Efficient Zero-shot Event Extraction with Context-Definition Alignment

Hongming Zhang, Wenlin Yao, Dong Yu
Tencent AI Lab, Bellevue, USA
{hongmzhang, wenlinyao, dyu}@global.tencent.com

Abstract

Event extraction (EE) is the task of identifying interested event mentions from text. Conventional efforts mainly focus on the supervised setting. However, these supervised models cannot generalize to event types out of the pre-defined ontology. To fill this gap, many efforts have been devoted to the zero-shot EE problem. This paper follows the trend of modeling event-type semantics but moves one step further. We argue that using the static embedding of the event type name might not be enough because a single word could be ambiguous, and we need a sentence to define the type semantics accurately. To model the definition semantics, we use two separate transformer models to project the contextualized event mentions and corresponding definitions into the same embedding space and then minimize their embedding distance via contrastive learning. On top of that, we also propose a warming phase to help the model learn the minor difference between similar definitions. We name our approach Zero-shot Event extraction with Definition (ZED). Experiments on the MAVEN dataset show that our model significantly outperforms all previous zero-shot EE methods with fast inference speed due to the disjoint design. Further experiments also show that ZED can be easily applied to the few-shot setting when the annotation is available and consistently outperforms baseline supervised methods.

1 Introduction

Event extraction, the task of identifying event mentions from documents and classifying them into pre-defined event types, is a fundamental NLP problem (Grishman et al., 2005). As a centric information extraction task, event extraction is the foundation of a series of event-centric NLP applications (Chen et al., 2021) including event relation extraction (Wang et al., 2020a), event schema induction (Li et al., 2020), and missing event prediction (Chaturvedi et al., 2017).

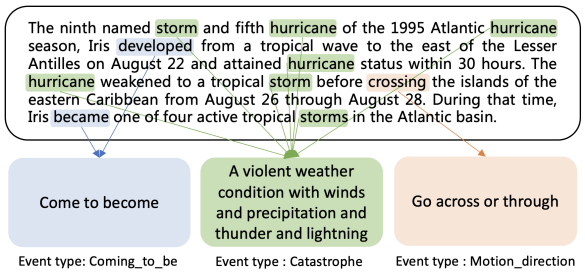

Figure 1: Zero-shot event extraction task demonstration. Given a corpus, the goal is to identify all event mentions that fit the target event definitions without using any annotation. The event definitions and corresponding mentions are indicated in the same color.

Traditional event extraction efforts (Wadden et al., 2019; Wang et al., 2019; Lin et al., 2020) mostly focus on learning to identify and classify events under a supervised learning setting, where a pre-defined event ontology and large-scale expert annotations is available. However, the learned supervised models cannot be easily applied to new event types out of the pre-defined ontology, limiting these models' usage in real applications.

Recently, large-scale pre-trained language models have demonstrated strong semantics representation capabilities and motivated a series of works to extract events in a zero-shot setting. For example, Du and Cardie (2020) propose to manually design templates for each event type to convert the event extraction problem into a question-answering (QA) task and then leverage QA models to extract events. Following that, Lyu et al. (2021) propose to verbalize candidate triggers and event types into hypothesis and premises and leverage pre-trained textual entailment models to extract events. However, as analyzed in (Lyu et al., 2021), these models heavily rely on the template design and often suffer from the domain-shifting problem between the original training task and the new task. Moreover, as these models require jointly encoding the event mentions and event types, the time complexity is

Findings of the Association for Computational Linguistics: EMNLP 2022, pages 7169–7179

$O(N * T)$, where N is the number of event mention candidates and T is the number of event types. Considering the low inference speed and high computation cost of inference with a deep model, such complexity could be a massive burden for real-time EE systems.

To avoid manually designing templates and to improve the inference efficiency, another line of work (Zhang et al., 2021) tries to leverage pre-trained language representation models (i.e., BERT (Devlin et al., 2019)) to acquire a contextualized event type representation. The model can decouple the mention and label representations during the inference time and predict the candidate trigger to the most similar event type based on the cosine similarity. As a result, this method could significantly reduce the inference time complexity from $O(N * T)$ to $O(N + T)$. However, as the experiments show, using only the label name might not lead to a good event-type representation because the selected words could be ambiguous.

In this work, we follow the trend of representation learning (Zhang et al., 2021; Gao et al., 2021) and move forward from representing each event type with a single name to a definition sentence. Specifically, we propose a three-stage event representation learning framework. In the offline pre-training phase, we leverage auto-extracted context-definition alignments to learn a definition encoding model that can encode the contextualized mentions and definitions into the same embedding space. In the second warming phase, we use the target event types to retrieve hard negative examples to further polish the model. In the end, we identify and classify event mentions based on the cosine similarity between the mention representation and corresponding event-type representations. As our system is a disjoint model, the inference time complexity is also $O(N+T)$. Experiments on MAVEN (Wang et al., 2020b), the largest EE dataset to the best of our knowledge, show that ZED outperforms all previous zero-shot approaches with high inference efficiency. Further experiments show that ZED could also be applied to the supervised setting, where it achieves comparable performance in the fully supervised setting and consistently outperforms baseline supervised models in the data-scarce learning settings. Specifically, with 10% of the training data, ZED could achieve over 95% of the full performance. All the collected alignment data, created definitions, and the code are available at: https://github.com/tencent-ailab/ZED.

2 Related Works

In this section, we introduce related works about event extractions, contrastive representation learning, and definition modeling.

2.1 Event Extraction

As a fundamental information extraction task (Chen et al., 2021), event extraction has attracted many efforts in the NLP community (Sundheim, 1992; Grishman and Sundheim, 1996; Riloff, 1996; Grishman et al., 2005; Chen et al., 2021; Hong et al., 2022). Recent success on the event extraction task mostly relies on employing either symbolic features (Ji and Grishman, 2008; Liao and Grishman, 2010; Liu et al., 2016) or distributed features (Chen et al., 2015; Nguyen et al., 2016; Liu et al., 2018; Zhang et al., 2019; Wadden et al., 2019; Lin et al., 2020) to learn supervised models with large-scale high-quality annotations. However, the requirement of a pre-defined ontology and corresponding annotations limits the application of these models in real applications.

To address this issue and extract unseen event types, Huang et al. (2018) propose a zero-shot event extraction task and use a transfer-learning framework to apply the model trained with seen event types to unseen ones. However, the prerequisite of their high performance is the similarity between seen and unseen event types. Recently, with the fast development of large-scale language models, several works (Du and Cardie, 2020; Lyu et al., 2021; Zhang et al., 2021) propose to leverage the pre-trained models to encode the label semantics either with templates or contextualized embeddings. In this work, we follow the effort of using deep models to model the label semantics but make a step further. Instead of directly using a pre-trained model, we train a disjoint context-to-definition alignment encoding model, which can effectively map the candidate event mentions and definitions into the same embedding space and thus more accurately and efficiently extract events for any arbitrarily defined event types.

2.2 Contrastive Representation Learning

The contrastive loss (Chopra et al., 2005) is one of the most popular training objectives for representa-

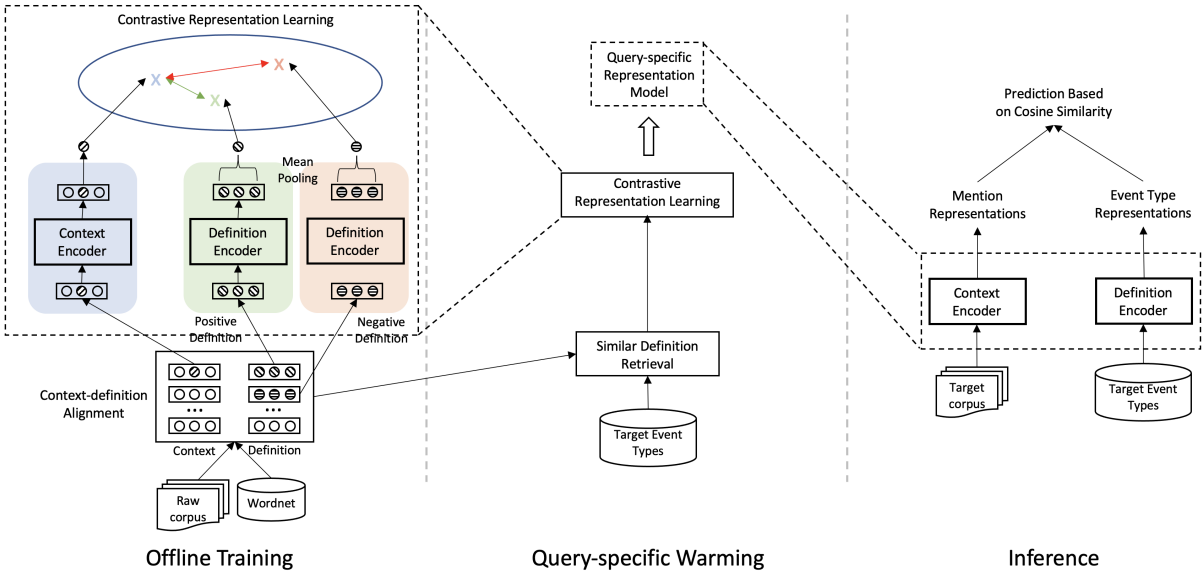

Figure 2: Overall framework of ZED. In the offline training phase, we train the separate context and definition encoders with auto-extracted context-definition alignment data. In the second warming phase, after knowing the target event types, as no annotation is provided, we first retrieve similar concepts from WordNet (Miller, 1998) and use corresponding alignment data to polish the representation model. In the last inference phase, after encoding all candidate event type definitions, for each candidate event mention, we will encode it with the context encoder and determine whether it belongs to one of the target event types based on the cosine similarity.

tion learning. The original contrastive loss and its variations (e.g., triplet loss (Schroff et al., 2015), lifted structured loss (Oh Song et al., 2016), N-pair loss (Sohn, 2016), and NCE loss (Gutmann and Hyvärinen, 2010)) have been shown helpful for a series of vision applications (Radford et al., 2021). After being introduced to the NLP community, the contrastive learning-based method also leads to the success of a series of representation learning tasks such as sentence representation (Gao et al., 2021). Different from previous works, where the anchors and positive/negative examples typically belong to the same category (e.g., image/sentence), we propose to use the contextualized token representation as the anchor and event type definition representations as the positive/negative examples to better solve the zero-shot event extraction task. Moreover, motivated by the success of the "pre-training+fine-tuning" paradigm, we propose a novel three-stage representation learning framework.

2.3 Definition Modeling

Humans are capable of understanding new concepts by reading their glosses or definitions. Thus, how to leverage the definitions and explanations from dictionaries to help understand human language is a long-standing question in the NLP community. Most of the previous efforts in this direction are working on the word sense disambiguation task (Luo et al., 2018; Huang et al., 2019; Blevins and Zettlemoyer, 2020; Kumar et al., 2019; Bevilacqua and Navigli, 2020; Yao et al., 2021; Su et al., 2022a,b). These models learn to map a token into the correct pre-defined synset by either jointly or disjointly encoding the tokens and definitions. Even though the setting of our model and these WSD models are similar, identifying event mentions that satisfy an arbitrary event type definition is a more challenging task (Senel et al., 2022). WSD aims to learn to distinguish the correct synset versus several (typically less than 10) other pre-defined synsets, while our goal is to align an event mention and the corresponding definition, where all other arbitrary definitions are considered to be the negative candidates. To address the engineering limitation that negative candidates exceed the GPU memory limitation, we propose a coarse-to-fine negative sampling strategy to help models learn the minor differences between similar definitions without forgetting the big picture.

3 Task Definition

We define the zero-shot event extraction task as follows. Given a document in the format of a sentence set $\mathcal{S}$ and event type set $\mathcal{E}$. Each event type $E \in \mathcal{E}$ is defined with a natural sentence d. The task is

Mention in Context	Definition
I love playing basketball.	get pleasure from
Bob is studying computer science.	be a student; follow a course of study; be enrolled at an institute of learning
I got promoted after many years of hard work	give a promotion to or assign to a higher position

Table 1: Demonstration of collected context and definition alignments. Target mentions are underlined.

to identify all mentions $\mathcal{M}_E$ in $\mathcal{S}$ that satisfy the definition of E for each $E \in \mathcal{E}$ without using direct annotations during the training phase.

4 Model

We present the model framework in Figure 2. Motivated by the success of the "pre-training + fine-tuning" learning paradigm, we propose to address the zero-shot event extraction problem with a three-stage framework. Technical details are as follows.

4.1 Offline Pre-training

The offline pre-training step aims to train a decent definition encoder to map the target mention representation and corresponding definitions into the same embedding space. To achieve this goal, as no annotation is provided, we first collect context-definition alignments and then train the encoder with a contrastive learning loss.

4.1.1 Data Preparation

We select all verbal synsets from the WordNet ontology (Miller, 1998) to form our open-world event definition set. In total, we collect 13,814 event synsets. After that, to collect large-scale alignment data between context and definitions, we apply the current state-of-the-art word sense disambiguation model (Yao et al., 2021) to the NYT corpus (Sandhaus, 2008) to align tokens in NYT with their correct definitions. We randomly select 10 context instances for each synset to speed up the training process. As a result, we collect 775K context-definition alignments. Examples of extracted alignments are presented in table 1.

4.1.2 Context-definition Alignment Encoding with Contrastive Learning

The goal of the context-definition alignment encoding is encoding the contextualized representation of the target mention and the sentence representation of the definition into the same embedding space and pushing them to be closer to each other because they should have similar semantic meanings. As this objective aligns well with the learning objective of the contrastive learning framework, we follow the standard contrastive learning framework (Chopra et al., 2005). Specifically, we denote the pre-processed context-definition alignment set as $\mathcal{T}$, where each instance $(S, i, j, D) \in \mathcal{T}$ contains context sentence S, which is a list of tokens $w_1^S, w_2^S, ..., w_n^S$, target word starting position i, target word ending position[1] j, and a definition sentence D, which is also a list of tokens $w_1^D, w_2^D, ..., w_m^D$. We follow the standard approach to get the contextualized word representation as the mean pooling of all sub-token representations:

$$\mathbf{e}_{S,i,j} = \frac{\sum_{i \le k \le j} \mathbf{e}_k}{j - i + 1}, \tag{1}$$

where $\mathbf{e}_k$ is the contextualized representation of token k produced by a transformer baseline language model (e.g., BERT (Devlin et al., 2019)). For the sentence encoding, we choose to use the average representation of all tokens as follows:

$$\mathbf{d}_D = \frac{\sum_{1 \le k \le m} FFN(\mathbf{e}_k)}{m}, \tag{2}$$

where FFN represents a two-layer feed-forward neural network and $\mathbf{e}_k$ is the token representation of token w_k.

Following the contrastive learning framework, during this step, we optimize the marginal ranking loss[2]. Assume that the set of randomly sampled negative definitions is $\mathcal{D}'$, for each $D' \in \mathcal{D}'$, we could follow equation 2 to compute its representation as $\mathbf{d}_{D'}$. For each instance $(S, i, j, D) \in \mathcal{T}$ and a randomly sampled negative definition set $\mathcal{D}'$, we minimize the following marginal ranking loss:

$$\frac{\sum_{D' \in \mathcal{D}'} max(0, \epsilon - (cos(\mathbf{e}_{S,i,j}, \mathbf{d}_D) - cos(\mathbf{e}_{S,i,j}, \mathbf{d}_{D'})))}{\|\mathcal{D}'\|}, \tag{3}$$

where max means the maximum operation, cos is the cosine similarity, and ϵ is the margin.

4.2 Query-specific Warming

After the pre-training phase, the model briefly understands how to project the contextualized event

[1] Each word could have multiple tokens because we follow the standard tokenization of BERT (Devlin et al., 2019).

[2] We chose ranking loss over entropy loss mainly because the alignment data we used for the pre-training is automatically collected, and the training signal may contain noise.

mentions and corresponding definitions into similar positions in the embeddings. However, its capability of distinguishing similar definitions is still limited because the previous negative sampling strategy does not encourage such capabilities. To address this issue, we introduce an additional warming phase to help models learn the minor difference between similar definitions.

Similar to how human beings understand new concepts by recalling relevant knowledge, we also retrieve relevant knowledge from $\mathcal{T}$ to further fine-tune the model. Specifically, assume that the set of interested event definitions is $\hat{\mathcal{D}}$, for each $\hat{D} \in \hat{\mathcal{D}}$, we first retrieve the most similar definition $\tilde{D}$ from the original definition set $\mathcal{D}$ by:

$$\tilde{D} = \arg\max_{D \in \mathcal{D}} sim(PLM(D), PLM(\hat{D})), \quad (4)$$

where sim is the similarity measurement and PLM represents the encoding with a pre-trained language model. In our experiment, we select cosine similarity as the similarity measurement and average contextualized token embedding encoded with BERT-base (Devlin et al., 2019) as the encoding. But other techniques could also be applied.

We thus denote the set of all retrieved relevant definitions as $\tilde{\mathcal{D}}$ and select a subset $\tilde{\mathcal{I}}$ of $\mathcal{I}$ such that all definitions in $\tilde{\mathcal{I}}$ belong to $\tilde{\mathcal{D}}$ to further fine-tune the model. After generating all the data, we will fine-tune all models following the loss function in Equation 3.

4.3 Inference

During the inference, we compute the representation for each candidate event mention in and target event type descriptions. After that, for each candidate mention, we compute its cosine similarity with all the target event-type representations. If the largest similarity is larger than a threshold t, this mention is identified and labeled as the most similar event type. Assume that the size of all candidate mentions and target event types are N and T, respectively. Compared with previous zero-shot models that rely on the joint encoding of the candidate mention and target event types (Du and Cardie, 2020; Lyu et al., 2021; Yao et al., 2021), we successfully reduce the computation complexity from $O(N*T)$ to $O(N+T)$. A numerical evaluation of the computation efficiency is shown in Section 6.2.

5 Experiments

This section introduces experiment details, including the selected baseline methods, experiment datasets, and implementation details.

5.1 Baseline Methods

In the past two years, the community has been devoting significant effort to solving the zero-shot event extraction problem with different approaches. Specifically, we select the following best-performing models as our baselines.

1. **Pre-trained Question Answering Models** (Du and Cardie, 2020) (QA): Most NLP tasks can be converted into a QA format and event extraction is not an exception. Motivated by this, Du and Cardie (2020) propose to design a question template for each target event type and directly ask a QA model to answer whether a mention is the target event.

2. **Pre-trained Textual Entailment Models** (Lyu et al., 2021) (TE): Motivated by the QA approach, Lyu et al. (2021) explore the possibility of utilizing a pre-trained textual entailment (TE) model to automatically extract events. Specifically, for each target event type, Lyu et al. (2021) manually design a template to convert it into a hypothesis, treat the target event mention as the premise, and ask the TE model whether the target event mention can entail an event type.

3. **Word Sense Disambiguation Models** (Yao et al., 2021) (WSD): Prior WSD works also heavily rely on the correctly modeling of the definitions, so conceptually they could also be applied to the event extraction task following our setup. There are mainly two key differences between our work and (Yao et al., 2021): (1) Yao et al. (2021) encode the context and definition jointly while our model encodes them separately; (2) Yao et al. (2021) aim at modeling the minor difference between different synsets of the same word while our work aims at modeling general definition semantics.

4. **Contextualized Label Embedding** (Zhang et al., 2021) (CLE): The last baseline we compare with is the contextualized label representation. Specifically, for each target event type, Zhang et al. (2021) generate a contextualized label representation by putting the label name back into contexts and directly extracting events

Model	Identification			Identification+Classification		
	P	R	F1	P	R	F1
Chance Performance	19.33	20.05	19.68	0.11	0.11	0.11
Most Popular Event Type	18.59	19.45	19.01	0.74	0.77	0.75
QA (Du and Cardie, 2020)	19.76	45.18	27.49	4.19	9.58	5.83
TE (Lyu et al., 2021)	20.20	32.83	25.01	4.59	7.46	5.68
WSD (Yao et al., 2021)	24.66	80.52	37.76	5.36	17.51	8.21
CLE (Zhang et al., 2021)	55.07	14.63	23.00	42.99	11.34	17.95
ZED	59.37	42.28	**49.39**	39.63	28.22	**32.96**

Table 2: Zero-shot Event identification and classification results on MAVEN (Wang et al., 2020b), which has 168 event types. Best F1 performances are indicated with bold font.

based on the similarity between the mention representation and event type representations.

Besides these baselines, we also present the "Chance" performance, where a mention is randomly selected following the percentage of gold mentions and randomly assigned an event type, and the "Most Popular Event Type" performance, where a mention is also randomly selected following the percentage of gold mentions and is always predicted to be the most popular event type.

5.2 Evaluation Dataset

We select MAVEN (Wang et al., 2020b) as the evaluation dataset due to its large-scale and balanced distribution. Specifically, MAVEN contains 186 unique event types selected from FrameNet (Baker et al., 1998) and 118,732 annotated event mentions, which is almost two magnitudes larger than the previous datasets such as ACE (Grishman et al., 2005). Moreover, MAVEN provides the official event mention candidates to evaluate the mention understanding capability of all event extraction models more fairly. As the original dataset only provides the event name in the format of a phrase (e.g., "Body_movement"), we directly use definitions from Wordnet as the description[3]. Examples of the event types and corresponding definitions are presented in Appendix Table 5.

5.3 Implementation Details

For baseline models, we conduct experiments with officially released code, hyperparameters, templates, and pre-trained models. For ZED, we use two separate encoders for the context and definition encoding. Both of them are initialized with BERT-base (Devlin et al., 2019). As no training set is needed in the zero-shot setting and the test set of MAVEN is not publicly available, we report the performance on the dev set. Specifically, we set the margin to 0.2 for the marginal ranking loss and set the number of negative examples to 2. The selection threshold at the inference phrase is set to be 0.7. We train the model with ten epochs for both the pre-training and warming phrases. We directly evaluate the last checkpoint to simulate the real application, where no dev set is available. All models are trained with Tesla P40 with batch size 16. The pre-training and warming phrases will take around 200 and 3 hours on a single GPU, respectively, but we could speed it up with multiple GPUs.

[3]For event names that have multiple synsets or not covered by WordNet, we manually select the most accurate description from WordNet.

	F1 (I)	F1 (I+C)
ZED	49.39	32.96
- Warming	47.86 (-1.53)	21.89 (-11.07)
- Strong Negative Sampling	48.91 (-0.48)	29.20 (-3.76)

Table 3: Ablation study. "I" and "C" represent the identification and classification, respectively.

6 Zero-shot Performance

The zero-shot performance of all models is presented in Table 2, from which we can make the following observations:

1. All models significantly outperform the naive baselines even though they do not use any annotations. This observation shows that current deep models can indeed learn rich semantics that could generalize outside of their original training goal.

2. The overall performance of pre-trained QA, TE, and WSD models is not satisfying because they suffer from domain shifting. For example, even though current deep-model-driven QA models have surpassed human performance on several

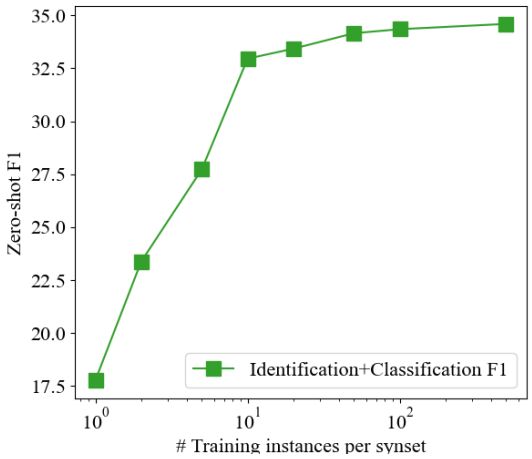

Figure 3: Effect of training instance number per definition. (Zero-shot performance on the Identification+Classification F1 is reported.)

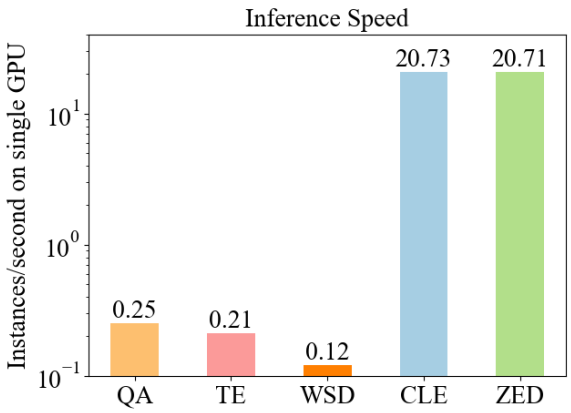

Figure 4: Inference Speed of all zero-shot models. For a fair comparison, we evaluate all models with the same GPU and use batch size 1. As our model is smaller than baseline models, we could use a larger batch size in real applications to further boost efficiency.

leaderboards, they are still not ready to be used as a general QA model for solving tasks that require deep understanding, such as zero-shot event extraction.

3. Compared with other methods, Contextualized label embedding achieves lower identification F1 but higher classification accuracy, which aligns with the original observation in (Zhang et al., 2021). The reason behind this is that due to the cone property of the BERT representation (i.e., most of the token representations of BERT are grouped in a small region), it is tough to determine the cosine similarity boundary of whether an event mention fits a specific event type. As a result, even though CLE could accurately identify high-confident mentions, it cannot handle boundary ones very well.

4. Compared with baseline methods, ZED could perform better on both the identification and classification tasks. The main reason is that we are using definitions to model the label semantics, which is more accurate than a single word.

6.1 Ablation Study

From the ablation study results in Table 3, we can see that if we remove the warming phase, the model's performance will drop on both the identification and classification, especially the classification step. This aligns well with our assumption that the model can learn to model the general definition semantics after the pre-training step but cannot distinguish minor differences very well. The performance drop of removing the strong negative sampling module indicates that strong negatives are crucial for the success of representation learning, which aligns well with previous observations (Clark et al., 2020).

Besides those ablation studies, we also show the impact of the pre-training data scale in Figure 3. As expected, the more data we use, the better performance we will get. However, the performance gain after 10 instances per synset is limited. As a result, we select 10 instances for each synset as the pre-training data for training efficiency.

6.2 Inference Efficiency

We present the inference speed of all evaluated models in Figure 4. As ZED adopts a disjoint encoding design, we successfully reduce the computation complexity from $O(N * T)$ to $O(N + T)$, where N is the number of event mentions and T number of event types. On Maven, which has 168 different event types, ZED could speed up the inference efficiency by almost two magnitudes.

7 Warming with Gold Annotation

ZED can also be adapted to a fully supervised or few-shot learning setting when the annotation is available. Specifically, during the warming phase of our model, we can replace the auto-retrieved examples with the annotated ones and fine-tune the model. In this section, we follow the benchmark paper (Wang et al., 2020b) to compare with

Model	Identification			Identification+Classification		
	P	R	F1	P	R	F1
DMBERT	73.37	87.82	79.95	61.20	73.25	66.69
BERT+CRF	75.19	81.80	78.35	64.40	70.28	**67.21**
ZED + Supervision	82.48	80.76	**81.61**	67.87	66.46	67.16

Table 4: Model Performance with full annotations, Best F1 performances are indicated with the bold font. "I" and "C" indicate the event identification and classification, respectively.

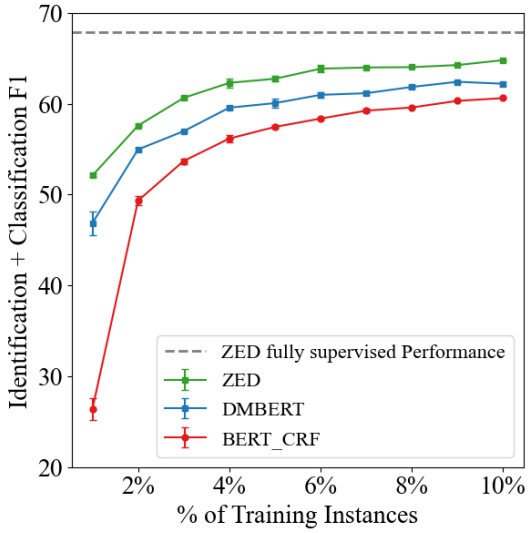

Figure 5: Model performance With limited annotation.

the recent language model-driven baselines[4]: DMBERT (Wang et al., 2019) and BERT (Devlin et al., 2019) + CRF (Lafferty et al., 2001), which also achieved the previous best performance. Please refer to the original papers for technical details of these baseline models. We implement all models with the officially released code[5] and report the average performance of five trials on the development set. All models are trained for ten epochs, and the final model is evaluated. Like the zero-shot setting, we also report the micro precision, recall, and F1 for both the "identification" and "identification+classification" settings. All hyper-parameters are based on the officially released code.

Results in Table 4 show that with the help of the pre-training step, our model can outperform all previous supervised models on the identification task and comparable performance on the classification task. This makes sense because a carefully designed deep model could learn to identify and classify event mentions well with the large-scale annotation provided by MAVEN.

However, we argue that such a large-scale annotation is often expensive in terms of money and time. The data-scarce learning setting might be more applicable in real applications. Thus, we also test the performance of these supervised settings under the data-scarce learning setting. Specifically, we randomly select 1% to 10% of the training sentences to be sampled from the training data and report the performances in Figure 5. Our model can constantly outperform baseline models with a small number of annotations. Especially when only 1% of the data is available, we only have 7.07 mentions per event type, ZED could achieve over 50 F1. With 10% of the training data, ZED could achieve over 95% of full supervised performance. These observations show that our framework could be applied to broader applications where limited or enough annotations are available besides the zero-shot setting. Besides that, another interesting finding is that even though "BERT+CRF" could outperform "DMBERT" slightly when enough annotation is available, which is consistent with the observations in (Wang et al., 2020b), its performance is worse under the data-scarce setting. This observation indicates that using CRF might not be the optimal option when the annotation scale is limited.

8 Conclusion

This paper proposes a novel zero-shot event extraction framework ZED. Given a set of interested event types in the format of definitions, ZED could automatically extract all the event mentions that fit the definitions from raw documents much better than previous methods. Experiments show that the proposed warming phase and the mixed strong negative examples sampling strategies contribute to the

[4]Several other recent works (e.g., OneIE (Lin et al., 2020)) further improves the performance on event extraction by utilizing the constraints between trigger and arguments. However, as such information is not available in MAVEN and extracting arguments is beyond the research scope of this paper, we cannot compare with them.

[5]https://github.com/THU-KEG/MAVEN-dataset

success of ZED. Additional experiments also show that ZED could be applied to the supervised setting. Thanks to the pre-training phase, it could achieve good performance under both the fully supervised and data-scarce settings.

Acknowledgements

We thank anonymous reviewers for their insightful comments and suggestions.

References

Collin F Baker, Charles J Fillmore, and John B Lowe. 1998. The berkeley framenet project. In *Proceedings of ACL 1998*, pages 86–90.

Michele Bevilacqua and Roberto Navigli. 2020. Breaking through the 80% glass ceiling: Raising the state of the art in word sense disambiguation by incorporating knowledge graph information. In *Proceedings of ACL 2020*, pages 2854–2864.

Terra Blevins and Luke Zettlemoyer. 2020. Moving down the long tail of word sense disambiguation with gloss informed bi-encoders. In *Proceedings of ACL 2020*, pages 1006–1017.

Snigdha Chaturvedi, Haoruo Peng, and Dan Roth. 2017. Story comprehension for predicting what happens next. In *Proceedings of EMNLP 2017*, pages 1603–1614.

Muhao Chen, Hongming Zhang, Qiang Ning, Manling Li, Heng Ji, Kathleen McKeown, and Dan Roth. 2021. Event-centric natural language processing. In *Proceedings of ACL 2021 Tutorial*, pages 6–14.

Yubo Chen, Liheng Xu, Kang Liu, Daojian Zeng, and Jun Zhao. 2015. Event extraction via dynamic multi-pooling convolutional neural networks. In *Proceedings of ACL 2015*, pages 167–176.

Sumit Chopra, Raia Hadsell, and Yann LeCun. 2005. Learning a similarity metric discriminatively, with application to face verification. In *Proceedings of CVPR 2005*, pages 539–546.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. ELECTRA: pre-training text encoders as discriminators rather than generators. In *Proceedings of ICLR 2020*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT 2019*, pages 4171–4186.

Xinya Du and Claire Cardie. 2020. Event extraction by answering (almost) natural questions. In *Proceedings of EMNLP 2020*, pages 671–683.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. Simcse: Simple contrastive learning of sentence embeddings. In *Proceedings of EMNLP 2021*, pages 6894–6910.

Ralph Grishman and Beth Sundheim. 1996. Message understanding conference- 6: A brief history. In *Proceedings of COLING 1996*, pages 466–471.

Ralph Grishman, David Westbrook, and Adam Meyers. 2005. Nyu's english ace 2005 system description. *ACE*, 5.

Michael Gutmann and Aapo Hyvärinen. 2010. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of IJCAI 2010*, pages 297–304.

Ruixin Hong, Hongming Zhang, Xintong Yu, and Changshui Zhang. 2022. Learning event extraction from a few guideline examples. *IEEE ACM Trans. Audio Speech Lang. Process.*, 30:2955–2967.

Lifu Huang, Heng Ji, Kyunghyun Cho, Ido Dagan, Sebastian Riedel, and Clare R. Voss. 2018. Zero-shot transfer learning for event extraction. In *Proceedings of ACL 2018*, pages 2160–2170.

Luyao Huang, Chi Sun, Xipeng Qiu, and Xuanjing Huang. 2019. Glossbert: BERT for word sense disambiguation with gloss knowledge. In *Proceedings of EMNLP-IJCNLP 2019*, pages 3507–3512.

Heng Ji and Ralph Grishman. 2008. Refining event extraction through cross-document inference. In *Proceedings of ACL 2008*, pages 254–262.

Sawan Kumar, Sharmistha Jat, Karan Saxena, and Partha P. Talukdar. 2019. Zero-shot word sense disambiguation using sense definition embeddings. In *Proceedings of ACL 2019*, pages 5670–5681.

John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of ICML 2001*, pages 282–289.

Manling Li, Qi Zeng, Ying Lin, Kyunghyun Cho, Heng Ji, Jonathan May, Nathanael Chambers, and Clare R. Voss. 2020. Connecting the dots: Event graph schema induction with path language modeling. In *Proceedings of EMNLP 2020*, pages 684–695.

Shasha Liao and Ralph Grishman. 2010. Using document level cross-event inference to improve event extraction. In *Proceedings of ACL 2010*, pages 789–797.

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

Shulin Liu, Yubo Chen, Shizhu He, Kang Liu, and Jun Zhao. 2016. Leveraging framenet to improve automatic event detection. In *Proceedings of ACL 2016*.

Xiao Liu, Zhunchen Luo, and Heyan Huang. 2018. Jointly multiple events extraction via attention-based graph information aggregation. In *Proceedings of EMNLP 2018*, pages 1247–1256.

Fuli Luo, Tianyu Liu, Qiaolin Xia, Baobao Chang, and Zhifang Sui. 2018. Incorporating glosses into neural word sense disambiguation. In *Proceedings of ACL 2018*, pages 2473–2482.

Qing Lyu, Hongming Zhang, Elior Sulem, and Dan Roth. 2021. Zero-shot event extraction via transfer learning: Challenges and insights. In *Proceedings of ACL 2021*, pages 322–332.

George A Miller. 1998. *WordNet: An electronic lexical database*. MIT press.

Thien Huu Nguyen, Kyunghyun Cho, and Ralph Grishman. 2016. Joint event extraction via recurrent neural networks. In *Proceedings of NAACL-HLT 2016*, pages 300–309.

Hyun Oh Song, Yu Xiang, Stefanie Jegelka, and Silvio Savarese. 2016. Deep metric learning via lifted structured feature embedding. In *Proceedings of CVPR 2016*, pages 4004–4012.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning transferable visual models from natural language supervision. In *Proceedings of ICML 2021*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR.

Ellen Riloff. 1996. Automatically generating extraction patterns from untagged text. In *Proceedings of AAAI 1996*, pages 1044–1049.

Evan Sandhaus. 2008. The new york times annotated corpus. *Linguistic Data Consortium, Philadelphia*, 6(12):e26752.

Florian Schroff, Dmitry Kalenichenko, and James Philbin. 2015. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of CVPR 2015*, pages 815–823.

Lütfi Kerem Senel, Timo Schick, and Hinrich Schütze. 2022. Coda21: Evaluating language understanding capabilities of nlp models with context-definition alignment. In *Proceedings of ACL 2022*.

Kihyuk Sohn. 2016. Improved deep metric learning with multi-class n-pair loss objective. In *Proceedings of NIPS 2016*.

Ying Su, Hongming Zhang, Yangqiu Song, and Tong Zhang. 2022a. Multilingual word sense disambiguation with unified sense representation. In *Proceedings of the COLING 2022*, pages 4193–4202.

Ying Su, Hongming Zhang, Yangqiu Song, and Tong Zhang. 2022b. Rare and zero-shot word sense disambiguation using z-reweighting. In *Proceedings of ACL 2022*, pages 4713–4723.

Beth Sundheim. 1992. Overview of the fourth message understanding evaluation and conference. In *Proceedings of MUC 1992*, pages 3–21.

David Wadden, Ulme Wennberg, Yi Luan, and Hannaneh Hajishirzi. 2019. Entity, relation, and event extraction with contextualized span representations. In *Proceedings of EMNLP-IJCNLP 2019*, pages 5783–5788.

Haoyu Wang, Muhao Chen, Hongming Zhang, and Dan Roth. 2020a. Joint constrained learning for event-event relation extraction. In *Proceedings of EMNLP 2020*, pages 696–706.

Xiaozhi Wang, Xu Han, Zhiyuan Liu, Maosong Sun, and Peng Li. 2019. Adversarial training for weakly supervised event detection. In *Proceedings of NAACL-HLT 2019*, pages 998–1008.

Xiaozhi Wang, Ziqi Wang, Xu Han, Wangyi Jiang, Rong Han, Zhiyuan Liu, Juanzi Li, Peng Li, Yankai Lin, and Jie Zhou. 2020b. MAVEN: A massive general domain event detection dataset. In *Proceedings of EMNLP 2020*, pages 1652–1671.

Wenlin Yao, Xiaoman Pan, Lifeng Jin, Jianshu Chen, Dian Yu, and Dong Yu. 2021. Connect-the-dots: Bridging semantics between words and definitions via aligning word sense inventories. In *Proceedings of EMNLP 2021*, pages 7741–7751.

Hongming Zhang, Haoyu Wang, and Dan Roth. 2021. Zero-shot label-aware event trigger and argument classification. In *Proceedings of ACL 2021 Findings.*, pages 1331–1340.

Tongtao Zhang, Heng Ji, and Avirup Sil. 2019. Joint entity and event extraction with generative adversarial imitation learning. *Data Intell.*, 1(2):99–120.

A MAVEN Ontology Demonstration

Name	Definition
Manufacturing	make or cause to be or to become
Achieve	to gain with effort
Communication	express in words
Employment	engage or hire for work
Process_start	take the first step or steps in carrying out an action
Theft	take without the owner's consent
Legal_rulings	pronounce a sentence in a court of law
Influence	have an effect upon
Give_up	give up, such as power, as of monarchs and emperors, or duties and obligations
Catastrophe	a violent weather condition

Table 5: Representative MAVEN event types and associated definitions. All used definitions will be released with the code.