by Gao et al. (2022), data augmentation methods generally fall into two categories: modifying existing examples and generating new data. The STAR model proposed by Ma et al. (2023a) introduces a method for generating new data using LLMs based on a given event structure to enhance low-resource information extraction performance. In our approach, we also adopt the latter strategy, but without constraints on the event structure, aiming to maintain diversity.

The Proposed Method

In this section, we introduce LLM-based techniques for event extraction, which fall into two main categories. The first strategy involves directly using LLMs by prompting them to extract event information from sentences. The second leverages LLMs' capabilities to create new labeled samples, thereby improving the performance of fine-tuned extraction methods. Together, these approaches highlight the diverse and integral roles that LLMs can fulfill in advancing the state of the art.

Prompting LLMs for Event Extraction

While there has been some research in open event extraction (Nguyen et al. 2016), the majority of studies have focused on datasets that adhere to specific schemas, such as ACE 2005 (Walker et al. 2006). In this study, we explore a novel dimension by testing the zero-shot and one-shot performance of LLMs on the task, assessing how the provision of specific examples influences the outcomes. We further examine both joint and pipeline methods to determine whether extracting triggers and arguments simultaneously or in separate stages affects the accuracy of the process.

Specifically, our exploration is divided into three main components as follows:

- **Joint and Pipeline Event Extraction**: We conducted experiments on both joint event extraction and pipeline event extraction. In the joint approach, we instruct the LLMs to simultaneously identify event types, triggers, and arguments. Conversely, in the pipeline approach, we

[1] https://github.com/shiqinghuayi19/LLMforEvent

Prompt for GPT-4

Could you create a succinct sentence that includes the **Justice.Acquit** event and then annotate it as per the provided examples?
sentence: Wouter Basson was acquitted in April 2002 on 46 charges, ranging from murder and drug trafficking to fraud and theft.
event_type: Justice.Acquit
trigger: acquitted
defendant: Wouter Basson
adjudicator: None
place: None

Prompt for PaLM and GPT-3.5-Turbo

Could you create a new and unique sentence that includes the **Ingestion** event, different from the example I've provided and then annotate it as per the provided examples?
The format is as follows
sentence:...
event_type:...
trigger:...
The example is as follows
sentence: It weakened further and was absorbed by a stationary trough near the South Island on March 12.
event_type: Ingestion
trigger: absorbed

Figure 1: Prompts utilized for guiding LLMs during the data annotation task.

adopt a two-stage strategy: first asking the LLMs to determine the event types and triggers, and then prompting them to return the corresponding arguments, given the sentence and event type information.

- **Zero-shot and One-shot Event Extraction**: We experimented with both zero-shot and one-shot event extraction. Since most datasets follow specific schemas, providing a specific example from the training dataset as a reference can be helpful for guiding the LLMs' extraction process.
- **Extraction of Multiple Events Simultaneously and Individually**: We explored two strategies: extracting all event types at once and extracting each event type one by one. General event extraction datasets often encompass numerous event types (e.g., 33 event types in ACE 2005 and 168 event types in MAVEN). In the one-shot setting, the prompt's length may become too extended. Therefore, we also investigated the extraction of each event separately, assessing the practicality and efficiency of both approaches.

Given that the language models produce free-text responses, extracting the desired information necessitates the use of regular expression patterns. This method enables a structured interpretation of the language models' outputs and allows for a robust analysis of their capabilities in detecting and classifying diverse event types. By examining these specific scenarios, our research aims to shed light on the effectiveness and limitations of current LLMs in event extraction, providing valuable insights to guide future advancements in this crucial area of study.

Empowering Event Extraction with LLM-based Annotators

In alignment with previous research (Qin et al. 2023; Ma et al. 2023b), a performance gap still exists between LLMs and fine-tuned models specifically designed for event extraction. To bridge this gap, we propose an integrative approach that leverages the strengths of both methods. The core concept consists of: 1) utilizing LLMs as expert annotators to generate labeled data that adheres to a specific schema, mirroring the structure of manually labeled data within a given dataset; and 2) fine-tuning specialized event extraction models using this enhanced and strategically labeled dataset.

Although LLMs possess extensive knowledge and can generate samples with annotations in the correct format, there may still be some semantic discrepancies with the current event extraction dataset. To mitigate potential disparities in data distribution, we employ targeted prompts to guide LLMs in generating labeled data that aligns closely with existing examples. We achieve this by incorporating a labeled sample from the current dataset, as illustrated in Figure 1, which displays two of the prompts we utilized. GPT-4 can discern the intended meaning even with just a basic instruction and without specific format stipulations. In contrast, PaLM and GPT-3.5-Turbo require more detailed prompts to function effectively.

Furthermore, for argument annotations, we include all potential argument roles of a specific event in our prompt, even when they may not be relevant to the specific example. This strategy aims to elicit the comprehensive annotation, recognizing that sentences generated by the LLMs might contain all types of arguments. This nuanced approach ensures both the richness and relevance of the annotated data, thereby fostering a more effective fine-tuning process.

Experimental Setup

In this section, we outline how we design our experiments. We first describe the datasets we studied and the LLMs we used. Then, we discuss the specific settings and techniques of the fine-tuning approaches considered.

Datasets

As a part of our current work, we conducted a comprehensive study focused on the advancement of sentence-level event extraction systems. We concentrate our analysis on two widely-recognized benchmark datasets, ACE 2005 (Walker et al. 2006) and MAVEN (Wang et al. 2020). To minimize the potential biases or variations introduced by preprocessing and evaluation methods, we have employed the OmniEvent framework[2], as proposed by Peng et al. (2023). This choice ensures a standardized and consistent

[2]https://github.com/THU-KEG/OmniEvent

Prompt for Zero-shot ED

Please analyze the following sentence to determine if it contains any of the listed events: [...]. If an event is detected, kindly provide its event type and trigger word/phrase, formatting your response as:
Event_Type: event type
Trigger: trigger word/phrase
If no event is identified, simply return **'**None**'.**
Sentence: But, well, the business is complicated and the business is tough.
Response:

Prompt for Zero-shot EAE

The sentence is understood to describe an event **Personnel.End-Position**, triggered by **'leaving'**. Analyze the sentence to identify any of the following arguments related to this event: **'person', 'entity', 'place'**. If you find any arguments, format your response as: {role_type: arguments}. If multiple arguments are related to this event, include them all. If no arguments are found, simply respond with 'None'.
Sentence: Davies is leaving to become chairman of the London School of Economics, one of the best - known parts of the University of London.
Response:

Figure 2: Prompts for zero-shot ED and zero-shot EAE

approach, allowing our analysis to focus more directly on the intrinsic performance of the models being studied.

Large Language Models

Our study is mainly directed toward the evaluation of three LLMs that have made considerable advancements in NLP. These models are noteworthy for their unique contributions and represent the forefront of current technology:

- **GPT-3.5-Turbo**[3], the most adept and economically viable version within the GPT-3.5 series (Xu et al. 2023).
- **GPT-4** (OpenAI 2023), a state-of-the-art model known for its computational power and adaptability, making it one of the most potent LLMs available in the market.
- **PaLM** (Chowdhery et al. 2023), renowned as one of the leading models, with an innovative architecture and exceptional performance in a wide array of complex tasks.

Through this research, our objective is to present a precise and systematic evaluation of the performance of these cutting-edge LLMs in the nuanced task of event extraction. We investigate their capabilities both as direct tools for extraction and as annotators. These models were specifically chosen for our evaluation due to their distinct architectures, differing capacities, and varied performance profiles. Together, these insights offer a comprehensive and multifaceted view of the current state of the field, highlighting both the capabilities and challenges of applying LLMs to event extraction.

Baseline Event Extraction Models

To test the quality of data labeled by LLMs, we include the testing of various classical approaches, encompassing different categories of methods such as classification-based methods, sequence-labeling methods, span prediction methods, and conditional generation methods (Peng et al. 2023).

- **BERT+CRF** (Wang et al. 2020), a sequence labeling model that integrates BERT (Devlin et al. 2019) as a feature extractor with the conditional random field (CRF) (Lafferty, McCallum, and Pereira 2001) to model structured output dependencies.
- **DMBERT** (Wang et al. 2019a,b), a classification model that adopts dynamic multi-pooling (Chen et al. 2015) operation on the hidden representations of BERT (Devlin et al. 2019).
- **CLEVE** (Wang et al. 2021), a contrastive pre-training framework for event extraction designed to better capture event knowledge from extensive unsupervised data.
- **EEQA** (Du and Cardie 2020), a span prediction method that approaches event extraction problems as a question-answering task.
- **Text2Event** (Lu et al. 2021), a conditional generation method incorporating constrained decoding and curriculum learning.

[3]https://platform.openai.com/docs/models/gpt-3-5

Experimental Results

As previously stated, we proposed two methods for leveraging LLMs in event extraction: one by using the LLMs to extract event information directly, and the other by employing them as annotators to enhance the performance of fine-tuned models. In this section, we will present a detailed analysis of the performance of these two approaches, along with corresponding insights and evaluations.

Large Language Models for Event Extraction

In this study, we explore the capability of LLMs in extracting event-related information. As previously discussed, we assessed the performance of several LLMs on the extraction task using different prompts. As illustrated in Figure 2, we present the prompts for zero-shot ED and EAE when simultaneously extracting multiple event types. For zero-shot ED, LLMs are presented with a list of event types from which they can choose. For zero-shot EAE, LLMs are supplied with the sentences accompanied by their corresponding gold-standard event types and trigger words. Furthermore, to furnish LLMs with schema information, we restrict the role types to those identified in the event-specific training samples. For one-shot ED, a single example per event type is provided. For one-shot EAE, we assume that the event types within the sentence are provided, and therefore, we only furnish LLMs with one example having the same event type.

Prompt

Could you create a succinct sentence that includes the **Justice.Acquit** event and then annotate it as per the provided examples?
sentence: Wouter Basson was acquitted in April 2002 on 46 charges, ranging from murder and drug trafficking to fraud and theft.
event_type: Justice.Acquit
trigger: acquitted
defendant: Wouter Basson
adjudicator: None
place: None

Response from GPT-4

Certainly! Here's a new sentence along with its annotation:
sentence: Jane Doe was acquitted of embezzlement charges by Judge Smith in New York City in June 2019.
event_type: Justice.Acquit
trigger: acquitted
defendant: Jane Doe
adjudicator: Judge Smith
place: New York City

Response from PaLM

sentence: Wouter Basson was acquitted in April 2002 on 46 charges, ranging from murder and drug trafficking to fraud and theft.
event_type: Justice.Acquit
trigger: acquitted
defendant: Wouter Basson
adjudicator: None
place: None

Response from GPT-3.5-Turbo

Wouter Basson was acquitted [Justice.Acquit] in April 2002 on 46 charges, ranging from murder and drug trafficking to fraud and theft.
[**event_type**: Justice.Acquit] [**trigger**: acquitted] [**defendant**: Wouter Basson] [**adjudicator**: None] [**place**: None]

Figure 3: Annotated example generated by LLMs

LLMs	M	Zero-shot ED			One-shot ED		
		P	R	F1	P	R	F1
PaLM	S	21.2	32.3	25.6	35.6	34.5	35.1
	L	31.9	48.6	38.5	45.1	43.7	44.4
GPT-3.5	S	14.1	10.4	12.0	39.5	29.5	33.8
	L	45.5	33.5	38.6	53.8	40.2	46.0
GPT-4	S	10.5	9.0	9.7	41.7	36.7	39.1
	L	56.1	49.1	52.4	60.6	53.3	56.7

Table 1: Performance of LLMs in zero-shot and one-shot ED on ACE 2005. In the 'S' mode, a correct prediction requires both event type and trigger to be accurate. Conversely, the 'L' mode considers a prediction correct based solely on accurate event type.

LLMs	Zero-shot EAE			One-shot EAE		
	P	R	F1	P	R	F1
PaLM	20.6	34.5	25.8	30.4	41.6	35.1
GPT-3.5	17.9	24.7	20.7	20.7	21.9	21.3
GPT-4	19.0	29.5	23.1	23.3	38.4	29.0

Table 2: Performance of LLMs in zero-shot and one-shot EAE on ACE 2005.

The performance of LLMs in zero-shot and one-shot pipeline event extraction on the ACE 2005 dataset is detailed in Tables 1 and 2, with a focus on the simultaneous extraction of multiple events. Owing to space limitations, the performance of LLMs for joint event extraction is detailed in the appendix. As demonstrated in the tables,

- Providing an example in the prompt enhances both ED and EAE performance.
- Performance experiences a significant decline when both event type and trigger word/phrase predictions are considered simultaneously, compared to focusing solely on the event type.
- Considering that fine-tuned models have already achieved an F1 performance of 80% in the ED task (Yang et al. 2019) and 60% in the EAE task (Wang et al. 2021), there remains a significant performance disparity between LLMs on the event extraction task and models specifically fine-tuned for it.

Upon further analysis, we discovered that the LLM predictions in failure cases are not always entirely inaccurate. LLMs appear predisposed to generating more expansive answers. For example, GPT-4 might recognize "divorce case" as the trigger phrase when the correct golden trigger is simply "divorce". It might also label "the Welches" as a Person, whereas the gold standard label is "Welches". Since we used the gold-standard event type and trigger for EAE evaluations, this nuance partly explains the less-than-stellar extraction performance.

We also evaluated PaLM's ED capabilities on MAVEN. We observed that the majority of the predictions adhere to the required format. Nonetheless, the performance leaves room for improvement:

- Out of 9,400 test samples, PaLM failed to make predictions for 6 tests.
- It identified 3,182 event types that are not recognized within the MAVEN event spectrum, such as "Being_a_member" and "Opening".

- When submitted to the competition system[4], the performance scores were as follows: precision at 21.8%, recall at 6.9%, and F1 score at 10.5%.

The unsatisfactory performance might be attributed to the overwhelming number of event types in the prompt. Evaluating the performance of each individual event type on popular LLMs should be a consideration for future work.

Large Language Models for Data Annotation

As outlined earlier, we evaluated the annotation capabilities of GPT-4, GPT-3.5-Turbo, and PaLM. To obtain consistent samples, we incorporated an example within the prompt. Our selection of these examples followed heuristic rules. Specifically, if an example contains a list of arguments that cover all roles, it is chosen. If no single sample for an event type encompasses all arguments, we collect several training samples until all role types are represented. The experimental outcomes reveal a performance hierarchy: GPT-4 is at the forefront, succeeded by PaLM, and then GPT-3.5-Turbo. Figure 3 depicts a representative example, showcasing the detailed response returned by each model. Notable observations from this figure are:

- **GPT-4**: While this model generally comprehends the requirements and can create a sentence with corresponding labels, it occasionally duplicates given sentences and annotations or responds with comments such as ”Certainly! You’ve provided the sentence and annotation already. If there is anything else you’d like me to do or add, please let me know!”.
- **PaLM**: This model recognizes the return format and provides sentences with annotations, although it has a tendency to repeat the given sentence.
- **GPT-3.5-Turbo**: In contrast to the others, GPT-3.5-Turbo often struggles to understand the task properly when using the prompt shown in Figure 3. A more explicit instruction is required to guide it in generating a distinct sentence and its corresponding annotation.

Beyond the evident shortcomings, the samples generated may be incompletely labeled. For instance, a sentence might contain two events with a causal relationship. However, if only one event is provided as an example, the samples returned by the LLMs typically label only one event. To provide a more detailed evaluation of the quality of labeled data generated by LLMs, we examined the performance of an event extraction model when fine-tuned with and without LLM-labeled data. Table 3 compares the performance of models fine-tuned with and without augmentation by GPT-4 labeled data to the ACE 2005 dataset. Due to space limitations, the performance comparisons for GPT-3.5-Turbo/PaLM-generated data augmentation, will be presented in the appendix.

Method	Dataset	ED P	ED R	ED F1
BERT+CRF	ACE 2005	64.5	68.5	66.4
	ACE 2005_DA	67.4	72.7	69.9
DMBERT	ACE 2005	61.6	75.2	67.7
	ACE 2005_DA	63.8	74.2	68.6
CLEVE	ACE 2005	65.5	77.4	71.0
	ACE 2005_DA	68.3	77.9	72.8
EEQA	ACE 2005	63.8	74.9	68.9
	ACE 2005_DA	66.9	72.2	69.5
Text2Event	ACE 2005	62.5	72.0	66.9
	ACE 2005_DA	64.2	71.7	67.8

Table 3: The performance comparison of ED methods fine-tuned with and without data augmentation. ‘ACE 2005_DA’ denotes the ACE 2005 training dataset enhanced with labeled data sourced from GPT-4.

Given the typically small sizes of event extraction datasets, preprocessing steps such as tokenization, sentence splitting, dependency parsing, and negative example selection can significantly influence model performance (Lai, Nguyen, and Nguyen 2020). To ensure uniformity in data preprocessing and evaluation, we employed the Omnievent framework for our experiments. We adopted the ”ACE-DYGIE” (Wadden et al. 2019) preprocessing strategy, excluding certain roles, such as those related to time, to emphasize roles with greater semantic distinctions. Apart from a few event types with over 300 labeled samples, we sourced 3-5 labeled examples from GPT-4 for all other events, totaling an additional 111 labeled samples. Given the high quality of GPT-4’s samples, our approach is iterative: we provided a sample from the training dataset to GPT-4 to generate a labeled example and then used this newly labeled sample in the prompt for the subsequent sample. As indicated in Tables 3 and 4, the inclusion of the GPT-4 labeled dataset leads to a notable improvement in performance most of the time. This finding demonstrates that the labeled data generated by GPT-4 is of sufficiently high quality to enhance the effectiveness of the fine-tuned model.

For the MAVEN dataset, we generated labeled data only with GPT-3.5-Turbo and PaLM due to budget constraints. We ranked all event types based on the number of their training samples, and for those not within the top 30 ranks, we randomly selected five to ten samples from the training data and generated labeled data for each. We used a filtering script to exclude samples deviating from the required format or exhibiting evident issues. The observed performance improvements are not as substantial, for example, from 67.99% to 68.04%, aligning with the quality analysis of the generated labeled data, suggesting that GPT-3.5-Turbo and PaLM can annotate data to some extent but not as effectively as GPT-4. Another contributing factor could be that MAVEN contains significantly more samples than the samples generated by LLMs, hence the improvement is not substantial.

Discussion

Although LLMs are widely employed in current applications, they may not serve as a flawless solution for all

[4]https://codalab.lisn.upsaclay.fr/competitions/395

Methods	Triggers	P	R	F1
BERT+CRF	Gold_ACE	66.9	62.3	64.5
	Predicted_ACE	50.1	64.9	56.5
	Gold_DA	65.4	65.1	65.3
	Predicted_DA	50.3	66.4	57.2
DMBERT	Gold_ACE	64.1	71.5	67.6
	Predicted_ACE	43.3	69.9	53.5
	Gold_DA	65.1	72.0	68.4
	Predicted_DA	45.9	70.1	55.5
CLEVE	Gold_ACE	70.8	75.0	72.8
	Predicted_ACE	51.1	73.7	60.4
	Gold_DA	69.0	76.0	72.3
	Predicted_DA	50.9	76.7	61.2
EEQA	Gold_ACE	70.7	54.2	61.4
	Predicted_ACE	44.3	56.3	49.6
	Gold_DA	73.1	58.9	65.2
	Predicted_DA	51.9	56.9	54.3
Text2Event	Gold_ACE	64.4	57.1	60.5
	Predicted_ACE	44.2	55.1	49.0
	Gold_DA	66.9	55.7	60.8
	Predicted_DA	48.8	56.3	52.3

Table 4: Performance of EAE with and without data augmented by GPT-4. We differentiate between EAE results using golden triggers and those using triggers predicted from prior ED. Specifically, 'Gold_ACE' refers to EAE performance using golden triggers on the original ACE 2005, while 'Predicted_DA' signifies EAE performance with predicted triggers on the augmented ACE 2005 dataset.

general-purpose NLP tasks (Qin et al. 2023). As demonstrated by our experiments, there remains a noticeable gap between fine-tuned event extraction models and LLMs. We believe this disparity arises not merely from capability differences, but also due to the following reasons:

- In datasets like ACE 2005, sentences are extracted from disparate articles, and their annotations are context-dependent. A specific case might be the word "this", triggering the "Conflict.Attack" event in the sentence "Nobody questions whether this is right or not.". Without access to the context of this sentence, determining what "this" refers to—and thus identifying the event—becomes exceedingly challenging.
- The prevailing evaluation approach heavily hinges on the principle of an exact match. While this method provides a straightforward way to gauge performance, it can inadvertently fail to recognize and credit precise predictions furnished by LLMs. Given that LLMs often produce semantically equivalent outputs that might not align verbatim with the gold standard, this metric can undersell their true capabilities. To more accurately reflect the proficiency and nuances of these models, there is a pressing need to adopt more flexible and suitable evaluation metrics, as also discussed by Wei et al. (2022a).
- LLMs' outputs might not always adhere to the specific format, potentially leading to the omission of some correct predictions.

Enhancing the performance of fine-tuned models through data augmentation also poses a challenge. An analysis of the benchmark dataset's data statistics reveals a trend where event types with fewer training samples also tend to have fewer test samples. This implies that the quantity and diversity of labeled samples, especially for specific events, play a crucial role in enhancing test performance. Additionally, as highlighted in DYGIE++ (Wadden et al. 2019), the small size and domain shift between the development and test splits in the ACE 2005 dataset can render selections based on the development dataset unreliable.

Future Work

Despite the great potential we discovered, our approach still has some room for improvement. Looking ahead, our plans for future research include exploring the following avenues:

- **Document-level Event Extraction**: Studying the practicality of extending our method to document-level extraction. While current methodologies are primarily sentence-focused, broader contexts are needed for more complete event understanding. For instance, a pronoun like "he" could be ambiguous without contextual clues.
- **Enhancing EAE**: Exploring methods to improve the extraction of event arguments. Recognizing an event without specific details might be insufficient. Therefore, studying accurate techniques for extracting specific information is paramount.
- **Prompt Investigation**: The choice of prompt is crucial to fully harness the capabilities of LLMs. Thus, exploring various prompt techniques to maximize the utility of LLMs warrants further study.
- **Experimenting with Open-source LLMs**: LLMs, although powerful, demand significant resources. Testing open-source models with fewer parameters, such as LLaMA (Touvron et al. 2023), which are suitable for environments with limited resources, will also be an essential part of our ongoing work.

Conclusions

In this study, we have concentrated on utilizing LLMs for sentence-level event extraction. To the best of our knowledge, this represents the first comprehensive exploration of employing LLMs for this task. Through rigorous testing with various LLMs and prompts, we assessed their performance in both joint and pipeline manners. Our findings revealed a noticeable gap between the performance of LLMs and that of fine-tuned models. Therefore, we proposed employing LLMs as expert annotators. This strategy yields labeled data that aligns with the benchmark dataset's distribution. By mitigating data scarcity and imbalance issues, this approach boosts the performance of the fine-tuned models. This underscores the effectiveness of LLMs as proficient annotators. Experiments on commonly used datasets affirmed the feasibility of this approach, highlighting its potential application in various domains.

Acknowledgments

This research was supported in part by National Robotics Programme under its Survey of Foundation Models for Robotics (Award M23NBK0091) project. Any opinions and conclusions expressed in this material are those of the authors and may not necessarily reflect the views of the programme or its host agency. No official endorsement should be inferred.

References

Baker, C. F.; Fillmore, C. J.; and Lowe, J. B. 1998. The Berkeley FrameNet Project. In *36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics, Volume 1*, 86–90. Montreal, Quebec, Canada: ACL.

Chen, Y.; Xu, L.; Liu, K.; Zeng, D.; and Zhao, J. 2015. Event Extraction via Dynamic Multi-Pooling Convolutional Neural Networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 167–176. Beijing, China: ACL.

Chowdhery, A.; Narang, S.; Devlin, J.; Bosma, M.; Mishra, G.; Roberts, A.; Barham, P.; Chung, H. W.; Sutton, C.; Gehrmann, S.; Schuh, P.; Shi, K.; Tsvyashchenko, S.; Maynez, J.; Rao, A.; Barnes, P.; Tay, Y.; Shazeer, N.; Prabhakaran, V.; Reif, E.; Du, N.; Hutchinson, B.; Pope, R.; Bradbury, J.; Austin, J.; Isard, M.; Gur-Ari, G.; Yin, P.; Duke, T.; Levskaya, A.; Ghemawat, S.; Dev, S.; Michalewski, H.; Garcia, X.; Misra, V.; Robinson, K.; Fedus, L.; Zhou, D.; Ippolito, D.; Luan, D.; Lim, H.; Zoph, B.; Spiridonov, A.; Sepassi, R.; Dohan, D.; Agrawal, S.; Omernick, M.; Dai, A. M.; Pillai, T. S.; Pellat, M.; Lewkowycz, A.; Moreira, E.; Child, R.; Polozov, O.; Lee, K.; Zhou, Z.; Wang, X.; Saeta, B.; Diaz, M.; Firat, O.; Catasta, M.; Wei, J.; Meier-Hellstern, K.; Eck, D.; Dean, J.; Petrov, S.; and Fiedel, N. 2023. PaLM: Scaling Language Modeling with Pathways. *Journal of Machine Learning Research*, 24: 240:1–240:113.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 4171–4186. Minneapolis, Minnesota: ACL.

Driess, D.; Xia, F.; Sajjadi, M. S. M.; Lynch, C.; Chowdhery, A.; Ichter, B.; Wahid, A.; Tompson, J.; Vuong, Q.; Yu, T.; Huang, W.; Chebotar, Y.; Sermanet, P.; Duckworth, D.; Levine, S.; Vanhoucke, V.; Hausman, K.; Toussaint, M.; Greff, K.; Zeng, A.; Mordatch, I.; and Florence, P. 2023. PaLM-E: An Embodied Multimodal Language Model. In *International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, 8469–8488. Honolulu, Hawaii: PMLR.

Du, X.; and Cardie, C. 2020. Event Extraction by Answering (Almost) Natural Questions. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, 671–683. Online: ACL.

Gao, J.; Yu, C.; Wang, W.; Zhao, H.; and Xu, R. 2022. Mask-then-Fill: A Flexible and Effective Data Augmentation Framework for Event Extraction. In *Findings of the Association for Computational Linguistics: EMNLP*, 4537–4544. Abu Dhabi, United Arab Emirates: ACL.

Gendron, G.; Bao, Q.; Witbrock, M.; and Dobbie, G. 2023. Large Language Models Are Not Strong Abstract Reasoners. arXiv:2305.19555.

Ichter, B.; Brohan, A.; Chebotar, Y.; Finn, C.; Hausman, K.; Herzog, A.; Ho, D.; Ibarz, J.; Irpan, A.; Jang, E.; Julian, R.; Kalashnikov, D.; Levine, S.; Lu, Y.; Parada, C.; Rao, K.; Sermanet, P.; Toshev, A.; Vanhoucke, V.; Xia, F.; Xiao, T.; Xu, P.; Yan, M.; Brown, N.; Ahn, M.; Cortes, O.; Sievers, N.; Tan, C.; Xu, S.; Reyes, D.; Rettinghouse, J.; Quiambao, J.; Pastor, P.; Luu, L.; Lee, K.; Kuang, Y.; Jesmonth, S.; Joshi, N. J.; Jeffrey, K.; Ruano, R. J.; Hsu, J.; Gopalakrishnan, K.; David, B.; Zeng, A.; and Fu, C. K. 2022. Do As I Can, Not As I Say: Grounding Language in Robotic Affordances. In *Conference on Robot Learning*, volume 205 of *Proceedings of Machine Learning Research*, 287–318. Auckland, New Zealand: PMLR.

Kojima, T.; Gu, S. S.; Reid, M.; Matsuo, Y.; and Iwasawa, Y. 2022. Large Language Models are Zero-Shot Reasoners. In *NeurIPS*.

Lafferty, J. D.; McCallum, A.; and Pereira, F. C. N. 2001. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, 282–289. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.

Lai, V. D.; Nguyen, T. N.; and Nguyen, T. H. 2020. Event Detection: Gate Diversity and Syntactic Importance Scores for Graph Convolution Neural Networks. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, 5405–5411. Online: ACL.

Liu, J.; Chen, Y.; Liu, K.; Bi, W.; and Liu, X. 2020. Event Extraction as Machine Reading Comprehension. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, 1641–1651. Online: ACL.

Liu, S.; Chen, Y.; He, S.; Liu, K.; and Zhao, J. 2016. Leveraging FrameNet to Improve Automatic Event Detection. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2134–2143. Berlin, Germany: ACL.

Lu, Y.; Lin, H.; Xu, J.; Han, X.; Tang, J.; Li, A.; Sun, L.; Liao, M.; and Chen, S. 2021. Text2Event: Controllable Sequence-to-Structure Generation for End-to-end Event Extraction. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 2795–2806. Online: ACL.

Ma, M. D.; Wang, X.; Kung, P.-N.; Brantingham, P. J.; Peng, N.; and Wang, W. 2023a. STAR: Improving Low-Resource Information Extraction by Structure-to-Text Data Generation with Large Language Models. In *NeurIPS 2023 Workshop on Synthetic Data Generation with Generative AI*.

Ma, Y.; Cao, Y.; Hong, Y.; and Sun, A. 2023b. Large Language Model Is Not a Good Few-shot Information Extractor, but a Good Reranker for Hard Samples! In *Findings of the Association for Computational Linguistics: EMNLP*, 10572–10601. Singapore: ACL.

Nguyen, K.-H.; Tannier, X.; Ferret, O.; and Besançon, R. 2016. A Dataset for Open Event Extraction in English. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation*, 1939–1943. Portorož, Slovenia: ELRA.

OpenAI. 2023. GPT-4 Technical Report. arXiv:2303.08774.

Peng, H.; Wang, X.; Yao, F.; Zeng, K.; Hou, L.; Li, J.; Liu, Z.; and Shen, W. 2023. The Devil is in the Details: On the Pitfalls of Event Extraction Evaluation. In *Findings of the Association for Computational Linguistics*, 9206–9227. Toronto, Canada: ACL.

Qin, C.; Zhang, A.; Zhang, Z.; Chen, J.; Yasunaga, M.; and Yang, D. 2023. Is ChatGPT a General-Purpose Natural Language Processing Task Solver? In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 1339–1384. Singapore: ACL.

Touvron, H.; Lavril, T.; Izacard, G.; Martinet, X.; Lachaux, M.-A.; Lacroix, T.; Rozière, B.; Goyal, N.; Hambro, E.; Azhar, F.; Rodriguez, A.; Joulin, A.; Grave, E.; and Lample, G. 2023. LLaMA: Open and Efficient Foundation Language Models. arXiv:2302.13971.

Wadden, D.; Wennberg, U.; Luan, Y.; and Hajishirzi, H. 2019. Entity, Relation, and Event Extraction with Contextualized Span Representations. In Inui, K.; Jiang, J.; Ng, V.; and Wan, X., eds., *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 5784–5789. Hong Kong, China: Association for Computational Linguistics.

Walker, C.; et al. 2006. ACE 2005 Multilingual Training Corpus. https://catalog.ldc.upenn.edu/LDC2006T06.

Wang, X.; Han, X.; Liu, Z.; Sun, M.; and Li, P. 2019a. Adversarial Training for Weakly Supervised Event Detection. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 998–1008. Minneapolis, Minnesota: ACL.

Wang, X.; Wang, Z.; Han, X.; Jiang, W.; Han, R.; Liu, Z.; Li, J.; Li, P.; Lin, Y.; and Zhou, J. 2020. MAVEN: A Massive General Domain Event Detection Dataset. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, 1652–1671. Online: ACL.

Wang, X.; Wang, Z.; Han, X.; Liu, Z.; Li, J.; Li, P.; Sun, M.; Zhou, J.; and Ren, X. 2019b. HMEAE: Hierarchical Modular Event Argument Extraction. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, 5777–5783. Hong Kong, China: ACL.

Wang, Z.; Wang, X.; Han, X.; Lin, Y.; Hou, L.; Liu, Z.; Li, P.; Li, J.; and Zhou, J. 2021. CLEVE: Contrastive Pre-training for Event Extraction. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 6283–6297. Online: ACL.

Wei, J.; Tay, Y.; Bommasani, R.; Raffel, C.; Zoph, B.; Borgeaud, S.; Yogatama, D.; Bosma, M.; Zhou, D.; Metzler, D.; Chi, E. H.; Hashimoto, T.; Vinyals, O.; Liang, P.; Dean, J.; and Fedus, W. 2022a. Emergent Abilities of Large Language Models. *Transactions on Machine Learning Research*.

Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; Ichter, B.; Xia, F.; Chi, E. H.; Le, Q. V.; and Zhou, D. 2022b. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. In *NeurIPS*.

Xu, F.; Lin, Q.; Han, J.; Zhao, T.; Liu, J.; and Cambria, E. 2023. Are Large Language Models Really Good Logical Reasoners? A Comprehensive Evaluation and Beyond. arXiv:2306.09841.

Yang, S.; Feng, D.; Qiao, L.; Kan, Z.; and Li, D. 2019. Exploring Pre-trained Language Models for Event Extraction and Generation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 5284–5294. Florence, Italy: ACL.

Zhang, Z.; Zhang, A.; Li, M.; and Smola, A. 2023. Automatic Chain of Thought Prompting in Large Language Models. In *The Eleventh International Conference on Learning Representations*. Kigali, Rwanda: OpenReview.net.

Zhao, Y.; Misra, I.; Krähenbühl, P.; and Girdhar, R. 2023. Learning Video Representations from Large Language Models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 6586–6597. Vancouver, BC: IEEE.