Multi-CoPED: A Multilingual Multi-Task Approach for Coding Political Event Data on Conflict and Mediation Domain

Erick Skorupa Parolin
erick.skorupaparolin@utdallas.edu
The University of Texas at Dallas
Richardson, Texas, USA

MohammadSaleh Hosseini
seyyedmohammadsaleh.hosseini@utdallas.edu
The University of Texas at Dallas
Richardson, Texas, USA

Yibo Hu
yibo.hu@utdallas.edu
The University of Texas at Dallas
Richardson, Texas, USA

Latifur Khan
lkhan@utdallas.edu
The University of Texas at Dallas
Richardson, Texas, USA

Patrick T. Brandt
pbrandt@utdallas.edu
The University of Texas at Dallas
Richardson, Texas, USA

Javier Osorio
josorio1@email.arizona.edu
The University of Arizona
Tucson, Arizona, USA

Vito D'Orazio
dorazio@utdallas.edu
The University of Texas at Dallas
Richardson, Texas, USA

ABSTRACT

Political and social scientists monitor, analyze and predict political unrest and violence, preventing (or mitigating) harm, and promoting the management of global conflict. They do so using *event coder systems*, which extract structured representations from news articles to design forecast models and event-driven continuous monitoring systems. Existing methods rely on expensive manual annotated dictionaries and do not support multilingual settings. To advance the global conflict management, we propose a novel model, **Multi-CoPED** (**Multi**lingual **Multi**-Task Learning BERT for **Co**ding **P**olitical **E**vent **D**ata), by exploiting multi-task learning and state-of-the-art language models for coding multilingual political events. This eliminates the need for expensive dictionaries by leveraging BERT models' contextual knowledge through transfer learning. The multilingual experiments demonstrate the superiority of Multi-CoPED over existing event coders, improving the absolute macro-averaged F1-scores by 23.3% and 30.7% for coding events in English and Spanish corpus, respectively. We believe that such expressive performance improvements can help to reduce harms to people at risk of violence.

CCS CONCEPTS

• **Applied computing** → **Computing in government**; Military; • **Computing methodologies** → **Information extraction**; *Multi-task learning*.

AIES'22, August 1–3, 2022, Oxford, United Kingdom

ACM ISBN 978-1-4503-9247-1/22/08…$15.00
https://doi.org/10.1145/3514094.3534178

KEYWORDS

artificial intelligence and geopolitics, political conflict, social conflict, event coding, natural language processing, transfer learning

ACM Reference Format:
Erick Skorupa Parolin, MohammadSaleh Hosseini, Yibo Hu, Latifur Khan, Patrick T. Brandt, Javier Osorio, and Vito D'Orazio. 2022. Multi-CoPED: A Multilingual Multi-Task Approach for Coding Political Event Data on Conflict and Mediation Domain. In *Proceedings of the 2022 AAAI/ACM Conference on AI, Ethics, and Society (AIES'22), August 1–3, 2022, Oxford, United Kingdom.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3514094.3534178

1 INTRODUCTION

Extracting political events from news articles is a crucial task in political science. Traditionally, conflict scholars rely on such data to analyze interactions among political entities across the globe and forecast events of political instability, such as civil conflicts and violence (e.g., rebellions, insurgencies, ethno-religious violence), domestic and international political conflicts, military operations, political cooperation and diplomatic affairs. Government agencies in the security sector and policy makers can use these predictions and findings to aid humanitarian and political crises.

Computerized event-coding systems are key components in the global violence management process, generating political event data to support modeling and analytics. State-of-the-art systems like PETRARCH [7], PETRARCH2 [47] and Universal PETRARCH [43] are utilized to identify, extract, and categorize conflict interactions from unstructured text and convert them to the form of a *who-did-what-to-whom* template.

Despite the large number of applications and research based upon *automated coders*, the technical methods employed have remained unchanged for more than two decades. These coding systems implement a pattern-matching approach, using external knowledge bases (KBs) to identify the presence of certain lexico-syntactic patterns in a sentence, indicating a particular semantic relationship between political entities. However, the complexity of an unstructured text generally exceeds the capacity of these dictionary-based

coders, often producing low performance results (low recall and accuracy). Updating and expanding these handcrafted pattern repositories is often too costly and time-consuming, rendering them obsolete in the context of rapidly changing conflict processes. Lastly, despite the efforts of coding event data in non-English languages [50–52], to the best of our knowledge, the systems and ontologies used in political science do not support coding event data on multilingual corpora, imposing severe limitations and bias on conflict analysis covering distinct areas in the globe.

Given the challenges and peculiarities inherent in coding political event data, traditional information extraction (IE) [15, 19–21, 63] and NLP approaches, such as semantic role labeling (SRL) [29, 72, 76, 95] cannot address the task (as discussed in Section 3). This prevents the usage of off-the-shelf approaches to generate political event data.

Recent advances in natural language processing (NLP) techniques open new possibilities to solve some of the core challenges of traditional event-coding approaches. In particular, transformer-based pre-trained language models [81], such as BERT [16], introduced a new approach to obtain state-of-the-art results on a wide range of NLP tasks [33, 35, 61, 83].

Our new automated event-coding system combines transfer learning and multi-task learning techniques to code event data from domain-specific multilingual corpora in a *who-did-what-to-whom* template. We explore transfer learning by leveraging multilingual pre-trained language models, providing promising results even when the training dataset is small.

We demonstrate the superiority of our approach through conducting extensive experiments on a corpus based on **CAMEO** (**C**onflict **a**nd **M**ediation **E**vent **O**ntology) [25]. Apart from the performance superiority, our proposed models favor a more comprehensive analysis in global level (through multilingual processing) and move beyond the biases introduced from existing computerized event-coding systems (see Subsection 2.4).

This paper makes multiple contributions bridging AI and geopolitics, and supporting advances in conflict analysis. First, an innovative model MTL-BERT based on multi-task learning for coding multilingual political event data is presented. Second, we design a novel approach **Multi-CoPED** (**Multi**lingual **Multi**-Task network for **Co**ding **P**olitical **E**vent **D**ata) to address the challenges of *event coding* (discussed in Section 3) by integrating CAMEO, MTL-BERT and our event parsing procedures (discussed in Subsection 4.2). Finally, we conduct multilingual experiments to compare the empirical results of current coding systems with ours.

2 PRELIMINARIES

2.1 Related Work

Conflict scholars often are interested in extracting events from text in a structured format to track conflict processes and violence using computational methods. Most previous works for coding event data are based on pattern matching approaches [47, 51, 52, 57], usually supported by large KBs or domain-specific ontologies.

Other studies have concentrated on political conflict event detection, relying on classical machine learning and deep learning (DL) techniques. Hanna [28] proposed a support vector machine (SVM)-based framework for coding protest events. Beieler [6] exploited convolutional neural networks (CNNs) to solely detect political event types in sentences, while Radford [59] trained a recurrent neural network (RNN) to identify indicators of protest events. O'Connor et al. [53] proposed an unsupervised model for extracting events occurred among major political actors from news corpus. Glavaš et al. [26] have applied semantic text representations and induced a joint multilingual semantic vector space to enable supervised learning (SVM and CNN) for topical coding of sentences from electoral manifestos of political parties in different languages (English, French, German and Italian). Osorio et al. [52] introduced a logistic regression-based framework to detect news documents related to conflict and use external dictionaries to extract events from text in Arabic.

Recent works [13, 32, 48, 55] utilize BERT, ELMo, and DistilBERT to extract representations from political documents that are later used as input features for traditional machine learning classifiers. In particular Parolin et al. [56] design a multi-label BERT-based network to extract events about organized crime. In political science, other work [49, 60] employs deep neural networks based on transformers for distinct tasks like events clustering and co-referencing.

In a broader view, coding political event data resembles information extraction (IE) related tasks in natural language processing area. Previous works employing deep neural network show promising results on event extraction by exploring RNNs [46, 93], graph neural networks [42, 45], and hybrid neural networks [31, 39, 92]. Most recently, transformer-based models have been proposed for event extraction [18, 37, 38, 40, 44, 82, 90], semantic role labeling (SRL) [54, 72], named entity recognition (NER) [16], and relationship extraction (RE) [4, 62, 84, 85, 87, 94]. Although these works have advanced research in standard IE tasks, none of these address all the key challenges of coding political event data (discussed in Section 3).

2.2 Ontologies and Current Approaches for Coding Political Event Data

A dominant ontology for political event data is CAMEO, which incorporates a KB of actor dictionaries (containing some 67K entries) and action-pattern dictionaries (about 14K verb phrases). The former acts as a data repository for political entities, such as country actors, international actors, military non-state actors, and general political agents, while the latter is used to store representations of political interactions. Overall, CAMEO covers more than 200 event types, each associated with a list of verbal pattern entries in the action repository. Despite the high granularity of event types offered by CAMEO verb dictionaries, conflict scholars traditionally use a higher level of event types, grouping the original types into five classes (otherwise known as **pentacodes**): *Make a Statement (0), Verbal Cooperation (1), Material Cooperation (2), Verbal Conflict (3), and Material Conflict (4).*

CAMEO is a static ontology where the knowledge rests. **Automated coders** are systems which are connected with CAMEO for processing input sentences and running inferences over such KBs. Current systems syntactically explore input sentences, trying to find matches of verbal patterns and actors in CAMEO repositories.

The **PETRARCH** family implements automated coder systems for CAMEO-based political event extraction. The main difference

between the coders in this family is how they syntactically process sentences to search for patterns. Whereas PETRARCH and PETRARCH2 extensively leverage the constituency parse tree of the input sentence, UPETRARCH implements dependency parse tree to determine the who-did-what-to-whom event codes and supports processing sentences in English and Spanish languages.

Suppose we have the following sentence as input to the event coder:

```
Obama said he would not provide support to Israel.
```

The event coder should output the following codes:

```
WHO: USAGOV      TO-WHOM: ISR
DID-WHAT: 3- VERBAL CONFLICT
```

Coding systems process data in sentence level and usually take the news publish date as additional input for retrieving the correct political role from repositories. In the example shown above, if the news publish date is Feb 2013, the system outputs *USAGOV* once it identifies *Obama* as the US President on that date. On the other hand, if the date is May 2021, the system outputs *USAELI*, recognizing *Obama* as part of US elite (former government officials, celebrities, etc). CAMEO maps 26 distinct political roles, which can be linked to an entity given a date interval.

2.3 Applications of Political Event Data

In practical applications, the structured event data provided by the automated coders (e.g., PETRARCH) serve as input for conflict and mediation studies. Previous work focused on designing forecasting models and early warning systems to predict inter- and intra-state political conflicts in many regions across the globe, such as Asian countries [66], the Cross-Straits [9, 10], the Balkans [65], between Israel and Palestine [68, 70], Southern Lebanon [64], the Middle East [67], and other regions [11, 12]. Other research and applications utilize such structured event data to analyze theories about the international mediation process [25, 69], monitor civil conflicts [3, 71], and even studying the effects and consequences of domestic conflicts [91].

2.4 Ethical Considerations on Political Event Data Research

Event data generation is a crucial process for understanding and managing inter- and intra-state conflict and violence. Scientists rely on such structured data to monitor, explain and forecast events of political instability, while policy makers leverage such findings on decision-making process for social and political crisis management.

However, studying political violence is a potentially controversial and sensitive topic. Language models such as those used here have been shown to encode biases from the training data (as discussed in previous work [1, 17, 24, 27, 58, 79, 80]). We follow standard social science practices to select corpora and training data to mitigate these issues [5]. Further we work to move beyond the biases introduced from dictionary-based methods (e.g., PETRARCH coders) by using deep learning and statistical methods, in line with the literature [86]. This is also part of the de-biasing of using multiple coders for the training data. Stundal et al. [78] show that using the machine coded event data seen here closely matches that coded by human rights experts in highly contested regions like Colombia.

Finally, there are application specific sensitivities to studying things like international relations and even political violence. That is why we bring together scholars and researchers from these domains and the NLP and computer science communities here to best inform how to derive and use this information from large corpora, recognizing that this is no substitute for deep ethnographic and field research.

3 CODING POLITICAL EVENT DATA AND CHALLENGES

The problem addressed in this paper is commonly known in social science studies as *event coding*, and it aims at extracting events in *who-did-what-to-whom* (otherwise known as *source-action-target*) template from sentence-level texts in domain-specific multilingual corpus. Following, we elaborate on the main challenges inherent to event coding task.

Multi-Source/Target Events: Conforming to political science applications, we assume that there is only one event type per sentence, yet *multiple events of the same type* can occur in the same sentence. To put the issue into perspective, considering the sentence:

```
Obama and Putin said they will attack Iraq.
```

we should have the following events as output:

```
WHO: USAGOV     TO-WHOM: IRQ
DID-WHAT: 0- MAKE A STATEMENT

WHO: RUSGOV     TO-WHOM: IRQ
DID-WHAT: 0- MAKE A STATEMENT
```

Note that in this example, there exist two coded events of the same type (*Make a Statement*), both of which have the same target (*Iraq*) but two different sources (*USA and Russian government entities*).

Reciprocal Events: Another type of sentences that generates multiple coded events as output are the ones involving *reciprocal* relations. For instance, in the sentence

```
French National Assembly president Laurent
Fabius held talks with leaders of Romania's
new government.
```

there will be two events of type *Verbal Cooperation*: the first will code French National Assembly president (FRAGOV) as source and Romania's new government (ROUGOV) as target; while the second will code the same event type in opposite direction (ROUGOV as source and FRAGOV as target).

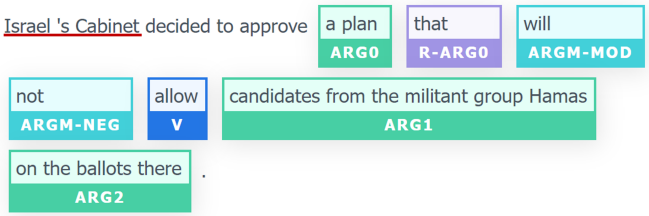
Multilingual: A crucial aspect that differentiates the problem introduced in this paper is the *multilingual* factor. Semantically equivalent sentences written in different languages should produce the same output codes. Considering our previous *"Obama and Putin"* example, we should obtain the same codes given the equivalent sentences in any language, such as Spanish or Portuguese (respectively exemplified below):

```
Obama y Putin dijeron que atacarán a Irak.
Obama e Putin disseram que vão atacar o Iraque.
```

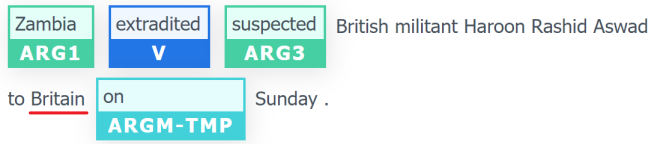
Traditional IE models do not suffice: Although our problem is restricted to labeling only two roles (source and target), there are certain challenges that differentiate coding event data from similar

traditional IE tasks, such as SRL (the closest to our problem definition). We examine these challenges and illustrate some examples in Fig. 1 to support our discussion.

First, *source*'s and *target*'s roles are not semantically fixed and may vary depending on the event type. Therefore, sources and targets are not necessarily associated with *agent* and *theme* in the context of semantic relations, as they can be represented by people, organizations, locations, nationalities, and religious or political groups depending on the event type.



(a) Deep Learning-based model for SRL captures the target *"the militant group Hamas"* but does not capture the source *"Israel's Cabinet"* as an argument.



(b) SRL model captures the source *"Zambia"* but does not capture target *"Britain"*.

Figure 1: Examples where SOTA SRL model [72] is not capable of correctly detecting sources and targets.

Second, sources and targets are not necessarily arguments which are associated with the same verb eventually triggering an event. For example, in the sentence expressed in Fig. 1a, traditional SRL approaches would retrieve the ground truth target "*the group Hamas*" as one of the arguments for the verb "*allow*" but would not retrieve the ground truth source "*Israel's Cabinet*". Eventually, there will be cases where sources and targets are not even part of any argument in any triggering verb (e.g., ground truth target "*Britain*" in Fig. 1b).

Finally, SRL models traditionally output arguments for each verb in an input sentence. Deciding which are the verbs to be triggered, analyzed and associated to each pentacode requires an extra complex task on the top of the SRL model.

Although the challenges aforementioned provide theoretical arguments making the usage of SRL (and general IE application) impractical for our problem, we empirically demonstrate it in Section 5.4 by taking a state-of-the-art SRL [72] implementation as one of our baselines.

4 MODEL DESCRIPTION

In this section we introduce our approach for coding conflict and mediation event data from multilingual corpus, addressing the challenges introduced in Section 3. Subsection 4.1 introduces the design for our Multi-Task Learning BERT model (MTL-BERT), which works as the basis of our framework; while Subsection 4.2 details our Multilingual Multi-Task network for Coding Political Event Data (Multi-CoPED), as our end-to-end CAMEO-based event coder.

4.1 Multi-Task Learning BERT (MTL-BERT)

In order to address *event coding* problem, we design our model to handle two tasks: (1) find the spans of texts that denote sources and targets; (2) detect the conflict action (pentacode) expressed in the sentences.

We formulate the first task as sequence labeling, in which each word in the sentence is assigned to a tag showing the type of that word for our purposes. We consider the following tags: **S** (Source), **T** (Target), **R** (Reciprocal), and **O** (Other). A word with a certain tag shows that it has occurred in the text span with the corresponding type. A word with tag R denotes that it is in a span that is both a source and a target (this can happen in reciprocal relations as discussed in Section 3).

The second task is formulated as classification, which detects the pentacode corresponding to the main conflict action expressed in the sentence, with five possible labels each showing the corresponding pentacode: *0 (Make a Statement), 1 (Verbal Cooperation), 2 (Material Cooperation), 3 (Verbal Conflict), and 4 (Material Conflict).*

Our model, **M**ulti-**T**ask **L**earning **BERT** (**MTL-BERT**), consists of three components (as depicted in Fig. 2a) introduced in the following paragraphs.

(i) Contextualized Word Embeddings Extractor using BERT: In this component, we utilize BERT [16] to effectively capture the contextualized semantic features of input sentences and their words. Given an input sentence S composed of a sequence of words $(s_1, s_2, \ldots, s_{|S|})$, we first tokenize it to N tokens using WordPiece tokenizer [88] and add BERT's special tokens [CLS] and [SEP] to the beginning and end of the said tokens. We then feed these $M = N+2$ tokens $(t_0, t_1, \ldots, t_{N+1})$ into BERT, which maps the embedding vectors of the mentioned tokens $(E_{[CLS]}, E_1, \ldots, E_N, E_{[SEP]})$ to their corresponding contextualized embedding vectors $T = (T_{[CLS]}, T_1, \ldots, T_N, T_{[SEP]})$.

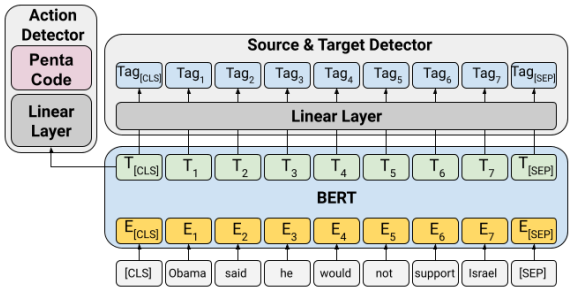
We feed the above-mentioned contextual representations $T \in \mathbb{R}^{M \times h}$ to both Source and Target Detector and Action Detector heads. Note that h denotes the dimension of the contextualized word embeddings.

(ii) Source and Target Detector: This module deals with the sequence labeling task and is essentially composed of M identical linear layers (i.e., with tied parameters), each getting its input from one of the contextualized word embeddings. The core linear layer is represented by the parameter matrix $W_{tag} \in \mathbb{R}^{|L| \times h}$, where $L = \{S, T, R, O, [CLS], [SEP]\}$ is the set of possible tags. Thus, for each token t_i, Source and Target Detector returns a tag $Tag_i \in L$ as follows:

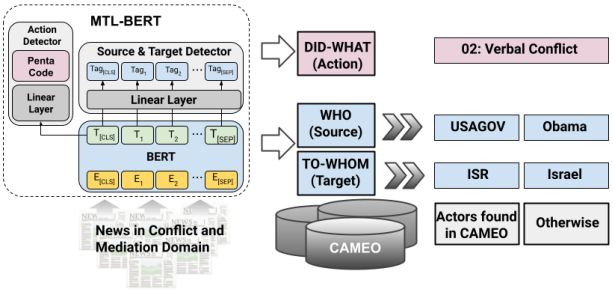
$$Tag_i = argmax\ softmax(W_{tag} \cdot T_i) \quad (1)$$

where argmax outputs the tag in L corresponding to the maximum element's index in softmax's output.

(iii) Action Detector: This component attends to the classification task and consists of one linear layer expressed by the matrix $W_{penta} \in \mathbb{R}^{|C| \times h}$, where $C = \{0, 1, 2, 3, 4\}$ corresponds to the set of pentacodes. As a result, for each input sentence S, Action Detector



(a) MTL-BERT implements two heads over the same BERT network: One dedicated to action detection (classification) and another to actors detection (seq. labeling).



(b) Multi-CoPED extends MTL-BERT by integrating it with CAMEO actor repositories to work as an end-to-end CAMEO-based event coder.

Figure 2: Design of the proposed models MTL-BERT and Multi-CoPED.

returns a pentacode c_S according to the following equation:

$$c_S = argmax\ softmax(W_{penta} \cdot T_{[CLS]}) \quad (2)$$

where $T_{[CLS]}$ carries the semantic representation of sentence S, and argmax functions similar to its counterpart in Eq. 1.

For training our model in a multi-task fashion, we use AdaMax [34] algorithm and update the model weights once per task in each epoch.

Finally, to address the multilingual aspect of our event coding problem, we utilize BERT multilingual pre-trained model as initial weights of MTL-BERT.

4.2 Multilingual Multi-Task BERT for Coding Political Event Data (Multi-CoPED)

In this subsection, we introduce Multi-CoPED as our end-to-end CAMEO-based event coder, shown in Fig. 2b. The design of our framework rests on extending MTL-BERT by connecting it with CAMEO's actor dictionaries through the procedure described in Algorithm 1.

Algorithm 1: The Multi-CoPED Framework

input : Sentence **sent**, publish date **pub_date**, MTL-BERT model **bert**
output: List of CAMEO-coded events in triplet format **triplets**

1 $tags, pentaCode \leftarrow bert(sent)$
2 $spans \leftarrow tokens2spans(tags)$
3 $S_{spans}, T_{spans}, R_{spans} \leftarrow adjustTags(spans)$
4 **if** $not\ empty(R_{spans})$ **then**
5 $\quad R_{codes} \leftarrow retrieve(R_{spans}, pub\_date)$
6 $\quad$ **return** $perm(R_{codes}, pentaCode)$ `// see (iv)`
7 **else**
8 $\quad S_{codes} \leftarrow retrieve(S_{spans}, pub\_date)$
9 $\quad T_{codes} \leftarrow retrieve(T_{spans}, pub\_date)$
10 $\quad$ **return** $cross(S_{codes}, T_{codes}, pentaCode)$ `// see (iv)`

Note that Multi-CoPED **does not** utilize CAMEO's action repositories: it simply relies on MTL-BERT to detect actions (pentacodes), instead of resorting to the lexico-syntactic patterns from CAMEO (like PETRARCH coders do). As a result, Multi-CoPED improves the performance on event coding task, reduces (or eliminates) the costs associated to repository maintenance/extension and allows language-agnostic application.

The four components described below summarize the steps performed along Multi-CoPED.

(i) Parsing Tags: We convert the tagged tokens output by MTL-BERT to text spans tagged with the same actor labels (S, R, or T). In this step (run in Line 2), we keep the sequence of contiguous tokens with the same label, say S, to compose a span assigned with the said label S. Note that, in the same sentence, we can have multiple spans of the same type.

(ii) Adjusting Tags: The existence of label R introduces the possibility of obtaining inconsistent set of actors as output from step (i). In this step (run in Line 3), we adjust such inconsistent cases by applying a simple heuristic, expressed in Table 1.

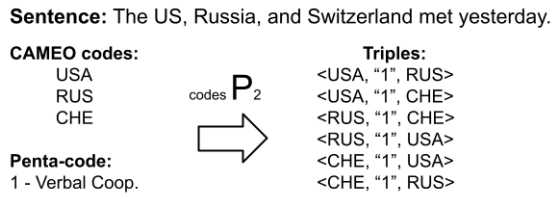
Table 1: Heuristic rules and examples of adjusted tags.

Inconsistent Tagging (Example)	Heusitic Rules	Adjusted Tags
[Syria]$_{[R]}$ says it won't accept any more refugees.	R → S	[Syria]$_{[S]}$
[Obama]$_{[R]}$, [Puting]$_{[R]}$ and [Leuthard]$_{[T]}$ met this afternoon.	S*, T+, RR+ → RR+	[Obama]$_{[R]}$, [Puting]$_{[R]}$, [Leuthard]$_{[R]}$
[Obama]$_{[S]}$, [Puting]$_{[R]}$ and [Leuthard]$_{[R]}$ met this afternoon.	S+, T*, RR+ → RR+	
Russia$_{[R]}$ closed its southern borders with Iran$_{[T]}$ and Turkey$_{[T]}$	T+, R → S, T+	Russia$_{[S]}$, Iran$_{[T]}$, Turkey$_{[T]}$
[Obama]$_{[S]}$ and [Puting]$_{[S]}$ said they will attack [Iraq]$_{[R]}$.	S+, R → S+, T	[Obama]$_{[S]}$, [Puting]$_{[S]}$, [Iraq]$_{[T]}$

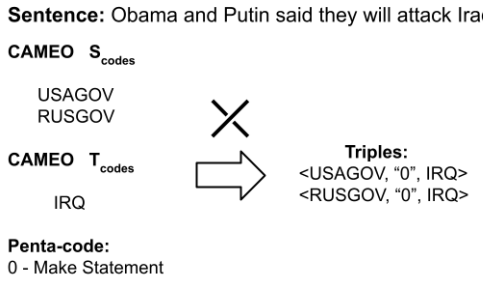
For instance, reciprocal events require at least two entities. In the case there is only one span labeled R (first example of Table 1), we convert it to S. As another example, if there exist two or more reciprocal spans (RR^+), zero or more source spans (S^*) and at least one target span (T^+), then we disregard the sources and targets, keeping only the reciprocal spans (second example of Table 1).

(iii) Retrieving CAMEO Codes: For each labeled span, we query CAMEO repositories to retrieve the codes corresponding to the actors expressed in that span. This search is executed through querying in the following order: domestic actors, international actors, military non-state actors and agents repositories. This step is run in Lines 5, 8, and 9. Moreover, the sentence (news) publish date must also be given as input to the function *retrieve* so that it returns the code corresponding to the political role of an actor on that date. In the case no actor is found in CAMEO repositories, *retrieve* returns the text span given as input, guaranteeing no events will be lost.

(iv) Generating Structured Events (SAT Triplets): In this step (run in Lines 6 and 10), we combine the retrieved codes from actors with the provided pentacode in such a way that outputs CAMEO-based triplets expressing the events in *who-did-what-to-whom* layout. In practice, at this step, we may have three cases of actors relationships: usual *source-target*, *multi source/target* or *reciprocal*. As shown in Fig. 3, depending on the relationship type, we combine the actors in a different way to output the correct event triplets. For *reciprocal* cases we permute the actors (through function *perm* in Algorithm 1); otherwise, we apply cross-product over source and target actors (through function *cross*).



(a) Generating reciprocal triplets: we permute the actors into all possible source-target pairs.



(b) Generating multi source/target triplets (also applied for usual source-target): we apply cross product over source and target actors.

Figure 3: Generating structured events in Source-Action-Target (SAT) triplets both for source-target and reciprocal cases.

5 EXPERIMENTS AND RESULTS

In this section, we describe the datasets, the baselines and the computational setup used in our experiments (subsections 5.1, 5.2 and 5.3 respectively). Lastly, in Subsection 5.4, we present the performance for the models proposed in Section 4.

5.1 Dataset

Due to lack of annotated datasets in conflict and mediation domain, we created a brand-new dataset for our application purpose. The test data is made up of the gold standard records originally used for validating PERTARCH2 and UPETRARCH[1], plus the annotated sentences available in CAMEO codebook [2], totaling 451 error-free annotated sentences in English language. The test dataset necessary for performing the multilingual experiments were obtained by manually translating the 451 English (EN) sentences into Portuguese (PT) and Spanish (ES) to obtain three parallel testing corpus (D_{EN}^{test}, D_{ES}^{test}, D_{PT}^{test}). Note that, translations were performed by native speakers of the target languages, in order to guarantee syntactic and semantic correctness on testing samples.

For obtaining the training (and validation) dataset, we collected 3,728 sentences (2,207 in English and 1,521 Spanish) by crawling newswire text data from various world-wide news agencies, and carefully pre-processing and filtering out out-of-domain news based on the metadata information. The annotations for pentacodes, sources, and targets were independently performed by six undergraduate students trained for such task and revised by a committee consisting of conflict specialists. Overall, we obtained good standards in terms of inter-annotation agreement (Fleiss' Kappa = 60.03% for pentacodes).

The distribution for pentacodes on training dataset is balanced, with 16.44% of the sentences labeled with the less frequent pentacode (Material Conflict) and 29.08% with the most frequent one (Verbal Conflict). On the other hand, the token labels distribution is quite unbalanced: only 1.98% of tokens are labeled as *reciprocal*, while 70.46% are labeled as *others*.

5.2 Baseline Models

As our main baselines, we adopted the current state-of-the-art frameworks for CAMEO-coded political event extraction. As previously introduced in Subsection 2.2, the PETRARCH family implements three distinct versions of coding mechanisms: **PERTARCH**, **PERTARCH2** and **UPETRARCH**. Although UPETRARCH is the only one which supports event coding in English and Spanish, we took all of them as reference for the experiments.

In addition, we implemented an **LSTM-based** model for sequence labeling tasks [30, 76]. Our model is a two-step pipeline: (1) a sequence classifier which implements a bidirectional LSTM (BiLSTM) layer and two dense layers trained to detect the pentacodes, followed by (2) a sequence tagger based on BiLSTM and Conditional Random Field (CRF). We concatenate the embedding of predicted pentacode as an additional feature with word embedding to feed the BiLSTM layer in step (2), following [29, 95]. Finally, we use Viterbi algorithm to obtain the most likely tag sequence. To support multilingual experiments on LSTM models, we used 300-dimensional English, Spanish, and Portuguese word embeddings pretrained on Wikipedia from fastText [8]. We followed [74] to align monolingual vectors from these three languages in a single vector space.

[1] https://github.com/openeventdata/petrarch2/blob/master/petrarch2/data/text/GigaWord.sample.PETR.xml
[2] https://parusanalytics.com/eventdata/data.dir/cameo.html

Finally, we experimented a state-of-the-art BERT-based model for SRL [72] implemented in AllenNLP toolkit [23]. Since SRL outputs only spans of texts as its arguments, it can be experimented merely for source and target detection, making its application impractical for pentacodes detection and end-to-end event coder purposes. For each sentence, SRL outputs multiple tuples of semantic roles (arguments). Out of all the arguments, most of the time, ARG0 and ARG1 correspond to the source and target, respectively. Therefore, we extract sources and targets from the outputs of SRL in two ways in which we consider ARG0 as the source and ARG1 as the target. In the first version, called **SRL-based**, we choose the first tuple that has both ARG0 and ARG1. In the second version, called **SRL-based-UB**, we choose the tuple having ARG0 and ARG1 that maximize Macro-F1 when calculated against the gold sources and targets. Note that SRL-based-UB works only as an upper-bound baseline since it is not implementable in practice. It simply shows what would be the best possible performance we could reach by using SRL model.

5.3 Setup

To conduct the experiments presented in this paper, we used a computer with one NVIDIA GeForce RTX 2080 GPU. We run 10 rounds of training process for each experimented model and report the averaged results observed on testing set. In each round, we generate different train/validation splits (75%/25% over the annotated data) and randomly initialize the model based on the seed assigned for that round. We train our models over 30 epochs and the best model of each round is selected based on F1-scores observed on their corresponding validation splits. We use the same random seeds for all models to make the comparison fair. Specifically for PETRARCH and SRL-based baselines, we run them over the test set only once (they do not require training process).

For MTL-BERT, we leverage multi-task learning implementations for transformers-based networks from previous works [41, 77] and perform the necessary adaptations to attend our design. We fine-tune MTL-BERT using both cased[3] and uncased[4] multilingual pre-trained versions. Since both models presented similar performance, we suppress the results for uncased model in Subsection 5.4, reserving space for this analysis in Appendix A.1.

5.4 Experiments on Conflict and Mediation Dataset

We design cross-language experiments to analyze the performance of models on multilingual application. In each evaluation, we train MTL-BERT and LSTM-based models on the same monolingual (EN/ES) or multilingual (EN+ES) dataset, and then evaluate them on each parallel testing datasets (D_{EN}^{test} , D_{ES}^{test}, D_{PT}^{test}). Once PETRARCH, PETRARCH2 and SRL models do not support multilingual application, we restrain our analysis on English language only.

Following, we analyze the performance of the models on Source and Target Detection (text spans format) and Action Detection (pentaclass format) tasks separately. For that purpose, we apply MTL-BERT and evaluate its performance versus the baselines. Then, we close our experiments by analyzing the performance of Multi-CoPED as an end-to-end CAMEO-based event coder.

Source and Target Detector: Given a sentence as the input, for the pattern matching approaches (PETRARCH, PETRARCH2 and UPETRARCH), we mark as Source (S) and Target (T) the *spans of text* returned by PETRARCH coders as source and target, respectively. For LSTM-based and MTL-BERT, we follow the same procedure explained in step (i) in Subsection 4.2 to obtain S, T, and R spans. For this evaluation, R is regarded as both an S and a T. Table 2 shows the F1-scores (in exact-match and partial-match manners) not only on S and T spans detection, but also overall (Macro-F1). MTL-BERT trained both on English and Spanish corpus consistently outperforms the other models, except for Source detection on Spanish testing corpus (for exact-match), where MTL-BERT trained on Spanish presented the best results.

Action Detector: Table 3 shows the F1-scores for action detection task. MTL-BERT models significantly outperform all the other models for all the pentacodes individually and collectively (Macro-F1), in all languages. In particular, MTL-BERT$_{(EN+ES)}$ improves the existing best coders PETRARCH (in English) and UPETRARCH (in Spanish), by absolute Macro-F1 increases of 25.9% and 31.9%, respectively.

End-to-end Event Coder: Finally, to evaluate Multi-CoPED as an end-to-end CAMEO-based event coder, we analyze the performance of this model in event level through example-based metrics [75].

Although we implemented the example-based measures to evaluate the end-to-end models in an overall manner, we too show the results broken by source, target, and action separately in Table 4. The results indicate that Multi-CoPED outperforms all the baselines, including the current systems for CAMEO-coded political event extraction both in English and Spanish languages.

Specifically, Multi-CoPED trained on English and Spanish corpus (Multi-CoPED$_{(EN+ES)}$) significantly outperforms the best results observed for the existing event coder systems (i.e. PETRARCH2 and UPETRARCH), improving the absolute overall F1-score by 23.3% and 30.7% on English and Spanish languages, respectively.

Overall Discussion and Findings: The empirical results discussed along this section show indications to support the following findings. First, the performance superiority of both MTL-BERT and Multi-CoPED against the baselines is statistically significant (at 0.001 level based on t-test) in all evaluated languages when looking at the Macro-F1 scores.

Second, our proposed models address the low-recall weakness typically associated to pattern-matching approaches (e.g., PETRARCH family). Recall figures observed under Overall column in Table 4 illustrate this effect as well as Tables 6 and 7 in Appendix A.2.

Third, our models keep high performance standards on Target detection, which is the most challenging task we experiment (due to the reasons discussed in Section 3).

Fourth, as expected, languages with larger lexical similarity (e.g., Spanish and Portuguese) show better results in cross-language experiments. In all experiments in this section, we see the models trained in Spanish performing better than those trained in English when looking at the performance over Portuguese testing set D_{PT}^{test}.

[3]https://huggingface.co/bert-base-multilingual-cased
[4]https://huggingface.co/bert-base-multilingual-uncased

Table 2: Experimental results for source and target detector. We compute F1-scores (both in Exact-Match and Partial-Match fashion) to evaluate performance of MTL-BERT vs. the baselines on testing set per language. Each column identified with a language code shows the models' performance on testing set in that particular language (D_{EN}^{test}, D_{ES}^{test} and D_{PT}^{test}). Bold values represent the highest scores among all the models per language per argument (source and target).

	F1-score **Exact-Match** on testing set by language									F1-score **Partial-Match** on testing set by language								
	Source			**Target**			**Macro-F1**			**Source**			**Target**			**Macro-F1**		
	EN	**ES**	**PT**	**EN**	**ES**	**PT**	**EN**	**ES**	**PT**	**EN**	**ES**	**PT**	**EN**	**ES**	**PT**	**EN**	**ES**	**PT**
PETRARCH [7]	45.6	-	-	32.6	-	-	39.1	-	-	50.2	-	-	41.4	-	-	45.8	-	-
PETRARCH2 [47]	34.1	-	-	22.6	-	-	28.3	-	-	36.5	-	-	30.3	-	-	33.4	-	-
UPETRARCH [43]	38.3	36.8	-	14.1	11.4	-	26.2	23.4	-	44.0	42.9	-	25.4	22.6	-	34.7	32.5	-
SRL-based [72]	73.4	-	-	12.7	-	-	43.0	-	-	83.4	-	-	36.1	-	-	59.7	-	-
SRL-based (UB)	77.4	-	-	20.0	-	-	48.7	-	-	82.8	-	-	39.9	-	-	61.4	-	-
LSTM-based$_{(EN)}$	85.7	80.0	81.8	64.4	43.2	34.9	75.0	61.6	58.3	89.1	85.8	86.6	75.6	59.2	55.9	82.3	72.5	71.3
LSTM-based$_{(ES)}$	68.6	85.5	84.7	35.7	61.2	44.8	52.1	73.4	64.7	80.0	88.7	88.4	55.8	73.2	67.1	67.9	81.0	77.7
LSTM-based$_{(EN+ES)}$	86.4	87.1	87.8	65.6	61.9	49.8	76.0	74.5	68.8	88.7	89.9	90.3	75.9	74.2	69.0	82.3	82.0	79.6
MTL-BERT$_{(EN)}$	89.6	87.4	87.1	74.6	65.9	62.7	82.1	76.7	74.9	93.1	91.4	92.0	83.3	78.8	79.2	88.2	85.1	85.6
MTL-BERT$_{(ES)}$	86.8	**90.9**	90.4	65.8	70.2	64.3	76.3	80.5	77.3	91.6	**93.4**	93.9	78.4	81.0	79.5	85.0	87.2	86.7
MTL-BERT$_{(EN+ES)}$	**90.3**	90.4	**90.6**	**75.8**	**71.7**	**66.6**	**83.0**	**81.1**	**78.6**	**93.6**	**93.4**	**94.1**	**83.8**	**82.1**	**81.4**	**88.7**	**87.7**	**87.8**

Table 3: Experimental results for pentacodes classification (action detector). Each column identified with a language code shows the F1-score for the models on testing set in that particular language (D_{EN}^{test}, D_{ES}^{test} and D_{PT}^{test}). Bold values represent the highest scores among all the models per language per pentacode.

	F1-score on testing set by language																	
	Make a Statement			**Verbal Cooperation**			**Material Cooperation**			**Verbal Conflict**			**Material Conflict**			**Macro F1**		
	EN	**ES**	**PT**	**EN**	**ES**	**PT**	**EN**	**ES**	**PT**	**EN**	**ES**	**PT**	**EN**	**ES**	**PT**	**EN**	**ES**	**PT**
PETRARCH [7]	60.7	-	-	59.0	-	-	43.0	-	-	56.9	-	-	68.5	-	-	57.6	-	-
PETRARCH2 [47]	53.6	-	-	58.3	-	-	54.2	-	-	54.1	-	-	67.4	-	-	57.5	-	-
UPETRARCH [43]	53.5	41.5	-	56.2	49.3	-	52.2	41.4	-	59.1	52.8	-	66.0	60.6	-	57.4	49.0	-
LSTM-based$_{(EN)}$	64.5	43.7	47.8	72.7	62.9	59.3	66.7	46.3	50.8	77.5	67.8	64.1	83.0	76.2	76.1	72.9	59.3	59.6
LSTM-based$_{(ES)}$	44.5	61.7	55.4	62.4	69.2	54.1	44.2	62.3	49.4	70.1	73.5	64.7	76.0	81.0	78.8	59.4	69.4	60.5
LSTM-based$_{(EN+ES)}$	64.5	65.1	59.1	74.9	71.8	63.8	66.8	65.7	60.6	79.7	78.5	71.1	84.0	83.0	81.1	74.1	72.8	67.1
MTL-BERT$_{(EN)}$	81.1	49.3	46.3	81.6	63.4	62.3	77.1	47.6	46.7	86.6	70.1	66.8	88.6	76.5	76.3	83.0	61.3	59.6
MTL-BERT$_{(ES)}$	50.1	75.6	66.1	63.0	79.1	72.8	52.2	71.5	55.1	73.4	80.4	74.6	77.9	84.1	81.6	63.2	78.1	70.0
MTL-BERT$_{(EN+ES)}$	**82.1**	**78.6**	**70.6**	**82.0**	**80.5**	**74.4**	**77.4**	**75.8**	**60.9**	**87.0**	**83.0**	**77.2**	**89.3**	**86.5**	**84.2**	**83.5**	**80.9**	**73.5**

Finally, training the models in multilingual corpus (EN+ES) consistently produces better results than training in monolingual dataset (ES or ES separately) on both D_{EN}^{test} and D_{ES}^{test}, and even on a third language (D_{PT}^{test}), which is not part of the training corpus. Such finding is consistent to previous studies in cross-lingual representations [2, 14, 22, 36, 73, 74, 89].

Qualitative Analysis: The superiority of MTL-BERT and Multi-CoPED goes beyond the figures denoted in the empirical experiments shown here. Apart from the advantage of supporting multilingual coding, the models proposed in this paper do not rely on extensive annotated action dictionaries including verbal phrase patterns, which are too expensive to update and maintain. Instead, our models explore not only the syntactic but also the semantic aspects of input text, by leveraging the default BERT pre-trained models through transfer learning (requiring a small annotated corpus to do a successful job). Further, as depicted in Fig. 2b, Multi-CoPED works as a hybrid model, which is able to output both the CAMEO codes and the spans of text for sources and targets. Such property can be potentially explored for retrieving new political actors and extending the CAMEO repositories.

6 CONCLUSIONS AND FUTURE WORK

Political event data generation is a crucial process for understanding and managing the global violence. Political scientists and government agencies in the security sector typically rely on computerized systems to gather and analyze event data on conflict processes and violence around the world. However, the existing event coder systems present major limitations in terms of cost, performance, and multilingual parsing. These shortcomings prevent the generation of accurate event data, causing significant impacts on forecasting and monitoring political conflict and violence events.

To overcome these key challenges and help advance the global violence management, we propose an innovative technique by combining state-of-the-art NLP models with multi-task learning approach to efficiently extract events in structured CAMEO-like format. Our proposed model, MTL-BERT, requires a small number of labeled data to provide high quality results for source, target, and action detection. We extend MTL-BERT by integrating it with CAMEO actor repositories through our novel procedure, called Multi-CoPED, which employs MTL-BERT as its main engine to compose an end-to-end CAMEO-based event coder.

The experiments on our multilingual corpus indicate that MTL-BERT outperforms all the baselines in actors and action detection,

Table 4: Results based on example-based metrics for Multi-CoPED and baseline models working as end-to-end CAMEO-based event coders. Table on the top shows performance over the English testing set (D_{EN}^{test}) while the bottom one shows the figures for the models over Spanish testing set (D_{ES}^{test}). We do not evaluate the results on (D_{PT}^{test}) since UPETRARCH does not support Portuguese language. Bold values represent the highest scores among all the models per component (Source, Target and Action).

Testing in English (D_{EN}^{test})	Source			Target			Action			Overall		
	Prec	Rec	F1	Prec	Rec	F1	Prec	Rec	F1	Prec	Rec	F1
PETRARCH [7]	62.1	46.2	53.0	51.0	38.0	43.6	62.3	46.4	53.2	34.3	22.9	27.5
PETRARCH2 [47]	53.2	51.7	52.4	45.9	44.6	45.3	46.5	45.2	45.8	50.4	34.9	41.2
UPETRARCH [43]	41.5	47.5	44.3	37.2	42.6	39.7	43.5	49.8	46.4	31.8	26.3	28.8
LSTM-based$_{(EN)}$	81.4	80.4	80.8	76.9	71.2	73.8	74.8	64.5	69.3	57.1	51.9	54.3
LSTM-based$_{(ES)}$	77.9	69.0	73.1	72.6	47.0	56.8	63.2	54.5	58.5	44.0	26.7	33.1
LSTM-based$_{(EN+ES)}$	83.3	80.4	81.8	79.0	72.2	75.4	76.2	65.7	70.6	59.7	53.2	56.2
Multi-CoPED$_{(EN)}$	90.4	84.6	87.4	79.8	81.3	80.5	84.1	72.6	77.9	59.8	65.7	62.6
Multi-CoPED$_{(ES)}$	88.3	82.4	85.2	72.5	75.4	73.9	66.6	57.4	61.7	41.0	48.1	44.3
Multi-CoPED$_{(EN+ES)}$	**91.1**	**84.7**	**87.8**	**81.4**	**82.7**	**82.0**	**84.6**	**73.0**	**78.4**	**62.2**	**67.1**	**64.5**

Testing in Spanish (D_{ES}^{test})	Source			Target			Action			Overall		
	Prec	Rec	F1	Prec	Rec	F1	Prec	Rec	F1	Prec	Rec	F1
UPETRARCH [43]	39.2	47.0	42.8	35.3	42.7	38.7	42.6	48.8	45.5	29.3	26.4	27.7
LSTM-based$_{(EN)}$	70.9	75.9	72.3	62.6	50.2	54.7	62.9	54.2	58.2	38.6	30.7	33.5
LSTM-based$_{(ES)}$	80.2	78.6	79.4	72.0	66.4	69.0	71.4	61.5	66.1	48.3	44.7	46.4
LSTM-based$_{(EN+ES)}$	78.8	79.1	78.8	70.8	67.9	69.2	74.8	64.5	69.3	50.4	48.7	49.4
Multi-CoPED$_{(EN)}$	83.1	81.1	82.1	73.6	77.1	75.3	64.5	55.6	59.7	38.4	47.2	42.3
Multi-CoPED$_{(ES)}$	**85.1**	**82.1**	**83.6**	74.1	79.2	76.5	79.1	68.2	73.2	52.0	60.8	56.0
Multi-CoPED$_{(EN+ES)}$	84.2	**82.1**	83.1	**75.7**	**80.2**	**77.9**	**81.8**	**70.5**	**75.8**	**54.6**	**62.9**	**58.4**

in all languages. Additionally, Multi-CoPED consistently shows the best results on generating CAMEO-coded political event data.

An open discussion for future work is to analyze how the performance of Multi-CoPED model will behave in multilingual corpus containing more languages with lower lexical similarity. Moreover, we intend to expand the case study to other micro domains in the political science sphere (e.g., terrorism, organized crime, insurgencies, protest movements, and multinational military exercises).

ACKNOWLEDGMENTS

The research reported herein was supported in part by NSF awards DMS-1737978, DGE-2039542, OAC-1828467, OAC-1931541, and DGE-1906630, ONR awards N00014-17-1-2995 and N00014-20-1-2738, Army Research Office Contract No. W911NF2110032 and IBM faculty award (Research).

REFERENCES

[1] Abubakar Abid, Maheen Farooqi, and James Zou. 2021. Persistent anti-muslim bias in large language models. In *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*. 298–306.

[2] Waleed Ammar, George Mulcaire, Yulia Tsvetkov, Guillaume Lample, Chris Dyer, and Noah A Smith. 2016. Massively multilingual word embeddings. *arXiv preprint arXiv:1602.01925* (2016).

[3] Benjamin E Bagozzi. 2015. Forecasting civil conflict with zero-inflated count models. *Civil Wars* 17, 1 (2015), 1–24.

[4] Livio Baldini Soares, Nicholas FitzGerald, Jeffrey Ling, and Tom Kwiatkowski. 2019. Matching the Blanks: Distributional Similarity for Relation Learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Florence, Italy, 2895–2905. https://doi.org/10.18653/v1/P19-1279

[5] Pablo Barberá, Amber E Boydstun, Suzanna Linn, Ryan McMahon, and Jonathan Nagler. 2021. Automated text classification of news articles: A practical guide. *Political Analysis* 29, 1 (2021), 19–42.

[6] John Beieler. 2016. Generating politically-relevant event data. *In Proceedings of the First Workshop on NLP and Computational Social Science* (2016), 37–42.

[7] John Beieler and Clayton Norris. 2014. Petrarch: Python Engine for Text Resolution And Related Coding Hierarchy. Available at https://github.com/openeventdata/petrarch (2020/05/15). Unpublished Manuscript.

[8] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching Word Vectors with Subword Information. *Transactions of the Association for Computational Linguistics* 5 (2017), 135–146.

[9] Patrick T Brandt, John R Freeman, Tse-min Lin, and Phillip A Schrodt. 2013. Forecasting conflict in the cross-straits: long term and short term predictions. In *Annual Meeting of the American Political Science Association*.

[10] Patrick T Brandt, John R Freeman, and Philip A Schrodt. 2011. Racing horses: constructing and evaluating forecasts in political science. In *28th summer meeting of the society for political methodology*. 39.

[11] Patrick T Brandt, John R Freeman, and Philip A Schrodt. 2011. Real time, time series forecasting of inter-and intra-state political conflict. *Conflict Management and Peace Science* 28, 1 (2011), 41–64.

[12] Patrick T Brandt, John R Freeman, and Philip A Schrodt. 2014. Evaluating forecasts of political conflict dynamics. *International Journal of Forecasting* 30, 4 (2014), 944–962.

[13] Berfu Büyüköz, Ali Hürriyetoğlu, and Arzucan Özgür. 2020. Analyzing ELMo and DistilBERT on Socio-political News Classification. In *Proceedings of the Workshop on Automated Extraction of Socio-political Events from News 2020*. European Language Resources Association (ELRA), Marseille, France, 9–18. https://www.aclweb.org/anthology/2020.aespen-1.4

[14] Alexis Conneau, Guillaume Lample, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2017. Word translation without parallel data. *arXiv preprint arXiv:1710.04087* (2017).

[15] Lei Cui, Furu Wei, and Ming Zhou. 2018. Neural open information extraction. *arXiv preprint arXiv:1805.04270* (2018).

[16] 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)*. Association for Computational Linguistics, Minneapolis, Minnesota, 4171–4186. https://doi.org/10.18653/v1/N19-1423

[17] Lucas Dixon, John Li, Jeffrey Sorensen, Nithum Thain, and Lucy Vasserman. 2018. Measuring and mitigating unintended bias in text classification. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*. 67–73.

[18] Xinya Du and Claire Cardie. 2020. Event Extraction by Answering (Almost) Natural Questions. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, Online, 671–683. https://doi.org/10.18653/v1/2020.emnlp-main.49

[19] Oren Etzioni, Anthony Fader, Janara Christensen, Stephen Soderland, et al. 2011. Open information extraction: The second generation. In *Twenty-Second International Joint Conference on Artificial Intelligence*.

[20] Anthony Fader, Stephen Soderland, and Oren Etzioni. 2011. Identifying relations for open information extraction. In *Proceedings of the 2011 conference on empirical methods in natural language processing*. 1535–1545.

[21] James Fan, David Ferrucci, David Gondek, and Aditya Kalyanpur. 2010. Prismatic: Inducing knowledge from a large scale lexicalized relation resource. In *Proceedings of the NAACL HLT 2010 first international workshop on formalisms and methodology for learning by reading*. 122–127.

[22] Manaal Faruqui and Chris Dyer. 2014. Improving vector space word representations using multilingual correlation. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*. 462–471.

[23] Matt Gardner, Joel Grus, Mark Neumann, Oyvind Tafjord, Pradeep Dasigi, Nelson F. Liu, Matthew Peters, Michael Schmitz, and Luke S. Zettlemoyer. 2017. AllenNLP: A Deep Semantic Natural Language Processing Platform. arXiv:arXiv:1803.07640

[24] Sahaj Garg, Vincent Perot, Nicole Limtiaco, Ankur Taly, Ed H Chi, and Alex Beutel. 2019. Counterfactual fairness in text classification through robustness. In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*. 219–226.

[25] Deborah J Gerner, Philip A Schrodt, Omür Yilmaz, and Rajaa Abu-Jabr. 2002. Conflict and mediation event observations (CAMEO): A new event data framework for the analysis of foreign policy interactions. *International Studies Association, New Orleans* (2002).

[26] Goran Glavaš, Federico Nanni, and Simone Paolo Ponzetto. 2017. Cross-Lingual Classification of Topics in Political Texts. In *Proceedings of the Second Workshop on NLP and Computational Social Science*. Association for Computational Linguistics, Vancouver, Canada, 42–46. https://doi.org/10.18653/v1/W17-2906

[27] Wei Guo and Aylin Caliskan. 2021. Detecting emergent intersectional biases: Contextualized word embeddings contain a distribution of human-like biases. In *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*. 122–133.

[28] Alex Hanna. 2017. Mpeds: Automating the generation of protest event data. Available at https://osf.io/preprints/socarxiv/xuqmv (2020/05/22). Unpublished Manuscript.

[29] Luheng He, Kenton Lee, Mike Lewis, and Luke Zettlemoyer. 2017. Deep semantic role labeling: What works and what's next. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 473–483.

[30] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9, 8 (1997), 1735–1780.

[31] Yu Hong, Wenxuan Zhou, Jingli Zhang, Guodong Zhou, and Qiaoming Zhu. 2018. Self-regulation: Employing a generative adversarial network to improve event detection. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 515–526.

[32] Yibo Hu, Mohammad Saleh 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*.

[33] Yibo Hu and Latifur Khan. 2021. Uncertainty-Aware Reliable Text Classification. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. 628–636.

[34] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[35] Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. 2017. Race: Large-scale reading comprehension dataset from examinations. *arXiv preprint arXiv:1704.04683* (2017).

[36] Guillaume Lample and Alexis Conneau. 2019. Cross-lingual language model pretraining. *arXiv preprint arXiv:1901.07291* (2019).

[37] Fayuan Li, Weihua Peng, Yuguang Chen, Quan Wang, Lu Pan, Yajuan Lyu, and Yong Zhu. 2020. Event Extraction as Multi-turn Question Answering. In *Findings of the Association for Computational Linguistics: EMNLP 2020*. Association for Computational Linguistics, Online, 829–838.

[38] Ying Lin, Heng Ji, Fei Huang, and Lingfei Wu. 2020. A Joint Neural Model for Information Extraction with Global Features. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 7999–8009.

[39] Jian Liu, Yubo Chen, and Kang Liu. 2019. Exploiting the ground-truth: An adversarial imitation based knowledge distillation approach for event detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 6754–6761.

[40] Jian Liu, Yubo Chen, Kang Liu, Wei Bi, and Xiaojiang Liu. 2020. Event extraction as machine reading comprehension. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 1641–1651.

[41] Xiaodong Liu, Pengcheng He, Weizhu Chen, and Jianfeng Gao. 2019. Multi-Task Deep Neural Networks for Natural Language Understanding. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 4487–4496. https://www.aclweb.org/anthology/P19-1441

[42] Xiao Liu, Zhunchen Luo, and Heyan Huang. 2018. Jointly Multiple Events Extraction via Attention-based Graph Information Aggregation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Brussels, Belgium, 1247–1256. https://doi.org/10.18653/v1/D18-1156

[43] J. Lu and Joydeep Roy. 2017. Universal Petrarch: Language-agnostic political event coding using universal dependencies. Available at https://github.com/openeventdata/UniversalPetrarch (2020/05/22).

[44] Qing Lyu, Hongming Zhang, Elior Sulem, and Dan Roth. 2021. Zero-shot event extraction via transfer learning: Challenges and insights. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*. 322–332.

[45] Thien Nguyen and Ralph Grishman. 2018. Graph convolutional networks with argument-aware pooling for event detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 32.

[46] Trung Minh Nguyen and Thien Huu Nguyen. 2019. One for all: Neural joint modeling of entities and events. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 6851–6858.

[47] Clayton Norris, Philip Schrodt, and John Beieler. 2017. PETRARCH2: Another Event Coding Program. *Journal of Open Source Software* 2, 9 (2017), 133. https://doi.org/10.21105/joss.00133

[48] Fredrik Olsson, Magnus Sahlgren, Fehmi ben Abdesslem, Ariel Ekgren, and Kristine Eck. 2020. Text Categorization for Conflict Event Annotation. In *Proceedings of the Workshop on Automated Extraction of Socio-political Events from News 2020*. European Language Resources Association (ELRA), Marseille, France, 19–25. https://www.aclweb.org/anthology/2020.aespen-1.5

[49] 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*. European Language Resources Association (ELRA), Marseille, France, 63–68. https://www.aclweb.org/anthology/2020.aespen-1.11

[50] Javier Osorio, Viveca Pavon, Sayeed Salam, Jennifer Holmes, Patrick T. Brandt, and Latifur Khan. 2019. Translating CAMEO verbs for automated coding of event data. *International Interactions* 45, 6 (2019), 1049–1064.

[51] Javier Osorio and Alejandro Reyes. 2017. Supervised Event Coding From Text Written in Spanish: Introducing Eventus ID. *Social Science Computer Review* 35, 3 (2017), 406–416. http://ssc.sagepub.com/content/early/2016/01/07/0894439315625475.abstract

[52] Javier Osorio, Alejandro Reyes, Alejandro Beltrán, and Atal Ahmadzai. 2020. Supervised Event Coding from Text Written in Arabic: Introducing Hadath. In *Proceedings of the Workshop on Automated Extraction of Socio-political Events from News 2020*. European Language Resources Association (ELRA), Marseille, France, 49–56. https://www.aclweb.org/anthology/2020.aespen-1.9

[53] Brendan O'Connor, Brandon M Stewart, and Noah A Smith. 2013. Learning to extract international relations from political context. *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics* 1 (2013), 1094–1104.

[54] Giovanni Paolini, Ben Athiwaratkun, Jason Krone, Jie Ma, Alessandro Achille, Rishita Anubhai, Cicero Nogueira dos Santos, Bing Xiang, and Stefano Soatto. 2021. Structured Prediction as Translation between Augmented Natural Languages. In *9th International Conference on Learning Representations, ICLR 2021*.

[55] Erick Skorupa Parolin, Yibo Hu, Latifur Khan, Javier Osorio, Patrick T Brandt, and Vito D'Orazio. 2021. CoMe-KE: A New Transformers Based Approach for Knowledge Extraction in Conflict and Mediation Domain. In *2021 IEEE International Conference on Big Data (Big Data)*. IEEE, 1449–1459.

[56] Erick Skorupa Parolin, Latifur Khan, Javier Osorio, Patrick Brandt, Vito D'Orazio, and Jennifer Holmes. 2021. 3M-Transformers for Event Coding on Organized Crime Domain. In *2021 IEEE International Conference on Data Science and Advanced Analytics (DSAA)*. IEEE, 1–10.

[57] Erick Skorupa Parolin, Latifur Khan, Javier Osorio, Vito D'Orazio, Patrick T Brandt, and Jennifer Holmes. 2020. HANKE: Hierarchical Attention Networks for Knowledge Extraction in political science domain. In *2020 IEEE 7th International Conference on Data Science and Advanced Analytics (DSAA)*. IEEE, 410–419.

[58] Desmond U Patton, William R Frey, Kyle A McGregor, Fei-Tzin Lee, Kathleen McKeown, and Emanuel Moss. 2020. Contextual analysis of social media: The promise and challenge of eliciting context in social media posts with natural language processing. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*. 337–342.

[59] Benjamin Radford. 2019. Multitask Models for Supervised Protest Detection in Texts. Available at https://arxiv.org/abs/2005.02954 (2020/05/22). Unpublished Manuscript.

[60] Benjamin Radford. 2020. Seeing the Forest and the Trees: Detection and Cross-Document Coreference Resolution of Militarized Interstate Disputes. In *Proceedings of the Workshop on Automated Extraction of Socio-political Events from News 2020*. European Language Resources Association (ELRA), Marseille, France, 35–41. https://www.aclweb.org/anthology/2020.aespen-1.7

[61] Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for SQuAD. *arXiv preprint arXiv:1806.03822* (2018).

[62] Oscar Sainz, Oier Lopez de Lacalle, Gorka Labaka, Ander Barrena, and Eneko Agirre. 2021. Label Verbalization and Entailment for Effective Zero-and Few-Shot Relation Extraction. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

[63] Michael Schmitz, Stephen Soderland, Robert Bart, Oren Etzioni, et al. 2012. Open language learning for information extraction. In *Proceedings of the 2012 joint conference on empirical methods in natural language processing and computational natural language learning*. 523–534.

[64] Philip A Schrodt. 1997. Early warning of conflict in southern lebanon using hidden markov models. In *American Political Science Association*.

[65] Philip A Schrodt. 2006. Forecasting conflict in the Balkans using hidden Markov models. In *Programming for peace*. Springer, 161–184.

[66] Philip A Schrodt. 2011. Forecasting political conflict in Asia using Latent Dirichlet Allocation models. In *Annual meeting of the European political science association, Dublin*.

[67] Philip A Schrodt and Deborah J Gerner. 1996. *Using cluster analysis to derive early warning indicators for political change in the Middle East, 1979-1996*. University of Kansas.

[68] Philip A Schrodt, Deborah J Gerner, and Omur Yilmaz. 2004. Using event data to monitor contemporary conflict in the israel-palestine dyad. *International Studies Association, Montreal, Quebec, Canada* (2004), 1–31.

[69] Philip A Schrodt, Ömür Yilmaz, and Deborah J Gerner. 2003. Evaluating "Ripeness" and "Hurting Stalemate" in Mediated International Conflicts: An Event Data Study of the Middle East, Balkans, and West Africa. In *Annual Meeting of the International Studies Association, Portland, OR, February (eventdata. parusanalytics. com/papers. dir/Schrodt. etal. ISA03. pdf)*.

[70] Robert Shearer. 2007. Forecasting Israeli-Palestinian conflict with hidden Markov models. *Military Operations Research* (2007), 5–15.

[71] Stephen M Shellman and Brandon M Stewart. 2007. Predicting risk factors associated with forced migration: An early warning model of Haitian flight. *Civil Wars* 9, 2 (2007), 174–199.

[72] Peng Shi and Jimmy Lin. 2019. Simple bert models for relation extraction and semantic role labeling. *arXiv preprint arXiv:1904.05255* (2019).

[73] Jasdeep Singh, Bryan McCann, Nitish Shirish Keskar, Caiming Xiong, and Richard Socher. 2019. Xlda: Cross-lingual data augmentation for natural language inference and question answering. *arXiv preprint arXiv:1905.11471* (2019).

[74] Samuel L Smith, David HP Turban, Steven Hamblin, and Nils Y Hammerla. 2017. Offline bilingual word vectors, orthogonal transformations and the inverted softmax. *arXiv preprint arXiv:1702.03859* (2017).

[75] Mohammad S Sorower. 2010. A literature survey on algorithms for multi-label learning. *Oregon State University, Corvallis* 18 (2010), 1–25.

[76] Gabriel Stanovsky, Julian Michael, Luke Zettlemoyer, and Ido Dagan. 2018. Supervised open information extraction. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. 885–895.

[77] Asa Cooper Stickland and Iain Murray. 2019. BERT and PALs: Projected Attention Layers for Efficient Adaptation in Multi-Task Learning. In *Proceedings of the 36th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 97)*. PMLR, 5986–5995.

[78] Logan Stundal, Benjamin E Bagozzi, John R Freeman, and Jennifer S Holmes. 2021. Human Rights Violations in Space: Assessing the External Validity of Machine-Geocoded versus Human-Geocoded Data. *Political Analysis* (2021), 1–17.

[79] Nathaniel Swinger, Maria De-Arteaga, Neil Thomas Heffernan IV, Mark DM Leiserson, and Adam Tauman Kalai. 2019. What are the biases in my word embedding?. In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*. 305–311.

[80] Aaron D Tucker, Markus Anderljung, and Allan Dafoe. 2020. Social and governance implications of improved data efficiency. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*. 378–384.

[81] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Neural Information Processing Systems (NIPS)* (2017), 5998–6008.

[82] David Wadden, Ulme Wennberg, Yi Luan, and Hannaneh Hajishirzi. 2019. Entity, relation, and event extraction with contextualized span representations. *arXiv preprint arXiv:1909.03546* (2019).

[83] Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. *arXiv preprint arXiv:1804.07461* (2018).

[84] Yucheng Wang, Bowen Yu, Yueyang Zhang, Tingwen Liu, Hongsong Zhu, and Limin Sun. 2020. TPLinker: Single-stage Joint Extraction of Entities and Relations Through Token Pair Linking. In *Proceedings of the 28th International Conference on Computational Linguistics*. International Committee on Computational Linguistics, Barcelona, Spain (Online), 1572–1582. https://www.aclweb.org/anthology/2020.coling-main.138

[85] Zhepei Wei, Jianlin Su, Yue Wang, Yuan Tian, and Yi Chang. 2020. A Novel Cascade Binary Tagging Framework for Relational Triple Extraction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. 1476–1488.

[86] John Wilkerson and Andreu Casas. 2017. Large-scale computerized text analysis in political science: Opportunities and challenges. *Annual Review of Political Science* 20 (2017), 529–544.

[87] Shanchan Wu and Yifan He. 2019. Enriching pre-trained language model with entity information for relation classification. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*. 2361–2364.

[88] Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144* (2016).

[89] Chao Xing, Dong Wang, Chao Liu, and Yiye Lin. 2015. Normalized word embedding and orthogonal transform for bilingual word translation. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 1006–1011.

[90] 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*. 5284–5294.

[91] James E. Yonamine. 2011. The Effects of Domestic Conflict on Interstate Conflict: An Event Data Analysis of Monthly Level Onset and Intensity. *Unpublished MA Thesis, Lockheed Martin* (2011), 1–2.

[92] Tongtao Zhang, Heng Ji, and Avirup Sil. 2019. Joint entity and event extraction with generative adversarial imitation learning. *Data Intell* 1, 2 (2019), 99–120.

[93] Yunyan Zhang, Guangluan Xu, Yang Wang, Xiao Liang, Lei Wang, and Tinglei Huang. 2019. Empower event detection with bi-directional neural language model. *Knowledge-Based Systems* 167 (2019), 87–97.

[94] Zexuan Zhong and Danqi Chen. 2021. A Frustratingly Easy Approach for Entity and Relation Extraction. In *North American Association for Computational Linguistics (NAACL)*.

[95] Jie Zhou and Wei Xu. 2015. End-to-end learning of semantic role labeling using recurrent neural networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. 1127–1137.

A APPENDIX

A.1 Cased vs. Uncased

Table 5 shows the performance of MTL-BERT considering multilingual pre-trained models both cased and uncased as initial weights on training (fine-tuning) process.

Table 5: Results for MTL-BERT utilizing cased and uncased multilingual BERT pre-trained models. Bold values represent the highest scores among all the models per language.

Source and Target Detection	S & T Detection			Action Detection		
	EN	ES	PT	EN	ES	PT
MTL-BERT$_{cased(EN)}$	82.1	76.7	74.9	83.0	61.3	59.6
MTL-BERT$_{cased(ES)}$	76.3	80.5	77.3	63.2	78.1	70.0
MTL-BERT$_{cased(EN+ES)}$	**83.0**	81.1	**78.6**	**83.5**	80.9	**73.5**
MTL-BERT$_{uncased(EN)}$	82.2	76.2	75.6	82.2	63.8	60.0
MTL-BERT$_{uncased(ES)}$	74.6	79.3	76.6	63.6	78.6	67.7
MTL-BERT$_{uncased(EN+ES)}$	82.9	**81.4**	78.5	83.3	**81.3**	**73.5**

A.2 Overall Performance (Recall and Precision)

Tables 6 and 7 show the performance for MTL-BERT and baselines for Source and Target Detector and Action Detector, respectively. We show precision and recall measures in order to complement the discussion over Tables 2 and 3 in Section 5.4, facilitating a comprehensive analysis.

Table 6: Evaluating precision and recall computed in exact-match fashion for MTL-BERT and baselines on Source and Target detection. Each column shows the performance over different testing set (D_{EN}^{test}, D_{ES}^{test} and D_{PT}^{test}). Bold values represent the highest scores among all the models per argument (Source and Target) per metric.

MODELS	D_{EN}^{test}				D_{ES}^{test}				D_{PT}^{test}			
	Source		Target		Source		Target		Source		Target	
	Prec	Rec	Prec	Rec	Prec	Rec	Prec	Rec	Prec	Rec	Prec	Rec
PETRARCH	55.1	38.9	42.2	26.6	-	-	-	-	-	-	-	-
PETRARCH2	52.5	25.2	33.8	17.0	-	-	-	-	-	-	-	-
UPETRARCH	49.8	31.1	22.5	10.3	47.2	30.3	18.9	8.1	-	-	-	-
SRL-based	77.5	69.7	13.0	12.4	-	-	-	-	-	-	-	-
SRL-based (UB)	88.6	68.7	22.4	18.1	-	-	-	-	-	-	-	-
LSTM-based$_{(EN)}$	87.6	83.9	66.1	62.8	81.8	78.3	51.8	37.1	85.1	78.8	44.1	28.9
LSTM-based$_{(ES)}$	75.7	62.8	44.0	30.1	86.9	84.2	63.9	58.8	87.9	81.6	51.8	39.6
LSTM-based$_{(EN+ES)}$	88.6	84.2	67.4	63.9	88.2	86.1	64.0	60.0	90.1	85.7	55.7	45.1
MTL-BERT$_{(EN)}$	90.5	88.6	73.6	75.7	86.4	88.5	63.9	68.1	85.9	88.3	60.6	65.0
MTL-BERT$_{(ES)}$	88.8	84.9	65.1	66.5	**91.7**	**90.0**	68.4	72.1	91.2	89.6	63.0	65.7
MTL-BERT$_{(EN+ES)}$	**91.3**	**89.3**	**74.8**	**76.8**	90.9	89.9	**70.3**	**73.2**	**91.4**	**89.8**	**65.7**	**67.6**

Table 7: Evaluating precision and recall for MTL-BERT and baseline models on action detection. Each split on the table shows the performance over different testing set (D_{EN}^{test}, D_{ES}^{test} and D_{PT}^{test}). Bold values represent the highest scores among all the models per pentacode per metric.

Testing in English (D_{EN}^{test})	Make a Statement		Verbal Cooperation		Material Cooperation		Verbal Conflict		Material Conflict	
	Prec	Rec	Prec	Rec	Prec	Rec	Prec	Rec	Prec	Rec
PETRARCH	60.7	60.7	69.5	51.3	55.6	35.1	80.5	44.0	81.0	59.3
PETRARCH2	63.4	46.4	78.7	46.3	66.7	45.6	86.8	39.3	79.8	58.3
UPETRARCH	**76.7**	41.1	58.9	53.8	51.7	52.6	80.5	46.7	74.4	59.3
LSTM-based$_{(EN)}$	60.4	69.9	77.5	68.9	64.2	69.9	78.0	77.1	85.4	81.0
LSTM-based$_{(ES)}$	44.9	45.9	72.3	55.1	39.6	51.7	71.2	69.5	77.1	75.3
LSTM-based$_{(EN+ES)}$	60.4	69.9	78.4	72.3	62.7	71.7	81.0	78.9	87.2	81.2
MTL-BERT$_{(EN)}$	75.5	87.6	80.1	**83.1**	**74.6**	80.0	89.6	**83.8**	91.6	85.6
MTL-BERT$_{(ES)}$	49.9	50.9	**83.2**	51.0	43.9	65.8	70.8	76.0	82.9	73.6
MTL-BERT$_{(EN+ES)}$	75.9	**89.4**	81.3	82.7	73.6	**82.0**	**90.5**	83.7	**92.6**	**86.0**
Spanish (D_{ES}^{test})	**Prec**	**Rec**	**Prec**	**Rec**	**Prec**	**Rec**	**Prec**	**Rec**	**Prec**	**Rec**
UPETRARCH	58.9	32.5	57.9	43.3	39.8	43.9	70.9	42.5	68.3	54.6
LSTM-based$_{(EN)}$	41.0	47.0	76.8	54.4	43.6	52.5	69.7	66.7	75.3	77.6
LSTM-based$_{(ES)}$	54.4	71.2	77.2	63.0	59.5	66.5	77.5	70.0	80.1	82.2
LSTM-based$_{(EN+ES)}$	57.0	75.6	81.3	64.3	61.0	71.6	82.4	75.5	83.1	82.9
MTL-BERT$_{((EN)}$	43.6	58.2	74.4	55.8	45.9	50.8	70.7	69.9	78.9	74.3
MTL-BERT$_{(ES)}$	68.7	84.0	**82.7**	76.0	63.3	81.9	85.9	75.6	86.4	81.8
MTL-BERT$_{(EN+ES)}$	**73.3**	**85.2**	81.8	**79.2**	**70.3**	**82.6**	**86.3**	**79.9**	**89.0**	**84.1**
Portuguese (D_{PT}^{test})	**Prec**	**Rec**	**Prec**	**Rec**	**Prec**	**Rec**	**Prec**	**Rec**	**Prec**	**Rec**
LSTM-based$_{(EN)}$	39.5	61.5	79.1	48.3	45.3	60.5	69.8	59.8	77.5	74.7
LSTM-based$_{(ES)}$	47.0	68.5	71.6	44.0	41.7	62.0	71.5	59.9	79.9	78.1
LSTM-based$_{(EN+ES)}$	49.3	73.7	77.7	54.3	**52.6**	71.8	78.8	65.2	81.5	**81.0**
MTL-BERT$_{(EN)}$	41.2	53.5	71.9	55.9	44.8	50.1	66.9	67.2	80.5	72.3
MTL-BERT$_{(ES)}$	56.6	79.8	80.4	66.7	47.0	67.2	79.7	70.3	89.6	75.1
MTL-BERT$_{(EN+ES)}$	**62.1**	**81.8**	**81.0**	**68.9**	52.5	**72.9**	**81.3**	**73.6**	**91.0**	78.4