EDM3: Event Detection as Multi-task Text Generation

Ujjwala Anantheswaran* **Himanshu Gupta†** **Mihir Parmar**
Kuntal Kumar Pal **Chitta Baral**
Arizona State University
{uananthe, hgupta35, mparmar3, kkpal, chitta}@asu.edu

Abstract

We present EDM3, a novel approach for Event Detection (ED) based on decomposing and reformulating ED, and fine-tuning over its atomic subtasks. EDM3 enhances knowledge transfer while mitigating the error propagation inherent in pipelined approaches. EDM3 infers dataset-specific knowledge required for the complex primary task from its atomic tasks, making it adaptable to any set of event types. We evaluate EDM3 on multiple ED datasets, achieving state-of-the-art results on RAMS (71.3% *vs.* 65.1% F1), and competitive performance on WikiEvents, MAVEN ($\Delta = 0.2\%$), and MLEE ($\Delta = 1.8\%$). We present an ablation study over rare event types (<15 instances in training data) in MAVEN, where EDM3 achieves $\sim$ 90% F1. To the best of the authors' knowledge, we are the first to analyze ED performance over non-standard event configurations (i.e., multi-word and multi-class triggers). Experimental results show that EDM3 achieves $\sim$ 90% exact match accuracy on multi-word triggers and $\sim$ 61% prediction accuracy on multi-class triggers [1]. This work establishes the effectiveness of EDM3 in enhancing performance on a complex information extraction task.

1 Introduction

Event Detection (ED) involves characterizing events occurring in unstructured text, by recognizing their event triggers and classifying their event types. ED is used extensively for downstream tasks such as information retrieval (Kanhabua and Anand, 2016), event prediction (Souza Costa et al., 2020), and argument detection (Cheng and Erk, 2018). Existing methods for ED (Liu et al., 2018; Nguyen and Grishman, 2018) cannot easily leverage pre-trained semantic knowledge (Lai et al., 2020; Paolini et al., 2021), failing to identify complex events or function in low-resource scenarios (Chen et al., 2015; Nguyen et al., 2016). Additionally, they lack the ability to generalize across domains such as biomedicine or cybersecurity. (He et al., 2022; Satyapanich et al., 2020). As a result, they may handicap comprehensive event extraction (Liu et al., 2020; Huang et al., 2020).[2]

Sentence		Large-scale hostilities mostly **ended** with the cease-fire agreements after the 1973 Yom Kippur **War**.
Discriminative	**ED**	… [mostly] [ended] [with] … [Kippur] [War] … [O] [*process_end*] [O] … [O] [*military_operation*]
EDM3	**EI**	**ended** | **War**
	EC	*process_end* | *military_operation*
	ED	**ended**->*process_end* | **War**->*military_operation*

Figure 1: Comparing label formulation for ED output in traditional discriminative approaches vs. EDM3. In EDM3, EI (Event Identification) and EC (Event Classification) labels are analogous strings with event triggers and types respectively, while ED output is a string with all triggers and their types.

To overcome these challenges, we propose EDM3 (**E**vent **D**etection by **M**ulti-task Text Generation over **three** subtasks), a novel approach based on decomposing an intricate primary task (ED) into its constituent atomic subtasks. We hypothesize that these subtasks (EI, EC) are less reliant on domain-specific knowledge than on semantic similarities (Pustejovsky, 1991), and hence simpler to learn. EDM3 involves training on these subtasks simultaneously in a non-pipelined, multi-task fashion. This diverges from the traditional discriminative token classification paradigm (Fig. 1). Unlike concurrent works such as InstructUIE (Wang et al., 2023) and UIE (Lu et al., 2022), which propose a unified model over multiple disparate language tasks, EDM3 focuses on a single complex task. This approach thus provides a framework adapt-

*Now at Microsoft Corporation

[1]Data and source code are available at https://github.com/ujjwalaananth/EDM3_EventDetection

†Currently in Amazon (The work was done prior to joining Amazon)

[2]Extended related work is discussed in Appendix §A

*Proceedings of the 13th Joint Conference on Lexical and Computational Semantics (*SEM 2024)*, pages 438 - 451
June 20-21, 2024 ©2024 Association for Computational Linguistics

Approaches	Datasets	Tasks Covered			Domain Generalization	Comparative Performance
		Identification	Classification	Detection		
Liu et al. (2022)	ACE, MAVEN	✗	✗	✓	✗	SOTA on MAVEN
Veyseh et al. (2021)	ACE, RAMS, CysecED	✓	✗	✓	✓	SOTA on ACE and CysecED Competitive on RAMS
He et al. (2022)	MLEE	✗	✗	✓	✗	SOTA on MLEE
EDM3 (Ours)	MAVEN, MLEE WikiEvents, RAMS	✓	✓	✓	✓	SOTA on RAMS Competitive on MLEE & MAVEN Benchmark on WikiEvents

Table 1: Comparison of EDM3 with other SOTA approaches highlighting the advantages of our approach. Columns 'Identification', 'Classification', and 'Detection' denote which tasks can be performed independently and end-to-end with the same model. We provide additional information to contextualize the performance metrics.

able to any independent complex task that can be decomposed into subtasks.

To evaluate EDM3, we conduct extensive experiments on RAMS, WikiEvents, MAVEN, and MLEE datasets. EDM3 achieves an F1 score of 71.3% on RAMS, surpassing the SOTA score by 6.2% points. EDM3 also achieves a competitive macro F1 score of 60.1% on MAVEN, compared to 60.3% (SOTA). We benchmark ED performance on WikiEvents with 60.7% F1 score. Finally, EDM3 achieves a competitive result of 78.1% F1 against 79.9% (SOTA) on the biomedical MLEE dataset. While other approaches use domain-specific embeddings and hand-crafted features, EDM3 uses a vanilla T5 model to obtain these results, supporting our hypothesis. Table 1 highlights the advantages of EDM3 over previous SOTA approaches.

We conduct investigations along multiple lines of inquiry to explore the efficacy of EDM3. We observe that our multi-tasking approach improves ED performance by 3-6%. We explore the efficacy of EDM3 in low-resource scenarios (evaluating rare event types). Experimental results reveal scores of $\sim 90\%$ F1 achieved over rare event types. We also evaluate its performance over multi-word and multi-class triggers, which while lacking in benchmark datasets, are common in real-world data. EDM3 achieves $\sim 90\%$ exact match accuracy on multi-word triggers and $\sim 61\%$ prediction accuracy on multi-class triggers. Finally, we discuss the importance of multi-sentence context. In summary, our contributions are as follows:

1. We propose EDM3, a novel training paradigm that generatively reformulates ED and its subtasks, and trains a single multi-task model that can perform them concurrently.
2. We obtain SOTA or competitive performances over various datasets across multiple domains.
3. Our analysis shows that EDM3 performs well for low-resource scenarios as well as non-standard event configurations.

2 Proposed Method

Given an input instance containing diversely-typed event triggers, we aim to capture all triggers present. We reformulate ED and its subtasks as sequence generation tasks. We use instructional prompts to train a model on all 3 generative tasks jointly to create a single multi-task model.

Task Decomposition ED is a multi-level task requiring both event identification and classification, which sequence labeling approaches conduct in a single step. We manually decompose ED into independent tasks to be carried out in parallel with the primary task, to augment the training process.

Generative reformulation The task labels are converted to delimited strings following a consistent pattern. The number of unique event types and triggers for an instance may differ, making all tasks notably distinct from one another, as opposed to ED being a linear combination of EI and EC.

Event Identification/Classification We represent the task output as a singly-delimited sequence of labels. An instance with x unique triggers and y unique event types would have the following label representations for the EI and EC tasks respectively:

$$T_1 \mid T_2 \mid T_3 \ldots T_x$$
$$E_1 \mid E_2 \mid E_3 \ldots E_y$$

Where T_i is the i^{th} event trigger occurring in an input instance and E_i is the i^{th} type of event occurring in the instance.

Event Detection Each label for ED contains 2 components: event trigger and type. The task output can be represented as a doubly-delimited se-

quence of events. We use -> as a delimiter between trigger and type. For an instance with x events:

$$T_1\text{->}E_1 \mid T_2\text{->}E_2 \mid T_3\text{->}E_3 \ldots T_x\text{->}E_x$$

Where T_x is the x^{th} event trigger and is of type E_x. For an example of an instance showing the reformulated outputs for all tasks, see Fig. 1.

Multi-Task Learning We posit that when trained over ED alongside its atomic tasks, a multi-task model gains significant transferable knowledge. In the case of rarer event types, modeling Event Classification (EC) separately improves the model's recognition of instances containing these events - leading to improved identification and detection. We use task-specific instructional prompts (natural language descriptions of how to perform each task with examples) to improve multi-tasking. To craft these instructional prompts, we follow the approach detailed by Wang et al. (2022b). The task-specific prompts and examples can be found in §B. For an example, see Fig. 2 in §C.

3 Results and Analysis

3.1 Results

We use EDM3 to train T5-base (220M). For experimental details, see §C. To compare our method fairly with established baselines, we evaluate our predictions by converting them to token-level labels. We report the average performance over 5 experimental runs.

RAMS We achieve 71.33% F1 score, which surpasses GPTEDOT by 6.2% (Table 2). Furthermore, the difference between precision and recall is much lower, indicating greater robustness.

WikiEvents We establish the benchmark performance of 60.7% F1 score (Table 3) on this dataset. We use single-task ED performance as a baseline to contextualize the benefits of EDM3. Over sentences with at least one event, we observe that the performance increases from 58.71% to 64.31% (Table 6) We show an example of improved ED using EDM3 in §C.1.

MAVEN We obtain a maximum F1 score of 62.66% (Table 4) which is influenced by severe class imbalance in the dataset. The competitive macro F1 score (60.1% vs. 60.3%) indicates better performance on rare classes. EDM3 also shows significant advantages in performing ED on multi-word triggers (Table 7 in §C).

Model	P	R	F1
DMBERT (Wang et al., 2019)	62.6	44.0	51.7
GatedGCN (Lai et al., 2020)	66.5	59.0	62.5
GPTEDOT (Veyseh et al., 2021)	55.5	**78.6**	65.1
EDM3	**71.6**	71.0	**71.3**

Table 2: Results on RAMS. All previous models are sentence-level BERT-based models.

Model	P	R	F1	W1
Single-task	60.0	49.6	54.3	52.1
EDM3	60.8	60.6	**60.7**	59.4

Table 3: Results on WikiEvents. W1: Weighted F1 %

Model	P	R	F1	F1*
SaliencyED (Liu et al., 2022)	**64.9**	**69.4**	**67.1**	60.3
EDM3	60.1	65.5	62.7	60.1

Table 4: Results on MAVEN. All results are on the publicly-available dev split. F1*: Macro F1 %

Model	P	R	F1
SVM2 (Zhou and Zhong, 2015) *	72.2	82.3	76.9
Two-stage (He et al., 2018) *	79.2	80.3	79.8
EANNP (Nie et al., 2015)	71.0	**84.6**	77.2
LSTM + CRF (Chen, 2019)	81.6	74.3	77.8
LSTM + CRF (Chen, 2019) **	81.8	77.7	79.7
BiLSTM + Att (He et al., 2022)	**82.0**	78.0	**79.9**
EDM3	75.9	80.4	78.1

Table 5: Results on MLEE dataset. * models using handcrafted features. All neural network-based models here use domain-specific embeddings. ** results when 4 biomedical datasets are used for transfer learning.

MLEE We compare with 1) labour-intensive approaches requiring creation of handcrafted features and 2) neural network-based models that use domain-specific embeddings obtained by parsing Pubmed or Medline abstracts. Our domain-agnostic approach achieves 78.1% F1 score, competitive with more sophisticated, domain-specific approaches (See Table 5). Our model also has higher recall (80.4%) than most approaches.

3.2 Analysis

In this work, we conduct various experiments to assess our approach in different scenarios.

Multi-tasking over EI and EC improves performance over ED Without instructional prompts,

Dataset	Single-task		EDM3 (tags)		EDM3 (instr)	
	All	Pos	All	Pos	All	Pos
MLEE	71.07	72.20	74.57	75.82	**77.09**	**78.45**
RAMS	63.21	63.21	67.66	67.66	**69.53**	**69.53**
MAVEN	58.10	59.18	62.29	63.56	**62.40**	**63.66**
WikiEvents	54.31	58.47	56.77	61.35	**58.71**	**64.31**

Table 6: Results on all datasets. Single-task: results using ED for training. EDM3 (tags): results from training with EI, EC, and ED. EDM3 (instr): using instructional prompts. All: performance on all input instances. Pos: performance on only event-containing instances.

EDM3 improves performance by at least 3% over single-tasking for all datasets. This can be attributed to the success of the subtask-level multi-tasking paradigm, with the improved performance and fewer false negatives due to training the model over EI and EC. Table 6 documents the metrics for single-task and multi-task models over all datasets.

EDM3 is well-suited to low-resource scenarios Despite its scope of 168 event types, Zhang et al. (2022) show that 18% of all event types in MAVEN have less than 100 annotated instances (Fig. 7 in §D). *Breathing* and *Extradition*, have less than 15 annotated event instances in more than 8K training sentences. Despite this, we see our model accurately identify all triggers of these event types in the testing split (see Fig. 3 in §C), achieving 100% testing precision on both, and 100% and 80% micro F1 score respectively.

Successful identification of multi-word triggers Multi-word event triggers, common in real-word data, comprise 3.42% and 3.38% of all triggers in MAVEN and RAMS respectively (see Table 10 in §D) Evaluating multi-word triggers as token classification yields misleading results as they represent the event type only when the entire phrase is annotated. For example, for the trigger "took place", the individual words are distinct from the event type denoted by the phrase. To evaluate performance on multi-word trigger phrases, we calculate exact match accuracy over them. We achieve nearly 91% and 89% on MAVEN and RAMS, respectively (Table 7 in §C), with incomplete predictions being similar to the ground truth ("assault vs the assault", "in touch" vs "been in touch").

Successful classification of multi-class triggers In a real-world ED scenario, event triggers may trigger multiple event classes in one context. 4% of all event triggers in RAMS can be classified as multi-class. (Table 10 in §C). See Fig. 4, where **purchasing** denotes both *transferownership* (arguments: *previous* and *current* owner) and *transfermoney* (arguments: *amount*). To accurately extract this event, it is necessary to capture all the senses of the trigger **purchasing**. Existing token classification methods perform event detection as multi-class, not multi-label classification. Generating sequences, as well as training over EC, enables our model to identify multi-class triggers. We achieve average prediction accuracy (% of types captured for a multi-class trigger) of 61% on RAMS, indicating the model can capture most of the senses in which each multi-class trigger functions.

Case Study: Multi-sentence context is vital to ED Consider these examples from WikiEvents:
Example 1: The whole building has **collapsed**.

Example 2: He chose **destruction**.

In Example 1, EDM3 extracts the token in bold as a relevant event trigger of the type *artifact existence*. However, this example is taken from a document primarily focused on $conflict$ events, with the triggers **bombing** and **explosion**. Therefore, **collapsed** becomes an auxiliary event that should not be predicted. Conversely, in Example 2, our model finds no salient event; however, the following sentences in the same document demonstrate that **destruction** is a salient event of type *artifact existence*. It is difficult for a sentence-level model to judge the saliency of an event without the context of its document or surrounding events, making it vital to include multi-sentence or document-level context.

4 Conclusion

In this paper, we propose EDM3, a domain-agnostic generative approach to the Event Detection task. EDM3 leverages a multi-tasking strategy that incorporates instructional prompts to improve model performance on imbalanced data and complex event instances. Our analysis shows an improvement in F1 score over single-task performance, supporting our main hypothesis viz. the effectiveness of breaking down complex generation tasks into subtasks that can support model learning on the primary task. Furthermore, our results highlight the potential for generative models in traditionally discriminative tasks like ED, paving the way for future advancements in the field.

Limitations

Our work demonstrates a prompted and generative approach on a single task, Event Detection, which can be easily adapted to other information retrieval tasks. Due to access issues, we were unable to use the ACE05 dataset. In lieu of this, we utilize 3 publicly-available general-domain datasets (RAMS, MAVEN, WikiEvent). Furthermore, there is a possibility of improving prompt quality further by analyzing the number and scope of examples required to achieve the best prompted performance. Finally, integrating domain knowledge could improve event-type classification, and we encourage future researchers to explore this area. Despite these limitations, our work provides a strong foundation for generative, instructional prompt-based frameworks for end-to-end Event Extraction and opens up exciting avenues for future research.

References

Ujjwala Anantheswaran, Himanshu Gupta, Kevin Scaria, Shreyas Verma, Chitta Baral, and Swaroop Mishra. 2024. A disturbance in the fours: Investigating the robustness of llms on math word problems.

Emanuela Boros, José G. Moreno, and Antoine Doucet. 2021. Event detection as question answering with entity information. *CoRR*, abs/2104.06969.

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, Beijing, China. Association for Computational Linguistics.

Rich Caruana. 1997. Multitask learning. *Mach. Learn.*, 28(1):41–75.

Yifei Chen. 2019. Multiple-level biomedical event trigger recognition with transfer learning. *BMC Bioinformatics*, 20.

Yubo Chen, Liheng Xu, Kang Liu, Daojian 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, Beijing, China. Association for Computational Linguistics.

Pengxiang Cheng and Katrin Erk. 2018. Implicit argument prediction with event knowledge. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 831–840, New Orleans, Louisiana. Association for Computational Linguistics.

Michael Crawshaw. 2020. Multi-task learning with deep neural networks: A survey.

Shumin Deng, Ningyu Zhang, Luoqiu Li, Chen Hui, Tou Huaixiao, Mosha Chen, Fei Huang, and Huajun Chen. 2021. OntoED: Low-resource event detection with ontology embedding. 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 2828–2839, Online. 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.

Seth Ebner, Patrick Xia, Ryan Culkin, Kyle Rawlins, and Benjamin Van Durme. 2020. Multi-sentence argument linking. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8057–8077, Online. Association for Computational Linguistics.

Himanshu Gupta, Kevin Scaria, Ujjwala Anantheswaran, Shreyas Verma, Mihir Parmar, Saurabh Arjun Sawant, Swaroop Mishra, and Chitta Baral. 2023. Targen: Targeted data generation with large language models. *ArXiv*, abs/2310.17876.

Xinyu He, Lishuang Li, Yang Liu, Xiaoming Yu, and Jun Meng. 2018. A two-stage biomedical event trigger detection method integrating feature selection and word embeddings. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 15(4):1325–1332.

Xinyu He, Ping Tai, Hongbin Lu, Xin Huang, and Yonggong Ren. 2022. A biomedical event extraction method based on fine-grained and attention mechanism. *BMC Bioinformatics*, 23(1):1–17.

Kung-Hsiang Huang, Mu Yang, and Nanyun Peng. 2020. Biomedical event extraction with hierarchical knowledge graphs.

Nattiya Kanhabua and Avishek Anand. 2016. Temporal information retrieval. In *Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '16, page 1235–1238, New York, NY, USA. Association for Computing Machinery.

Viet Dac Lai and Thien Huu Nguyen. 2019. Extending event detection to new types with learning from keywords. *CoRR*, abs/1910.11368.

Viet Dac Lai, Tuan Ngo Nguyen, and Thien Huu Nguyen. 2020. Event detection: Gate diversity and syntactic importance scores for graph convolution neural networks. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5405–5411, Online. Association for Computational Linguistics.

Sha Li, Heng Ji, and Jiawei Han. 2021. Document-level event argument extraction by conditional generation. *ArXiv*, abs/2104.05919.

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.

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.

Jian Liu, Yufeng Chen, and Jinan Xu. 2022. Saliency as evidence: Event detection with trigger saliency attribution. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4573–4585, Dublin, Ireland. Association for Computational Linguistics.

Xiao Liu, Zhunchen Luo, and Heyan Huang. 2018. Jointly multiple events extraction via attention-based graph information aggregation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1247–1256, Brussels, Belgium. Association for Computational Linguistics.

Nicholas Lourie, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2021. Unicorn on rainbow: A universal commonsense reasoning model on a new multitask benchmark.

Yaojie Lu, Hongyu Lin, Xianpei Han, and Le Sun. 2019. Distilling discrimination and generalization knowledge for event detection via delta-representation learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4366–4376, Florence, Italy. 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.

Yaojie Lu, Qing Liu, Dai Dai, Xinyan Xiao, Hongyu Lin, Xianpei Han, Le Sun, and Hua Wu. 2022. Unified structure generation for universal information extraction. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5755–5772, Dublin, Ireland. 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.

Thien Nguyen and Ralph Grishman. 2018. Graph convolutional networks with argument-aware pooling for event detection. *Proceedings of the AAAI Conference on Artificial Intelligence*, 32(1).

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.

Yifan Nie, Wenge Rong, Yiyuan Zhang, Yuanxin Ouyang, and Zhang Xiong. 2015. Embedding assisted prediction architecture for event trigger identification. *Journal of bioinformatics and computational biology*, 13 3:1541001.

Giovanni Paolini, Ben Athiwaratkun, Jason Krone, Jie Ma, Alessandro Achille, Rishita Anubhai, Cicero Nogueira dos Santos, Bing Xiang, and Stefano Soatto. 2021. Structured prediction as translation between augmented natural languages. In *9th International Conference on Learning Representations, ICLR 2021*.

Mihir Parmar, Swaroop Mishra, Mirali Purohit, Man Luo, M. Hassan Murad, and Chitta Baral. 2022. Inboxbart: Get instructions into biomedical multi-task learning.

James Pustejovsky. 1991. The syntax of event structure. *Cognition*, 41(1):47–81.

Sampo Pyysalo, Tomoko Ohta, Makoto Miwa, Han-Cheol Cho, Jun'ichi Tsujii, and Sophia Ananiadou. 2012. Event extraction across multiple levels of biological organization. *Bioinformatics*, 28(18):i575–i581.

Taneeya Satyapanich, Francis Ferraro, and Tim Finin. 2020. Casie: Extracting cybersecurity event information from text. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):8749–8757.

Jinghui Si, Xutan Peng, Chen Li, Haotian Xu, and Jianxin Li. 2022. Generating disentangled arguments with prompts: A simple event extraction framework that works.

Tarcísio Souza Costa, Simon Gottschalk, and Elena Demidova. 2020. Event-qa: A dataset for event-centric question answering over knowledge graphs. In *Proceedings of the 29th ACM international conference on information & knowledge management*, pages 3157–3164.

Christoph Tillmann and Hermann Ney. 2003. Word Reordering and a Dynamic Programming Beam Search Algorithm for Statistical Machine Translation. *Computational Linguistics*, 29(1):97–133.

Meihan Tong, Bin Xu, Shuai Wang, Yixin Cao, Lei Hou, Juanzi Li, and Jun Xie. 2020. Improving event detection via open-domain trigger knowledge. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5887–5897, Online. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Amir Pouran Ben Veyseh, Viet Dac Lai, Franck Dernoncourt, and Thien Huu Nguyen. 2021. Unleash gpt-2 power for event detection. In *ACL*.

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 W. 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, Singapore. Association for Computational Linguistics.

Sijia Wang, Mo Yu, Shiyu Chang, Lichao Sun, and Lifu Huang. 2021. Query and extract: Refining event extraction as type-oriented binary decoding. *CoRR*, abs/2110.07476.

Sijia Wang, Mo Yu, and Lifu Huang. 2022a. The art of prompting: Event detection based on type specific prompts.

Xiao Wang, Weikang Zhou, Can Zu, Han Xia, Tianze Chen, Yuansen Zhang, Rui Zheng, Junjie Ye, Qi Zhang, Tao Gui, et al. 2023. Instructuie: Multi-task instruction tuning for unified information extraction. *arXiv preprint arXiv:2304.08085*.

Xiaozhi Wang, Xu Han, Zhiyuan Liu, Maosong Sun, and Peng Li. 2019. Adversarial training for weakly supervised event detection. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 998–1008, Minneapolis, Minnesota. Association for Computational Linguistics.

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

Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Atharva Naik, Arjun Ashok, Arut Selvan Dhanasekaran, Anjana Arunkumar, David Stap, Eshaan Pathak, Giannis Karamanolakis, Haizhi Lai, Ishan Purohit, Ishani Mondal, Jacob Anderson, Kirby Kuznia, Krima Doshi, Kuntal Kumar Pal, Maitreya Patel, Mehrad Moradshahi, Mihir Parmar, Mirali Purohit, Neeraj Varshney, Phani Rohitha Kaza, Pulkit Verma, Ravsehaj Singh Puri, Rushang Karia, Savan Doshi, Shailaja Keyur Sampat, Siddhartha Mishra, Sujan Reddy A, Sumanta Patro, Tanay Dixit, and Xudong Shen. 2022b. Super-NaturalInstructions: Generalization via declarative instructions on 1600+ NLP tasks. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5085–5109, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Tianbao Xie, Chen Henry Wu, Peng Shi, Ruiqi Zhong, Torsten Scholak, Michihiro Yasunaga, Chien-Sheng Wu, Ming Zhong, Pengcheng Yin, Sida I. Wang, Victor Zhong, Bailin Wang, Chengzu Li, Connor Boyle, Ansong Ni, Ziyu Yao, Dragomir Radev, Caiming Xiong, Lingpeng Kong, Rui Zhang, Noah A. Smith, Luke Zettlemoyer, and Tao Yu. 2022. Unifiedskg: Unifying and multi-tasking structured knowledge grounding with text-to-text language models.

Sen Yang, Dawei Feng, Linbo Qiao, Zhigang Kan, and Dongsheng Li. 2019. Exploring pre-trained language models for event extraction and generation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5284–5294, Florence, Italy. Association for Computational Linguistics.

Hongming Zhang, Haoyu Wang, and Dan Roth. 2021. Zero-shot Label-aware Event Trigger and Argument Classification. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1331–1340, Online. Association for Computational Linguistics.

Wenlong Zhang, Bhagyashree Ingale, Hamza Shabir, Tianyi Li, Tian Shi, and Ping Wang. 2022. Event detection explorer: An interactive tool for event detection exploration.

Deyu Zhou and Dayou Zhong. 2015. A semi-supervised learning framework for biomedical event extraction

based on hidden topics. *Artificial intelligence in medicine*, 64 1:51–8.

Appendix

A Related Work

Transformer-based models (Vaswani et al., 2017) have been at the forefront of many language tasks due to the wealth of pretrained knowledge. Models using BERT (Yang et al., 2019; Wang et al., 2019) treat ED as word classification, in graph-based architectures (Wadden et al., 2019; Lin et al., 2020). Models that improve ED performance for low resource settings include Lu et al. (2019); Deng et al. (2021). Other works (Tong et al., 2020; Veyseh et al., 2021) generate ED and EI samples respectively to augment training data. Many models frame ED as a question-answering task (Du and Cardie, 2020; Boros et al., 2021; Wang et al., 2021; Liu et al., 2020). APEX (Wang et al., 2022a) augments input with type-specific prompts. With the advent of more powerful sequence-to-sequence models such as T5, there has been an increased interest in formulating event detection and event extraction as sequence generation tasks (Paolini et al., 2021; Lu et al., 2021; Si et al., 2022)

Multi-Task Learning is a training paradigm in which a single machine learning model is trained on multiple separate tasks (Caruana, 1997; Crawshaw, 2020). Across domains, models trained on multiple disparate tasks are better performing due to shared learning. Multi-Task learning has been leveraged to great effect in Xie et al. (2022); Lourie et al. (2021), and in specific domains as well (Chen, 2019; Parmar et al., 2022). This paradigm is also the basis of the generative T5 model. Paolini et al. (2021) carried out multi-task learning experiments over a number of information retrieval tasks. Specifically for Event Detection, multi-tasking over ED subtasks is implemented in GPTEDOT (Veyseh et al., 2021), where EI is used to augment ED performance. This is because the simplicity of EI makes it easier to evaluate the quality of generated data. However, there is a risk of introducing noise or generating low-quality samples due to the characteristics of the source data.

Prompt engineering Prompt-based models have been used for Event Detection and Event Extraction as well. Prompt Engineering has been leveraged to great effect to generate data (Gupta et al., 2023; Anantheswaran et al., 2024) to improve existing data quality or dearth. More recently, Si et al. (2022) used predicted labels from earlier in the pipeline as prompts for later stages of trigger identification and argument extraction, while Wang et al. (2022a), following the example of other works that use prototype event triggers (Wang and Cohen, 2009; Bronstein et al., 2015; Lai and Nguyen, 2019; Lyu et al., 2021; Liu et al., 2020; Zhang et al., 2021) from the dataset, used triggers as part of tailored prompts for each event type in the schema. In proposing EDM3, we are the first to explore the efficacy of instructional prompts for ED.

B Natural Language Prompts

For each task, we provide a natural language instruction followed by a general domain example in conjunction with a biomedical domain example as part of the instructional prompt. We choose instances that are complex, i.e. have multiple labels, or multi-word or multi-class labels.

B.1 Event Identification

Instruction You are given a text as input. The text gives information about ongoing events. An event trigger is a word or phrase that most clearly expresses the event occurrence. Your task is to identify the words or phrases that are event triggers for events in the text, where event type is not given. If there are no events, print NONE.

General example INPUT: The information minister alleged that oil smuggled into Turkey was bought by the Turkish president's son , who owns an oil company . Mr al - Zoubi said in an interview , All of the oil was delivered to a company that belongs to the son of Recep (Tayyip) Erdogan . This is why Turkey became anxious when Russia began delivering airstrikes against the IS infrastructure and destroyed more than 500 trucks with oil already.</s>OUTPUT: smuggled</s>EXPLANATION: The event describes goods being moved. The exact trigger from the text that describes this event is "smuggled".

Biomedical example INPUT: Left ventricular weight, body weight, and their ratio were not significantly altered by alinidine treatment.</s>OUTPUT: treatment | altered</s>EXPLANATION: the words "treatment" and "altered" are salient words describing important events.

B.2 Event Classification

Instruction This input text gives information about specific types of ongoing events. The output

should be the types of events occurring in the text. If there are no events, print NONE.

General example INPUT: The leaflets carried several messages to the citizens attempting to reassure them that the advancing army " would not target civilians , " but warned them to avoid the known locations of Isis militants . The military operation is the most complex carried out in Iraq since US forces withdrew from the country in 2011 . Last week , the UN said it was bracing itself for the world's biggest and most complex humanitarian effort following the battle , which it expects will displace up to one million people and see civilians used as human shields.</s>OUTPUT: conflict.</s>EXPLANATION: The event triggered by "battle" refers to an event of the type "conflict" which refers to a serious disagreement between two or more entities.

Biomedical example INPUT: Left ventricular weight, body weight, and their ratio were not significantly altered by alinidine treatment.</s>OUTPUT: planned_process | regulation</s>EXPLANATION: The input contains multiple events of planned_process and regulation type.

B.3 Event Detection

Instruction The text given as input discusses ongoing events. An event trigger is a word or phrase that most clearly expresses the event occurrence. Generate output in the format [event trigger->event type] for all events in the text. If there are no events, print NONE.

General example INPUT: The Organization for Security and Cooperation In Europe 's (OSCE) Office for Democratic Institutions and Human Rights and the OSCE High Commissioner on National Minorities issued a report in September saying that since Russia 's land grab , fundamental freedoms had " deteriorated radically " for many in Crimea , especially for pro - Ukrainian activists , journalists , and the Crimean Tatar community.</s>OUTPUT: land grab->transaction.exchangebuysell</s>EXPLANATION: In this text, the event being discussed is the "land grab", which functions as the event trigger. The type of event it describes is a transaction, in which ownership of entities is transferred.

Biomedical example INPUT: Left ventricular weight, body weight, and their ratio were not significantly altered by alinidine treatment.</s>OUTPUT: treatment->planned_process | altered->regulation</s>EXPLANATION: The word "treatment" in the input denotes a planned process, while the "altered" indicates the sentence talks about regulation.

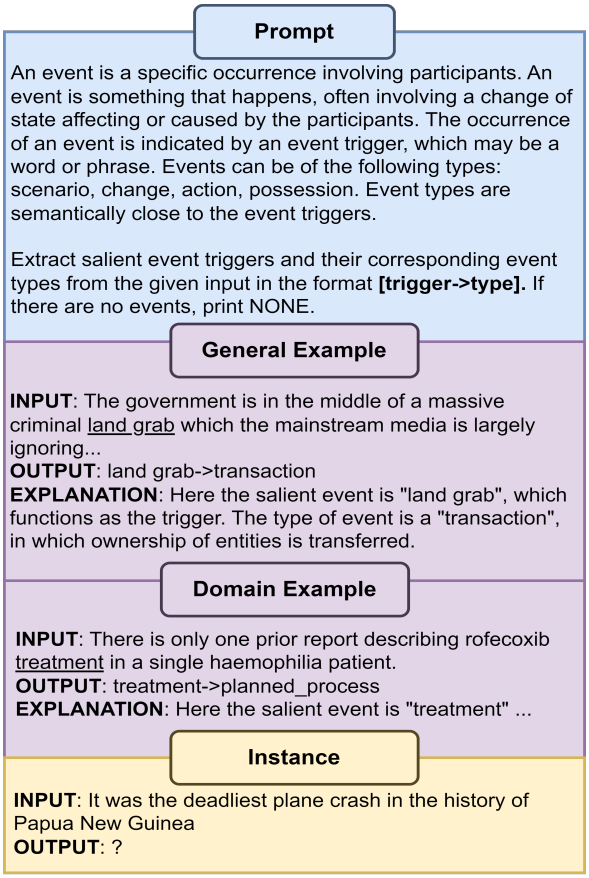

Figure 2: An example of an input instance for reformulated generative ED. The input comprises a task definition followed by diverse domain examples before the input sentence containing the events to be detected.

C Extended Analysis

Hyperparameters GPU: 2x NVIDIA GTX1080 GPUs. We train for 50 epochs with a batch size of 1. We use beam search decoding (Tillmann and Ney, 2003) during inference to generate output sequences. For beam search decoding, we use 50 beams.

…Osman Hussein was **arrested** in … **extradited** to the UK

Approach	Output
Single-task	arrested>arrest
EDM3	arrested->arrest extradited->extradition

Figure 3: Result on event type *extradition* , which has only 11 annotated instances.

…He cut his teeth in the 90s **purchasing** and producing the Miss Universe pageant, then made…

Approach	Output
Single-task	purchasing->transaction.transferownership
EDM3	purchasing->transaction.transferownership purchasing->transaction.transfermoney
Gold	purchasing->transaction.transferownership purchasing->transaction.transfermoney

Figure 4: EDM3 improving prediction on multi-class triggers.

C.1 EDM3 improves single-task ED performance on WikiEvents

Input:
Police in Calais have dispersed a rowdy anti-migrant protest with tear gas after clashes with protesters and detained several far-right demonstrators.

Single-task:
detained->movement.transportperson

EDM3:
detained->movement.transportperson | **clashes ->conflict.attack**

Gold:
detained->movement.transportperson | **clashes ->conflict.attack**

C.2 Negative instances hamper ED performance

From the dataset statistics in Table 9, we see that the WikiEvents dataset has close to 54% instances that have no annotated events, i.e. negative instances. We hypothesize that this detracts from the model's ability to discern relevant events and their types, and instead emphasizes the binary classification task of identifying event presence. We analyze the effect of negative examples further experimentally (Table 6). The consistent trend of higher Pos scores indicates that, given a sentence, our approach is better at extracting its events accurately as opposed to identifying whether it contains an event.

Dataset	#mwt		EM acc %
	Train	Test	
MAVEN	2442	633	90.84
RAMS	228	20	88.89

Table 7: Results on multi-word triggers. #mwt: number of multi-word triggers in training and testing data. EM acc %: exact match accuracy, i.e. percentage of multi-word triggers in test data predicted by our model.

The difference between both metrics is stark in the case of WikiEvents. We observe increased performance (60.71% to 65.67% after beam search decoding) over WikiEvents, which is significantly higher than what we observe on other datasets. From further analysis, we find that training on only positive examples improves the ED performance on event sentences by nearly 5%. Furthermore, despite the fact that MAVEN has 168 event types and WikiEvents has only 49 (Table 8), the ED performance on MAVEN (62.4%) is higher than on WikiEvents (58.7%). This indicates that rather than the complexity of the ED task, the distribution of positive and negative instances may hamper the model's ability to perform the task.

We attribute this to the much higher share of negative instances in this dataset. The performance drops over non-event sentences as the model may predict event occurrence based on salient events in the sentence, that are important in the context of the sentence alone but are divorced from the subject of the document, and therefore annotated as non-events. We explore this further in our discussion of the need for multi-sentence context, which may be a way to counter the negative impact of a high proportion of non-event sentences on our ED model.

C.3 Annotation issues

We present an approach that accurately extracts text terms for event annotations while preserving case sensitivity, a crucial factor in distinguishing different event triggers. Improper extraction or human error can lead to errors in existing annotations. Our approach can identify such errors by highlighting discrepancies in the case of event triggers. Addi-

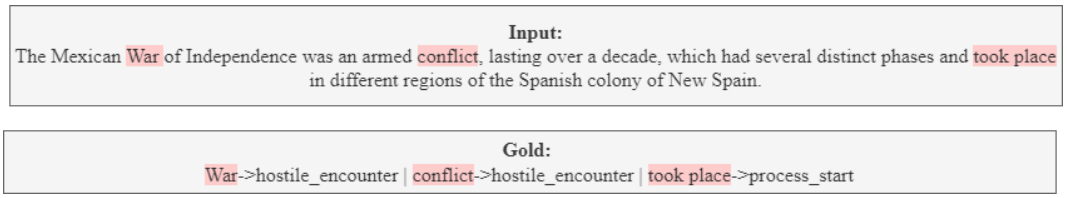

Figure 5: Example of an event with multi-word trigger (2 words)

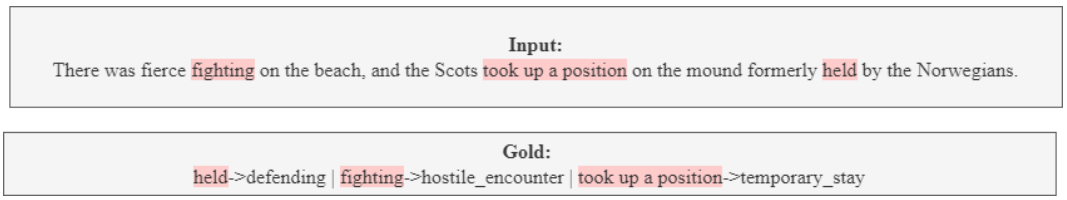

Figure 6: Example of an event with multi-word trigger (4 words)

tionally, we observe an ambiguity in some annotation schema, particularly in MAVEN, where the extensive coverage of event types results in overlapping event type definitions. For instance, the event types motion, self_motion, and motion_direction exhibit minor differences, leading to inconsistent annotations. This ambiguity introduces noise into the classification and ED subtasks. Our proposed model resolves this issue and accurately extracts all events in the corpus. We provide examples that demonstrate the improved ED performance achieved through multi-tasking.

D Data

The datasets we choose to demonstrate our approach on span a range of characteristics, from sentence-level to multi-sentence level, with varying proportions of non-event instances. We also include a biomedical domain dataset to illustrate the adaptability of our approach. In Table 8, we note the document and event instance statistics across datasets. Table 9 delineates the dataset statistics post-data processing. We note the average and maximum number of events and distinct event types that occur per data instance for each dataset. We evaluate on two-level event type labels for RAMS and WikiEvents.

MAVEN Wang et al. (2020) proposed this dataset with the idea of combating data scarcity and low coverage problem in prevailing general domain event detection datasets. The high event coverage provided by MAVEN results in more events per sentence on average, including multi-word triggers, as compared to other general domain ED datasets (more details in App. C.3). The dataset, reflective of real-world data, has a long tail distribution (see Fig. 7).

We follow the example of SaliencyED (Liu et al., 2022) and evaluate our model performance on the development split of the original MAVEN dataset.

WikiEvents Existing work on this dataset proposed by Li et al. (2021) focuses exclusively on document-level argument extraction and event extraction.

Sentences without any event occurrences make up nearly half of the entire dataset (see Table 9). In the absence of existing baselines, we establish the benchmark performances on sentence-level ED on this dataset for future researchers.

RAMS This dataset, created by Ebner et al. (2020), is primarily geared towards the task of multi-sentence argument linking. The annotated argument roles are in a 5-sentence window around the related event trigger.

In its native form, the dataset is geared towards multi-sentence argument role linking. Using the original configuration allows us to test the efficacy of our model on the multi-sentence level. Furthermore, on the sentence level, the dataset is imbalanced: 77% of the sentences contain no events. Training a model on this incentivizes event occurrence detection over ED.

MLEE This biomedical ED corpus by Pyysalo et al. (2012) is taken from PubMed abstracts centered around tissue-level and organ-level processes.

The majority of the datasets used in this work are Event Extraction (EE) datasets, maintaining the scope of possible extensions of the proposed reformulation and multi-tasking approach to EE.

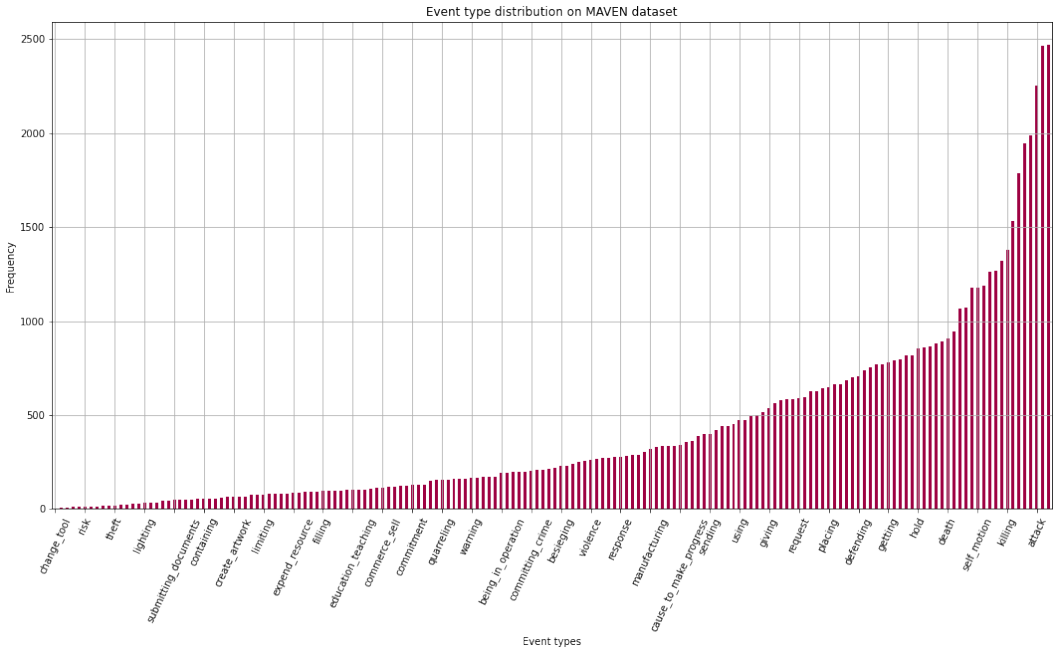

Figure 7: Distribution of event types in MAVEN. The distribution is a long-tail distribution, indicating strong class imbalance.

Dataset	Docs			#triggers	#types
	Train	Dev	Test		
MLEE	131	44	87	8014	30
RAMS	3194	399	400	9124	38
MAVEN	2913	710	857	118732	168
WikiEvents	206	20	20	3951	49

Table 8: Dataset statistics, including number of documents per data split, as well as number of event triggers and unique event types across the dataset.

Dataset	Neg (%)	Events per row		Types per row		#zs
		Avg	Max	Avg	Max	
MLEE	18.22	2.867	16	2.369	9	3
RAMS	0	1.066	6	1.061	4	0
MAVEN	8.64	2.433	15	2.314	15	0
WikiEvents	54.11	1.671	7	1.429	6	1

Table 9: Dataset statistics (post-processing) for training. Neg%: Proportion of input instances with no event occurrences. Events per row: Number of event triggers per input instance. Types per row: Number of unique event types per input instance. #zs: Number of event types in test split not seen during training.

Dataset	Multi-word triggers		Multi-class triggers	
	%instances	%rows	%instances	%rows
RAMS	3.38	2.89	**3.97**	**3.72**
MAVEN	**3.42**	**7.39**	0.06	0.13

Table 10: Statistics on multi-word and multi-class triggers in all datasets. %instances: the % of total triggers present. %rows: the % of all input instances that contain at least 1 multi-word or multi-class trigger.

Category	Event type	Example triggers
Anatomical	cell_proliferation	proliferation, proliferate, growing
	development	formation, progression, morphogenesis
	blood_vessel_development	angiogenic, angiogenesis
	death	death, apoptosis, survival
	breakdown	dysfunction, disrupting, detachment
	remodeling	remodeling, reconstituted
	growth	proliferation, growth, regrowth
Molecular	synthesis	production, formation, synthesized
	gene_expression	expression, expressed, formation
	transcription	expression, transcription, mRNA
	catabolism	disruption, degradation, depleted
	phosphorylation	phosphorylation
	dephosphorylation	dephosphorylation
General	localization	migration, metastasis, infiltrating
	binding	interactions, bind, aggregation
	regulation	altered, targeting, contribute
	positive_regulation	up-regulation, enhancement, triggered
	negative_regulation	inhibition, decrease, arrests
Planned	planned_process	treatment, therapy, administration

Table 11: Event types in MLEE, along with example triggers.

Event type	Frequency	Example triggers
process_start	2468	began, debut, took place
causation	2465	resulted in, caused, prompted
attack	2255	bombing, attacked, struck
hostile_encounter	1987	fought, conflict, battle
motion	1944	fell, pushed, moved
catastrophe	1785	explosion, hurricane, flooded
competition	1534	event, championships, match
killing	1380	killed, murder, massacre
process_end	1323	closing, complete, ended
statement	1269	asserted, proclaimed, said

Table 12: Top 10 event types in MAVEN, along with example triggers.

Event type	Frequency	Example triggers
conflict.attack	721	massacre, battle, bombing
movement.transportperson	491	smuggling, walked, incarcerate
transaction.transfermoney	482	reimbursed, paid, purchasing
life.die	442	die, murder, assassinating
life.injure	422	surgery, injured, brutalized
movement.transportartifact	367	imported, trafficking, smuggling
transaction.transferownership	327	auction, donated, acquire
contact.requestadvise	250	advocating, recommending, urged
contact.discussion	249	discuss, meet, negotiated
transaction.transaction	211	funded, donated, seized

Table 13: Top 10 event types in RAMS, along with example triggers.

Event type	Frequency	Example triggers
conflict.attack	1188	explosion, shot, attack
contact.contact	530	met, said, been in touch
life.die	501	killed, died, shot
life.injure	273	injuring, wounded, maimed
movement.transportation	212	transferred, brought, arrived
justice.arrestjaildetain	176	arrested, capture, caught
artifactexistence.damagedestroydisabledismantle	103	damaged, destruction, removed
justice.investigatecrime	102	analysis, discovered, investigation
justice.chargeindict	96	charged, accused, alleged
artifactexistence.manufactureassemble	82	construct, make, build

Table 14: Top 10 event types in WikiEvents, along with example triggers.