The Art of Prompting: Event Detection based on Type Specific Prompts

Sijia Wang♣, Mo Yu♠, Lifu Huang♣
♣Virginia Tech, ♠WeChat AI
{sijiawang, lifuh}@vt.edu, moyymyu@tencent.com

Abstract

We compare various forms of prompts to represent event types and develop a unified framework to incorporate the event type specific prompts for supervised, few-shot, and zero-shot event detection. The experimental results demonstrate that a well-defined and comprehensive event type prompt can significantly improve event detection performance, especially when the annotated data is scarce (few-shot event detection) or not available (zero-shot event detection). By leveraging the semantics of event types, our unified framework shows up to 22.2% F-score gain over the previous state-of-the-art baselines.

1 Introduction

Event detection (ED) (Grishman, 1997; Chinchor and Marsh, 1998; Ahn, 2006) is the task of identifying and typing event mentions from natural language text. Supervised approaches, especially deep neural networks (Chen et al., 2020; Du and Cardie, 2020; Lin et al., 2020; Liu et al., 2020; Lyu et al., 2021), have shown remarkable performance under a critical prerequisite of a large amount of manual annotations. However, they cannot be effectively generalized to new languages, domains or types, especially when the annotations are not enough (Huang et al., 2016; Huang and Ji, 2020; Lai et al., 2020b; Shen et al., 2021) or there is no annotation available (Lyu et al., 2021; Zhang et al., 2021b; Pasupat and Liang, 2014).

Recent studies have shown that both the accuracy and generalizability of ED can be improved via leveraging the semantics of event types based on various forms of prompts, such as event type specific queries (Lyu et al., 2021; Du and Cardie, 2020; Liu et al., 2020), definitions (Chen et al., 2020), structures (Lin et al., 2020; Wang et al., 2019), or a few prototype event triggers (Wang and Cohen, 2009; Dalvi et al., 2012; Pasupat and Liang, 2014; Bronstein et al., 2015; Lai and Nguyen, 2019; Zhang et al., 2021b; Cong et al., 2021). These studies further encourage us to take another step forward and think about the following three questions: (1) does the choice of prompt matter when the training data is abundant or scarce? (2) what’s the best form of ED prompt? (3) how to best leverage the prompt to detect event mentions?

To answer the above research questions, we conduct extensive experiments with various forms of prompts for each event type, including (a) event type name, (b) prototype seed triggers, (c) definition, (d) event type structure based on both event type name and its predefined argument roles, (e) free parameter based continuous soft prompt, and (f) a more comprehensive event type description (named APEX prompt) that covers all the information of prompts (a)-(d). We observe that (1) by considering the semantics of event types with most forms of prompts, especially seed triggers and the comprehensive event type descriptions, the performance of ED under all settings can be significantly improved; (2) Among all forms of event representations, the comprehensive description based prompts show to be the most effective, especially for few-shot and zero-shot ED; (3) Different forms of event type representations provide complementary improvements, indicating that they capture distinct aspects and knowledge of the event types.

The contributions of this work are as follows:

• We investigate various prompts to represent event types for both supervised and weakly supervised ED, and prove that a well-defined and comprehensive event type prompt can dramatically improve the performance of ED and the transferability from old types to new types.

• A unified framework is developed to leverage the semantics of event types with prompts for supervised, few-shot, and zero-shot ED, and demonstrate

---

1The source code, model checkpoints and data are publicly available at https://github.com/VT-NLP/Event_APEX.
state-of-the-art performance with up to 22.2% F-score improvement over the strong baseline methods.

2 Related Work

**Supervised ED:** Most of the existing Event Detection studies follow a supervised learning paradigm (Ji and Grishman, 2008; Liao and Grishman, 2010; McClosky et al., 2011; Li et al., 2013; Chen et al., 2015; Cao et al., 2015; Feng et al., 2016; Yang and Mitchell, 2016; Nguyen et al., 2016; Zhang et al., 2017; Lin et al., 2020; Wang et al., 2021b). However, they cannot be directly applied to detect new types of events. Recently studies have shown that, by leveraging the semantics of event types based on type-specific questions (Du and Cardie, 2020; Liu et al., 2020; Li et al., 2020; Lyu et al., 2021) or seed event triggers (Bronstein et al., 2015; Lai and Nguyen, 2019; Wang et al., 2021a), the event detection performance can be improved. However, it is still unknown whether they are the best choices for representing the semantics of event types.

**Few-shot ED:** Two primary learning strategies in few-shot classification tasks are Meta-Learning (Kang et al., 2019; Li et al., 2021; Xiao and Marlet, 2020; Yan et al., 2019; Chowdhury et al., 2021) and Metric Learning (Sun et al., 2021; Wang et al., 2020b; Zhang et al., 2021a; Agarwal et al., 2021). Several studies have exploited metric learning to align the semantics of candidate events with a few examples of the novel event types for few-shot event detection (Lai et al., 2020a; Deng et al., 2020; Lai et al., 2020b; Cong et al., 2021; Chen et al., 2021; Shen et al., 2021).

**Zero-shot ED:** Huang et al. (2018) first exploited zero-shot event extraction by leveraging Abstract Meaning Representation (Banarescu et al., 2013) to represent event mentions and types into a shared semantic space. Recent studies (Zhang et al., 2021b; Lyu et al., 2021) further demonstrate that by leveraging a large external corpus with abundant anchor triggers, zero-shot event detection can also be achieved with decent performance without using any training data.

**Prompt Learning** Prompt learning aims to learn a task-specific prompt while keeping most of the model’s parameters frozen (Li and Liang, 2021; Hambardzumyan et al., 2021; Brown et al., 2020). It has shown competitive performance in many applications of natural language processing (Raffel et al., 2020; Brown et al., 2020; Shin et al., 2020; Jiang et al., 2020; Lester et al., 2021; Schick and Schütze, 2021b). Previous work either used a manual (Petroni et al., 2019; Brown et al., 2020; Schick and Schütze, 2021b) or automated approach (Jiang et al., 2020; Yuan et al., 2021; Li and Liang, 2021) to create prompts.

3 Problem Formulation

Here, we first define each setting of the event detection task and then describe the various forms of event type prompts.

3.1 Settings of ED

For supervised ED (SED), we follow the conventional supervised event detection setting where the training, validation, and evaluation data sets cover the same set of event types. The goal is to learn a model \( f \) to identify and classify event mentions for the target event types.

For few-shot ED (FSED), there are two separate training data sets for few-shot event detection: (1) A large-scale data set \( D_{\text{base}} = \{(x_i, y_i)\}_{i=1}^M \) that covers the old event types (named base types) where \( M \) denotes the number of base event types; (2) a smaller data set \( D_{\text{novel}} = \{(x_j, y_j)\}_{j=1}^{N \times K} \) that covers \( N \) novel event types, with \( K \) examples each. Note that the base and novel event types are disjoint except for the Other class. The model \( f \) will be first optimized on \( D_{\text{base}} \), and then further fine-tuned on \( D_{\text{novel}} \). The goal is to evaluate the generalizability and transferability of the model from base event types to new event types with few annotations.

For zero-shot ED (ZSED), the training data sets are the only difference between zero-shot and few-shot event detection. In zero-shot event detection, there is only a large-scale base training data set \( D_{\text{base}} = \{(x_i, y_i)\}_{i=1}^M \) for the base event types. The model \( f \) will be only optimized on base event types and evaluated on the novel types.

3.2 Event Type Prompts

We compare the following five forms of prompts to represent the event types: (a) **Event Type Name** is the event class name, usually consisting of one to three tokens. (b) **Definition** can be a short sentence that formally describes the meaning of the event types. (c) **Prototype Seed Triggers** a list of
tokens or phrases that are frequently identified as event triggers. (d) Event Type Structure consists of event key argument roles, indicating the core participants of the target event type. (e) Prompts can also be Continuous Soft Prompt, that is, a free vector of parameters to represent each event type. (f) We further define a more comprehensive description APEX Prompts that is manually written and covers all previous prompts except soft prompts. Examples of all event type prompts are shown in Figure 1 and Appendix A. Detailed prompt token selection is in Appendix B.

4 A Unified Framework for ED

We adapt (Wang et al., 2021a) and design a unified event detection framework (as shown in Figure 1) which leverages event type specific prompts to detect events under supervised, few-shot, and zero-shot settings. Formally, given an input sentence \( W = \{w_1, w_2, \ldots, w_n\} \), we take each event type prompt \( T^t = \{\tau^t_1, \tau^t_2, \ldots, \tau^t_m\} \) as a query of \( M \) tokens to extract triggers for event type \( t \). Specifically, we first concatenate them into a sequence [CLS] \( \tau^t_1 \ldots \tau^t_m \) [SEP] \( w_1 \ldots w_n \) [SEP]. We use a pre-trained BERT encoder (Devlin et al., 2019) to get contextual representations for the input sentence \( W = \{w_0, w_2, \ldots, w_n\} \) as well as the event type prompt \( T = \{\tau^t_0, \tau^t_1, \ldots, \tau^t_m\} \).

Given a prompt of each event type, we aim to extract corresponding event triggers from the input sentence. To achieve this goal, we need to capture the semantic correlation of each input token to the event type. Thus we learn a weight distribution over the sequence of contextual representations of the event type prompt, to obtain event type \( t \) aware contextual representation \( A^t_i = \sum_{j=1}^{T^t} \alpha_{ij} \cdot \tau^j \), where \( \alpha_{ij} = \cos(w_i, \tau^j) \), where \( \tau^j \) is the contextual representation of the \( j \)-th prompt token. \( \cos(\cdot, \cdot) \) is the cosine similarity function between two vectors.

With that, the event type aware contextual representation \( A^t_i \) will be concatenated with the original contextual representation \( w_i \) from the encoder, and classified into a binary label, indicating whether it is a candidate trigger of event type \( t \) or not:

\[
\tilde{y}^t_i = U_o([w_i; A^t_i; P_j]),
\]

where \([; ;]\) denotes concatenation operation, \( U_o \) is a learnable parameter matrix for event trigger detection, and \( P_j \) is the one-hot part-of-speech (POS) encoding of word \( w_i \). For continuous soft prompt based event detection, we follow Li and Liang (2021) where a prefix index \( q \) is prepended to the input sequence \( W' = [q; W] \). The prefix embedding is learned by \( q = \text{MLP}_\theta(Q_q[q]) \), where \( Q_q \in \mathbb{R}^{|Q| \times k} \) denotes the embedding lookup table for the vocabulary of prefix indices. Both \( \text{MLP}_\theta \) and \( Q_q \) are trainable parameters. Detailed learning strategy is in Appendix C.

5 Experiment Setup

We perform experiments on three public benchmark datasets, including ACE05-E+ (Automatic Content Extraction), ERE (Entity Relation Event) (Song et al., 2015), and MAVEN (Wang et al., 2020a). On each dataset, we conduct experiments for SED, FSED, and ZSED. For SED, we use the same data split as the previous studies (Li et al., 2013; Wadden et al., 2019; Lin et al., 2020; Du and Cardie, 2020; Lin et al., 2020; Nguyen et al., 2021; Wang et al., 2020a) on all the three benchmark datasets. For FSED and ZSED on MAVEN, we follow the previous study (Chen et al., 2021) and choose 120 event types with the most frequent mentions as the base event types and the rest 45 event types as novel ones. For FSED and ZSED on ACE and ERE, previous studies (Lai et al., 2020b,a;
Table 1: Results of event detection (ED) on ACE05 (F1-score, %). * indicates evaluation on our data set split based on the authors’ public implementations.

| Method         | SED  | FSED | ZSED |
|----------------|------|------|------|
| Previous SOTA  | 73.3 | 35.2*| 49.1*|
| (Nguyen et al., 2021) | (Lai et al., 2020b) | (Chen et al., 2021) |
| (a) Event type name | 72.2 | 52.7 | 49.8 |
| (b) Definition   | 73.1 | 46.7 | 45.5 |
| (c) Seed triggers| 73.7 | 53.8 | 49.6 |
| (d) Event structure | 72.8 | 50.4 | 48.0 |
| (e) Soft prompt  | 68.1 | 48.2 | -   |
| Majority voting of (a-e) | 73.9 | 52.1 | 48.7 |
| (f) APEX Prompt  | 74.9 | 57.4 | 51.2 |

Table 2: Results of event detection (ED) on ERE (F1-score, %).

| Method         | SED  | FSED | ZSED |
|----------------|------|------|------|
| Previous SOTA  | 59.4 | 33.0*| 41.2*|
| (Lu et al., 2021) | (Lai et al., 2020b) | (Zhang et al., 2021b) |
| (a) Event type Name | 58.2 | 44.8 | 40.5 |
| (b) Definition   | 57.9 | 44.2 | 40.4 |
| (c) Seed triggers| 60.4 | 50.4 | 46.2 |
| (d) Event structure | 59.1 | 48.5 | 48.7 |
| (e) Soft prompt  | 55.6 | 41.7 | -   |
| Majority voting of (a-e) | 60.2 | 47.9 | 45.6 |
| (f) APEX Prompt  | 63.4 | 52.6 | 48.9 |

MAVEN are shown in Table 1-3, from which we see that (1) the APEX prompt achieves the best performance among all the forms of prompts under all the settings of the three benchmark datasets. Compared with the previous state of the art, the APEX prompt shows up to 4% F-score gain for SED (on ERE), 22.2% F-score gain for FSED (on ACE), and 19.7% F-score gain for ZSED (on MAVEN); (2) All the forms of prompts provide significant improvement for FSED and ZSED, demonstrating the benefit of leveraging the semantics of event types via various forms of prompts. (3) Except APEX, seed triggers provide more improvements than other forms of event type prompts under most settings, suggesting its potential to represent the semantics of event types accurately. (4) Continuous soft prompt does not provide comparable performance as other forms of event type representations, which proves the necessity of leveraging event type specific prior knowledge to the representations; (5) The majority voting does not show improvement over individual prompts since each prompt captures a particular aspect of the event type semantics.

**Supervised Event Detection** By carefully investigating the event mentions that are correctly detected by the APEX prompt while missed by other prompts, we find that the APEX prompt is more effective in detecting two types of event mentions: homonyms (multiple-meaning words) and intricate words. General homonyms are usually hard to be detected as event mentions as they usually have dozens of meanings in different contexts. For example, consider the following two examples: (i) Airlines are getting [Transport:Movement] flyers to destinations on time more often. (ii) If the board cannot vote to give [Transaction:Transfer-Money] themselves present money. Here, “get” and “give”
are not detected based on the event type name or seed triggers but are correctly identified by the definition and APEX prompts. The definition and APEX prompts make 10% and 7% fewer false predictions than seed triggers on general homonyms. For intricate words, their semantics cannot be captured with an individual prompt. In the following two examples: (i) *It is reasonable, however, to reimburse board members for legitimate expenses* (ii) *ever having discussed being compensated by the board in the future*, "reimburse" and "compensated" indicate sophisticated meaning of Transaction:Transfer-Money, which may not be captured by prompts, such as seed triggers. With the event definition and the argument roles in the APEX prompt, the highly correlated contexts, such as “board members” and “legitimate expenses”, can help the model correctly detect reimburse as an event mention of Transaction:Transfer-Money.

**Few-shot Event Detection** Figure 2 shows the F-score distribution of all novel types based on various forms of event type prompts, from which we observe that: (1) The event type name, seed triggers, and APEX prompt generally perform better than definition and structure, as they carry more straightforward semantics of event types. (2) Event type name based prompts show lower performance on Personnel:End-Position, Personnel:Start-Position and Transaction:Transfer-Money than other event types, as the semantics of these event type names are less indicative than other event types. (3) Seed trigger based prompts perform worse than event type name and APEX prompts on two event types, Life:injure and Life:die, probably because the prototype seed triggers are not properly selected. (4) The structure based prompt outperforms the other prompts on Life:Injure as Life:Injure events require the existence of a person or victim. (5) APEX prompt shows consistently (almost) best performance on all the event types because it combines all the information of other prompts. (6) We also observe that the performance of Life:Be-Born, Life:Die, Life:Marry, and Personnel:Elect based on various forms of prompts are consistently better than the other types as the intrinsic semantics of those types the corresponding event triggers are concentrated.

**Zero-shot Event Detection** The proposed prompt-based method is more affordable to be generalized compared with the prior state-of-the-art zero-shot approach (Zhang et al., 2021b). The average length of created APEX prompts is less than 20 tokens. Thus manually creating them will not take much human effort. On the contrary, Zhang et al. (2021b) requires an extensive collection of anchor sentences to perform zero-shot event detection, e.g., 4,556,237 anchor sentences for ACE and ERE. This process is time-consuming and expensive.

**7 Conclusion**

We investigate a variety of prompts to represent the semantics of event types, and leverage them with a unified framework for supervised, few-shot and zero-shot event detection. Experimental results demonstrate that, a well-defined and comprehensive description of event types can significantly improve the performance of event detection, especially when the annotations are limited (few-shot event detection) or even not available (zero-shot event detection), with up to 22.2% F-score gain over the prior state of the art.
**Limitations**

We have demonstrated that an accurate description can perform better for both supervised and weakly supervised event detection. However, the event types from most existing ontologies are not properly defined. For example, in ACE annotation guideline (Linguistic Data Consortium, 2005), transfer-money is defined as “giving, receiving, borrowing, or lending money when it is not in the context of purchasing something”. However, it is hard for the model to interpret it accurately, especially the constraints “not in the context of purchasing something”. In addition, many event types from MAVEN, e.g., Achieve, Award, and Incident, are not associated with any definitions. A potential future research direction is to leverage mining-based approaches or state-of-the-art generators to automatically generate a comprehensive event type description based on various sources, such as annotation guidelines, example annotations, and external knowledge bases.

**Acknowledgments**

We thank the anonymous reviewers and area chair for their valuable time and constructive comments. This research is based upon work supported by the Amazon Research Award.

**References**

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

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

Collin F Baker, Charles J Fillmore, and John B Lowe. 1998. The berkeley frame net project. In 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics, Volume 1, pages 86–90.

Laura Banarescu, Claire Bonial, Shu Cai, Madalina Georgescu, Kira Griffitt, Ulf Hermjakob, Kevin Knight, Philipp Koehn, Martha Palmer, and Nathan Schneider. 2013. Abstract Meaning Representation for sembanking. In Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse, pages 178–186, Sofia, Bulgaria. Association for Computational Linguistics.

Ofer Bronstein, Ido Dagan, Qi Li, Heng Ji, and Anette Frank. 2015. Seed-based event trigger labeling: How far can event descriptions get us? In Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers), pages 372–376.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D 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 Ziegler, Jeffrey Wu, Clemens Winter, Chris 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. In Advances in Neural Information Processing Systems, volume 33, pages 1877–1901. Curran Associates, Inc.

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

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

Yubo Chen, Liheng Xu, Kang Liu, Daoqian Zeng, and Jun Zhao. 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), pages 167–176.

Yunmo Chen, Tongfei Chen, Seth Ebner, Aaron Steven White, and Benjamin Van Durme. 2020. Reading the manual: Event extraction as definition comprehension. In Proceedings of the Fourth Workshop on Structured Prediction for NLP, pages 74–83, Online. Association for Computational Linguistics.

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

Arkabandhu Chowdhury, Mingchao Jiang, and Chris Jermaine. 2021. Few-shot image classification: Just use a library of pre-trained feature extractors and a simple classifier. abs/2101.00562.

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

Bhavana Dalvi, William W. Cohen, and Jamie Callan. 2012. Websets: extracting sets of entities from
the web using unsupervised information extraction. ArXiv, abs/1307.0261.

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

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 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), pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Xinya Du and Claire Cardie. 2020. Event extraction by answering (almost) natural questions. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 671–683, Online. Association for Computational Linguistics.

Xiaocheng Feng, Lifu Huang, Duyu Tang, Heng Ji, Bing Qin, and Ting Liu. 2016. A language-independent neural network for event detection. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pages 66–71, Berlin, Germany. Association for Computational Linguistics.

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

Karen Hambardzumyan, Hrant Khachatryan, and Jonathan May. 2021. WARP: Word-level Adversarial ReProgramming. In Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pages 4921–4933, Online. Association for Computational Linguistics.

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

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

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), 2160–2170, Melbourne, Australia. Association for Computational Linguistics.

Heng Ji and Ralph Grishman. 2008. Refining event extraction through cross-document inference. In Proceedings of ACL-08: Hlt, pages 254–262.

Zhengbao Jiang, Frank F. Xu, J. Araki, and Graham Neubig. 2020. How can we know what language models know? Transactions of the Association for Computational Linguistics, 8:423–438.

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

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

Viet Dac Lai and Thien Huu Nguyen. 2019. Extending event detection to new types with learning from keywords. arXiv preprint arXiv:1910.11368.

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

Brian Lester, Rami Al-Rfou, and Noah Constant. 2021. The power of scale for parameter-efficient prompt tuning. In EMNLP.

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

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

Qi Li, Heng Ji, and Liang Huang. 2013. Joint event extraction via structured prediction with global features. In Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 73–82, Sofia, Bulgaria. Association for Computational Linguistics.

Xiang Lisa Li and Percy Liang. 2021. Prefix-tuning: Optimizing continuous prompts for generation. 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), abs/2101.00190.
Shasha Liao and Ralph Grishman. 2010. Using document level cross-event inference to improve event extraction. In Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics, 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 the 58th Annual Meeting of the Association for Computational Linguistics, pages 7999–8009, Online. Association for Computational Linguistics.

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

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

Yaojie Lu, Hongyu Lin, Jin Xu, Xianpei Han, Jialong Tang, Annan Li, Le Sun, Meng Liao, and Shaoyi Chen. 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), pages 2795–2806, Online. Association for Computational Linguistics.

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.

David McClosky, Mihai Surdeanu, and Christopher D Manning. 2011. Event extraction as dependency parsing. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, pages 1626–1635.

Minh Van Nguyen, Viet Dac Lai, and Thien Huu Nguyen. 2021. Cross-task instance representation interactions and label dependencies for joint information extraction with graph convolutional networks.

Thien Huu Nguyen, Kyunghyun Cho, and Ralph Grishman. 2016. Joint event extraction via recurrent neural networks. In Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 300–309, San Diego, California. Association for Computational Linguistics.

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

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. Language models as knowledge bases? In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pages 2463–2473, Hong Kong, China. Association for Computational Linguistics.

Colin Raffel, Noam M. Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. JMLR.

Timo Schick and Hinrich Schütze. 2021a. Few-shot text generation with pattern-exploiting training.

Timo Schick and Hinrich Schütze. 2021b. It’s not just size that matters: Small language models are also few-shot learners. In Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics, pages 2339–2352.

Shirong Shen, Tongtong Wu, Guilin Qi, Yuan-Fang Li, Gholamreza Hafiziri, and Sheng Bi. 2021. Adaptive knowledge-enhanced bayesian meta-learning for few-shot event detection. In Findings of the Association for Computational Linguistics, page 2417–2429. Association for Computational Linguistics (ACL). Annual Meeting of the Association for Computational Linguistics and the International Joint Conference on Natural Language Processing 2021, ACL-IJCNLP 2021 ; Conference date: 01-08-2021 Through 06-08-2021.

Taylor Shin, Yasaman Razeghi, Robert L. Logan IV, Eric Wallace, and Sameer Singh. 2020. AutoPrompt: Eliciting knowledge from language models with automatically generated prompts. In Empirical Methods in Natural Language Processing (EMNLP).

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 erc: annotation of entities, relations, and events. In Proceedings of the the 3rd Workshop on EVENTS: Definition, Detection, Coreference, and Representation, pages 89–98.

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

David Wadden, Ulme Wennberg, Yi Luan, and Hannaneh Hajishirzi. 2019. Entity, relation, and event
extraction with contextualized span representations. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pages 5784–5789, Hong Kong, China. Association for Computational Linguistics.

Richard C Wang and William Cohen. 2009. Character-level analysis of semi-structured documents for set expansion. In Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing, pages 1503–1512.

Sijia Wang, Mo Yu, Shiyu Chang, Lichao Sun, and Lifu Huang. 2021a. Query and extract: Refining event extraction as type-oriented binary decoding. arXiv preprint arXiv:2110.07476.

Xiao Wang, Houye Ji, Chuan Shi, Bai Wang, Yanfang Ye, Peng Cui, and Philip S Yu. 2019. Heterogeneous graph attention network. In The World Wide Web Conference, WWW ’19, page 2022–2032, New York, NY, USA. Association for Computing Machinery.

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

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

Ziqi Wang, Xiaozhi Wang, Xu Han, Yankai Lin, Lei Hou, Zhiyuan Liu, Peng Li, Juanzi Li, and Jie Zhou. 2021b. CLEVE: Contrastive Pre-training for Event Extraction. In Proceedings of ACL-IJCNLP, pages 6283–6297, Online. Association for Computational Linguistics.

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

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

Bishan Yang and Tom M. Mitchell. 2016. Joint extraction of events and entities within a document context. In Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 289–299, San Diego, California. Association for Computational Linguistics.

Weizhe Yuan, Graham Neubig, and Pengfei Liu. 2021. BARTScore: Evaluating generated text as text generation. In Advances in Neural Information Processing Systems.
A  APEX prompt examples for ACE

Table 4 and Table 5 show APEX prompt examples for ACE events.

B  Prompt Token Selection

In our experiments, the event type names and event type structures are automatically extracted from the target event ontology, such as ACE (Linguistic Data Consortium, 2005), ERE (Song et al., 2015) and MAVEN (Wang et al., 2020a). The prototype seed triggers are automatically selected from the annotated data for supervised and few-shot event extraction. For zero-shot event extraction, we manually select $R$ words from the NLTK synonyms of each event type as its prototype seed triggers. The definitions and APEX prompts are based on the official annotation guides for each target event ontology (Linguistic Data Consortium, 2005; Song et al., 2015; Wang et al., 2020a) and the available definitions in FrameNet (Baker et al., 1998) with manual editing.

C  Learning Strategy

The learning strategy varies for supervised, few-shot, and zero-shot learning. For supervised learning, we optimize the following objective for event trigger detection: $L = \frac{1}{|T| |N|} \sum_{t \in T} \sum_{i=1}^{|N|} y_i \cdot \log \tilde{y}_i$, where $T$ is the set of target event types and $N$ is the set of tokens from the training dataset. $y_i$ denotes the ground truth label vector. For few-shot event detection, we optimize the model on both base training data set and the smaller training data set for novel event types: $L = \frac{1}{|T| |N^B|} \sum_{t \in T^B} \sum_{i=1}^{|N^B|} y_i \cdot \log \tilde{y}_i - \beta \frac{1}{|T| |N^N|} \sum_{t \in T^N} \sum_{i=1}^{|N^N|} y_i \cdot \log \tilde{y}_i$, where $T^B$ and $N^B$ denote the set of base event types and tokens from the base training data set, respectively. $T^N$ is the set of novel event types. $N^N$ is the set of tokens from the training data set for novel event types. $\beta$ is a hyper-parameter to balance the two objectives. For zero-shot event detection, as we only have the base training data set, we minimize the following objective: $L = \frac{1}{|T| |N^B|} \sum_{t \in T^B} \sum_{i=1}^{|N^B|} y_i \cdot \log \tilde{y}_i$.

D  Dataset

After defining the base and novel event types, we create the training, validation, and evaluation split for all three datasets. We use the sentences with only base event type mentions as the base training data set for few-shot event detection, and randomly select 10 sentences with novel event type mentions as the additional smaller training data set. We use the sentences with both base and novel event type mentions as the development set and use the remaining sentences with only novel event type mentions as the evaluation dataset. For few-shot and zero-shot event detection and remove the instances with novel event mentions from the training set. We randomly split the sentences of each set for both zero-shot and few-shot event detection. Table 6 shows the detailed data statistics for all the three datasets under the few-shot and zero-shot event extraction settings.

E  Hyperparameters and Evaluation

For a fair comparison with the previous baseline approaches, we use the same pre-trained bert-large-uncased model for fine-tuning and optimizing our model with BertAdam. For supervised event detection, we optimize the parameters with grid search: training epoch is 3, learning rate $\in \{3e-6, 1e-4\}$, training batch size $\in \{8, 12, 16, 24, 32\}$, dropout rate $\in \{0.4, 0.5, 0.6\}$. The running time is up to 3 hours on one Quadro RTX 8000. For evaluation, we use the same criteria as previous studies (Li et al., 2013; Chen et al., 2015; Nguyen et al., 2016; Lin et al., 2020): an event mention is correct if its span and event type match a reference event mention.
| Event Rep Type | Comprehensive Prompt |
|----------------|-----------------------|
| **Business:** Declare-Bankruptcy | Declare Bankruptcy [SEP] bankruptcy bankruptcies bankrupting [SEP] Organization request legal protection from debt collection at a Place |
| **Business:** End-Org | End Organization [SEP] dissolving disbanded [SEP] an Organization goes out of business at a Place |
| **Business:** Merge-Org | Merge Organization [SEP] merging merger [SEP] two or more Organizations come together to form a new organization at a Place |
| **Business:** Start-Org | Start Organization [SEP] founded [SEP] an Agent create a new Organization at a Place |
| **Conflict:** Attack | Attack [SEP] invaded airstrikes overthrew ambushed [SEP] An Attacker physically attacks a Target with Instrument at a Place |
| **Conflict:** Demonstrate | Demonstrate [SEP] demonstrations protest strikes riots [SEP] Entities come together in a Place to protest or demand official action |
| **Contact:** Meet | Meet [SEP] reunited retreats [SEP] two or more Entities come together at same Place and interact in person |
| **Contact:** Phone-Write | Phone Write [SEP] emailed letter [SEP] phone or written communication between two or more Entities |
| **Justice:** Acquit | Acquit [SEP] acquitted [SEP] a trial of Defendant ends but Adjudicator fails to produce a conviction at a Place |
| **Justice:** Appeal | Appeal [SEP] appeal [SEP] the decision for Defendant of a court is taken to a higher court for Adjudicator review with Prosecutor |
| **Justice:** Arrest-Jail | Arrest Jail [SEP] arrested locked [SEP] the Agent takes custody of a Person at a Place |
| **Justice:** Charge-Indict | Charge Indict [SEP] indictment [SEP] a Defendant is accused of a crime by a Prosecutor for Adjudicator |
| **Justice:** Convict | Convict [SEP] pled guilty convicting [SEP] an Defendant found guilty of a crime by Adjudicator at a Place |
| **Justice:** Execute | Execute [SEP] death [SEP] the life of a Person is taken by an Agent at a Place |
| **Justice:** Extradite | Extradite [SEP] extradition [SEP] a Person is sent by an Agent from Origin to Destination |
| **Justice:** Fine | Fine [SEP] payouts financial punishment [SEP] a Adjudicator issues a financial punishment Money to an Entity at a Place |
| **Justice:** Pardon | Pardon [SEP] pardoned lift sentence [SEP] an Adjudicator lifts a sentence of Defendant at a Place |
| **Justice:** Release-Parole | Release Parole [SEP] parole [SEP] an Entity ends its custody of a Person at a Place |
| **Justice:** Sentence | Sentence [SEP] sentenced punishment [SEP] the punishment for the defendant is issued by a state actor |
| **Justice:** Sue | Sue [SEP] lawsuits [SEP] Plaintiff initiate a court proceeding to determine the liability of a Defendant judge by Adjudicator at a Place |
| **Justice:** Trial-Hearing | Trial Hearing [SEP] trial hearings [SEP] a court proceeding initiated to determine the guilty or innocence of a Person with Prosecutor and Adjudicator at a Place |
| **Life:** Be-Born | Be Born [SEP] childbirth [SEP] a Person is born at a Place |
| **Life:** Die | Die [SEP] deceased extermination [SEP] life of a Victim ends by an Agent with Instrument at a Place |

Table 4: APEX templates for ACE event types

1296
| Event Rep Type | Comprehensive Prompt |
|----------------|-----------------------|
| Life:Divorce   | Divorce [SEP] people divorce [SEP] two Person are officially divorced at a place |
| Life:Injure    | Injure [SEP] hospitalised paralyzed dismember [SEP] a Victim experiences physical harm from Agent with Instrument at a Place |
| Life:Marry     | Marry [SEP] married marriage marry [SEP] two Person are married at a Place |
| Movement:Transport | Transport [SEP] arrival travels penetrated expelled [SEP] an Agent moves an Artifact from Origin to Destination with Vehicle at Price |
| Personnel:Elect | Elect [SEP] reelected elected election [SEP] a candidate Person wins an election by voting Entity at a Place |
| Personnel:End-Position | End Position [SEP] resigning retired resigned [SEP] a Person stops working for an Entity or change office at a Place |
| Personnel:Nominate | Nominate [SEP] nominate [SEP] a Person is nominated for a new position by another Agent at a Place |
| Personnel:Start-Position | Start Position [SEP] hiring rehired recruited [SEP] a Person begins working for an Entity or change office at a Place |
| Transaction:Transfer-Money | Transfer Money [SEP] donations reimbursing deductions [SEP] transfer Money from the Giver to the Beneficiary or Recipient at a Place |
| Transaction:Transfer-Ownership | Transfer Ownership [SEP] purchased buy sell loan [SEP] buying selling loaning borrowing giving receiving of Artifacts from Seller to Buyer or Beneficiary at a Place |

Table 5: APEX templates for ACE event types (continued)

| Dataset          | ACE05-E+ | ERE-EN | MAVEN |
|------------------|----------|--------|-------|
| # Types          |          |        |       |
| Base              | 18       | 25     | 120   |
| Novel            | 10       | 10     | 45    |
| # Mentions       |          |        |       |
| Base              | 3572     | 5449   | 93675 |
| Novel            | 1724     | 3183   | 3201  |
| Train            |          |        |       |
| Few-shot         | 3216     | 3886   | 88085 |
| Zero-shot        | 3116     | 3786   | 87635 |
| Validation       |          |        |       |
| (51%/49%)        | 900      | 2797   | 3883  |
| (53%/47%)        |          |        |       |
| (71%/23%)        |          |        |       |
| Evaluation       | 1195     | 2012   | 1652  |

Table 6: Data statistics for ACE2005, ERE and MAVEN datasets under few-shot/zero-shot event detection settings.
ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
   Section 8

☑ A2. Did you discuss any potential risks of your work?
   Section 8

☑ A3. Do the abstract and introduction summarize the paper’s main claims?
   Abstract and Section 1

☒ A4. Have you used AI writing assistants when working on this paper?
   Left blank.

B ☑ Did you use or create scientific artifacts?
   Section 5 and Appendix C

☑ B1. Did you cite the creators of artifacts you used?
   Section 5 and Appendix C

☑ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
   Section 5

☑ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
   Section 5

☒ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
   Left blank.

☑ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
   Section 5

☑ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
   Appendix C

C ☑ Did you run computational experiments?
   Section 5 and Appendix D

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
   Appendix D

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.
C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
   *Section 5 and Appendix D*

C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
   *Appendix D*

C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
   *Appendix B*

D Did you use human annotators (e.g., crowdworkers) or research with human participants?
   *Left blank.*

   D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
      *No response.*

   D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants’ demographic (e.g., country of residence)?
      *No response.*

   D3. Did you discuss whether and how consent was obtained from people whose data you’re using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
      *No response.*

   D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
      *No response.*

   D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
      *No response.*