Negative documents are positive: Improving event extraction performance using overlooked negative data

Osman Mutlu
Koç University
Rumelifeneri Yolu 34450
Sarıyer, İstanbul/Turkey
omutlu@ku.edu.tr

Ali Hürriyetoğlu
KNAW Humanities Cluster DHLab
Oudezijds Achterburgwal 185, 1012DK
Amsterdam, the Netherlands
ali.hurriyetoglu@dh.huc.knaw.nl

Abstract

The scarcity of data poses a significant challenge in closed-domain event extraction, as is common in complex NLP tasks. This limitation primarily arises from the intricate nature of the annotation process. To address this issue, we present a multi-task model structure and training approach that leverages the additional data, which is found as not having any event information at document and sentence levels, generated during the event annotation process. By incorporating this supplementary data, our proposed framework demonstrates enhanced robustness and, in some scenarios, improved performance. A particularly noteworthy observation is that including only negative documents in addition to the original data contributes to performance enhancement. When training the model with only 80% of the original data alongside negative documents, the outcome closely paralleled employing the entire original data set without any negative documents. Our findings offer promising insights into leveraging extra data to mitigate data scarcity challenges in closed-domain event extraction.

1 Introduction

Closed-domain event extraction is a specialized task in Natural Language Processing (NLP) that focuses on automatically identifying and extracting specific events or occurrences from text within a restricted domain, such as biomedical research, financial markets, political events, or sports (Xiang and Wang, 2019; Parolin et al., 2021). It plays a crucial role in capturing and categorizing relevant events, their attributes, and relationships, enabling applications such as information retrieval (Abuleil and Evens, 2004), trend analysis (Cheng et al., 2022; Wang et al., 2012), and knowledge base construction (Schrodt and Idris, 2014; Hürriyetoğlu et al., 2021; Jenkins et al., 2023). However, despite the advancements in NLP models, the scarcity of annotated data poses a persistent bottleneck in achieving accurate and reliable event extraction models. The limited availability of annotated data, crucial for training and evaluating such models, hinders their performance and generalizability (Caselli et al., 2021; Hu et al., 2022).

The annotation process plays a vital role in event extraction, requiring domain experts to meticulously label relevant events and their associated attributes. However, this process is often labor-intensive, time-consuming, and expensive (Pustejovsky and Stubbs, 2012). The complexity and diversity of event types further complicate the task, as events can vary in structure, context, and representation. Moreover, the need for inter-annotator agreement adds to the complexity, requiring multiple annotators to reach a consensus on the event labels. These challenges contribute to the limited availability of annotated data, restricting the performance and generalizability of event extraction models.

To overcome the data scarcity challenge, we propose a model structure and training schema that harnesses the additional data generated as a natural by-product of the annotation process. Specifically, we utilize coarse-grained data that classifies documents or sentences as containing an event or not, as shown in Table 1. The first example shows the inherent document and sentence labels in a token-annotated document, while the second example is a document with no event information. This data can be easily generated from already annotated documents for event extraction, and one could easily gather more samples without token-level annotations. Labeling such data is relatively painless, effectively circumventing most of the aforementioned issues with annotating event extraction documents. Thus, achieving a higher data quality is considerably cheaper and easier. We analyze the trade-off between using token annotations and

Proceedings of the 6th Workshop on Challenges and Applications of Automated Extraction of Socio-political Events from Text, pages 124–135, Varna, Bulgaria, Sep 7, 2023.
https://doi.org/10.26615/978-954-452-089-2_017

Document No	Sentence No	Sentence	Sentence label	Document label
1	1	He said the union had already send a statutory letter to the Uber office here in connection with the strike.	Negative	Positive
	2	The leaders of the union also said the local taxi drivers had launched an **attack** against the online taxi drivers at the airport.	Positive	
	3	The online taxi drivers have been having a tough time for the last one year.	Negative	
	4	Uber and Ola are two prominent online taxi service providers in Kochi.	Negative	
	5	Earlier, some trade unions representing local taxi operators had come out in **protest** against the online taxi networks such as Uber and Ola.	Positive	
2	1	Tributes paid to Field Marshal Cariappa, students sing prayers at his 'samadi'	Negative	Negative
	2	Madikeri: Rich tributes were paid to the late Field Marshal K.M.Cariappa at "Roshanara" here, where his "samadhi" is located, to observe the birth anniversary of one of the great soldiers of the country.	Negative	
	3	Prayers in different languages were rendered by students of the Bharatiya Vidya Bhavan-Kodagu Vidyalaya (BVB-KV) and family members of the late Field Marshal.	Negative	

Table 1: A table that consists of 2 sample documents from ACL CASE 2021 shared task. The first document is positive and token-annotated. The second document has no event information, therefore negative. The event triggers are shown in bold, and event arguments are underlined.

coarse-grained labels, evaluating performance variations with different ratios of these data types.

In our training schema, we incorporate the extra coarse data as two auxiliary tasks alongside the main event extraction task: document binary classification and sentence binary classification. By utilizing this supplementary data, our approach aims to augment the training set and enhance the performance and robustness of the event extraction model. The integration of this additional data has yielded promising results, effectively addressing the limitations caused by the lack of annotated data in closed-domain event extraction.

This study contributes to the field by providing a practical solution to the data scarcity problem in closed-domain event extraction. By leveraging the extra data generated during the annotation process, we strive to advance the state-of-the-art in event extraction, paving the way for more accurate and efficient systems across various domains. The outcomes of our research have potential implications for numerous downstream applications, ultimately benefiting various sectors that rely on event extraction for knowledge extraction and decision-making processes (Hogenboom et al., 2016).

The following section provide a brief overview on studies related to our study. Next we provide details of the multi-task model and the data we utilize for our experiments in Sections 3 and 4 respectively. The experimental setting is described in Section 5 in terms of a baseline and three experiment sets. We report results of our experiments in Section 6 and summarize our findings in Section 7.

2 Related Work

The performance of event extraction has been significantly depended on the amount of relevant data utilized for creating an event extraction system (Chen and Ji, 2009; Hsu et al., 2022). The variety of the data contributes to the performance and generalizability of an event extraction system as well (Yörük et al., 2022).

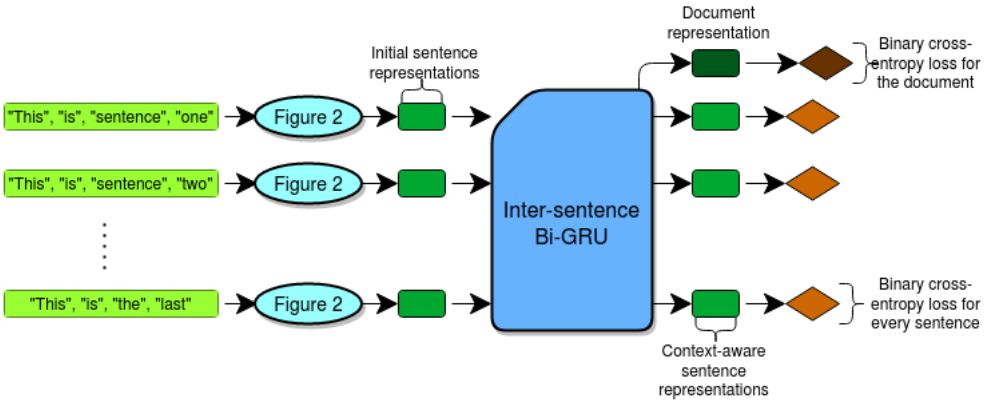

Figure 1: The main structure of the multi-task model. After creating embeddings for each sentence in the document (see Figure 2), these are sent through a bi-GRU to get a context-aware representation for each sentence and a single representation for the whole document. The losses for the document and the sentences are obtained by passing these embeddings through their respective classification layers.

The design of an event extraction system is another major determiner of the performance (Pei et al., 2023). The design should be able to utilize and encode as much as information available in the data for the target task. Consequently, syntax-oriented rule creation (Fleissner and Fang, 2012; Oostdijk et al., 2016), joint learning (Chen et al., 2018), multi-task learning (drissiya El-allaly et al., 2021), and pre-trained architectures (Yang et al., 2019) have been developed and successfully applied in many event extraction scenarios.

We follow these approaches both for data by increasing the size and variety of the data and benefiting from multi-task learning that is based on pre-trained architectures. Although in different domains, both Rei and Søgaard (2019) and Tong et al. (2021) are highly similar to our approach. They both adopt a multi-task structure within a joint learning framework, leveraging data at multiple levels of granularity. However, our approach diverges from theirs primarily in terms of incorporating document-level information, alongside sentence-level. An innovative aspect of our study is the revelation that integrating negative documents substantially augments performance and robustness, particularly in scenarios with limited data availability.

3 Model Structure

To solve event extraction tasks using deep learning techniques, they are commonly approached as token classification problems. In token classification, each word or token in the input text is assigned a label indicating its role in the event extraction process. One popular labeling scheme is the BIO format (Ramshaw and Marcus, 1995), which stands for "Beginning, Inside, and Outside." In this format, each token is labeled as either B-event, I-event, or O. The B-event label denotes the beginning of an event mention, the I-event label indicates that the token is inside the event mention, and the O label signifies that the token is outside any event mention. By converting the event annotations into the BIO format, deep learning models can be trained to recognize and classify tokens based on their involvement in events, facilitating the automated extraction of important information from text.

The model structure is designed to effectively leverage document and sentence-level information, alongside the main task of token classification, in a coherent manner. To achieve this, our model[1] predicts labels and trains on all three levels simultaneously, enabling comprehensive learning. Inspired by ScopeIt (Patra et al., 2020), our multi-task architecture, illustrated in Figure 1, enables the creation of representations for tokens, sentences, and documents to then put these through the respective classification layers for each task. We build on their model structure by adding the facilities for the token classification task. So, our model trains on the two auxiliary tasks, document and sentence classification tasks, in addition to the primary token classification task.

The model processes each sentence, with its split tokens, using a transformers-based encoder[2] to ob-

[1] https://github.com/OsmanMutlu/ms_thesis

[2] https://huggingface.co/

tain representations for individual tokens. To address the limited input problem of the encoder, each sentence is processed independently. Within each sentence, a bidirectional Gated Recurrent Unit (bi-GRU) (Cho et al., 2014), dubbed intra-sentence bi-GRU, is employed to further enhance token representations and generate a representation for the entire sentence by concatenating the last hidden states from both directions of the bi-GRU. These sentence embeddings are further enriched with contextual information by passing them through a second bi-GRU, named inter-sentence bi-GRU. Additionally, a single representation for the entire document is obtained by concatenating the last hidden states from the inter-sentence bi-GRU in both directions. Each representation, whether for tokens, sentences, or documents, is then passed through their respective classification layers to calculate the corresponding losses. The document and sentence tasks employ binary cross-entropy loss, while the token task utilizes categorical cross-entropy loss. The losses from each task are combined, yielding a final loss value for backpropagation (Rumelhart et al., 1986).

It is important to note that we maintain a consistent model structure across all our experiments, even if document or sentence loss is not calculated in certain scenarios. This ensures a standardized approach and facilitates fair comparisons across different variations of the model.

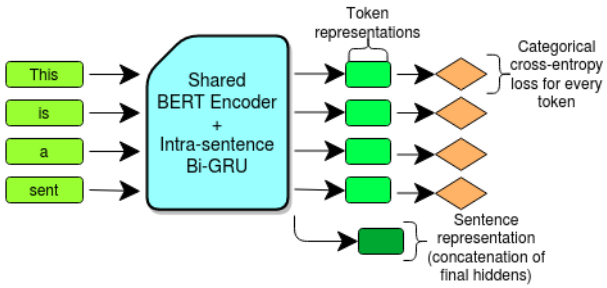

Figure 2: Each sentence of the document goes through a shared transformers-based encoder and a bi-GRU to produce embeddings for each token and the sentence. A categorical cross-entropy loss is calculated for each token after passing their embeddings through a classification layer.

4 Data

We leverage the data provided by the ACL CASE 2021 shared task (Hürriyetoğlu et al., 2021), which focuses on detecting protest events in the languages English, Portuguese, Spanish, and Hindi. The shared task encompasses four sequential steps, representing different stages of a real-world event extraction pipeline (Duruşan et al., 2022). For our experiments, we specifically utilize a subset of the English training data from subtask four, along with the corresponding English test data.

The training data consists of 717 token-annotated documents. These annotations were distributed in BIO format, meaning there are no overlapping labels for any individual token, effectively turning this task into token classification. The test set, which remains the same across all experiments since token classification serves as the primary task, includes 179 token-annotated documents. The distribution of token labels for the training and test sets is outlined in Table 2.

4.1 Inherent coarse-grained data

As mentioned earlier, our training schema takes advantage of the additional data inherent in the token-annotated documents. From a document classification perspective, the training set contains 717 positive documents, as all documents have at least one token annotation. Conversely, there are no negative documents. Regarding sentence classification, out of 14.06 sentences on average per document, 29% are positively labeled as they contain at least one token annotation. This translates to 2,893 positive and 7,191 negative sentences. It's worth noting that the statistics for the test set are irrelevant for coarse-grained data, given that token classification is the primary task.

4.2 Negative documents

Some of our experiments (explained in section 5) uses extra data that is not any part of the original 717 token-annotated documents. This extra data is sourced from subtask 1 of the same shared task and consists of 717 negatively labeled documents, indicating the absence of token annotations. These negative documents emerge as a by-product of the annotation process. When selecting documents for token-level annotation, the non-selected ones inadvertently contribute to the creation of negative documents. This set of 717 negative documents were randomly selected out of 7,412 negative documents in training set of subtask 1.

	etime	fname	organizer	participant	place	target	trigger
Train	1,071	1,089	1,187	2,435	1,436	1,334	4,096
Test	260	224	223	542	313	286	929

Table 2: The distribution of token labels for the training and test sets of subtask 4 of ACL CASE 2021 shared task.

5 Experimental Setup

Aside from the baseline, we conducted three main sets of experiments to address three key research questions, with each subsequent set incorporating additional data. In the first set, we utilized the inherent coarse-grained data available in token-annotated documents. In the second set, we introduced negative documents to balance the positive ones and further explored the effects of the document classification task. In the third set, we removed some of the 717 documents to be used as extra coarse-grained data without token annotations.

For each experiment set, we conducted three experiments based on different combinations of losses in addition to the token classification loss: only sentence classification loss (variation 1), only document classification loss (variation 2), and both sentence and document classification losses (variation 3). This approach allowed us to assess the individual effects of each auxiliary task introduced. Although some weights of the model may not update in certain cases due to the architecture, we maintained the same model for all experiments to ensure fair comparisons. Each experiment was run three times to calculate average performance and standard deviation scores. Additionally, we gradually decreased the amount of data in each experiment to measure the influence of data size on model performance.

Listed below are the parameters employed for our model. It's important to note that no parameter-specific experiments were conducted to fine-tune these values. They remain consistent throughout all experiments, thereby minimizing the potential impact of parameter variations. The selection of these parameters was driven by pragmatic considerations, encompassing factors such as data size, GPU capacity, and practical feasibility. The parameter settings are as follows:

- *Number of training epochs*: 30
- *Pretrained transformers model*: sentence-transformers/paraphrase-xlm-r-multilingual-v1
- *Learning rate for the encoder*: 2e-5
- *Learning rate for the general model*: 1e-4 (same as ScopeIt)
- *Batch size of documents*: 16
- *Maximum num of sentences in a document*: 200
- *Maximum token length of a sentence*: 128
- *Number of GRU layers*: 2
- *Size of GRU hidden layer*: 512
- *Development data*: random selection of 10% from the training data

Baseline:

As for the baseline, our model was trained using the 717 span-annotated documents. It's important to note that for the baseline model, the inter-sentence bi-GRU and MLPs for document and sentence classification did not train, as we solely utilized the loss for the primary task. However, the same model structure was retained to facilitate a fair comparison. To evaluate our experiments, we use a Python implementation [3] of the original [4] conlleval evaluation script, which we simply refer to as the F1 score.

Experiment Set 1:

In Experiment Set 1, we focused on the inherent information present in token annotations, aforementioned in Section 4.1, without incorporating any additional coarse-grained data. This allowed us to measure the impact of introducing auxiliary tasks to the baseline model without modifying the existing data. This reference point was important for comparing loss variations in the other two experiment sets and determining whether the fine-grained task of token classification inherently encompasses the coarser tasks during training.

[3] https://github.com/sighsmile/conlleval, accessed on July 6, 2023.

[4] www.cnts.ua.ac.be/conll2000/chunking/conlleval.txt, accessed on July 6, 2023.

Experiment Set 2:

Experiment Set 2 addressed the limitation of introducing the document classification task in the baseline model, where all 717 token-annotated documents were positive. To balance out the positive documents and mitigate the training challenges, we introduced negative documents obtained from another subtask of the same shared task as mentioned in Section 4.2. As this change only affected the calculation of document classification loss, there was no need to repeat this experiment for loss variation 1 (only sentence classification loss).

Experiment Set 3:

Finally, in Experiment Set 3, we investigated the effects of including extra coarse-grained data. To simulate a real-world scenario where researchers decide how many documents to annotate, we modified the data size reduction scenario. Instead of completely discarding a certain percentage of the data, we utilized that percentage of documents as extra training data for the sentence and document classification tasks. This experiment set aims to answer the following question; in a scenario where token-annotated data is small, and the training curve does not indicate data saturation for token classification, would easy-to-label coarse-grained data improve the model performance?

6 Results and Discussion

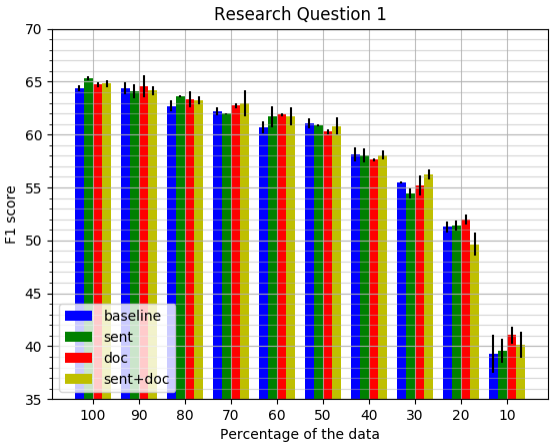

Figure 3: Results from experiment set 1. The black line in each bar indicates the standard deviation. "sent", "doc," and "sent+doc" is for variation 1, 2, and 3 for loss calculation, respectively.

Experiment Set 1:

The results obtained from the initial experiment set, depicted in Figure 3, closely align with our baseline performance, with minor fluctuations attributable to the standard deviation from three runs. Notably, we observe that incorporating document and sentence classification tasks alone does not yield any improvement in the absence of new data introduced to our model. This suggests that during training for the token classification task, the internal representations of our model already encompass the essential information for coarser tasks.

Experiment Set 2:

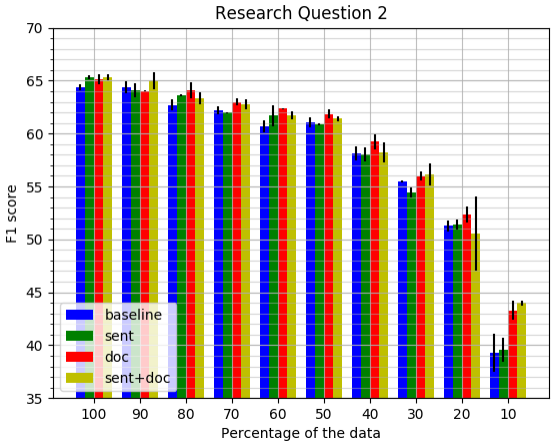

Figure 4: Results from experiment set 2. The black line in each bar indicates the standard deviation. "sent", "doc," and "sent+doc" is for variation 1, 2, and 3 for loss calculation, respectively.

Figure 4 illustrates a clear improvement in results, particularly evident when the data size is reduced to at least 80%. The introduction of negative documents to balance the positive ones is responsible for enhancing the model's performance. Since acquiring negative documents is relatively straightforward – they naturally arise during the document selection process for token-level annotations – this method offers a quick and effective way to boost existing event extraction models. This outcome represents a significant finding from our experiments; even in documents with no information related to events, the model can still exhibit improvements.

Experiment Set 3:

Figure 5 demonstrates a substantial overall gain. Notably, we observe that with only 60% of the

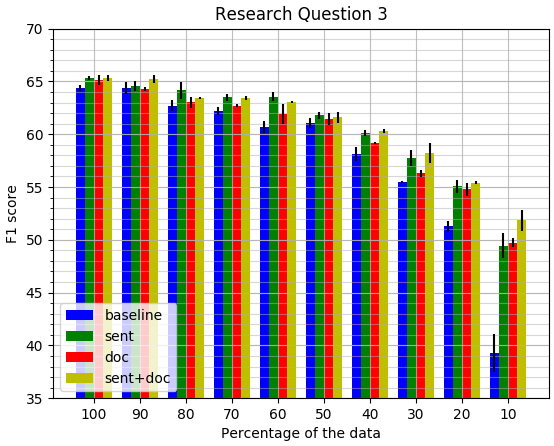

Figure 5: Results from experiment set 3. The black line in each bar indicates the standard deviation. “sent”, “doc,” and “sent+doc” is for variation 1, 2, and 3 for loss calculation, respectively.

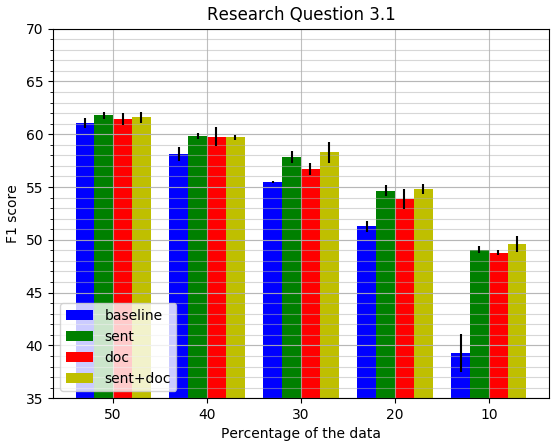

Figure 6: Results from experiment set 3.1. The black line in each bar indicates the standard deviation. “sent”, “doc,” and “sent+doc” is for variation 1, 2, and 3 for loss calculation, respectively.

717 documents token-annotated, and the remaining 40% having only document and sentence labels, we still achieve results comparable to having all documents token-annotated. Additionally, a general trend emerges, indicating that as token-annotated data decreases and extra coarse-grained data increases, the improvements from the baseline become more pronounced. This trend is further investigated in the experiment set 3.2. Experiment Set 3 involves two variables: token-annotated data size and extra coarse-grained data size. To clarify the impact of each, we conduct experiment set 3.1, where we fix the extra coarse-grained data size and focus solely on changes in token-annotated data size.

Experiment Set 3.1:

Starting with 50% of the data, we fix the discarded 50% as extra coarse-grained data and use it in all subsequent runs. By doing so, we can analyze performance changes between experiments without confusion as to whether the change originated from alterations in extra data size or token data size. As shown in Figure 6, the results align with the original experiment set 3, confirming that the improvement increases as the token data size decreases. Comparing the yellow line representing 10% of the data from this experiment with the same data size in the experiment set 3 reveals that having even more extra coarse-grained data than 50% could lead to further performance gains.

Experiment Set 3.2:

Designed to measure the impact of utilizing coarse-grained data in scenarios akin to few-shot learning settings, this experiment set presents noteworthy results, as depicted in Figure 7. The model exhibits significant improvement over the baseline, suggesting that leveraging coarse-grained data enhances the model’s robustness, even with minimal data sizes.

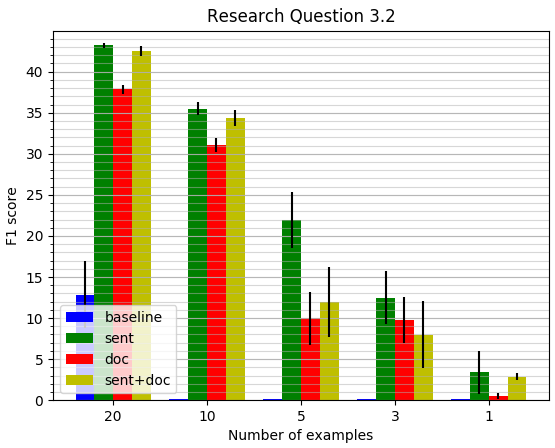

Figure 7: Results from experiment set 3.2. The black line in each bar indicates the standard deviation. “sent”, “doc”, and “sent+doc” is for variation 1, 2, and 3 for loss calculation, respectively.

7 Conclusion and Future Work

In this study, we addressed the challenge of data scarcity in closed-domain event extraction, a common hurdle in complex NLP tasks. Through our proposed multi-task model structure and training approach, we successfully leveraged additional data generated during the token annotation process. The inclusion of this supplementary data, particularly negative documents without event information, proved to be crucial in enhancing the performance and robustness of our event extraction model.

Our experiments demonstrated that introducing extra coarse-grained data, which identifies documents and sentences without events, significantly contributed to performance improvements. The integration of document and sentence classification tasks alongside token classification did not yield noticeable benefits on their own, reaffirming that the internal representations of our model already encompassed essential information for coarser tasks. Remarkably, even in scenarios where only a portion of the data was token-annotated, the model's performance remained comparable to situations with complete token annotations. We observed a clear trend of increasing performance gains as the token-annotated data size decreased and the extra coarse-grained data size increased. This trend was further reinforced when examining few-shot learning settings, where leveraging coarse-grained data notably enhanced the model's robustness even with minimal data sizes.

In conclusion, our findings offer promising insights into mitigating data scarcity challenges in closed-domain event extraction by effectively utilizing extra data obtained during the annotation process. This practical solution opens the door to more robust and efficient event extraction systems across various domains, with implications for knowledge extraction and decision-making processes. We utilized gold-standard data throughout all our experiments. We will be investigating the possible usage of silver coarse-grained data, which does not even require the considerable ease of labeling documents or sentences. We also plan to include more event information extraction data sets to test our hypothesis further.

References

Saleem Abuleil and Martha Evens. 2004. Events extraction and classification for arabic information retrieval systems. In *16th IEEE International Conference on Tools with Artificial Intelligence*, pages 769–773.

Tommaso Caselli, Osman Mutlu, Angelo Basile, and Ali Hürriyetoğlu. 2021. PROTEST-ER: Retraining BERT for protest event extraction. In *Proceedings of the 4th Workshop on Challenges and Applications of Automated Extraction of Socio-political Events from Text (CASE 2021)*, pages 12–19, Online. Association for Computational Linguistics.

Guandan Chen, Wenji Mao, Qingchao Kong, and Han Han. 2018. Joint learning with keyword extraction for event detection in social media. In *2018 IEEE International Conference on Intelligence and Security Informatics (ISI)*, page 214–219. IEEE Press.

Zheng Chen and Heng Ji. 2009. Can one language bootstrap the other: A case study on event extraction. In *Proceedings of the NAACL HLT 2009 Workshop on Semi-supervised Learning for Natural Language Processing*, pages 66–74, Boulder, Colorado. Association for Computational Linguistics.

Wai Khuen Cheng, Khean Thye Bea, Steven Mun Hong Leow, Jireh Yi-Le Chan, Zeng-Wei Hong, and Yen-Lin Chen. 2022. A review of sentiment, semantic and event-extraction-based approaches in stock forecasting. *Mathematics*, 10(14).

Kyunghyun Cho, Bart van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. 2014. On the properties of neural machine translation: Encoder–decoder approaches. In *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation*, pages 103–111, Doha, Qatar. Association for Computational Linguistics.

Fırat Duruşan, Ali Hürriyetoğlu, Erdem Yörük, Osman Mutlu, Çağrı Yoltar, Burak Gürel, and Alvaro Comin. 2022. Global contentious politics database (glocon) annotation manuals.

Ed drissiya El-allaly, Mourad Sarrouti, Noureddine En-Nahnahi, and Said Ouatik El Alaoui. 2021. Mttlade: A multi-task transfer learning-based method for adverse drug events extraction. *Information Processing & Management*, 58(3):102473.

Sebastian Fleissner and Alex Chengyu Fang. 2012. A syntax-oriented event extraction approach. In *KDIR 2012 - Proceedings of the International Conference on Knowledge Discovery and Information Retrieval*, pages 336–339. 4th International Conference on Knowledge Discovery and Information Retrieval, KDIR 2012 ; Conference date: 04-10-2012 Through 07-10-2012.

Frederik Hogenboom, Flavius Frasincar, Uzay Kaymak, Franciska de Jong, and Emiel Caron. 2016. A survey of event extraction methods from text for decision support systems. *Decision Support Systems*, 85:12–22.

I-Hung Hsu, Kuan-Hao Huang, Elizabeth Boschee, Scott Miller, Prem Natarajan, Kai-Wei Chang, and Nanyun Peng. 2022. DEGREE: A data-efficient generation-based event extraction model. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1890–1908, Seattle, United States. Association for Computational Linguistics.

Yibo Hu, MohammadSaleh Hosseini, Erick Skorupa Parolin, Javier Osorio, Latifur Khan, Patrick Brandt, and Vito D'Orazio. 2022. ConfliBERT: A pre-trained language model for political conflict and violence. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5469–5482, Seattle, United States. Association for Computational Linguistics.

Ali Hürriyetoğlu, Osman Mutlu, Erdem Yörük, Farhana Ferdousi Liza, Ritesh Kumar, and Shyam Ratan. 2021. Multilingual protest news detection - shared task 1, CASE 2021. In *Proceedings of the 4th Workshop on Challenges and Applications of Automated Extraction of Socio-political Events from Text (CASE 2021)*, pages 79–91, Online. Association for Computational Linguistics.

Ali Hürriyetoğlu, Erdem Yörük, Osman Mutlu, Fırat Duruşan, Çağrı Yoltar, Deniz Yüret, and Burak Gürel. 2021. Cross-Context News Corpus for Protest Event-Related Knowledge Base Construction. *Data Intelligence*, 3(2):308–335.

Chris Jenkins, Shantanu Agarwal, Joel Barry, Steven Fincke, and Elizabeth Boschee. 2023. Massively multi-lingual event understanding: Extraction, visualization, and search.

Nelleke Oostdijk, Ali Hürriyetoglu, Marco Puts, Piet Daas, and Antal van den Bosch. 2016. Information extraction from social media : A linguistically motivated approach. In *Actes de la conférence conjointe JEP-TALN-RECITAL 2016. Volume 10 : RisqueTAL*, pages 23–33, Paris, France. Association pour le Traitement Automatique des Langues. Extraction d'information des réseaux sociaux : une approche motivée linguistiquement.

Erick Skorupa Parolin, Latifur Khan, Javier Osorio, Patrick T. Brandt, Vito D'Orazio, and Jennifer Holmes. 2021. 3m-transformers for event coding on organized crime domain. In *2021 IEEE 8th International Conference on Data Science and Advanced Analytics (DSAA)*, pages 1–10.

Barun Patra, Vishwas Suryanarayanan, Chala Fufa, Pamela Bhattacharya, and Charles Lee. 2020. ScopeIt: Scoping task relevant sentences in documents. In *Proceedings of the 28th International Conference on Computational Linguistics: Industry Track*, pages 214–227, Online. International Committee on Computational Linguistics.

Kevin Pei, Ishan Jindal, Kevin Chen-Chuan Chang, ChengXiang Zhai, and Yunyao Li. 2023. When to use what: An in-depth comparative empirical analysis of OpenIE systems for downstream applications. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 929–949, Toronto, Canada. Association for Computational Linguistics.

James Pustejovsky and Amber Stubbs. 2012. *Natural Language Annotation for Machine Learning: A guide to corpus-building for applications*. " O'Reilly Media, Inc.".

Lance Ramshaw and Mitch Marcus. 1995. Text chunking using transformation-based learning. In *Third Workshop on Very Large Corpora*.

Marek Rei and Anders Søgaard. 2019. Jointly learning to label sentences and tokens. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01):6916–6923.

David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. 1986. Learning representations by back-propagating errors. *nature*, 323(6088):533–536.

Philip A. Schrodt and Muhammed Yassin Idris. 2014. Three's a charm?: Open event data coding with el:diablo, petrarch, and the open event data alliance.

Yiqi Tong, Yidong Chen, and Xiaodong Shi. 2021. A multi-task approach for improving biomedical named entity recognition by incorporating multi-granularity information. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 4804–4813.

Xiaofeng Wang, Matthew S. Gerber, and Donald E. Brown. 2012. Automatic crime prediction using events extracted from twitter posts. In *Social Computing, Behavioral - Cultural Modeling and Prediction*, pages 231–238, Berlin, Heidelberg. Springer Berlin Heidelberg.

Wei Xiang and Bang Wang. 2019. A survey of event extraction from text. *IEEE Access*, 7:173111–173137.

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.

Erdem Yörük, Ali Hürriyetoğlu, Fırat Duruşan, and Çağrı Yoltar. 2022. Random sampling in corpus design: Cross-context generalizability in automated multicountry protest event collection. *American Behavioral Scientist*, 66(5):578–602.

A Detailed Results

In this chapter of the appendices, tables with detailed results for all the experiments is listed. Each table contains a column named "exp_base" referring to the same baseline results for reference. "sent", "doc" and "sent+doc" columns represent the usage of only sentence classification loss (variation 1), only document classification loss (variation 2), and both sentence and document classification loss (variation 3) in addition to token classification loss when training, respectively.

Acknowledgments

We acknowledge the funding of this study by the European Research Council (ERC) Starting Grant 714868 awarded to Dr. Erdem Yörük for his project Emerging Welfare. Additionally, this research has been supported by the European Union's Horizon 2020 program project ODEUROPA under grant agreement number 101004469.

#num of documents	exp_base	sent	doc	sent+doc
717	64.3420 ± 0.2921	65.3298 ± 0.1894	64.7353 ± 0.2444	64.8228 ± 0.3512
645	64.3836 ± 0.5550	64.0928 ± 0.6175	64.5901 ± 1.0282	64.1613 ± 0.4205
573	62.7110 ± 0.5453	63.6060 ± 0.0960	63.3323 ± 0.7521	63.2426 ± 0.3936
501	62.1977 ± 0.3972	61.9715 ± 0.0423	62.7257 ± 0.2095	62.9463 ± 1.2105
430	60.6711 ± 0.5854	61.6818 ± 1.0062	61.8722 ± 0.1173	61.7387 ± 0.8262
358	61.0600 ± 0.4680	60.8886 ± 0.0812	60.2817 ± 0.2523	60.8147 ± 0.8071
286	58.1234 ± 0.6325	58.0093 ± 0.6460	57.6096 ± 0.1267	58.0629 ± 0.4495
215	55.4766 ± 0.1252	54.4221 ± 0.4860	55.2202 ± 0.9379	56.1987 ± 0.4816
143	51.2678 ± 0.5567	51.4187 ± 0.4740	51.9452 ± 0.4664	49.6417 ± 1.0795
71	39.2988 ± 1.8238	39.5794 ± 1.1141	41.0561 ± 0.8125	40.1392 ± 1.2548

Table 3: Detailed results for experiment set 1, which focuses on the effects of our auxiliary tasks without any data addition.

#num of documents	exp_base	sent	doc	sent+doc
717	64.3420 ± 0.2921	65.3298 ± 0.1894	65.1503 ± 0.4439	65.3221 ± 0.2928
645	64.3836 ± 0.5550	64.0928 ± 0.6175	64.0194 ± 0.0571	65.0234 ± 0.8096
573	62.7110 ± 0.5453	63.6060 ± 0.0960	64.1297 ± 0.7640	63.3397 ± 0.5576
501	62.1977 ± 0.3972	61.9715 ± 0.0423	63.0002 ± 0.3124	62.7757 ± 0.5070
430	60.6711 ± 0.5854	61.6818 ± 1.0062	62.3406 ± 0.0275	61.7372 ± 0.3482
358	61.0600 ± 0.4680	60.8886 ± 0.0812	61.8366 ± 0.4351	61.4128 ± 0.2562
286	58.1234 ± 0.6325	58.0093 ± 0.6460	59.2479 ± 0.7000	58.2156 ± 0.9521
215	55.4766 ± 0.1252	54.4221 ± 0.4860	55.9617 ± 0.4265	56.1648 ± 1.0320
143	51.2678 ± 0.5567	51.4187 ± 0.4740	52.3773 ± 0.7498	50.5637 ± 3.4775
71	39.2988 ± 1.8238	39.5794 ± 1.1141	43.2710 ± 0.8947	43.9792 ± 0.2619

Table 4: Detailed results for experiment set 2, which focuses on the effect of adding negatively labeled documents with no event information.

#num of token annotated documents	#num of extra auxiliary data	exp_base	sent	doc	sent+doc
717	0	64.3420 ± 0.2921	65.3298 ± 0.1894	65.1503 ± 0.4439	65.3221 ± 0.2928
645	72	64.3836 ± 0.5550	64.5663 ± 0.5121	64.2650 ± 0.1842	65.2259 ± 0.3991
573	144	62.7110 ± 0.5453	64.1659 ± 0.8118	63.0057 ± 0.4911	63.4717 ± 0.0919
501	216	62.1977 ± 0.3972	63.4803 ± 0.3426	62.7125 ± 0.1686	63.4292 ± 0.1548
430	287	60.6711 ± 0.5854	63.5683 ± 0.4099	61.9322 ± 0.9456	63.0599 ± 0.0922
358	359	61.0600 ± 0.4680	61.7831 ± 0.3405	61.4165 ± 0.5701	61.5980 ± 0.4963
286	431	58.1234 ± 0.6325	60.1496 ± 0.2897	59.1478 ± 0.1192	60.2890 ± 0.2159
215	502	55.4766 ± 0.1252	57.7715 ± 0.7474	56.2983 ± 0.3384	58.2433 ± 0.9674
143	574	51.2678 ± 0.5567	55.0612 ± 0.6047	54.7686 ± 0.6523	55.4178 ± 0.1332
71	646	39.2988 ± 1.8238	49.4441 ± 1.1924	49.7398 ± 0.4377	51.8297 ± 0.9912

Table 5: Detailed results for experiment set 3, which focuses on the effects of adding extra coarse-grained data.

#num of token annotated documents	#num of extra auxiliary data	exp_base	sent	doc	sent+doc
358	359	61.0600 ± 0.4680	61.7831 ± 0.3405	61.4165 ± 0.5701	61.5980 ± 0.4963
286	359	58.1234 ± 0.6325	59.8364 ± 0.2759	59.7761 ± 0.8661	59.7066 ± 0.2504
215	359	55.4766 ± 0.1252	57.8553 ± 0.5770	56.6985 ± 0.5273	58.2742 ± 0.9637
143	359	51.2678 ± 0.5567	54.6631 ± 0.5420	53.9005 ± 0.9373	54.7954 ± 0.4511
71	359	39.2988 ± 1.8238	49.0730 ± 0.3271	48.7882 ± 0.2006	49.6189 ± 0.7479

Table 6: Detailed results for experiment set 3.1, which is variation of experiment set 3 where extra data size is fixed.

#num of token annotated documents	exp_base	sent	doc	sent+doc
20	12.8237 ± 4.0776	43.2263 ± 0.3178	37.8599 ± 0.5719	42.5216 ± 0.5838
10	0.0000 ± 0.0000	35.4900 ± 0.7729	31.1247 ± 0.8497	34.3691 ± 1.0103
5	0.0000 ± 0.0000	21.9365 ± 3.4075	9.9743 ± 3.1956	11.9494 ± 4.2202
3	0.0000 ± 0.0000	12.4881 ± 3.2203	9.7442 ± 2.7849	7.9693 ± 4.0605
1	0.0000 ± 0.0000	3.3871 ± 2.5824	0.5398 ± 0.3380	2.8559 ± 0.4523

Table 7: Detailed results for experiment set 3.2, which is variation of experiment set 3 with tiny data sizes.