CASE 2021 Task 2: Zero-Shot Classification of Fine-Grained Sociopolitical Events with Transformer Models

Benjamin J. Radford
University of North Carolina at Charlotte
benjamin.radford@uncc.edu

Abstract

We introduce a method for the classification of texts into fine-grained categories of sociopolitical events. This particular method is responsive to all three Subtasks of Task 2, *Fine-Grained Classification of Socio-Political Events*, introduced at the CASE workshop of ACL-IJCNLP 2021. We frame Task 2 as textual entailment: given an input text and a candidate event class ("query"), the model predicts whether the text describes an event of the given type. The model is able to correctly classify in-sample event types with an average F_1-score of 0.74 but struggles with some out-of-sample event types. Despite this, the model shows promise for the zero-shot identification of certain sociopolitical events by achieving an F_1-score of 0.52 on one wholly out-of-sample event class.

1 Introduction

We introduce a method for the classification of text excerpts into fine-grained categories of sociopolitical events. This particular method is responsive to all three Subtasks of Task 2, *Fine-Grained Classification of Socio-Political Events*, introduced at the Challenges and Applications of Automated Extraction of Socio-political Events from Text (CASE) workshop of ACL-IJCNLP 2021 (Haneczok et al., 2021). We frame Task 2 as textual entailment: given an input text and a candidate event class ("query"), the model predicts whether the text describes an event of the given type. Because the query is given as an arbitrary sequence of character tokens, the model is open-ended with respect to query and can, theoretically, predict classes completely out-of-sample.

Three shared task challenges were introduced at CASE: (1) Multilingual Protest News Detection, (2) Fine-Grained Classification of Socio-Political Events, and (3) Discovering Black Lives Matter Events in United States. The second of these is further divided into three Subtasks: (1) supervised text classification of 25 event types, (2) unsupervised text classification of three additional event types, and (3) unsupervised text classification of a further two additional event types. No training data were provided by the shared task organizers; teams were given only the 25 initial event type descriptions. These event types were drawn from the Armed Conflict Location and Event Data Project (ACLED) event ontology (Raleigh et al., 2010). The subsequent five event types introduced by Subtasks 2 and 3 were provided by the shared task organizers immediately prior to the response submission deadline.

2 Data

We downloaded all ACLED events and the corresponding source texts within which those events were discovered. Source texts are short excerpts from news articles and are typically no more than a few sentences in length. We use this event-text corpus as training data for our model. These events and sentences represent only the 25 event types of Subtask 1. Event types by subtask are given in the second column of Table 1. The exact text representation of each event class in Table 1 is the query given to the model. No additional event class descriptors are included. Clearly some event types (e.g., *Abduction forced disappearance*) are more descriptive than others (e.g., *Attack*, *Other*). We partition the 1,127,635 ACLED events into training (80%), validation (10%), and test (10%) sets. However, due to time limitations, neither the validation set nor full test set were used.

To generate training data for the model, we pair every text excerpt with all 24 of the Subtask 1 event types that are not described by the excerpt and assign these artificial pairs a value of zero. We then

Subtask	Event class	Precision	Recall	F_1-score	Support
1	Disrupted weapons use	0.971	0.569	0.717	58
1	Abduction forced disappearance	0.714	0.750	0.732	20
1	Agreement	1.000	0.516	0.681	31
1	Air drone strike	0.786	0.917	0.846	36
1	Armed clash	0.449	0.924	0.604	66
1	Shelling artillery missile attack	0.646	0.861	0.738	36
1	Attack	0.333	0.852	0.479	27
1	Change to group activity	0.571	0.533	0.552	30
1	Chemical weapon	0.867	0.703	0.776	37
1	Arrests	0.684	0.382	0.491	34
1	Excessive force against protesters	0.833	0.652	0.732	23
1	Government regains territory	0.780	0.842	0.810	38
1	Grenade	0.949	0.771	0.851	48
1	Headquarters or base established	0.870	0.909	0.889	22
1	Mob violence	0.314	0.647	0.423	17
1	Non state actor overtakes territory	0.810	0.708	0.756	24
1	Non violent transfer of territory	0.714	0.476	0.571	21
1	Other	0.000	0.000	0.000	8
1	Peaceful protest	0.689	0.895	0.779	57
1	Looting property destruction	0.143	0.048	0.071	21
1	Protest with intervention	0.548	0.773	0.642	22
1	Remote explosive landmine IED	0.522	0.972	0.680	36
1	Sexual violence	0.955	0.913	0.933	23
1	Suicide bomb	0.946	0.854	0.897	41
1	Violent demonstration	0.642	0.642	0.642	53
1	*micro avg*	*0.739*	*0.739*	*0.739*	*829*
1	*macro avg*	*0.770*	*0.697*	*0.698*	*829*
1	*weighted avg*	*0.798*	*0.739*	*0.736*	*829*
2	Organized crime	0.500	0.103	0.171	29
2	Natural disaster	0.562	0.243	0.340	37
2	Man made disaster	0.167	0.019	0.034	52
2	*micro avg*	*0.658*	*0.658*	*0.658*	*947*
2	*macro avg*	*0.648*	*0.632*	*0.613*	*947*
2	*weighted avg*	*0.670*	*0.658*	*0.635*	*947*
3	Attribution of responsibility	0.167	0.071	0.100	28
3	Diplomatic event	0.511	0.523	0.517	44
3	*micro avg*	*0.629*	*0.629*	*0.629*	*1019*
3	*macro avg*	*0.621*	*0.602*	*0.582*	*1019*
3	*weighted avg*	*0.644*	*0.629*	*0.605*	*1019*

Table 1: Event types by subtask. Precision, recall, F_1-score, and support given by class. Averages are given by subtask. Class-wise values are all derived from the single result set for Subtask 3. Averages per subtask are derived from the result set for each particular subtask.

```
('<s> On 16 June, AQAP armed
men peacefully took control and
deployed on Al-Rawdah district
from Houthi forces. No further
info was provided. </s> Non
violent transfer of territory
</s>', 1.0)
```

Figure 1: A correct input text-query pair from ACLED. The first tuple element is a single text string containing a special token <s>, the input sentence, a delimiter </s>, the query, and a final delimiter </s>. The second tuple element is the target value for the text-query pair: 1.0 if correct, 0.0 if incorrect.

assign all observed pairs, text excerpts paired with the correct event type, a value of one. The model's job is to take a text-query pair and predict whether it is a correct pair or an incorrect pair. An example text-query pair from an ACLED event is given in Figure 1.

3 Model

We select a pre-trained RoBERTa model as the base of our solution.[1] RoBERTa is a transformer-based language model that is initially trained on a very large English language corpus and can then be fine-tuned to specific tasks with fewer training examples (Liu et al., 2019). We take the final layer hidden states for each token and apply global max pooling (i.e., find the element-wise maximum for each dimension of the hidden states). We add a fully-connected dense layer with a single neuron and sigmoid activation function to this pooled value. We use Adam to minimize the binary cross-entropy loss of our model (Kingma and Ba, 2015). We train the model for a single epoch with a learning rate of 5×10^{-5} and use a variable batch size to manage memory usage.

During the inference stage, the model must select a single class to best represent each text. All possible queries are appended to an input text and every pair is passed to the model independently. The model produces a prediction between 0.0 and 1.0 for each pair and the event class associated with the text-query pair that receives the highest predicted value is chosen. However, the *Other* category may result in misclassifications: can the model distinguish an out-of-sample class, like those from Subtasks 2 and 3, from the in-sample class *Other*? A second aggregation rule is therefore applied: if the greatest predicted value is associated with the *Other* class, the next highest probability class is inspected. If this runner-up class is out-of-sample (i.e., not present in Subtask 1), then it is chosen. If the runner-up class is present in Subtask 1, then the class *Other* is chosen. Results presented for Subtasks 2 and 3 are derived from this second aggregation method.

4 Results

Model performance is given in Table 1. When constrained to the initial 25 event types (Subtask 1), the model achieves average F_1-scores of between 0.70 and 0.74 depending on the method chosen for averaging. These values drop with the introduction of additional out-of-sample event types, averaging between 0.58 and 0.66 for Subtasks 2 and 3. Zero-shot performance on the five out-of-sample event types varies substantially: the F_1-scores for *Natural disaster* and *Diplomatic event* are 0.34 and 0.52, respectively, values that that fall within the typical range of in-sample event types. The model fares relatively poorly on the remaining out-of-sample types. The results are nearly identical when using the first aggregation method that does not correct for the *Other* category present in Subtask 1.

Comparison of predictions against target classes reveals that class overlap may to be blame for some of the poor out-of-sample performance. For example, the model correctly identifies *Organized crime* only 10% of the time and often misclassifies it as *Arrest* (21%), *Mob violence* (21%), and *Looting property destruction* (17%). One example of this, drawn from the test set, is given in Table 2 row a. The excerpt describes the detention of 34 persons by border guards as part of an enforcement action against an international gang. The model predicts *Arrest* but the given label is *Organized crime*. Another example given in Table 2 row b describes an event in which police recovered $500,000 in stolen property after an investigation into a breaking and entering event. The model predicted *Looting property destruction* but the given label is once again *Organized crime*. The model often misclassifies *Man made disaster* as one of *Remote explosive landmine IED* (37%), *Attack* (19%), and *Natural disaster* (12%). One such example relating to the 2020 Beirut port explosion is given in Table 2 row c. Clearly this is a *Man made disaster*, but it also

[1] Available here: https://huggingface.co/roberta-base.

ID	Text	Prediction	True Label
a.	Polish border guards have detained 34 people from the Middle East, including four women and four children, who were traveling in a trailer of the lorry that came from Turkey via Slovakia, authorities said on Saturday. The event is linked to a known international gang involved in facilitating illegal migration.	Arrest	Organized crime
b.	Toronto police identified five suspects in connection to a residential break and enter investigation dubbed 'Project High Class.' Police said in a media release they recovered $500,000 in stolen property. Toronto police Inspector Joanne Rudnick is expected to provide further information on the investigation at 10:30 a. [sic]	Looting property destruction	Organized crime
c.	On 4 August, two large explosions hit the city of Beirut, reportedly caused by large quantities of ammonium nitrate being stored in a warehouse in Beirut Port.	Remote explosive landmine IED	Natural disaster

Table 2: Examples of incorrectly classified texts.

describes an explosion that is conceivably "remote" (though not intentional).

5 Conclusion

Failure to account for ambiguity between event classes is likely to be an issue for the next generation of automated fine-grained event classification models. In the case of the model presented here, predictions are not necessarily calibrated properly: the model has no ability to specify that a text does not describe one and only one event type. This is enforced by the fact that a final classification is chosen by identifying the maximum value among all text-query pair predictions. Were the model calibrated by class, we would hope that predicted values greater than 0.5 denote a positive class membership and values below this threshold denote non-membership. In that case, multiple classes (or no class) could be indicated by the model for a single text. However, given the zero-shot nature of Subtasks 2 and 3, we were unable to calibrate those particular classes. Furthermore, the organizers have specified that all texts should be assigned one and only one label. However, it seems clear from inspection of the errors that the given ontology does not describe a mutually exclusive set of classes. Accounting for hierarchical or complementary classes within the ontology may help to produce more useful or consistent event coding models. Doing so will require a novel technique for selecting predicted classes in which each class prediction is not made independently of the other classes (as is the case here).

One solution may be to pose all queries to the model simultaneously. A single input example would comprise the source text concatenated with every possible event class: `<s> text </s> cat1 cat2... </s>`. The model would then output a vector of probabilities the same length (in tokens) as the input sequence. Classes for the source text would be chosen by inspecting this probability vector and selecting categories corresponding to relatively high probability-valued sub-sequences. When appropriate, the model may weight multiple (or no) class tokens very highly. Queries could be shuffled per source text to prevent the model from learning offset values for common classes rather than attending to the query texts themselves.

Despite the poor out-of-sample performance of this particular model on certain zero-shot event categories, the model's performance in-sample and on *Natural disaster* and *Diplomatic event* suggests that transformers will play a major role in future event coding systems. With additional time and resources, it is likely that substantial improvements are possible to the model described here. In fact, the performance of this model, given zero hyperparameter tuning or model search, suggests that the upper limit for transformer performance on this task is likely very high.[2]

Acknowledgments

We thank the organizers of CASE 2021 and three anonymous reviewers for their thoughtful feedback.

References

J Haneczok, G Jacquet, J Piskorsk, and N Stefanovitch. 2021. Fine-grained event classification in news-like text snippets shared task 2, case 2021. *Proceedings of the Workshop on Challenges and Applications of Automated Extraction of Socio-political Events from Text (CASE 2021), co-located with the Joint Conference of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (ACL-IJCNLP 2021)*.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

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. *CoRR*, abs/1907.11692.

Clionadh Raleigh, Andrew Linke, Håvard Hegre, and Joakim Karlsen. 2010. Introducing acled-armed conflict location and event data. *Journal of Peace Research*, 47:651–660.

[2] Due to time and resource constraints, we trained only one model and performed no out-of-sample evaluation prior to test set submission.