Automated Extraction of Socio-political Events from Text

Dominik Stammbach
ETH Zürich
dominik.stammbach@gess.ethz.ch

Didem Durukan
University of Zürich
eminedidem.durukan@student.ethz.ch

He Liu
University of Zürich
heliuhe@student.ethz.ch

1 Introduction

The need for precise and high-quality information about a wide variety of events ranging from political violence, environmental catastrophes, conflict, to international economic and health crises has been increasing. Furthermore, the documentation of such conflicts, e.g. through traditional media, NGOs and social media also is at an all time high, the further the digital age proceeds. This leads to an ever-growing amount of conflicts being described in an abundance of data, and thus calls for automation of extracting such information at scale.The resulting data is important for preventing or resolving conflicts, or improving the lives of and protect citizens in a variety of ways. Researchers also have an interest, as this data can lead to valuable social science experiments and provide insights to prevent further conflicts. These reasons combined are the cause why governments, local and global NGOs and researchers might be highly interested in gathering such socio-political information about events.

We hope that Automated Extraction of Socio-political events provides the resources for large scale socio-political event information collection across sources, countries, and languages (Hürriyetoğlu et al., 2020). Hence our choice of tackling this problem and dataset. In this project, our goal is to extract these socio-political events from raw text. We use manually annotated data, namely the CASE dataset[1], which has been used to organize a shared task and a workshop event at ACL-IJCNLP 2021.

The dataset is described in more detail in (Hürriyetoğlu et al., 2020a). As mentioned in the proposal we have four subtasks that we want to work on (we only tackle the English version of the subtask – the dataset offers the same task across different langauges and provides another two tasks which we did not conisder because this might go beyond the scope of this project).

1. Document classification (Does a news article contain information about a past or ongoing event?)
2. Sentence classification (Does a sentence contain information about a past or ongoing event?)
3. Event sentence coreference identification (Which event sentences (subtask 2) are about the same event?)
4. Event extraction (What is the event trigger and its arguments?)

Given the annotated data, we tackle the task as a supervised learning problem, using recent methods in NLP, namely transformer-based architectures such as RoBERTa (Liu et al., 2019) and BigBird (Zaheer et al., 2021).

2 The Tasks in More Detail

2.1 Subtask 1

Subtask 1 is a straight forward sequence classification problem. We have 9.3K annotated English documents (and 1K Spanish, 1.5K Portuguese documents). The task is to determine whether a document contains information about an event. The mean length of the documents is 171 tokens. Furthermore, the class distribution is imbalanced and we only have 1.9K documents in which an event of interest is described (20% of all documents). An example of a document containing an event of interest is *"PAU staff's hunger strike continues - Indian Express Express News Service , Express News Service : Ludhiana, Tue Dec 22 2009, 04:49 hrs The chain hunger strike against the s"*. Judging from

[1] https://emw.ku.edu.tr/case-2021/

System	Task 1 (F1)	Task 2 (F1)
Random baseline	45.79	45.06
TF-IDF SVM	75.45	69.55
RoBERTa-base	85.97	88.91
BigBird-base	87.01	88.43
Roberta-base pretrained on Task 1	na	88.74
BigBird-base pretrained on Task 1	na	88.48
hline BigBird-large	87.38	n/a
Debertav2-xl-mnli	n/a	89.42

Table 1: First results

this example, we see that the data has probably been automatically preprocessed and is not entirely clean.

2.2 Subtask 2

Subtask 2 is similar in spirit. We are trying to answer the following question: Does a sentence contain information about a past or ongoing event, hence it is a sequence classification task again, but on shorter sequences of text. An example of a sentence of interest is *”There was no formal notice of strike and this has created a lot of hassle for the public.”*.

2.3 Subtask 3

In this subtask, we are expected to determine the relationship between events and cluster the sentences in paragraphs based on the event categories they include. This task is a coreference resolution problem, and the analyzing process includes: (a) finding out the event tokens in sentences; (b) figuring out the coreference between event tokens in different sentences; (c) clustering sentences from the coreference. Unfortunately, we encountered difficulties in understanding the meaning of *sentence_no*. We plan to look into details and solve this problem in the remaining time of the project.

2.4 Subtask 4

Finally, in subtask4 we plan to extract fine-graned arguments of interest of a specific event, e.g. ”What is the event trigger and its arguments?” This again is a classification task, but we produce a classification for each token in a sequence.

To evaluate, we use the official macro-F1 for evaluating subtasks 1 and 2, and accuracy for subtask 4. We report these metrics for a random split of the train set (75%-25% train-test split) which we used to determine the architecture choice. And we also report results of our system’s performance on the official blind testset[2] of the shared task.

3 Dev Results and experiment

3.1 Subtask 1 and 2

We fine-tuned two transformer models on subtask 1 (document classification) and subtask 2 (sentence classification). We find that both transformer approaches outperform a TF-IDF SVM by at least 10% points. BigBird works better on subtask 1, presumably because many of the documents exceed the maximum sequence length of 512 subwords for RoBERTA. BigBird has access to this additional information which seems to help for this task. For sentence classification (shorter sequences) RoBERTa and BigBird perform very similarly, which is to be expected since both have access to the whole sequence. For the sentence classification task, we find that larger models perform better, and we obtain the best results using a Debertav2-xl model which was previously fine-tuned on the MNLI dataset.

Additionally, we further fine-tuned our checkpoints obtained by training on subtask 1 on subtask 2, however we get close to identical results (slightly worse in fact). We speculate that subtask 2 is contained in itself and models converge to a decent solution in any case, without the need for pre-training on document classification first. We observe this pattern for both transformer models.

3.2 Subtask 3

We explored the training data of this subtask and found that there are 62 different *sentence_no* in the training data’s 596 entities and the larger the number the fewer the occurrence (Figure 1 and 2). The numbers are not consecutive and the numbers

[2] https://competitions.codalab.org/competitions/31639#results

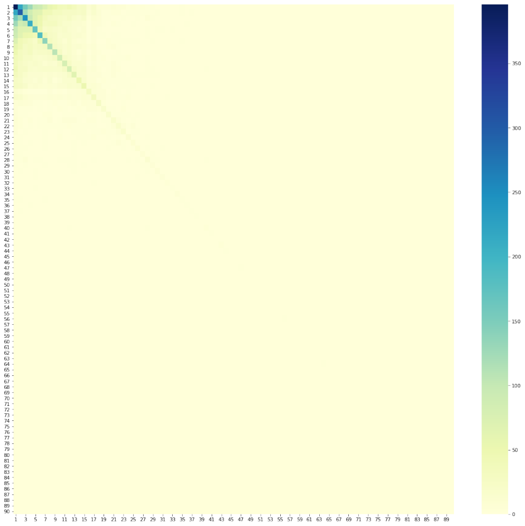


Figure 1: Subtask 3 Co-occurrence Matrix

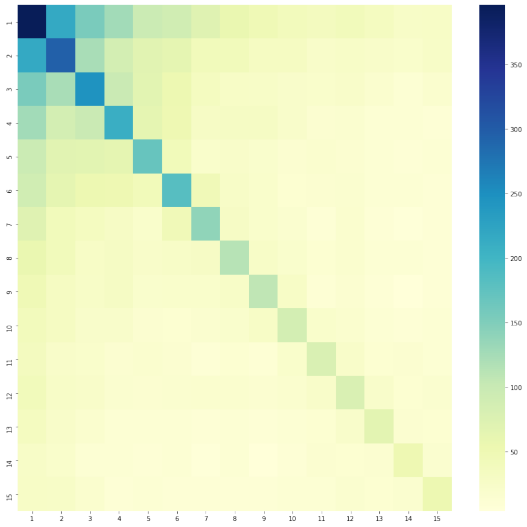


Figure 2: Subtask 3 Co-occurrence Matrix for 1-15

larger than 15 occur occasionally. Some numbers even exist in only one cluster or always be clustered standalone.

If *sentence_no* in the dataset refers to the kind of events, we can assume that some events are the superset of other events, and some events may have the same level as other events so they are never in the same cluster. Figure 3 is a bigram occurrence map for events showed more than twice.

We evaluated a pre-trained SpanBERT model but its performance is even worse than always clustering all sentences together. This model preferably outputs smaller clusters, which is not the case for training data, and we still have no idea how to train the model. We are still trying to find an appropriate training strategy.

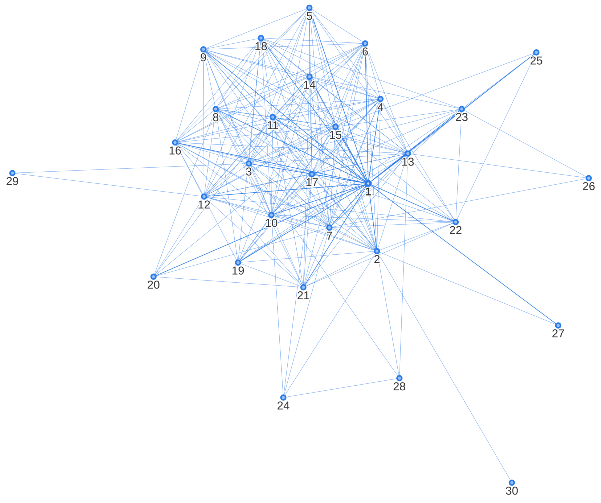


Figure 3: Subtask 3 connections between sentence numbers

System	Task 4 (Acc)
bigbird-roberta-base, 3 epochs	28.34%
bigbird-roberta-base, 20 epochs	74.08%

Table 2: First results

3.3 Subtask 4

For subtask 4, we again fine-tune a bigbird checkpoint in NER-style fashion. We first started with a RoBERTa-checkpoint, but found out that roughly 20% of the sequences are above 512 tokens, so we switched to bigbird tu avoid sliding window approaches. We performed mostly experiments regarding the number of epochs, becaues training for too little doesn't yields poor performance.

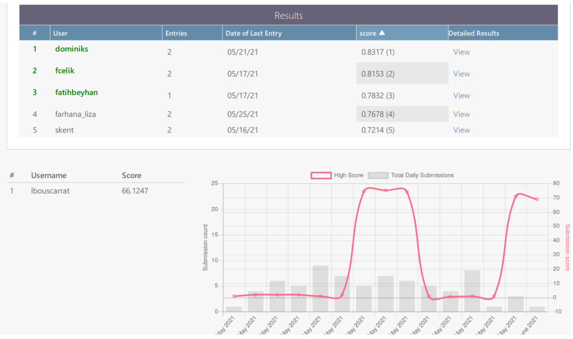
We selected 25% train split to fine-tune number of epochs and then trained a bigbird-roberta-large model using our best configuration (20 epochs) on all training data and used that to predict the test data.

4 Test results

At the time of submission, our approaches yielded best results for subtask 1 and 2 and somewhat competitive results for subtask 4. We show the leaderboards in the following, our account for submitting is *dominiks*

5 Future Work

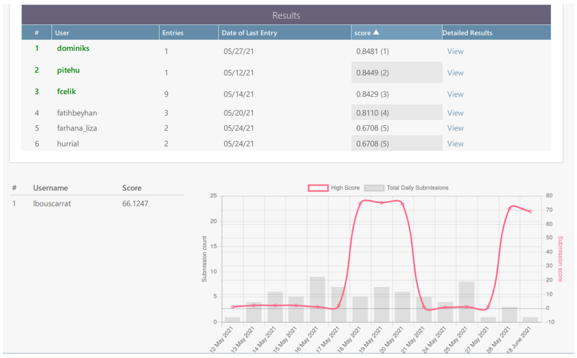
In the remainder of this project, we plan to develop a convincing solution for Subtask 3 and also perhaps further work on our system for Subtask 4. We also plan to write up the project report in more



Results					
#	User	Entries	Date of Last Entry	score ▲	Detailed Results
1	dominiks	2	05/21/21	0.8317 (1)	View
2	fcelik	2	05/17/21	0.8153 (2)	View
3	fatihbeyhan	1	05/17/21	0.7832 (3)	View
4	farhana_liza	2	05/25/21	0.7678 (4)	View
5	skent	2	05/16/21	0.7214 (5)	View

#	Username	Score
1	lbouscarrat	66.1247

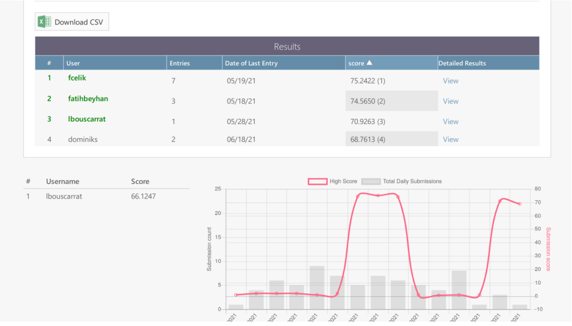
Figure 4: Subtask 1



Results					
#	User	Entries	Date of Last Entry	score ▲	Detailed Results
1	dominiks	1	05/27/21	0.8481 (1)	View
2	pitehu	1	05/12/21	0.8449 (2)	View
3	fcelik	9	05/14/21	0.8429 (3)	View
4	fatihbeyhan	3	05/20/21	0.8110 (4)	View
5	farhana_liza	2	05/24/21	0.6708 (5)	View
6	hurrial	2	05/24/21	0.6708 (5)	View

#	Username	Score
1	lbouscarrat	66.1247

Figure 5: Subtask 2



Download CSV

Results					
#	User	Entries	Date of Last Entry	score ▲	Detailed Results
1	fcelik	7	05/19/21	75.2422 (1)	View
2	fatihbeyhan	3	05/18/21	74.5650 (2)	View
3	lbouscarrat	1	05/28/21	70.9263 (3)	View
4	dominiks	2	06/18/21	68.7613 (4)	View

#	Username	Score
1	lbouscarrat	66.1247

Figure 6: Subtask 4

detail. Lastly, we plan to document what methods have been used by other researchers, especially the ones participating in the shared task. However, system descriptions haven't been released yet. To summarize, We think we are on track time-wise and have completed roughly 66% of the project.

6 Related Works

A similar study about extracting events from text has been covered in a shared task in 2019 (Hürriyetoğlu et al., 2020b), i.e. the competition was to solve Subtask 1,2 and 4, however on a smaller scale (3.5K training documents compared to the SCALE data with 9.3K English training documents). The winning system back then used GRUs to tackle Sbutask 1 and 2, achieving an average F1 of 0.69 (Hürriyetoğlu et al., 2020b).

The state-of-the-art in sequence classification has shifted a lot since then; These days, we use Transformer-based architectures pre-trained on language modeling tasks such as (Devlin et al., 2019; Liu et al., 2019), which are likely to work well for extracting event information.

For Subtask 3, we still find the method to solve the problem. The problem can be regarded as a version of zero-shot textual entailment, e.g. if a sentence entails another, they are likely to end up in a similar document cluster. This has been shown to work for zero-shot relation classification (Obamuyide and Vlachos, 2018) and to work for zero-shot document classification (Yin et al., 2019). Because this would be computationally too expensive, we can retrieve candidate sentences using dense embeddings (Reimers and Gurevych, 2019) and then ask for entailment for all sentences in the candidate set.

Additionally, it might be solved as a typical coreference resolution problem. (Joshi et al., 2019) applied BERT to analyzing coreference on the GAP and OntoNotes and reached state-of-the-art and they stated that BERT-large is particularly good at distinguishing between related but distinct entities but requires improvement in modelling document-level context. In (Joshi et al., 2020), they extended BERT by masking contiguous random spans and training the span boundary representations to predict the entire content of the masked span and outperformed BERT on coreference resolution. We will follow their ideas for our project.

For Subtask 4, we plan to treat it as a NER-type task, similar to (Devlin et al., 2019). It is perceivable that multi-task training with other tasks helps to generalize and increase results on this task.

References

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Ali Hürriyetoğlu, Vanni Zavarella, Hristo Tanev, Erdem Yörük, Ali Safaya, and Osman Mutlu. 2020. Automated extraction of socio-political events from news (AESPEN): Workshop and shared task report. In *Proceedings of the Workshop on Automated Extraction of Socio-political Events from News 2020*, pages 1–6, Marseille, France. European Language Resources Association (ELRA).

Ali Hürriyetoğlu, Erdem Yörük, Deniz Yüret, Osman Mutlu, Çağrı Yoltar, Fırat Duruşan, and Burak Gürel. 2020a. Cross-context news corpus for protest events related knowledge base construction.

Ali Hürriyetoğlu, Erdem Yörük, Deniz Yüret, Çağrı Yoltar, Burak Gürel, Fırat Duruşan, Osman Mutlu, and Arda Akdemir. 2020b. Overview of clef 2019 lab protestnews: Extracting protests from news in a cross-context setting.

Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S. Weld, Luke Zettlemoyer, and Omer Levy. 2020. SpanBERT: Improving pre-training by representing and predicting spans. *Transactions of the Association for Computational Linguistics*, 8:64–77.

Mandar Joshi, Omer Levy, Luke Zettlemoyer, and Daniel Weld. 2019. BERT for coreference resolution: Baselines and analysis. 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 5803–5808, Hong Kong, China. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach.

Abiola Obamuyide and Andreas Vlachos. 2018. Zero-shot relation classification as textual entailment. In *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*, pages 72–78, Brussels, Belgium. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Wenpeng Yin, Jamaal Hay, and Dan Roth. 2019. Benchmarking zero-shot text classification: Datasets, evaluation and entailment approach.

Manzil Zaheer, Guru Guruganesh, Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, and Amr Ahmed. 2021. Big bird: Transformers for longer sequences.