ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) | 979-8-3503-4485-1/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICASSP48485.2024.10448190

UNLEASHING TRIGGER-FREE EVENT DETECTION: REVEALING EVENT CORRELATIONS VIA A CONTRASTIVE DERANGEMENT FRAMEWORK

Hongzhan Lin[♠] *Haiqin Yang*[♡✉] *Ziyang Luo*[♠] *Jing Ma*[♠✉]

♠ Hong Kong Baptist University ♡International Digital Economy Academy

ABSTRACT

Event detection (ED), detecting events with specified types observed in given texts, is critical to many downstream applications. Existing ED methods generally require high-quality triggers annotated by human experts, which is labor-intensive, especially for those nontrivial texts about breaking events. In this paper, we propose a novel trigger-free ED framework that detects multiple events from a given text without pre-defined triggers. Specifically, we first shed light on the event correlations with input texts using a joint embedding paradigm. Next, we devise derangement-based contrastive learning to model fine-grained correlations between multi-event instances. Since events in training benchmarks are usually imbalanced, we further design a simple yet effective event derangement module for balanced training. Experimental results on two benchmarks show that our trigger-free method is remarkably competitive to state-of-the-art trigger-based baselines.

Index Terms— Event detection, trigger-free paradigm, multi-event correlations, balanced training

1. INTRODUCTION

The era of information explosion has necessitated the development of effective automatic event detection systems. Event detection (ED) aims to spot events together with specific types from texts, which assists human beings in reading and digesting mass information. In recent years, ED approaches have been successfully applied in many NLP tasks [1, 2, 3, 4], e.g., adverse drug event discovery, rumor events detection, court decision event identification, financial event extraction, etc.

Previous work proposed to detect event types with a set of annotated and pre-defined triggers [5], where the triggers can be words or phrases providing the *most clear* indication of an event occurrence [6]. Table 1 exemplifies two sentences with detected events and triggers. However, trigger-based methods [5, 7] are effective only if the trigger annotations are of high quality, which requires daunting manual effort [8], especially for detecting events in some texts about trending topics or breaking news. Therefore, it is crucial to develop trigger-free paradigms for event detection. Recently, [9] designed a sequence-to-structure generation model learning from coarse

s_1: And they sent him to Baghdad and killed. **Triggers**: sent, killed **Events**: Transport, Die
s_2: An FBI team is heading to Saudi Arabia to help investigate the attacks. **Triggers**: heading, attacks **Events**: Transport, Attack

Table 1. Examples of Event Detection task where events and triggers in the same color are of the same type.

parallel text-record annotations without the labeled trigger off-sets, but trigger words are still required to be specified manually. To alleviate tedious trigger annotations, [10] proposed an LSTM-based model for event detection using a binary classification for all the possible event types without explicit triggers, which however leads to an exponential-sized output space and the inductive bias of independent event type might suppress the multi-event detection.

In this paper, we assume that the semantic correlations between the event types and given context could implicitly infer all the possible events, to relieve the heavy dependency on trigger annotations. Intuitively, from Table 1, we observe that: 1) The event types are semantically closer to their corresponding triggers, so the correlations of context to all event types can strengthen discriminative feature learning for event detection, and the event-event correlations can further indicate latent inter-dependent features among multiple events; 2) Although two instances can share some common events, there may be some unique events in each instance, which indicates distinct trigger clues in the context. For example, both instances s_1 and s_2 contain the same event "Transport" but their trigger sets differ, where the trigger co-occurrence (e.g., "sent" and "killed" from s_1, "heading" and "attacks" from s_2) in different individual text instances can be largely varied. Therefore, we aim to capture such intricate event correlations among the multi-event instances to implicitly infer the co-occurrence clues of potential triggers specific to each instance, substantially facilitating the trigger-free ED task with a more general framework.

To this end, we propose a Contrastive Derangement framework with Event correlations (CoDE) to detect all events without triggers. Specifically, we first introduce a joint embedding paradigm to exploit the semantic correlations of the event types integrated with context information. Then we propose derangement-based contrastive learning to distinguish further

✉ Corresponding authors: hqyang@ieee.org, majing@hkbu.edu.hk.

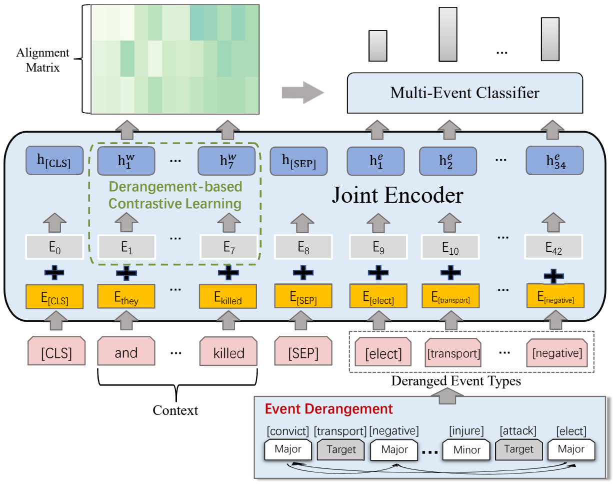

Fig. 1. The framework of our CoDE model.

co-occurrence inherent in implicit trigger clues from the holistic context in an unsupervised manner, which can enhance representation learning for instances with overlapping events. Finally, a multi-event classifier is devised to align the context to event types. On the other hand, since the events are unevenly distributed in real-world data following the Matthew effect [11], we develop an event derangement module for balanced model training by avoiding excessive updates on the major events. Experiments on two benchmarks demonstrate the effectiveness of our method for event detection. Further gradient explanation indicates that our trigger-free model can persuasively spot and link triggers to the corresponding events.

2. TRIGGER-FREE EVENT DETECTION

2.1. Problem Definition

Suppose an event detection dataset with N sentence-event pairs $\{(x_i, y_i)\}_{i=1}^{N}$, where $x_i = w_{i1}\, w_{i2}\, \ldots\, w_{i|x_i|}$ is a sentence represented as a word sequence, $y_i \subseteq \mathcal{S}$ records all the event types observed in x_i, and $\mathcal{S} = \{e_1, e_2, \ldots, e_n\}$ denotes the event tag set that consists ideally of all possible n event types, including an additional "negative" event for sentences without any event. The goal of *trigger-free event detection* [10] is to train a model for detecting all the corresponding event type(s) observed in a sentence, which could be formulated as a multi-label classification task.

2.2. Our CoDE Model

Figure 1 illustrates an overview of our proposed framework, which consists of four parts: (1) a Joint Encoder to learn the global embedding between texts and events; (2) Derangement-based Contrastive Learning (DCL) to distinguish indicative clues unique to individual instance; (3) an Event Derangement Module (EDM) to mitigate the imbalanced event types; and (4) a Multi-Event Classifier to align the context to event types.

Joint Encoder. We utilize BERT [12] to learn the token embeddings due to its superior performance on sequence representation learning. For classification tasks, a special token "[CLS]" is put at the beginning of the text and the output vector of "[CLS]" is designed to correspond to the final text representation. Different from this habitual operation, we unite the input text with all event types, which are packed into a single sequence and separated by "[SEP]". With both texts and event tokens as the input, we utilize BERT as the context-event joint encoder to learn the event correlations with the text.

Given a dataset with its initial sequence of event types $S_{\text{init}} = \{e_1, \ldots, e_n\}$, for a sentence $x = \{w_1, \ldots, w_{|x|}\}$, we construct $\{[\text{CLS}], w_1, \ldots, w_{|x|}, [\text{SEP}], [e_1], \ldots, [e_n]\}$ as input. Via BERT, we obtain the output contextualized hidden representations as $\{h_{[\text{CLS}]}, h_1^w, \ldots, h_{|x|}^w, h_{[\text{SEP}]}, h_1^e, \ldots, h_n^e\}$. Such an input structure could guarantee that context and event types are embedded jointly in the same space and explicitly make the token-level representations aware of semantic correlations of event types.

DCL. Generally, the co-occurrence of different event triggers is specific to each instance even with overlapping events. To model such complex correlations among the multi-event instances, we perform unsupervised contrastive learning to better capture the semantic clues of triggers from the holistic discourse. Theoretically, on top of the average pooling of the hidden states $H_w = \{h_1^w, \ldots, h_{|x|}^w\} \in \mathbb{R}^{|x| \times d}$ of the text x, we construct the contrastive objective as follows:

$$\mathcal{L} = -\log \frac{exp(\tilde{h}_i \cdot \ddot{h}_i)}{\sum_{k=1}^{B} \mathbb{1}_{[i \neq k]} (exp(\tilde{h}_i \cdot \tilde{h}_k) + exp(\tilde{h}_i \cdot \ddot{h}_k))}, \quad (1)$$

where $\tilde{h}_i \in \mathbb{R}^d$ is the mean-pooled representations of the sentence x_i in a batch of B training examples, and $\ddot{h}_i$ denotes its augmented example. For data augmentation, inspired by [13], we implement context derangement by passing the shuffled position ids of context to the embedding layer while keeping the order of the token ids unchanged, for producing hard augmented examples in an unsupervised manner while avoiding the co-occurrence between potential trigger tokens being destroyed. In this way, we could distinguish the potential trigger clues of individual instances, by maximizing the agreement between one sample and its augmented version with the same event triggers as it, while keeping it distant from other negative samples in the same batch.

EDM. During training, the event derangement module is further proposed to mitigate the imbalanced learning issue. In particular, we first characterize each event type as a major or minor event by drawing the practice of the previous literature [14]. we derange the sequence S_{init} by switching a set of sampled major events other than the golden truth, which is conducted with probability q only when there is one major event being the target (i.e., the ground-truth) at least. The event derangement module can prohibit the model from excessively learning major events, which works similarly to

Algorithm 1 Event Derangement

Input: Input sentence x; The initial event sequence S_{init}; The sequence of all event types in descending order S_{SA}; Possibility q; Number r

Output: Deranged sequence of event tokens S_{O}

1: Initialize E_{GT} as the set of the ground truth event types implied by x
2: Initialize E_{D} with r events that are not in E_{GT}, from the beginning of S_{SA}
3: Initialize $E_{tmp} = \emptyset$ `# a helper set to record the selected event types during derangement`
4: Initialize $S_{\text{O}} = []$
5: Generate $rand$ uniformly from [0, 1]
6: **if** $E_{\text{GT}} \cap E_{\text{Major}} \neq \emptyset$ and $rand < q$ **then**
7:     **for** e_{curr} in S_{init} **do**
8:         **if** e_{curr} in E_{D} **then**
9:             Randomly select e from E_{D} and $e \neq e_{curr}$ and $e \notin E_{tmp}$
10:             Append e to S_{O}
11:             Add e to E_{tmp}
12:         **else**
13:             Append e_{curr} to S_{O}
14:         **end if**
15:     **end for**
16: **else**
17:     $S_{\text{O}} = S_{\text{init}}$
18: **end if**
19: Return S_{O}

under-sampling the training instances of major events to make the training process more balanced.

We provide the pseudocode about EDM in Algo. 1. In our algorithm, we first sort all event types in descending order with respect to the number of instances in each class and obtain the sorted sequence S_{SA}. The set of candidate event types E_{D} only consists of r events from the beginning of the sequence S_{SA} (i.e., top-r event types in frequency), excluding those in E_{GT} (see line 2 of Algo. 1). From line 6 of Algo. 1, we can know that the derangement procedure is conducted with probability q only when the target (i.e., the ground-truth) events are the major events [14] (i.e., the top event types accounting for half of the total instances). From lines 7-10 of Algo. 1, we derange the sequence S_{init} by switching different event types in E_{D}.

Multi-Event Classifier. To explicitly model token-level semantic alignment between text and event types for context reasoning, we evolve contextual hidden states into event types:

$$\begin{aligned} \mathcal{A} &= \text{softmax}\left(H_w {H_e}^{\top}\right) \in \mathbb{R}^{|x| \times n}, \\ H_e^w &= \text{ReLU}\left(\mathcal{A}^{\top} H_w W\right) \in \mathbb{R}^{n \times d}, \end{aligned} \tag{2}$$

where H_e^w represents each event token by aggregating features of context tokens, $W \in \mathbb{R}^{d \times d}$ is a trainable matrix. $\mathcal{A}$ is the alignment matrix between context and event types. $H_e = \{h_1^e, \ldots, h_n^e\}$ denotes the hidden states of the event types. Finally, we use an Add&Norm layer to fuse the features of event types and then make a prediction:

$$\begin{aligned} H_o &= \text{LayerNorm}\left(H_e + H_e^w\right), \\ \hat{p} &= \text{softmax}\left(\text{MLP}\left(H_o\right)\right), \end{aligned} \tag{3}$$

where H_o is the enhanced representations of event types and $\hat{p} \in \mathbb{R}^n$ is a low-dimensional vector for event prediction. Our model can be trained by minimizing the Cross-Entropy loss:

$$\mathcal{L}_{CE} = -\sum_{i=1}^{n}\left(\,\overline{p}_i \log\left(\hat{p}_i\right) + (1 - \overline{p}_i)\log(1 - \hat{p}_i)\right), \tag{4}$$

where $\hat{p}_i = P(e_i|x)$ is the probability of e_i predicted by the model, $\overline{p}_i \in \{0, 1\}$ is the true categorical information of e_i.

Model Training. We jointly train the model with the cross-entropy and contrastive objectives on each sample:

$$\mathcal{L} = \mathcal{L}_{CE} + \gamma\mathcal{L}, \tag{5}$$

where γ is an automatic ramp-up adjustment [15] slowly increasing from 0 to 1 during training. The derangement probability q is set to 0.2 and the number of deranged tokens r is set to 24. We implement the *bert-base-uncased* model as the backbone. The batch size is set to 8. ADAM [16] is the optimizer with the learning and dropout rates of 2e-5 and 0.1.

3. EXPERIMENTS

3.1. Experimental Setups

Datasets and Evaluation. We conducted experiments on two benchmark datasets: ACE2005 [17] and TAC2015 [18]. ACE2005 consists of 8 event main types and 33 subtypes. The data distribution of event subtypes is heavily imbalanced (Imbalance Ratio, IR≈605.5). For a fair comparison, we follow the evaluation setting of previous work [10, 19]. TAC2015 is annotated with event nuggets in 38 types. We process the data following [20]. The data distribution is more balanced than ACE2005 with IR ≈ 61.5. We use micro-average **P**recision, **R**ecall, and **F1** scores to evaluate the model performance.

Baselines. We compare our model with several representative state-of-the-art baselines: in terms of low-resource ED model: TBNNAM [10], TEXT2EVENT [9], DEGREE [21]; And for trigger-based ED model: BERT_Trigger [22], DMCNN [6], JMEE [23], GCN-ED [24], HPNet [25], DNR [5]. We report the results in the same data setting from the corresponding paper, where trigger-based ED models use annotated triggers.

3.2. Event Detection Performance

Table 2 reports the overall performance of our proposed model ED_CoDE, ED_DRC (w/o DCL) and ED_RC (w/o DCL&EDM), on the ACE2005 and TAC2015, respectively ($p < 0.05$ under t-test). It shows that although the vanilla framework removing components, ED_RC, does not have access to the triggers, it could outperform the low-resource ED models in the first group and attain a competitive performance

Methods	ACE2005 (%)			TAC2015 (%)		
	P	R	F1	P	R	F1
TBNNAM	76.2	64.5	69.9	-	-	-
TEXT2EVENT	69.6	74.4	71.9	-	-	-
DEGREE	-	-	73.3	-	-	-
BERT_Trigger†	71.7	73.7	72.3	-	-	-
DMCNN†	75.6	63.6	69.1	71.3	45.8	55.8
JMEE†	76.3	71.3	73.7	69.7	47.0	56.1
GCN-ED†	77.9	68.8	73.1	70.3	50.6	58.8
HPNet†	80.1	75.7	77.8	70.9	54.8	61.8
DNR†	81.2	82.4	**81.8**	71.2	60.9	65.7
BERT Finetune	72.8	68.7	70.7	75.8	59.9	67.0
Our ED_RC	76.9	72.3	74.7	74.3	63.4	68.4
Our ED_DRC	79.5	76.8	78.1	**78.1**	62.7	69.6
Our ED_CoDE	78.3	80.9	**79.6**	76.8	**66.9**	**71.5**

Table 2. Event detection results on ACE2005 and TAC2015, respectively. † indicates requiring given annotated triggers.

Model	DMCNN	JMEE	DNR	CoDE
1/1	74.3	75.2	<u>78.9</u>	**75.8**
1/N	50.9	72.7	<u>84.0</u>	**83.4**
All	69.1	73.7	<u>81.8</u>	**79.6**

Table 3. Results of F1 score (%) on single-event sentences (1/1) and multi-event sentences (1/N).

compared with the trigger-based baselines in the second group. The result also demonstrates that the **joint embedding** paradigm with the **context-event alignment** in the multi-event classifier is capable of learning correlations between context and event types even without trigger annotations. After including EDM, ED_DRC could further improve the performance with remarkable gains, which demonstrates the effectiveness of the **event derangement** module. It could be observed that state-of-the-art ED systems require annotated triggers. Despite the more challenging setting, our trigger-free ED_CoDE could achieve a promising F1 score to the trigger-based model DNR, which explicitly relies on additional external knowledge of trigger annotations, on ACE2005 and even much better performance than it on TAC2015. It indicates that the **DCL** mechanism could effectively alleviate the trigger absence issue. The overall performance consistently verifies that the components in our framework complement each other.

3.3. Qualitative Analysis

Performance w.r.t. Multiple Events. In this subsection, we further dive into the ability of our model to accurately identify multiple events. Following the setting of previous studies [23], we divide the test data into **1/1** and **1/N** parts to evaluate the effectiveness of our model against different numbers of events. Table 3 reads that our trigger-free model could achieve competitive performance compared with trigger-based SOTA baselines. Especially in the 1/N data split, the performance of our model is extremely closer to the trigger-based model DNR. Furthermore, the performance in the 1/N data split exceeds that in 1/1 because our model explicitly makes the token-level representations aware of semantic correlations of multiple event types. Generally, the event type is determined by the corresponding event arguments, usually, the triplet of {subject, predicate, object}, from a given sentence. If we need to annotate triggers, we need at least one more pass of scanning the sentence to determine the trigger word(s) and the corresponding position(s) after determining the event type. Hence, the time cost is approximately double. Thus, our model could accurately determine the multiple event types in one sentence with approximately the same degree of F1 score as trigger-based models meanwhile saving at least half the time cost due to our trigger-free paradigm.

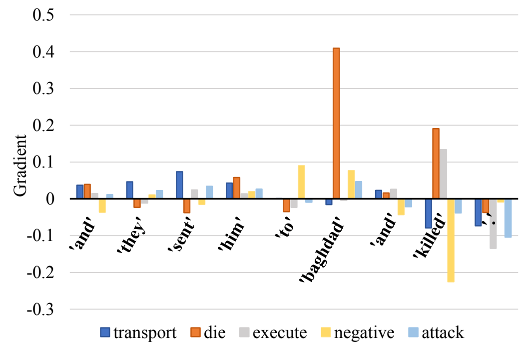

Fig. 2. Gradient visualization of words in a sentence from ACE2005 with respect to five typical event types.

Gradient Visualization. To investigate how the model understands input texts and identifies event types, we computed the gradients concerning the embeddings of the text within the intermediate model variant ED_DRC. Those gradients quantify the influence of changes in the tokens on the predictions, which has been verified as a more stable method to explain the attention-based model [26] than the attention weights in BERT. Here, we pick the example in Sec. 1 and select five events as shown in Figure 2. It shows that for the target event of "Die", the model can automatically focus on its trigger word "killed" instead of the trigger word "sent", while for the target event "Transport", the trigger "sent" is also notified by the model. In terms of other non-target events, the model attains low gradients on the triggers or gets high gradients on unrelated tokens (e.g., "to"). But the larger gradient on the location "baghdad" than trigger "killed" for the "Die" target event, highlights the further necessity to augment ED_DRC with DCL mechanism for better trigger discovery. Overall, these interesting observations indicate that the model can successfully link triggers to the corresponding target events, with great potential to extract the key structural information of events with gradients.

4. CONCLUSION

We dive into the limitation of trigger annotations for event detection and propose a novel contrastive derangement framework with event correlations towards trigger-free event detection, to overcome the challenges of the trigger absence, multi-event issue, and imbalanced event distribution. Results on two public benchmarks confirm the advantages of our model.

5. REFERENCES

[1] Nigel Collier, Son Doan, Ai Kawazoe, Reiko Matsuda Goodwin, Mike Conway, Yoshio Tateno, Quoc-Hung Ngo, Dinh Dien, Asanee Kawtrakul, Koichi Takeuchi, et al., "Biocaster: detecting public health rumors with a web-based text mining system," *Bioinformatics*, 2008.

[2] Erwin Filtz, María Navas-Loro, Cristiana Santos, Axel Polleres, and Sabrina Kirrane, "Events matter: Extraction of events from court decisions," *Legal Knowledge and Information Systems*, 2020.

[3] Xin Liang, Dawei Cheng, Fangzhou Yang, Yifeng Luo, Weining Qian, and Aoying Zhou, "F-hmtc: Detecting financial events for investment decisions based on neural hierarchical multi-label text classification," in *IJCAI*, 2020, pp. 4490–4496.

[4] Hongzhan Lin, Jing Ma, Liangliang Chen, Zhiwei Yang, Mingfei Cheng, and Chen Guang, "Detect rumors in microblog posts for low-resource domains via adversarial contrastive learning," in *NAACL*, 2022.

[5] Jinzhi Liao, Xiang Zhao, Xinyi Li, Lingling Zhang, and Jiuyang Tang, "Learning discriminative neural representations for event detection," in *SIGIR*, 2021.

[6] Yubo Chen, Liheng Xu, Kang Liu, Daojian Zeng, and Jun Zhao, "Event extraction via dynamic multi-pooling convolutional neural networks," in *ACL-IJCNLP*, 2015.

[7] Xiao Liu, He-Yan Huang, Ge Shi, and Bo Wang, "Dynamic prefix-tuning for generative template-based event extraction," in *ACL*, 2022, pp. 5216–5228.

[8] Ujjwala Anantheswaran, Himanshu Gupta, Mihir Parmar, Kuntal Kumar Pal, and Chitta Baral, "Edm3: Event detection as multi-task text generation," *arXiv preprint arXiv:2305.16357*, 2023.

[9] Yaojie Lu, Hongyu Lin, Jin Xu, Xianpei Han, Jialong Tang, Annan Li, Le Sun, Meng Liao, and Shaoyi Chen, "Text2event: Controllable sequence-to-structure generation for end-to-end event extraction," in *ACL*, 2021.

[10] Shulin Liu, Yang Li, Feng Zhang, Tao Yang, and Xinpeng Zhou, "Event detection without triggers," in *NAACL-HLT*, 2019, pp. 735–744.

[11] Matjaž Perc, "The matthew effect in empirical data," *Journal of The Royal Society Interface*, 2014.

[12] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova, "BERT: pre-training of deep bidirectional transformers for language understanding," in *NAACL-HLT*, 2019.

[13] Haejun Lee, Drew A Hudson, Kangwook Lee, and Christopher D Manning, "Slm: Learning a discourse language representation with sentence unshuffling," in *EMNLP*, 2020.

[14] Qi Dong, Shaogang Gong, and Xiatian Zhu, "Imbalanced deep learning by minority class incremental rectification," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2019.

[15] Jun Zhao, Tao Gui, Qi Zhang, and Yaqian Zhou, "A relation-oriented clustering method for open relation extraction," in *EMNLP*, 2021.

[16] Diederik P. Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," in *ICLR*, 2015.

[17] George R Doddington, Alexis Mitchell, Mark A Przybocki, Lance A Ramshaw, Stephanie M Strassel, and Ralph M Weischedel, "The automatic content extraction (ace) program-tasks, data, and evaluation.," in *LREC*, 2004.

[18] Joe Ellis, Jeremy Getman, Dana Fore, Neil Kuster, Zhiyi Song, Ann Bies, and Stephanie M. Strassel, "Overview of linguistic resources for the TAC KBP 2015 evaluations: Methodologies and results," in *TAC*. 2015, NIST.

[19] Jian Liu, Yubo Chen, Kang Liu, Wei Bi, and Xiaojiang Liu, "Event extraction as machine reading comprehension," in *EMNLP*, 2020.

[20] Haoruo Peng, Yangqiu Song, and Dan Roth, "Event detection and co-reference with minimal supervision," in *EMNLP*, 2016.

[21] I-Hung Hsu, Kuan-Hao Huang, Elizabeth Boschee, Scott Miller, Prem Natarajan, Kai-Wei Chang, and Nanyun Peng, "DEGREE: A data-efficient generation-based event extraction model," in *NAACL*, 2022.

[22] Xinya Du and Claire Cardie, "Event extraction by answering (almost) natural questions," in *EMNLP*, 2020.

[23] Xiao Liu, Zhunchen Luo, and He-Yan Huang, "Jointly multiple events extraction via attention-based graph information aggregation," in *EMNLP*, 2018.

[24] Thien Huu Nguyen and Ralph Grishman, "Graph convolutional networks with argument-aware pooling for event detection," in *AAAI*, 2018.

[25] Peixin Huang, Xiang Zhao, Ryuichi Takanobu, Zhen Tan, and Weidong Xiao, "Joint event extraction with hierarchical policy network," in *COLING*, 2020.

[26] Julius Adebayo, Justin Gilmer, Michael Muelly, Ian J. Goodfellow, Moritz Hardt, and Been Kim, "Sanity checks for saliency maps," in *NIPS*, 2018.