DAAI at CASE 2021 Task 1: Transformer-based Multilingual Socio-political and Crisis Event Detection

Hansi Hettiarachchi, Mariam Adedoyin-Olowe, Jagdev Bhogal, Mohamed Medhat Gaber
School of Computing and Digital Technology, Birmingham City University, UK
hansi.hettiarachchi@mail.bcu.ac.uk
{mariam.adedoyin-olowe,jagdev.bhogal,mohamed.gaber}@bcu.ac.uk

Abstract

Automatic socio-political and crisis event detection has been a challenge for natural language processing as well as social and political science communities, due to the diversity and nuance in such events and high accuracy requirements. In this paper, we propose an approach which can handle both document and cross-sentence level event detection in a multilingual setting using pretrained transformer models. Our approach became the winning solution in document level predictions and secured the 3^{rd} place in cross-sentence level predictions for the English language. We could also achieve competitive results for other languages to prove the effectiveness and universality of our approach.

1 Introduction

With technological advancements, today, we have access to a vast amount of data related to social and political factors. These data may contain information on a wide range of events such as political violence, environmental catastrophes and economic crises which are important to prevent or resolve conflicts, improve the quality of life and protect citizens. However, with the increasing data volume, manual efforts for event detection have become too expensive making the requirement of automated and accurate methods crucial (Hürriyetoğlu et al., 2020).

Considering this timely requirement, CASE 2021 Task 1: Multilingual protest news detection is designed (Hürriyetoğlu et al., 2021). This task is composed of four subtasks targeting different data levels. Subtask 1 is to identify documents which contain event information. Similarly, subtask 2 is to identify event described sentences. Subtask 3 targets the cross-sentence level to group sentences which describe the same event. The final subtask is to identify the event trigger and its arguments at the entity level. Since a news article can contain one or more events and a single event can be described together with some previous or relevant details, it is important to focus on different data levels to obtain more accurate and complete information.

This paper describes our approach for document and cross-sentence level event detection including an experimental study. Our approach is mainly based on pretrained transformer models. We use improved model architectures, different learning strategies and unsupervised algorithms to make effective predictions. To facilitate the effortless generalisation across the languages, we do not use any language-specific processing or additional resources. Our submissions achieved the 1^{st} place in document level predictions and 3^{rd} place in cross-sentence level predictions for the English language. Demonstrating the universality of our approach, we could obtain competitive results for other languages too.

The remainder of this paper is organised as follows. Section 2 describes the related work done in the field of socio-political event detection. Details of the task and datasets are provided in Section 3. Section 4 describes the proposed approaches. The experimental setup is described in Section 5 followed by results and evaluation in Section 6. Finally, Section 7 concludes the paper. Additionally, we provide our code to the community which will be freely available to everyone interested in working in this area using the same methodology[1].

2 Related Work

In early work, the majority of event detection approaches were data-driven and knowledge-driven (Hogenboom et al., 2011). Since the data-driven approaches are only based on the statistics of the

[1]The GitHub repository is publicly available on https://github.com/HHansi/EventMiner

Proceedings of the 4th Workshop on Challenges and Applications of Automated Extraction of Socio-political Events from Text (CASE) , pages 120–130

underlying corpus, they missed the important semantical relationships. The knowledge-driven or rule-based approaches were proposed to tackle this limitation, but they highly rely on the targeted domains or languages (Danilova and Popova, 2014).

Later, there was a more focus on traditional machine learning-based models (e.g. support vector machines, decision trees) including different feature extraction techniques (e.g. natural language parsing, word vectorisation) (Schrodt et al., 2014; Sonmez et al., 2016). Also, there was a tendency to apply deep learning-based approaches (e.g. CNN, FFNN) too following their success in many information retrieval and natural language processing (NLP) tasks (Lee et al., 2017; Ahmad et al., 2020). However, these approaches are less expandable to low-resource languages, due to the lack of training data to fine-tune the models.

Targeting this major limitation, in this paper we propose an approach which is based on pretrained transformer models. Due to the usage of general knowledge available with the pretrained models and their multilingual capabilities, our approach can easily support event detection in multiple languages including low-resource languages.

3 Subtasks and Data

CASE 2021 Task 1: Multilingual protest news detection is composed of four subtasks targeting event information at document, sentence, cross-sentence and token levels (Hürriyetoğlu et al., 2021). Mainly the socio-political and crisis events which are in the scope of contentious politics and characterised by riots and social movements are focused. Among these subtasks, we participated in subtask 1 and subtask 3 which are further described below.

Subtask 1: Document Classification Subtask 1 is designed as a document classification task. Participants need to predict a binary label of '1' if the news article contains information about a past or ongoing event and '0' otherwise. To preserve the multilinguality of the task, four different languages English, Spanish, Portuguese and Hindi have been considered for data preparation. Comparatively, a high number of training instances were provided with English than Spanish and Portuguese. No training data were provided for the Hindi language. For final evaluations, test data were provided without labels. The data split sizes in each language are summarised in Table 1.

Language	Train	Test
English (en)	9324	2971
Spanish (es)	1000	250
Portuguese (pt)	1487	372
Hindi (hi)	-	268

Table 1: Data distribution over train and test sets in subtask 1

Subtask 3: Event Sentence Coreference Identification (ESCI) Subtask 3 is targeted at the cross-sentence level with the intention to identify the coreference of sentences or sentences about the same event. Given event-related sentences, the targeted output is the clusters which represent separate events. As training data, per instance, a set of sentences and corresponding event clusters were provided as shown below:

```
{"sentence_no":[1,2,3],
"sentences":[
    "Maoist banners found 10th
       April 2011 05:14 AM
       KORAPUT : MAOIST banners
       were found near the
       District Primary Education
        Project ( DPEP ) office
       today in which the ultras
       threatened to kill Shikhya
        Sahayak candidates ,
       outsiders to the district
       , who have been selected
       to join the service here
       .",
    "Maoists , in the banners ,
       have also demanded release
        of hardcore cadre Ghasi
       who was arrested by police
        earlier this week .",
    "Similar banners were also
       found between Sunki and
       Ampavalli where Maoists
       also blocked road by
       felling trees ."],
"event_clusters":[[1,2],[3]]}
```

Listing 1: Subtask 3 training data sample

Data from three different languages: English, Spanish and Portuguese were provided. A few training data instances are available with non-English languages as summarised in Table 2. Simi-

lar to subtask 1, test datasets were provided with no labels (event clusters) to use with final evaluations.

Language	Train	Test
English (en)	596	100
Spanish (es)	11	40
Portuguese (pt)	21	40

Table 2: Data distribution over train and test sets in subtask 3

4 Methodology

The main motivation behind the proposed approaches for event document identification and event sentence coreference identification is the recent success gained by transformer-based architectures in various NLP and information retrieval tasks such as language detection (Jauhiainen et al., 2021) question answering (Yang et al., 2019) and offensive language detection (Husain and Uzuner, 2021; Ranasinghe and Zampieri, 2021). Apart from providing strong results compared to RNN based architectures, transformer models like BERT (Devlin et al., 2019) and XLM-R (Conneau et al., 2020) provide pretrained language models that support more than 100 languages which is a huge benefit when it comes to multilingual research. The available models have been trained on general tasks like language modelling and then can be fine-tuned for downstream tasks like text classification (Sun et al., 2019). Depending on the nature of the targeted subtask, we involved different transformer models along with different learning strategies to extract event information as mentioned below.

4.1 Subtask1: Document Classification

Document classification can be considered as a sequence classification problem. According to recent literature, transformer architectures have shown promising results in this area (Ranasinghe et al., 2019b; Hettiarachchi and Ranasinghe, 2020).

Transformer models take an input of a sequence and output the representations of the sequence. The input sequence could contain one or two segments separated by a special token [SEP]. In this approach, we considered a whole document or a news article as a single sequence and no [SEP] token is used. As the first token of the sequence, another special token [CLS] is used and it returns a special embedding corresponding to the whole sequence which is used for text classification tasks (Sun et al., 2019). A simple softmax classifier is added to the top of the transformer model to predict the probability of a class. The architecture of the transformer-based sequence classifier is shown in Figure 1.

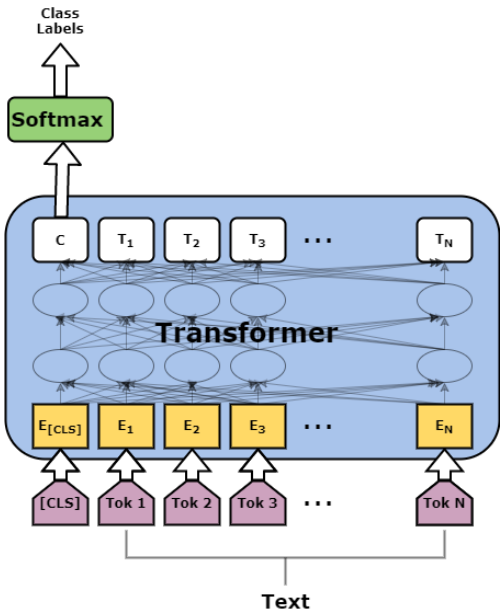

Figure 1: Text Classification Architecture

Unfortunately, the majority of transformer models such as BERT (Devlin et al., 2019) and XLM-R (Conneau et al., 2020) fails to process documents with a higher sequence length than 512. This limitation is introduced due to the self-attention operation used by these architectures which scale quadratically with the sequence length (Beltagy et al., 2020). Therefore, we specifically focused on improved transformer models targetting long documents: Longformer (Beltagy et al., 2020) and BigBird (Zaheer et al., 2020). Longformer utilises an attention mechanism that scales linearly with sequence length and BigBird utilises a sparse attention mechanism to handle long sequences.

Data Preprocessing: We applied a few preprocessing techniques to data before inserting them into the models. All the selected techniques are language-independent to support multilingual experiments. Analysing the datasets, there were documents with very low sequence length (< 5) and they were removed. Further, URLs were removed and repeating symbols more than three times (e.g. =====) were replaced by three occurrences (e.g. ===) because they are uninformative.

4.2 Subtask3: ESCI

Event Sentence Coreference Identification (ESCI) can be considered as a clustering problem. If a set of sentences are assigned to clusters based on their semantic similarity, each cluster will represent separate events. To perform clustering, each sentence needs to be mapped to an embedding which preserves its semantic details.

4.2.1 Sentence Embeddings

Different approaches were proposed to obtain sentence embeddings by previous research. Based on the word embedding models such as GloVe (Pennington et al., 2014), the average of word embeddings over a sentence was used. Later, more improved architectures like InferSent (Conneau et al., 2017) which is based on a siamese BiLSTM network with max pooling, and Universal Sentence Encoder (Cer et al., 2018) which is based on a transformer network and augmented unsupervised learning were developed. However, with the improved performance on NLP tasks by transformers, there was a tendency to input sentences into models like BERT and get the output of the first token ([CLS]) or the average of output layer as a sentence embedding (May et al., 2019; Qiao et al., 2019). These approaches were found as worse than average GloVe embeddings due to the architecture of BERT which was designed targeting classification or regression tasks (Reimers et al., 2019).

Considering these limitations and characteristics of transformer-based models, Reimers et al. (2019) proposed a new architecture named Sentence Transformer (STransformer), a modification to the transformers to derive semantically meaningful sentence embeddings. According to the experimental studies, STransformers outperformed average GloVe embeddings, specialised models like InferSent and Universal Sentence Encoder, and BERT embeddings (Reimers et al., 2019). Considering these facts, we adopt STransformers to generate sentence embeddings in our approach.

STransformer creates a siamese network using transformer models like BERT to fine-tune the model to produce effective sentence embeddings. A pooling layer is added to the top of the transformer model to generate fixed-sized embeddings for sentences. The siamese network takes a sentence pair as the input and passes them through the network to generate embeddings (Ranasinghe et al., 2019a). Then compute the similarity between embeddings using cosine similarity and compare the value with the gold score to fine-tune the network. The architecture of STransformer is shown in Figure 2.

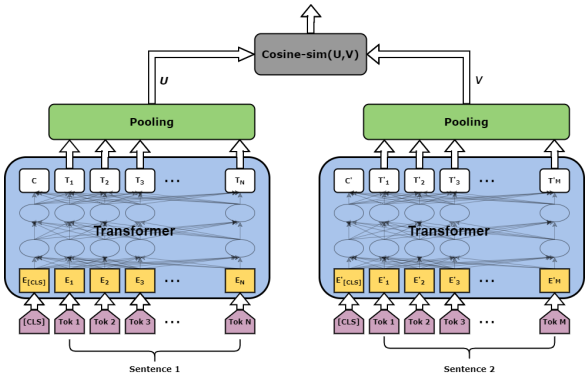

Figure 2: Siamese Sentence Transformer (STransformer) Architecture

Data Formatting: To facilitate the STransformer fine-tuning or training, we formatted given sentences into pairs and assigned the similarity of '1' if both sentences belong to the same cluster and '0' if not. During the pairing, the order of sentences is not considered. Thus, for n sentences, $(n \times (n-1))/2$ pairs were generated. For example, sentence pairs and labels generated for the data sample given in Listing 1 are shown in Table 3.

Sentence 1	Sentence 2	Label
1	2	1
1	3	0
2	3	0

Table 3: Sentence pairs and labels of data sample in Listing 1

4.2.2 Clustering

As clustering methods, we focused on hierarchical clustering and the pairwise prediction-based clustering approach proposed by Örs et al. (2020). Hierarchical clustering is widely used with event detection approaches over flat clustering because flat clustering algorithms (e.g. K-means) require the number of clusters as an input which is unpredictable (Hettiarachchi et al., 2021). Considering the availability of training data and recent successful applications, the pairwise prediction-based clustering approach is focused.

Hierarchical Clustering: For the hierarchical clustering algorithm, we used Hierarchical Agglomerative Clustering (HAC). Each sentence is

converted into embeddings to input to the clustering algorithm. HAC considers all data points as separate clusters at the beginning and then merge them based on cluster distance using a linkage method. The tree-like diagram generated by this process is known as a dendrogram and a particular distance threshold is used to cut it into clusters (Manning et al., 2008). For the distance metric, cosine distance is used, because it proved to be effective for measurements in textual data (Mikolov et al., 2013; Antoniak and Mimno, 2018) and a variant of it is used with STransformer models. For the linkage method, single, complete and average schemes were considered for initial experiments and the average scheme was selected among them because it outperformed others. We picked the optimal distance threshold automatically using the training data. If training data is further split into training and validation sets to use with STransformers, only the validation set is used to pick the cluster threshold, because the rest of the data is known to the embedding generated model.

Pairwise Prediction-based Clustering: We used the pairwise prediction-based clustering algorithm proposed by Örs et al. (2020) which became the winning solution of the ESCI task in the AESPEN-2020 workshop (Hürriyetoğlu et al., 2020). Originally this algorithm used the BERT model to predict whether a certain sentence pair belongs to the same event or not. In this research, we used STransformers to make those predictions except general transformers. Since a STransformer model is designed to obtain embeddings, to derive labels (i.e. '1' if the sentence pair belong to the same event and '0' if not) from them we used cosine similarity with a threshold. The optimal value computed during the model evaluation process is used as the threshold.

5 Experimental Setup

This section describes the learning configurations, transformer models and hyper-parameters used for the experiments.

5.1 Learning Configurations

We focused on different learning configurations depending on data and model availability, and multilingual setting. Considering the availability of data and models, we used the following configurations for the experiments.

Pretrained (No Learning): Pretrained models are used without making any modifications to them to make the predictions. In this case, models pretrained using a similar objective to the target objective need to be selected.

Fine-tuning: Under fine-tuning, we retrain an available model to a downstream task or the same task model already trained. This learning allows the model to be familiar with the targeted data.

From-scratch Learning: Models are built from scratch using the targeted data. This procedure helps to mitigate the unnecessary biases made by the data used to train available models.

Language Modelling (LM): In LM, we retrain the transformer model on the targeted dataset using the model's initial training objective before fine-tuning it for the downstream task. This step helps increase the model understanding of data (Hettiarachchi and Ranasinghe, 2020).

For multilingual data, the following configurations are considered to support both high- and low-resource languages.

Monolingual Learning: In monolingual learning, we build the model from the training data only from that particular language.

Multilingual Learning: In multilingual learning, we concatenate available training data from all languages and build a single model.

Zero-shot Learning: In zero-shot learning, we use the models fine-tuned for the same task using training data from other language(s) to make the predictions. The multilingual and cross-lingual nature of the transformer models has provided the ability to do this (Ranasinghe et al., 2020; Hettiarachchi and Ranasinghe, 2021).

5.2 Transformers

We used monolingual and multilingual general transformers as well as pretrained STransformers for our experiments.

General Transformers: As monolingual models, we used transformer models built for each of the targeted languages. For English, BigBird (Zaheer et al., 2020), Longformer (Beltagy et al., 2020) and BERT English (Devlin et al., 2019) models were considered. For Spanish, BETO (Canete et al., 2020) and for Portuguese, BERTimbau (Souza

Seq. Length	Model	Macro R	Macro P	Macro F1
256	BERT-large-cased	0.8717	0.8489	0.8595
	BigBird-roberta-large	0.8790	0.9119	**0.8941**‡
	Longformer-base	0.8800	0.8868	0.8833
512	BERT-large-cased	0.8697	0.8683	0.8690
	BigBird-roberta-base	0.8763	0.9018	0.8882‡
	Longformer-base	0.8608	0.9100	0.8824‡
700	BigBird-roberta-base	0.8770	0.8807	0.8788
	Longformer-base	0.8748	0.8846	0.8796

Table 4: Results: Macro Recall (R), Precision (P) and F1 of document classification experiments for English using different sequence lengths and models. Best is in Bold and submitted systems are marked with ‡.

	Model	Training Data	Macro R	Macro P	Macro F1
English	BERT-multilingual-cased	en+es+pt	0.8505	0.8567	0.8536
	XLM-R-base	en+es+pt	0.8280	0.8727	0.8476
Spanish	BETO-cased	es	0.6944	0.8681	0.7475‡
	BERT-multilingual-cased	es	NT	NT	NT
	BERT-multilingual-cased	en+es+pt	0.7831	0.8111	0.7962‡
	XLM-R-base	es	NT	NT	NT
	XLM-R-base	en+es+pt	0.7888	0.8530	**0.8167**‡
Portuguese	BERTimbau-large	pt	0.7672	0.8900	0.8126‡
	BERT-multilingual-cased	pt	0.7595	0.8331	0.7896
	BERT-multilingual-cased	en+es+pt	0.8384	0.8890	**0.8611**‡
	XLM-R-base	pt	NT	NT	NT
	XLM-R-base	en+es+pt	0.7845	0.8449	0.8104‡

Table 5: Results of multilingual document classification experiments. Training Data column summarises the language(s) of used datasets to train models. Due to training data limitations, a few models were found to be not trainable and they are indicated with NT. Best is in Bold and submitted systems are marked with ‡.

et al., 2020) models which are variants of the BERT model were considered. As multilingual models, BERT multilingual version and XLM-R (Conneau et al., 2020) models were used. Among these models, a higher sequence length than 512 is only supported by BigBird and Longformer models available for English. We used HuggingFace's Transformers library (Wolf et al., 2020) to obtain the models.

Sentence Transformers: STransformers provide pretrained models for different tasks[2]. Among them, we selected the best-performed models trained for semantic textual similarity (STS) and duplicate question identification, because these areas are related to the same event prediction.

5.3 Hyper-parameter Configurations

We used a Nvidia Tesla K80 GPU to train the models. Each input dataset is divided into a training set and a validation set using a 0.9:0.1 split. We predominantly fine-tuned the learning rate and the number of epochs of the model manually to obtain the best results for the validation set. For document classification, we obtained $1e^{-}5$ as the best value for the learning rate and 3 as the best value for the number of epochs. The same learning rate was found as the best value for STransformers with epochs of 5. For the sequence length, different values have experimented with document classification and they are further discussed in Section 6.1. A fixed sequence length of 136 was used for ESCI considering its data.

To improve the performance of document classification, we used the majority-class self-ensemble approach mentioned in (Hettiarachchi and Ranasinghe, 2020). During the training, we trained three models with different random seeds and considered the majority-class returned by the models as the final prediction.

To train STransformers, we selected the online contrastive loss, an improved version of the con-

[2]Sentence Transformer pretrained models are available on https://www.sbert.net/docs/pretrained_models.html

	Model	Training Data	Seq. Length	Macro F1
English	Best System			0.8455
	BigBird-roberta-large	en	256	
	BigBird-roberta-base	en	512	0.8220
Spanish	Best System			0.7727
	XLM-R-base	en+es+pt	512	0.6931
	BERT-multilingual-cased	en+es+pt	512	0.6886
Portuguese	Best System			0.8400
	XLM-R-base	en+es+pt	512	0.8243
	BERT-multilingual-cased	en+es+pt	512	0.7982
Hindi	Best System			0.7877
	XLM-R-base	en+es+pt	512	0.7707
	BERT-multilingual-cased	en+es+pt	512	0.4647

Table 6: Document classification results for test data

trastive loss function. The contrastive loss function learns the parameters by reducing the distance between neighbours or semantically similar embeddings and increasing the distance between non-neighbours or semantically dissimilar embeddings (Hadsell et al., 2006). The online version automatically detects the hard cases (i.e. negative pairs with a low distance than the largest distance of positive pairs and positive pairs with a high distance than the lowest distance of negative pairs) in a batch and calculates the loss only for them.

6 Results and Evaluation

In this section, we report the conducted experiments and their results.

6.1 Subtask1: Document Classification

Task organisers used Macro F1 as the evaluation metric for subtask 1. Since only the training data were released, we separated a dev set from each training dataset to evaluate our approach. Depending on the data size, 20% from English and 10% from other-language training data were separated as dev data.

Initially, we analysed the performance of fine-tuned document classifiers for English using BERT and improved transformer models for long documents, along with varying sequence length. Considering the sequence length distribution in data, we picked the lengths of 256, 512 and 700 for these experiments. The obtained results are summarised in Table 4. Even though we targeted *large* versions of the models (e.g. BigBird-roberta-large), due to the resource limitations, we had to use *base* versions (e.g. BigBird-roberta-base) for some experiments. According to the results, BERT models improve the F1 when we increase the sequence length. In contrast to it, both BigBird and Longformer models have higher F1 with low sequence lengths.

For predictions in Spanish and Portuguese documents, we fine-tuned the models using both monolingual and multilingual learning approaches. Since transformers with the maximum sequence length of 512 are used, we fixed the sequence length to 512 based on the findings in English experiments. The obtained results and training configurations are summarised in Table 5. For the high-resource language (i.e. English), multilingual learning returns a low F1 than monolingual learning. However, low-resource languages show a clear improvement in F1 with multilingual learning. Since there were no training data for the Hindi language, the best multilingual models were picked to apply the zero-shot learning approach.

We report the results we obtained for test data in Table 6. According to the results, our approach which used the BigBird model became the best system for the English language. For other languages, multilingual learning performed best. Among models, XLM-R outperformed the BERT-multilingual model. Compared to the best systems submitted, our approach has very competitive results for these languages too.

6.2 Subtask3: ESCI

To evaluate subtask 3 responses, organisers used CoNLL-2012 average score[3] (Pradhan et al., 2014). Similar to subtask 1, for evaluation purpose, we separated 20% from the English training dataset as dev data. There were no sufficient data in other

[3]The implementation of the scorer is available on https://github.com/LoicGrobol/scorch

	Base Model	STransformer	Clustering	CoNLL Average Score
Pretrained	DistilBERT-base-uncased	quora-distilbert-base	HAC	0.8360
	MPNet-base	stsb-mpnet-base-v2	HAC	0.8360
Fine-tune	DistilBERT-base-uncased	quora-distilbert-base	HAC	0.8392
	DistilBERT-base-uncased	quora-distilbert-base	(Örs et al., 2020)	0.8376
	MPNet-base	stsb-mpnet-base-v2	HAC	0.8370
	MPNet-base	stsb-mpnet-base-v2	(Örs et al., 2020)	0.8264
From-scratch	BERT-large-cased	-	HAC	**0.8688**‡
	BERT-large-cased	-	(Örs et al., 2020)	0.8656‡
LM + From-scratch	BERT-large-cased	-	HAC	0.8543‡
	BERT-large-cased	-	(Örs et al., 2020)	0.8328

Table 7: Results of ESCI for English along with different strategies experimented. Best is in Bold and submitted systems are marked with ‡.

	Base Model	STransformer	Clustering	CoNLL Average Score
Pretrained	DistilBERT-base-uncased	quora-distilbert-multilingual	HAC	0.8360
Fine-tune	DistilBERT-base-uncased	quora-distilbert-multilingual	HAC	0.8423‡
	DistilBERT-base-uncased	quora-distilbert-multilingual	(Örs et al., 2020)	0.8362
From-scratch	BERT-multilingual-cased	-	HAC	**0.8464**‡
	BERT-multilingual-cased	-	(Örs et al., 2020)	0.8414
	XLM-R-large	-	HAC	0.8360
	XLM-R-large	-	(Örs et al., 2020)	0.8350

Table 8: Results of ESCI for English using multilingual models. Best is in Bold and submitted systems are marked with ‡.

languages for further splits.

For the English language, we experimented with the clustering approaches using the embeddings generated by different STransformer models. Initially, we focused on pretrained models and their fine-tuned versions on task data. Later we built STransformers from scratch using general transformer models and further integrated LM too. The obtained results and corresponding model details are summarised in Table 7. According to the results, STransformers build from scratch outperformed the pretrained and fine-tuned models. LM did not improve the results and it is possible when data is not enough for modelling. Among the clustering algorithms, HAC showed the best results.

We could not train any STransformer for other languages because the organisers provided a limited number of labelled instances for those languages. We used pretrained multilingual models and adhering to zero-shot learning, fine-tuned them using English data. Further English data were used to build STransformers from scratch too. All the evaluations were also done on English data and best-performing systems were chosen to make predictions for other languages. The obtained results are summarised in Table 8. Similar to the English monolingual scenario, from-scratch multilingual models performed best.

We report the results for test data in Table 9. According to the results, for all languages, we could obtain competitive results compared to the results of the best-submitted system. Since our approach can be easily extended to different languages with very few training instances, we believe the results are at a satisfactory level.

7 Conclusions

In this paper, we presented our approach for document and cross-sentence level subtasks of CASE 2021 Task 1: Multilingual protest news detection. We mainly used pretrained transformer models including their improved architectures for long document processing and sentence embedding generation. Further, different learning strategies: monolingual, multilingual and zero-shot and, classification and clustering approaches were involved. For document level predictions, our approach achieved the 1^{st} place for the English language while being within the top 4 solutions for other languages. For cross-sentence level predictions, we secured the

	Model	Clustering	CoNLL Average Score
English	Best System		0.8444
	BERT-large-cased$_{from-scratch}$	HAC	0.8040
	BERT-large-cased$_{from-scratch}$	(Örs et al., 2020)	0.7951
Spanish	Best system		0.8423
	quora-distilbert-multilingual$_{fine-tune(en)}$	HAC	0.8183
	BERT-multilingual-cased$_{from-scratch(en)}$	HAC	0.8167
Portuguese	Best System		0.9303
	quora-distilbert-multilingual$_{fine-tune(en)}$	HAC	0.9023
	BERT-multilingual-cased$_{from-scratch(en)}$	HAC	0.9023

Table 9: ESCI results for test data

3^{rd} place for the English language with competitive results for other languages. Despite that, our approach can support multiple languages with low or no training resources.

As future work, we hope to further improve semantically meaningful sentence embedding generation using improved architectures, learning strategies and ensemble methods. Also, we would like to analyse the impact of different clustering approaches on cross-sentence level predictions.

References

Faizan Ahmad, Ahmed Abbasi, Brent Kitchens, Donald A Adjeroh, and Daniel Zeng. 2020. Deep learning for adverse event detection from web search. *IEEE Transactions on Knowledge and Data Engineering*.

Maria Antoniak and David Mimno. 2018. Evaluating the stability of embedding-based word similarities. *Transactions of the Association for Computational Linguistics*, 6:107–119.

Iz Beltagy, Matthew E Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.

José Canete, Gabriel Chaperon, Rodrigo Fuentes, and Jorge Pérez. 2020. Spanish pre-trained bert model and evaluation data. *PML4DC at ICLR*, 2020.

Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St John, Noah Constant, Mario Guajardo-Céspedes, Steve Yuan, Chris Tar, et al. 2018. Universal sentence encoder. *arXiv preprint arXiv:1803.11175*.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Alexis Conneau, Douwe Kiela, Holger Schwenk, Loïc Barrault, and Antoine Bordes. 2017. Supervised learning of universal sentence representations from natural language inference data. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 670–680, Copenhagen, Denmark. Association for Computational Linguistics.

Vera Danilova and Svetlana Popova. 2014. Sociopolitical event extraction using a rule-based approach. In *OTM Confederated International Conferences" On the Move to Meaningful Internet Systems"*, pages 537–546. Springer.

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.

Raia Hadsell, Sumit Chopra, and Yann LeCun. 2006. Dimensionality reduction by learning an invariant mapping. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pages 1735–1742. IEEE.

Hansi Hettiarachchi, Mariam Adedoyin-Olowe, Jagdev Bhogal, and Mohamed Medhat Gaber. 2021. Embed2detect: Temporally clustered embedded words for event detection in social media. *Machine Learning*, pages 1–39.

Hansi Hettiarachchi and Tharindu Ranasinghe. 2020. InfoMiner at WNUT-2020 task 2: Transformer-based covid-19 informative tweet extraction. In *Proceedings of the Sixth Workshop on Noisy User-generated Text (W-NUT 2020)*, pages 359–365, Online. Association for Computational Linguistics.

Hansi Hettiarachchi and Tharindu Ranasinghe. 2021. TransWiC at SemEval-2021 Task 2: Transformer-based Multilingual and Cross-lingual Word-in-Context Disambiguation. In *Proceedings of SemEval*.

Frederik Hogenboom, Flavius Frasincar, Uzay Kaymak, and Franciska De Jong. 2011. An overview of event extraction from text. In *DeRiVE@ ISWC*, pages 48–57. Citeseer.

Ali Hürriyetoğlu, Osman Mutlu, Farhana Ferdousi Liza, Erdem Yörük, 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)*, online. Association for Computational Linguistics (ACL).

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.

Fatemah Husain and Ozlem Uzuner. 2021. Leveraging offensive language for sarcasm and sentiment detection in arabic. In *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, pages 364–369.

Tommi Jauhiainen, Tharindu Ranasinghe, and Marcos Zampieri. 2021. Comparing approaches to dravidian language identification. In *Proceedings of the 7th Workshop on NLP for Similar Languages, Varieties and Dialects*.

Kathy Lee, Ashequl Qadir, Sadid A Hasan, Vivek Datla, Aaditya Prakash, Joey Liu, and Oladimeji Farri. 2017. Adverse drug event detection in tweets with semi-supervised convolutional neural networks. In *Proceedings of the 26th international conference on world wide web*, pages 705–714.

Christopher D Manning, Prabhakar Raghavan, and Hinrich Schütze. 2008. *Introduction to information retrieval*. Cambridge university press.

Chandler May, Alex Wang, Shikha Bordia, Samuel Bowman, and Rachel Rudinger. 2019. On measuring social biases in sentence encoders. 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 622–628.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Faik Kerem Örs, Süveyda Yeniterzi, and Reyyan Yeniterzi. 2020. Event clustering within news articles. In *Proceedings of the Workshop on Automated Extraction of Socio-political Events from News 2020*, pages 63–68.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Sameer Pradhan, Xiaoqiang Luo, Marta Recasens, Eduard Hovy, Vincent Ng, and Michael Strube. 2014. Scoring coreference partitions of predicted mentions: A reference implementation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 30–35, Baltimore, Maryland. Association for Computational Linguistics.

Yifan Qiao, Chenyan Xiong, Zhenghao Liu, and Zhiyuan Liu. 2019. Understanding the behaviors of bert in ranking. *arXiv preprint arXiv:1904.07531*.

Tharindu Ranasinghe, Constantin Orasan, and Ruslan Mitkov. 2019a. Semantic textual similarity with Siamese neural networks. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019)*, pages 1004–1011, Varna, Bulgaria. INCOMA Ltd.

Tharindu Ranasinghe, Constantin Orasan, and Ruslan Mitkov. 2020. TransQuest: Translation quality estimation with cross-lingual transformers. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 5070–5081, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Tharindu Ranasinghe and Marcos Zampieri. 2021. MUDES: Multilingual detection of offensive spans. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Demonstrations*, pages 144–152, Online. Association for Computational Linguistics.

Tharindu Ranasinghe, Marcos Zampieri, and Hansi Hettiarachchi. 2019b. BRUMS at HASOC 2019: Deep learning models for multilingual hate speech and offensive language identification. In *In Proceedings of the 11th annual meeting of the Forum for Information Retrieval Evaluation*.

Nils Reimers, Iryna Gurevych, Nils Reimers, Iryna Gurevych, Nandan Thakur, Nils Reimers, Johannes Daxenberger, 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.

Philip A Schrodt, John Beieler, and Muhammed Idris. 2014. Three'sa charm?: Open event data coding with el: Diablo, petrarch, and the open event data alliance. In *ISA Annual Convention*. Citeseer.

Cagil Sonmez, Arzucan Özgür, and Erdem Yörük. 2016. Towards building a political protest database to explain changes in the welfare state. In *Proceedings of the 10th SIGHUM Workshop on Language*

Technology for Cultural Heritage, Social Sciences, and Humanities, pages 106–110.

Fábio Souza, Rodrigo Nogueira, and Roberto Lotufo. 2020. BERTimbau: pretrained BERT models for Brazilian Portuguese. In *9th Brazilian Conference on Intelligent Systems, BRACIS, Rio Grande do Sul, Brazil, October 20-23 (to appear)*.

Chi Sun, Xipeng Qiu, Yige Xu, and Xuanjing Huang. 2019. How to fine-tune bert for text classification? In *Chinese Computational Linguistics*, pages 194–206, Cham. Springer International Publishing.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Wei Yang, Yuqing Xie, Aileen Lin, Xingyu Li, Luchen Tan, Kun Xiong, Ming Li, and Jimmy Lin. 2019. End-to-end open-domain question answering with BERTserini. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 72–77, Minneapolis, Minnesota. Association for Computational Linguistics.

Manzil Zaheer, Guru Guruganesh, Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, et al. 2020. Big bird: Transformers for longer sequences. *arXiv preprint arXiv:2007.14062*.