3M-Transformers for Event Coding on Organized Crime Domain

Erick Skorupa Parolin*, Latifur Khan*, Javier Osorio‡, Patrick T. Brandt†, Vito D'Orazio†, Jennifer Holmes†
Department of Computer Science, School of Economic, Political and Policy Sciences†*
The University of Texas at Dallas, Richardson, Texas
School of Government and Public Policy‡, *University of Arizona*, Tucson, Arizona
{erick.skorupaparolin, lkhan, pbrandt, dorazio, jholmes}@utdallas.edu, josorio1@arizona.edu

2021 IEEE 8th International Conference on Data Science and Advanced Analytics (DSAA) | 978-1-6654-2099-0/21/$31.00 ©2021 IEEE | DOI: 10.1109/DSAA53316.2021.9564232

***Abstract*—Political scientists and security agencies increasingly rely on computerized event data generation to track conflict processes and violence around the world. However, most of these approaches rely on pattern-matching techniques constrained by large dictionaries that are too costly to develop, update, or expand to emerging domains or additional languages. In this paper, we provide an effective solution to those challenges. Here we develop the 3M-Transformers (Multilingual, Multi-label, Multi-task) approach for Event Coding from domain specific multilingual corpora, dispensing external large repositories for such task, and expanding the substantive focus of analysis to organized crime, an emerging concern for security research. Our results indicate that our 3M-Transformers configurations outperform state-of-the-art usual Transformers models (BERT and XLM-RoBERTa) for coding events on actors, actions and locations in English, Spanish, and Portuguese languages.**

***Index Terms*—event coding, transfer learning, natural language processing, organized crime, deep neural networks, multi-task learning**

I. Introduction

The rapid evolution of data volume, variety, and velocity available from many different sources requires developing new techniques to extract and collect knowledge from such sources. This is particularly critical for political scientists studying violent conflicts at a global scale due to the rapidly increasing variety of actors, behaviors, and locations they study. Conflict scholars have developed computerized approaches to code event data [1]–[7], which consists of automatically identifying events from unstructured input text and extract structured data, providing a description of *someone doing something to someone else in a given location and time*. However, studying conflict processes from a computational social science perspective represents additional challenges to natural language processing (NLP). First, incidents of organized violence are ubiquitous around the world, which requires NLP tools capable of multilingual processing. Furthermore, conflict processes often involve a plurality of armed actors engaging in a variety of violent tactics, thus NLP tools need to perform well on identifying and extracting all these multiple event components reported in conflict news articles.

State-of-the-art generation of conflict datasets generally rely on computerized event coding systems, which identify, extract and categorize conflict interactions from unstructured text, converting them into computer friendly structured representations. Automated coder systems such as PETRARCH, PETRARCH2 [8], Universal PETRARCH [1], Eventus ID [6], and Hadath [7], typically require extensive external dictionaries that serve as knowledge bases to extract critical facts from raw text, such as, *who* are the actors, in *what* type of actions are they involved, and *where* did the event occur.

The dominant ontology for political event data is CAMEO (Conflict and Mediation Event Observations) [9], a coding structure that incorporates a knowledge base composed of actor dictionaries (containing about 67K entries) and action-pattern dictionaries (about 14K verb phrases). The former acts as a data repository for political entities, while the latter is used to store representations of political actions or interactions.

Technically, these dictionaries inform a pattern-matching approach aiming at identifying the presence of certain lexico-syntactic patterns in a sentence, indicating a particular semantic relationship between two nouns. Unfortunately, the complexity of unstructured text generally exceeds the capacity of these dictionary-based coders, often producing low-recall results. In addition, updating and extending these hand-built pattern repositories is too expensive and time-consuming, which quickly renders them obsolete in context of rapidly changing conflict processes. Furthermore, despite the efforts of coding event data in non-English languages [6], [7], [10], to the best of our knowledge, the systems and ontologies used in political science do not support coding events on multilingual corpora, which imposes limitations when working with sources coming from different countries and languages.

Recent advances in deep neural network and natural language understanding techniques open new possibilities to solve some of the core challenges of the traditional event coding approach. In particular, *transformers* [11] based architectures (such as BERT [12] and XLM-RoBERTa) introduced a new approach of pre-training language models for obtaining state-of-the-art results on a wide range of NLP tasks.

In this paper, we introduce the 3M-Transformers (**M**ultilingual, **M**ulti-label, **M**ulti-task) by effectively combining transfer learning and multi-task learning techniques for event coding from multilingual, domain-specific corpora. We explore transfer learning by leveraging pre-trained transformers based models, which in general provide good results requiring small size annotated dataset. Our approach

[1] https://github.com/openeventdata/UniversalPetrarch

still employs parallel residual adapters attached to the transformers based network, favoring multi-task learning and allowing the network to learn document representations in a more effective manner for the purpose of event coding task.

We demonstrate the superiority of 3M-Transformers on a real-world dataset in organized crime domain, which represents a type of conflict process increasingly attracting the attention of political scientists. Although we focus our experiments on this specific domain, the design of our approach is flexible enough to generalize over small size annotated corpora in any domain.

Overall, this paper includes four key contributions. First, by focusing on English, Spanish and Portuguese languages, we design an innovative supervised approach for event coding in multilingual, domain-specific corpora inspired by the combination of techniques from distinct domains, including: natural language understanding, multi-task learning, transfer learning and computer vision concepts (parallel attention levels, residual adapters, and squeeze and excitation). Second, in contrast of the heavy reliance on large actor and action dictionaries, our approach requires a minimalist use of class categories. Third, our application extends the menu of actors and actions traditionally studied in political science by advancing the analysis of organized crime using computational social sciences. Finally, we provide a human-annotated organized crime dataset[2], which may help advance other event extraction approaches and social science studies.

Based on the experiments performed in this paper, our 3M-Transformers models consistently outperform state-of-the-art transformers models in all evaluated performance measures for event coding task. Furthermore, the models maintain good performance values when analyzing the results separately by language, showing evidences that 3M-Transformers models generalize well in multilingual corpora.

The rest of the paper is structured as follows. Section II discusses previous works related to information extraction, extensively covering previous efforts involving coding event data in political and social sciences. Section III outlines the problem addressed in this research and defines our modeling strategy. Section IV describes process for obtaining and annotating the organized crime dataset. Section V provides detailed design of the 3M-Transformers. Section VI presents the results of the proposed approach, and Section VII concludes the paper.

II. Related Works

Generally, event extraction consists of detecting the existence of an event reported in the text through an event trigger component, that works together with argument and role detectors for capturing potential arguments related to such event. Recent works employing deep neural networks show remarkable results on event extraction related tasks by exploring recurrent neural networks [13], [14], graph neural networks [15], [16], hybrid neural networks [17]–[19] and attention mechanism [20].

From the perspective of political and social sciences, conflict scholars are primarily interested on extracting events from text, in such a *structured* format, that allows tracking conflict processes and violence through computational methods. For that purpose, most of the previous works for coding event data are based on pattern matching approach [6]–[8], usually supported by lengthy external repositories or domain-specific ontologies [9]. Some previous efforts [21]–[26] focus on proposing automatic methods for maintaining and extending such ontologies in order to improve the coding event process in political science domain.

In essence, identifying and properly coding events of political violence and protest can also be expressed as classification task. Traditionally, studies focused on classification task rely on classical machine learning and deep learning techniques. Hanna [2] proposes a framework based on support vector machines (SVM) for identifying and coding protest events. Beieler [3] resorts to the application of convolutional neural networks (CNN) to classify pre-selected sentences into the *QuadClass* political events, while Radford [4] trains a recurrent neural networks (RNN) to identify indicators of protest events in English text data. O'Connor et al. [27] describe an unsupervised model for detecting events between major political actors from news corpora. Glavaš et al. [28] develop a multilingual framework based on 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. [7] introduce a logistic regression based framework to code events from conflict related news in Arabic language.

From the perspective of knowledge representation and ontology-based tools applied to crime domain, most efforts focus on modeling and constructing ontologies for criminal law and legal domains [29]–[32]. Some of these works dedicated to ontology modeling cover more specific categories of crime, like social media related crimes [33], property crime [34], or organized crime [35].

With recent advances in deep neural networks and techniques like transferring learning, innovative natural language understanding tools are driving real transformations in NLP applications and delivering state-of-the-art results on a wide range of natural language tasks in various domains. Specifically in political science, recent works [36], [37] utilize BERT, ELMo and DistilBERT as resource for extracting representations from documents, which are latter used as input features for traditional machine learning non-linear classifiers. Still in political science field, other works [38], [39] evidence the power of transformers tools by applying them in distinct tasks like events clustering and coreference resolution.

Outside political and social sciences, recent works focus on proposing approaches for event extraction on multilingual data. Zhu et al. [40] explore machine translation through Google Translate for obtaining text representations with bilingual word features for latter training Chinese and English corpora

[2]https://figshare.com/s/73f02ab8423bb83048aa

altogether. Di et al. [41] proposes a cross-lingual transfer learning method for information extraction using transformer as encoder [42] and position-wise feed-forward sub-layers to capture the event trigger and output the argument roles for the given input sentence. Some efforts [43]–[45] propose deep learning approaches for information extraction related tasks on cross-lingual corpora and other works directly apply cross-lingual bootstrapping on multilingual corpora without resorting to machine translation [46]–[48].

The method introduced in this paper improves the efforts aforementioned by combining state-of-the-art natural language understanding models with multi-task learning for coding document level events in *structured* format from raw-text data. In addition to the multilingual flexibility, our method bypasses high cost and extensive human effort associated with creating and updating external knowledge bases and repositories.

III. Problem Definition and Proposed Modeling Solution

Event coding task aims at extracting structured representations of events in a predetermined template, commonly in triplet format like (*who, did-what, to-whom*) or (*who, did-what, where*). These event representations are usually expressed as a group of codes or categories, each of which corresponding to the argument identified for their roles (who, did-what or where). The main purpose of event coding task is to provide enough amount of structured data, allowing conflict scientists to apply computational methods to analyze, monitor and design forecasting models for conflict processes and violence involving social and political entities across the globe.

Therefore, we define the event coding problem addressed in this paper as a specific case of document-level event extraction. Given an input document, the event coder identifies the arguments and actions that fill any *event component* associated with the event expressed in the document (e.g. who, did-what or where), outputting the codes (or categories) corresponding to such arguments. For example, suppose that we have the following snippet of text in a document:

```
"... members of the criminal MS-13 gang killed
their victims and threw their bodies into
a canyon in the Angeles National Forest, ..."
```

Assuming that we are interested in coding events in (*who, did-what, where*) template, the desired output for such document containing the snippet above should be:

WHO: 32 - MS13 **DID-WHAT:** 04 - HOMICIDES
WHERE: 16 - USA

Note that the output of event coding task is always expressed in structured format (through categories or classes), instead of spans of text for each event component (or role). The exact same event should be coded in a document containing the following sentence:

```
"Mara Salvatrucha gang members were indicted
for six murders on Long Island ..."
```

Given political scientists' interest on coding events around the world, one of the challenges is to extract information from multilingual corpora. Another challenge refers to the need of annotating different event dimensions, such as actor, action, and location of each event. Finally, given the complexity of the behaviors of interest, each dimension can be assigned to multiple labels (e.g., an action can be labeled as "homicide", "kidnapping", "extortion", etc.).

The primary goal of our work is to design an event extraction solution that facilitates coding document level events on multilingual domain-specific news articles via supervised learning, dispensing the use of large dictionaries as knowledge bases. Although our supervised learning approach requires a minimally annotated data as training input, it dismisses the construction of wide-ranging knowledge bases, usually required on pattern-matching based techniques.

To facilitate the usage of supervised learning as our proposed modeling solution, we convert the domain of all annotation components (entities, actions, and locations) that can be potentially captured as part of an event into a labelset. Thus, instead of a set of annotations, each document will be associated to a set of labels (based on a pre-defined structure) corresponding to its annotations. Lastly, we train a multi-label classifier to predict such labels assigned to text documents.

Formally speaking, let $D = \{(x_1, y_1), \ldots, (x_n, y_n)\}$ be a multilingual domain-specific corpora composed by n news articles where $X = \{x_1, \ldots, x_n\}$ corresponds to the raw-text corpora and $Y = \{y_1, \ldots, y_n\}$ refers to the event annotations assigned to their corresponding documents. Each labelset y_i keeps the record of the annotated event components (entity, action, and location) as event in the news article x_i through a binary word of size k, where each one of its positions flags the existence of a particular entity, action, and location. In other words, $y_i \in \{0, 1\}^k$ with labelset L and $|L| = k$. We want to construct a multi-label classifier, h, for predicting the multiple binary labels that may be assigned to each document. Hence, by correctly predicting such labels, we successfully capture the events from the corpora. Although we aim to demonstrate the model's performance on a real-world dataset (see Section IV), the design introduced in Section V is flexible enough to generalize over annotated corpora in any domain and any event template.

IV. Data Description

The dataset in this study consists of news articles reporting organized crime activity in both English and Spanish. The corpora came from the Insightcrime[3] web page through the Open Event Data Alliance web-scraper program [4]. InsightCrime is a foundation specialized in studying and reporting organized criminal activity involving gangs, cartels and other non-state armed actors operating in Latin America and Caribbean. For constructing our dataset, we collected $19,940$ documents from July 2004 to March 2020 ($13,236$ in English and $6,704$ in Spanish).

As part of the labeling process, a political science committee composed of three experts annotated the ground-truth tags

[3]https://www.insightcrime.org

[4]https://github.com/openeventdata/scraper

corresponding to the **criminal entities** and **crime categories** reported in the documents, as well as the **locations** (in country level) of the occurrence. The coders worked independently on annotations, reaching excellent standards in terms of inter-annotation agreement, as we show below:

- Exact Match Ratio: 97.6%
- Cohen's Kappa (Annotator 1 & Annotator 2): 99.44%
- Cohen's Kappa (Annotator 1 & Annotator 3): 99.50%
- Cohen's Kappa Measure (Annotator 2 & Annotator 3): 99.16%
- Fleiss' Kappa: 99.37%

Note that, because we work with multi-label data and the annotated labels are not mutually exclusive, we computed the kappa statistics separately for each and every label, and then averaged them. In the few cases of disagreements, we considered the majority of votes for each instance as final ground-truth annotations.

Since the documents contain unstructured text and the nature of their description varies, a fixed format for assigning the annotations is not viable. To address this, the coders assigned up to three labels per event component (crime category, criminal entities and location) for each news story. This means, for example, that a document may report a specific crime occurring in a given location without mentioning the perpetrators (sometimes the authors of the crime are unknown), while another document may report three related crimes (e.g. arms trafficking, extortion, and homicides) occurring in distinct locations (e.g. El Salvador and Honduras) involving two gangs (e.g. Mara Salvatrucha and Barrio 18).

From the $19,940$ scrapped InsighCrime documents, we manually annotated a random sample of $2,533$ news articles. Overall, the annotations assigned to documents contain 16 distinct crime categories, 41 criminal entities, and 33 countries (including the U.S. and some European countries). This is a minimalist classification set when compared to the CAMEO dictionaries and other domain-specific repositories in political science sphere. Fig. 1 illustrates the percentage of documents annotated with each crime category.

Due to space limitations, we omit the plots of criminal entities and countries in this section. Instead, we report here the most and least frequent annotations for these two dimensions. For criminal entities, the most frequent annotations are FARC (5.17%), Cartel de Sinaloa (4,78%) and Zetas (4,07%), while the least frequent are Barrio Azteca (0.04%), Los Machos (0.04%) and Leones (0.04%). For countries, the most frequent are Mexico (24.99), Colombia (17.37%) and USA (12.23%), while the least frequent are the UK (0.12%), Portugal (0.08%) and Germany, (0.04%).

Given the imbalanced distribution observed in all event components, we re-arranged the annotations by grouping the least frequent tags into larger groups. We generated a preliminary grouping through unsupervised learning (k-means++) running over the documents representations obtained through the BERT base pre-trained model (bert-base-multilingual-cased [5]). Then, political science experts fine-tuned the initial grouping suggestions and designed the final grouping sets.

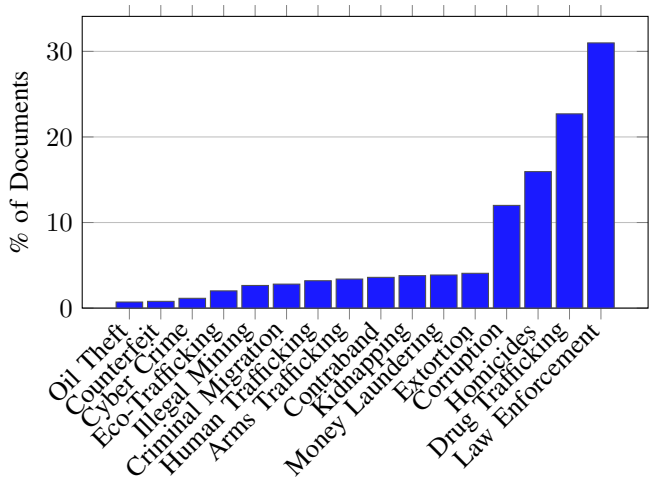

Fig. 1: Distribution of documents by category of crime.

The final crime category groups used in the study are:

- Drug Trafficking;
- Corruption;
- Law Enforcement;
- Homicides;
- Economic Related Crimes (Counterfeit, Extortion, Kidnapping, Money Laundering and Cyber Crime);
- Natural Resources Crimes (Eco-Trafficking, Illegal Mining and Oil Theft);
- Crime Mobility (Arms Trafficking, Contraband, Criminal Migration, Human Trafficking).

For criminal entities, political science experts grouped together names of criminal organizations, leaders, and main members belonging to major gangs or organized criminal groups, which totaled 15 criminal organizations. For countries, we kept the original label for each Latin American country, but grouped together all European countries in a single category given their low individual frequency. This yielded a total of 16 location groups.

Finally, due to the limited number of manually-annotated tags, we applied a semi-supervised label propagation algorithm [49] to propagate the human annotations to the whole corpora. We obtained the document features through the BERT base pre-trained model following the same procedure implemented for clustering analysis as mentioned above.

Technically, let $(x_1, y_1), \ldots, (x_l, y_l)$ be the annotated documents, where $Y_l = \{y_1, \ldots, y_l\}$ and $l = 2,533$. Let $(x_{l+1}, y_{l+1}), \ldots, (x_{l+u}, y_{l+u})$ be the scraped and non-annotated documents, where $Y_u = \{y_{l+1}, \ldots, y_{l+u}\}$ and $u = 17,407$. Lastly, let $X = \{x_1, \ldots, x_{l+u}\}$ where $x_i \in \mathbb{R}^{d_m}$ corresponds to the documents representations with dimension size d_m extracted through the BERT model. We use label propagation to estimate Y_u from X and Y_l. By using the whole

[5]https://huggingface.co/bert-base-multilingual-cased

scraped corpora X with their corresponding propagated labels as major part of training set, we increased (on average) 2 percentage points in F1-score for all the models experimented in Section VI (taking as baseline the results obtained with the annotated documents only).

For the experiments reported in this paper, we perform 14 rounds of cross-validation, by randomly splitting the annotated data between train (60%), development (20%) and test (20%) in each round. All the remaining scraped documents not belonging to the annotated portion is reserved only for training purpose in all the rounds. Thus, 100% of the testing set in all the rounds is composed by annotated data only, **ensuring the veracity of performance measures reported in our experiments**.

In order to better analyze the multilingual functionality of the proposed method and to cover all the major languages spoken in Latin America, we also scraped news articles in Portuguese. As data source, we have chosen the crimes section of three prestigious news websites in Brazil: El Pais[6], Veja[7] and UOL[8]. Only one coder annotated documents in Portuguese, therefore such dataset is used only for validating the proposed methodology, discussed in Section VI.

V. Approach and Design

As described in Section III, we devise a modeling solution based on supervised learning for coding event data from multilingual news articles. The design of our proposed approach is detailed in Subsections V-A to V-C, while subsection V-D closes this section by briefly describing the baseline methods used as reference for comparison in our experiments.

A. *3M-Transformers for Event Coding*

In this paper, we design the 3M-Transformers (**M**ultilingual, **M**ulti-label, **M**ulti-task) by combining multi-task learning with machine transfer learning techniques to efficiently extract events in structured format from multilingual corpora.

We employ transfer learning by leveraging the transformers based architectures as part of our design, requiring only a small annotated corpora for fine-tuning pre-trained multilingual models.

Furthermore, 3M-Transformers explore multi-task learning to extract different representations from the same document through residual adaptations (each one adjusted to each event component) over the same transformers based network as base of the model. Although we work with a domain-specific corpora, we conjecture that the feature spaces and data distribution characteristics for each of event component are not the same.

Therefore, our proposed architecture implements parallel *residual adapters* [50] attached to transformers based models to favor multi-task learning. Fig. 2 illustrates the 3M-Transformers design: we keep BERT model (inner gray block) as base of the network and connect in parallel, for each event component, adapters which will attend and independently learn their own concepts. Thus, given the same document as input, such network outputs different representations (one for each event component), which will feed their corresponding linear layers (again, one Feed Forward Network for each event component), each one working as multi-label classifiers.

Without loss of generality, the network in Fig. 2 illustrates the event code extraction in the (*who, did-what, where*) template, which resembles the output format expected from using as input the dataset described in Section IV.

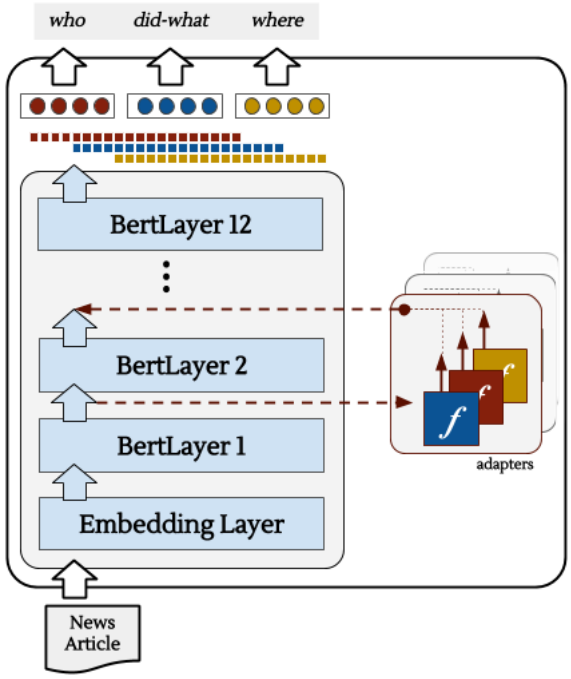

Fig. 2: 3M-Transformers design (with BERT)

Technically, the original BERT base architecture is composed by 12 layers, each of which implements *self-attention* layer over the input hidden representation, followed by a layer-norm (LN) with residual connection, as expressed next:

$$BertLayer(h) = LN(h + SelfAttention(h)) \quad (1)$$

where h corresponds to the hidden representation given as input.

As expressed in Fig. 2, we attach a residual adapter function (for each event component) in parallel to each BertLayer before applying the layer-norm (LN), which results as follows:

$$h_{out} = LN(h_{in} + SelfAttention(h_{in}) + f(h_{in})) \quad (2)$$

where h_{out} is the output hidden representation obtained after the layer-norm over the original BERT layer output and the parallel residual adapter function $f(\cdot)$.

Because BERT, as well as other transformers based architectures, are prominent in recent NLP researches and our design is based on its original implementation[9], we omit further detailed description about these model architectures.

[6]https://brasil.elpais.com/
[7]https://www.uol.com.br/
[8]https://veja.abril.com.br/

[9]https://github.com/huggingface/transformers

B. *Residual Adapter Configurations*

We implement two residual adapter functions as essential components of 3M-Transformers. The first one is based on the *projected attention layer* [51], which consists of a low-dimensional multi-head attention layer that is attached in parallel to BERT layers. This residual adapter function is expressed as follows:

$$f_{pal}(h) = V^D MH(V^E h), \tag{3}$$

where $h \in \mathbb{R}^{d_m}$ represents the hidden layer that is first encoded through a matrix multiplication with $V^E \in \mathbb{R}^{d_s \times d_m}$, which in turn will feed a low dimensional multi-head attention layer $MH(\cdot)$ based on transformers architecture. Finally $V^D \in \mathbb{R}^{d_m \times d_s}$ serves as decoder and returns the original dimension of h to feed the layer norm layer expressed in Eq. 2. Although we attach $f(\cdot)$ as a residual adapter module over each one of the 12 BERT-layers, in this configuration both V^E and V^D are shared across the BERT layers.

Because we dedicate one task for each event component and the majority of the network parameters are shared among tasks, our second configuration was designed seeking the recalibration of the feature responses in each BERT-layer for different tasks. Therefore, we borrow *squeeze-and-excitation* [52] concept from computer vision domain and tailor it to natural language processing application to devise the following *SE-based* residual adapter function:

$$f_{se}(H) = F_e(F_s(H), W) \odot h0, \tag{4}$$

Different than $f_{pal}(\cdot)$, function $f_{se}(\cdot)$ receives $H = [h0, \ldots, hd_T] \in \mathbb{R}^{d_m \times d_T}$ as input, where $d_m = 768$ (standard BERT hidden size) and $d_T = 512$ (maximum number of tokens in the document). In other words, $f_{se}(\cdot)$ receives the representations of all the tokens in the input document, instead of pooling them and working only with the hidden state corresponding to the first special token ([CLS]), usually output as hidden state in each BertLayer (as described in section 3 of [12]). Therefore, such residual adapter configuration allows learning document representations in a more effective manner, by exploring hidden representations of all the tokens in the input document for all the transformers hidden layers.

Precisely, $F_s(\cdot)$ averages the representations of all d_T tokens to obtain $z \in \mathbb{R}^{d_m}$, where the i-the element of z is simply computed as follows:

$$z_i = \frac{1}{d_T} \sum_{t=1}^{d_T} H_{it}, \tag{5}$$

The excitation step represented by the function F_e learns a nonlinear and non-mutually-exclusive relationship between features through a simple gated mechanism using the sigmoid activation, as expressed in Eq. 6:

$$F_e(z, W) = \sigma(W_2(\delta(W_1 z)), \tag{6}$$

where z represents the output from the squeeze step, δ refers to the ReLU function, $W_1 \in \mathbb{R}^{\frac{d_m}{r} \times d_m}$ and $W_2 \in \mathbb{R}^{d_m \times \frac{d_m}{r}}$. In practice, we have two fully connected layers around the non-linearity, where the first one performs a dimensionality reduction with parameters W_1 with reduction ratio r while the second executes a dimensionality increasing with parameter W_2. As output, function $F_e(\cdot)$ returns a vector of dimension d_m.

The final output of $f_{se}(\cdot)$ is obtained by rescaling the hidden state corresponding to the special token ([CLS]) represented as $h0$ in Eq. 4 with the activations output from $F_e(\cdot)$ through a feature-wise multiplication (Hadamard product). Note that $h0 \in \mathbb{R}^{d_m}$ is indeed exactly the same hidden representation as h expressed in Eq. 3.

Finally, $f_{se}(\cdot)$ serves as an adapter block plugged as $f(\cdot)$ in Eq. 2 to compose an additional configuration of 3M-Transformers.

Attaching such external adapters to transformers base architectures results in a slight complexity increase in terms of number of parameters in the overall network. Specifically for PALs models, such complexity increasing may be expressed as follows:

$$EC \times (2d_m d_s + 12 \times 3d_s^2), \tag{7}$$

while the same increasing in complexity order is expressed as follows for SE configurations:

$$EC \times (12 \times (2d_m(d_m/r))), \tag{8}$$

where EC means the number of event components (or event tasks), d_m represents the default base transformers hidden dimension with 12 layers, d_s defines the dimensions of V^D and V^E on *PAL* designs and r expresses the reduction ratio on configurations using *SE* based adapters.

In our experiments we set $d_s = 204$ and $r = 18$. Considering $EC = 3$ and $d_m = 768$, we have an increase of **no more than 2.1%** in total parameters on the original BERT base (110 million parameters) when applying *SE-based* adapter.

C. *Transformers architectures*

We combine the residual adapter configurations expressed in previous subsection to two transformers based architectures: BERT and XLM-RoBERTa. Later in Section VI, we explicitly call each experimented model based on their transformer base architecture followed by the suffix "***_SE***" or "***_PAL***" depending on the adapter they used.

To address the multilingual aspect of the problem, we simply used the BERT multilingual and XLM-RoBERTa base pre-trained models as initial weights for fine-tuning our 3M-Transformers implementations (see Subsection VI-A).

D. *Baseline Approaches*

As baseline approaches, we considered three transformers based implementations. The first method consists on **Multi-label BERT for Sequence Classification**, where we simply fine tune the BERT base multilingual pre-trained model for extracting documents representations followed by a sequence multi-label classifier on the top of that. Latter in Section VI, we refer to this model as **Simple-BERT**.

In the same manner as we did for **Simple-BERT**, we create **Simple-XLM-RoBERTa** by replacing BERT by XLM-RoBERTa, as our second baseline.

Lastly, we implement the event coding approach introduced in [37], which explores BERT pre-trained model for extracting documents representations and apply SVM for predicting the event components. We slightly adapt the original implementation by using the multilingual pre-trained model instead of using the common BERT base trained in English language only.

To the best of our knowledge, there is no organized crime ontology or knowledge base which supports coding events on multilingual corpora, precluding any implementation based on pattern-matching approach as baseline.

VI. EXPERIMENTS

In this section, we describe the computational setup used on the experiments and the metrics applied for performance evaluation (Subsections VI-A and VI-B respectively). Lastly, in Subsection VI-C, we present the results of our experiments run over the real-world organized crime dataset.

A. Setup

To conduct the experiments presented in this paper we used a computer with NVIDIA Quadro RTX 8000 GPU. The base architecture (excluding the residual adapters) for the BERT and XLM-RoBERTa networks were entirely based on their original design and implementation, publicly available on transformers repository[10]. We fine-tune all of our 3M-Transformers configurations for 30 epochs with mini-batch size of 4 and gradient accumulation set to 8 given the large value of maximum sequence length $d_T = 512$, required for dealing with long text sizes in organized crime domain. We choose the best model found during training step based on F1-score observed on development set. We use Adam optimizer with learning rate set to $1e-5$, $\beta_1 = 0.9$, $\beta_2 = 0.999$ and $\epsilon = 1e-6$. Given the high imbalance rate of the input data, we experimented training the models with *focal loss* function [53], but we observed better results using *binary cross entropy* as loss function.

We train our models in multi-task fashion based on *round robin* approach: the batches of each task are simply selected in the same order in a cycle, which means that every epoch will select equivalent number of examples for all the three tasks (event components) during training process.

Furthermore, we use the following publicly available pre-trained models as initial weights on the bases of the architectures detailed in the Subsections V-B and V-D:

- BERT base multilingual cased[11]: Pre-trained on 104 languages, with 12 layers (12 attention heads each) and hidden states dimensions of size $d_m = 768$.
- XLM-RoBERTa base multilingual cased[12]: Pre-trained on 100 languages, with 12 layers (12 attention heads each) and hidden states dimensions of size $d_m = 768$.

For all 3M-Transformers implementations (which are based on multi-task technique), we set the number of steps per epoch to $5,000$. It totals $20,000$ cycles per epoch (mini-batch size $\times$ steps per epoch), which is equivalent to the training set size, allowing a fair comparison with the baseline methods. Furthermore, for each configuration, we report the average of the results on the testing sets over the 14 cross-validation rounds (as described in Section IV).

Specifically for *SE-based* configurations, we set the reduction rate $r = 18$, which presented, on average, the best results over the other values experimented for this hyper-parameter ($[6, 12, 18, 24]$).

On configurations using *project attention layers*, we use 6 attention-heads with $d_s = 204$ for each block denoted as $MH(\cdot)$ expressed in Eq. 3.

We apply *k-means++* and *label propagation* implementations from *Scikit-learn* library to perform some steps of dataset construction, as detailed in Section IV.

B. Performance Evaluation

Dealing with multi-label classification requires using suitable performance measures for properly evaluating the trained models. Therefore, following previous references [54], we adopt **example-based** measures for evaluating the models:

$$A = \frac{1}{n}\sum_{i=1}^{n}\frac{|Y_i \cap Z_i|}{|Y_i \cup Z_i|}, \qquad P = \frac{1}{n}\sum_{i=1}^{n}\frac{|Y_i \cap Z_i|}{|Z_i|}$$

$$R = \frac{1}{n}\sum_{i=1}^{n}\frac{|Y_i \cap Z_i|}{|Y_i|}, \qquad F1 = \frac{1}{n}\sum_{i=1}^{n}\frac{2\,|Y_i \cap Z_i|}{|Y_i| + |Z_i|}$$

$$MR = \frac{1}{n}\sum_{i=1}^{n} I(Y_i = Z_i)$$

where n represents the size of the multi-labeled corpora composed by instances (x_i, Y_i), $1 \leq i \leq n$, $x_i \in \mathbb{R}^{d_m}$ denoting the document representations, $Y_i \in \{0,1\}^k$ with labelset L and $|L| = k$. Assuming h is our multi-label classifier, we let $Z_i = h(x_i) = \{0,1\}^k$ be the set of label memberships predicted by h for the example x_i.

The measures accuracy (A), precision (P), recall (R) and F1-score (F1) account for partial correctness. Alternatively, the exact match ratio (MR) represents the most strict measure, taking into consideration the indicator function I, which doesn't distinguish complete incorrect and partially correct.

C. Experiments on Organized Crime Dataset

We start by evaluating the performance measures for the configurations detailed in Section V compared to the baseline methods.

[10]https://huggingface.co/transformers/

[11]https://huggingface.co/bert-base-multilingual-cased

[12]https://huggingface.co/xlm-roberta-base

TABLE I: 3M-Transformers vs. baselines: overall performance for coding event data on organized crime dataset

MODELS	A	P	R	F1	MR
Simple-BERT	72.53	83.93	83.23	80.99	39.77
Simple-XLM-Ro	72.70	84.06	83.49	81.23	39.81
BERT_SVM [37]	53.43	78.23	60.11	63.33	23.62
3M-BERT_PAL	74.40	86.49	83.26	82.35	43.61
3M-BERT_SE	**74.44**	86.34	**83.55**	**82.39**	43.21
3M-XLM-Ro_PAL	74.27	86.58	83.51	82.32	**43.64**
3M-XLM-Ro_SE	73.98	**86.78**	83.53	82.31	42.49

Results in **bold** font indicate the best performing model.

Table I shows the overall results obtained by evaluating the event components (event tasks) altogether. Overall, **3M-BERT_SE** outperforms all the experimented models in accuracy, recall and F1-score, showing more than three percentage points improvement in exact match ratio (MR) when comparing to Simple-BERT and Simple-XLM-Ro.

Furthermore, all 3M-Transformers models consistently presented better results than those obtained by fine-tuning state-of-the-art transformers models (BERT and XLM-RoBERTa) for the purpose of event coding task. The performance superiority is statistically significant when looking at the most strict measure (Match Ratio): both 3M-BERT_SE and 3M-BERT_PAL outperform Simple-BERT as well as 3M-XLMro_PALs and 3M-XLMro_SE significantly outperform Simple-XLM-Ro at 0.001 level (based on t-test).

Although 3M-Transformers models learn the parameters of all event tasks altogether along the same training process, we can evaluate how is the performance of these models broken by task. For that purpose, Table II presents the performance measures of each model, at the event components level.

For properly understanding this analysis, it is necessary to add an extra detail about the organized crime dataset. Because the nature of events related to organized crime varies, there may be documents without annotations corresponding to one of the event components. For instance, there may be documents reporting a specific type of crime which occurred in a place, without necessarily reporting any criminal entity. Therefore, computing the performance measures and reading the analysis presented in Table II may require a deeper interpretation.

By definition, we use *precision* measure in our context to evaluate the percentage of event elements which were correctly predicted. Therefore, following the formula indicated in Subsection VI-B, the computation of the *precision* measure ignores those instances where $|Z_i| = 0$, as well as the *recall* measure ignores cases where $|Y_i| = 0$.

Because only approximately 30% of the annotated documents contain criminal entities assigned to them, we see that MR measures for all models are greater than F1-score measures in the "*Criminal Entity (who)*" event task. In practice, it means that models can correctly identify documents without criminal entities, which contributes for increasing the MR but do not count to F1-score (because $|Y_i| + |Z_i| = 0$ for those cases).

Even though none of the models consistently obtained the best performance measures cross-event tasks, we note that the best values for each measure are highly concentrated in the 3M-Transformers models.

Although we focus the performance discussion on *example-based* measures (introduced in previous subsection), 3M-Transformers models also presented good results along all event components when analyzing *label-based* [54] measures. When comparing 3M-BERT_SE against Simple-BERT, the former presented better results in F1-score for 6 out of the 7 labels related to *crime categories*, 9 out of the 16 labels in *location* component and 9 out of the 15 labels in *criminal entities* component.

In order to evaluate the multilingual capabilities of 3M-Transformers, we analyze separately the performance measures across languages. As discussed in Section IV, the InsightCrime corpora is composed by documents both in English (EN) and Spanish (ES) languages, which were used together for fine-tuning all the models presented so far. To perform a more comprehensive evaluation, we add to this analysis an extra organized crime corpora in Portuguese (PT) language (see Section IV) and fine-tune (with only 3 epochs) the previous models with the training portion of the Portuguese corpora.

TABLE III: 3M-Transformers' overall performance by language (3M-BERT_SE and 3M-XLM-Ro_SE)

MEASURES	3M-BERT_SE			3M-XLM-Ro_SE		
	EN	ES	PT	EN	ES	PT
A	75.17	73.27	64.90	75.09	72.05	65.86
P	86.70	86.17	91.92	87.20	86.63	91.21
R	84.44	81.72	67.39	84.05	81.05	68.63
F1	83.16	81.14	73.20	83.00	80.45	73.88
MR	42.77	44.94	37.86	43.50	41.41	38.94

Table III summarizes the overall performance of **3M-BERT_SE** and **3M-XLM-Ro_SE** by language, showing that both models generalize well in all experimented languages. Although we observe better performance in EN and ES languages (as expected), the models achieve good performance in PT corpora with only a small extra fine-tuning effort. Such results serve as empirical evidences to support not only the 3M-Transformers' multilingual capability for coding events task but also the robustness of these models to generalize on input corpora from distinct sources with different distributions over the labelsets (Portuguese corpora).

VII. Conclusions and Future Work

Political scientists and government agencies in the security sector are in constant need of gathering and analyzing event data on conflict processes and violence around the world. To that end, researchers increasingly rely on computer generated data. However, most of these event coding protocols require costly development, maintenance, and expansion of dictionaries of actors, actions, and locations. These hurdles often prevent the generation of timely and accurate structured event data to track conflict and violence.

In this paper, we propose an innovative technique to address key challenges of computerized event data generation from

TABLE II: 3M-Transformers vs. baselines: performance by task (event component level)

MODELS	Criminal Entity (who)					Crime Category (did-what)					Location (where)				
	A	P	R	F1	MR	A	P	R	F1	MR	A	P	R	F1	MR
Simple-BERT	62.03	73.73	**80.31**	63.90	85.99	63.70	76.34	73.40	67.30	55.26	89.22	93.59	94.18	91.55	82.22
Simple-XLM-Ro	60.34	72.49	79.25	62.21	85.17	64.80	76.42	74.22	68.31	56.51	89.14	93.58	94.04	91.50	82.02
BERT_SVM [37]	34.49	73.97	39.95	36.25	80.11	48.06	67.58	51.79	50.10	45.08	69.88	87.75	74.26	73.08	60.82
3M-BERT_PAL	**66.07**	**81.91**	78.85	**68.22**	**88.25**	65.37	**78.38**	73.96	68.69	57.82	90.18	94.81	94.06	92.31	83.80
3M-BERT_SE	65.60	81.29	78.49	67.69	88.07	65.39	78.24	**74.49**	68.77	57.51	**90.27**	94.65	**94.50**	**92.44**	83.67
3M-XLM-Ro_PAL	63.60	80.28	76.81	65.64	87.44	**65.56**	78.12	74.29	**68.92**	**58.07**	89.79	94.79	94.15	92.02	83.30
3M-XLM-Ro_SE	64.06	80.74	77.04	66.08	87.64	64.33	77.81	74.08	68.32	56.60	90.13	**95.07**	94.24	92.24	**84.05**

Results in **bold** font indicate the best performing model.

multilingual domain-specific corpora. We do so combining state-of-the-art natural language understanding models with multi-task learning approach to efficiently extract events in structured format. We demonstrate the application of such proposed architecture through a real-world case study focused on organized crime.

3M-Transformers (**M**ultilingual, **M**ulti-label, **M**ulti-task) for event coding implements transfer learning technique by leveraging multilingual transformers models, which provide high quality results through fine-tuning process over small number of labeled data. Furthermore, 3M-Transformers incorporate parallel residual adapters dedicated to better explore the feature spaces of each event component through multi-task learning. We propose the SE-based residual adapter by borrowing squeeze-and-excitation concept from computer vision domain and tailoring it to natural language processing application. SE-based adapters allow learning document representations in a more effective manner, by exploring hidden representations of all the tokens in the input document for all the transformers hidden layers.

Our experiments on organized crime show indications that 3M-Transformers outperform state-of-the-art usual transformers models for coding event data, by a minimal complexity increasing in number of parameters. **3M-BERT_SE** consistently shows better results than **Simple-BERT** as well as **3M-XLM-Ro_SE** consistently outperforms **Simple-XLM-Ro** with statistical significance.

Related to the multilingual challenge of event coding, 3M-Transformers report good performance in both English and Spanish languages, and do well in capturing the semantics of other languages outside training corpora (Portuguese) requiring only a small number of fine-tuning epochs. Results for the Portuguese corpora are likely to increase once we have enough annotated data on this language to use it as part of training set.

An open discussion for future work is to analyze how the performance of 3M-Transformers models will behave by increasing the heterogeneity level (in terms of data source and number of distinct languages) of the input corpora. Furthermore, we intend to expand the case study to other micro domains in political science sphere (e.g. terrorism, insurgencies, protest movements and multinational military exercises).

ACKNOWLEDGEMENTS

The research reported herein was supported in part by NSF awards OAC-1931541, OAC-1828467, DMS-1737978, DGE-2039542, 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] P. A. Schrodt, "Keynote abstract: Current open questions for operational event data," in *Proceedings of the Workshop on Automated Extraction of Socio-political Events from News 2020*. Marseille, France: European Language Resources Association (ELRA), May 2020, p. 8.

[2] A. Hanna, "Mpeds: Automating the generation of protest event data," Available at https://osf.io/preprints/socarxiv/xuqmv (2021/08/07), 2017, unpublished Manuscript.

[3] J. Beieler, "Generating politically-relevant event data," *In Proceedings of the First Workshop on NLP and Computational Social Science*, pp. 37–42, 2016.

[4] B. Radford, "Multitask models for supervised protest detection in texts," Available at https://arxiv.org/abs/2005.02954 (2021/08/07), 2019, unpublished Manuscript.

[5] S. Salam, P. Brandt, J. Holmes, and L. Khan, "Distributed framework for political event coding in real-time," *Proceedings of the 2018 conference on Electrical Engineering and Computer Science (EECS)*, 2018.

[6] J. Osorio and A. Reyes, "Supervised Event Coding From Text Written in Spanish: Introducing Eventus ID," *Social Science Computer Review*, vol. 35, no. 3, pp. 406–416, 2017.

[7] J. Osorio, A. Reyes, A. Beltrán, and A. Ahmadzai, "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*. Marseille, France: European Language Resources Association (ELRA), May 2020, pp. 49–56.

[8] C. Norris, P. Schrodt, and J. Beieler, "Petrarch2: Another event coding program," *Journal of Open Source Software*, vol. 2, no. 9, p. 133, 2017. [Online]. Available: https://doi.org/10.21105/joss.00133

[9] D. J. Gerner, P. A. Schrodt, O. Yilmaz, and R. Abu-Jabr, "Conflict and mediation event observations (cameo): A new event data framework for the analysis of foreign policy interactions," *International Studies Association, New Orleans*, 2002.

[10] J. Osorio, V. Pavon, S. Salam, J. Holmes, P. T. Brandt, and L. Khan, "Translating CAMEO verbs for automated coding of event data," *International Interactions*, vol. 45, no. 6, pp. 1049–1064, 2019.

[11] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, and J. Brew, "Huggingface's transformers: State-of-the-art natural language processing," *ArXiv*, vol. abs/1910.03771, 2019, unpublished Manuscript.

[12] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018, unpublished Manuscript.

[13] T. M. Nguyen and T. H. Nguyen, "One for all: Neural joint modeling of entities and events," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 01, 2019, pp. 6851–6858.

[14] Y. Zhang, G. Xu, Y. Wang, X. Liang, L. Wang, and T. Huang, "Empower event detection with bi-directional neural language model," *Knowledge-Based Systems*, vol. 167, pp. 87–97, 2019.

[15] X. Liu, Z. Luo, and H. Huang, "Jointly multiple events extraction via attention-based graph information aggregation," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Brussels, Belgium: Association for Computational Linguistics, Oct.-Nov. 2018, pp. 1247–1256.

[16] T. Nguyen and R. Grishman, "Graph convolutional networks with argument-aware pooling for event detection," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, 2018.

[17] Y. Hong, W. Zhou, J. Zhang, G. Zhou, and Q. Zhu, "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)*, 2018, pp. 515–526.

[18] J. Liu, Y. Chen, and K. Liu, "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, no. 01, 2019, pp. 6754–6761.

[19] T. Zhang, H. Ji, and A. Sil, "Joint entity and event extraction with generative adversarial imitation learning," *Data Intell*, vol. 1, no. 2, pp. 99–120, 2019.

[20] S. Liu, Y. Chen, K. Liu, and J. Zhao, "Exploiting argument information to improve event detection via supervised attention mechanisms," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2017, pp. 1789–1798.

[21] M. Du and R. Yangarber, "Acquisition of domain-specific patterns for single document summarization and information extraction," *Proceedings of the 2nd International Conference on Artificial Intelligence and Pattern Recognition*, p. 30, 2015.

[22] E. S. Parolin, S. Salam, L. Khan, P. Brandt, and J. Holmes, "Automated verbal-pattern extraction from political news articles using cameo event coding ontology," *International Conference on Intelligent Data and Security*, pp. 258–266, 2019.

[23] E. S. Parolin, L. Khan, J. Osorio, V. D'Orazio, P. Brandt, and J. Holmes, "Hanke: Hierarchical attention networks for knowledge extraction in political science domain," *IEEE International Conference on Data Science and Advanced Analytics (DSAA)*, 2020.

[24] R. Yangarber, R. Grishman, P. Tapanainen, and S. Huttunen, "Automatic acquisition of domain knowledge for information extraction," *Proceedings of the 18th conference on Computational linguistics*, vol. 2, pp. 940–946, 2000.

[25] E. Riloff, R. Jones *et al.*, "Learning dictionaries for information extraction by multi-level bootstrapping," *Proceedings of the 16th National Conference on Artificial Intelligence*, pp. 474–479, 1999.

[26] R. Huang and E. Riloff, "Multi-faceted event recognition with bootstrapped dictionaries," *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 41–51, 2013.

[27] B. O'Connor, B. M. Stewart, and N. A. Smith, "Learning to extract international relations from political context," *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, vol. 1, pp. 1094–1104, 2013.

[28] G. Glavaš, F. Nanni, and S. P. Ponzetto, "Cross-lingual classification of topics in political texts," in *Proceedings of the Second Workshop on NLP and Computational Social Science*. Vancouver, Canada: Association for Computational Linguistics, Aug. 2017, pp. 42–46.

[29] C. Asaro, M. A. Biasiotti, P. Guidotti, M. Papini, M.-T. Sagri, D. Tiscornia *et al.*, "A domain ontology: Italian crime ontology," *Proceedings of the ICAIL 2003 Workshop on Legal Ontologies & Web based legal information management*, pp. 1–7, 2003.

[30] A. Gangemi, M.-T. Sagri, and D. Tiscornia, "Metadata for content description in legal information," *Proceedings of the 14th International Workshop on Database and Expert Systems Applications*, 2003.

[31] R. Hoekstra, J. Breuker, M. Di Bello, A. Boer *et al.*, "The lkif core ontology of basic legal concepts." *Proceedings of the Workshop on Legal Ontologies and Artificial Intelligence Techniques (LOAIT)*, vol. 321, pp. 43–63, 2007.

[32] R. Rubino, A. Rotolo, and G. Sartor, "An owl ontology of norms and normative judgements," *Proceedings of the 5th Legislative XML Workshop*, pp. 173–187, 2007.

[33] E. Kalemi, S. Yildirim-Yayilgan, E. Domnori, and O. Elezaj, "Smont: an ontology for crime solving through social media," *International Journal of Metadata, Semantics and Ontologies*, vol. 12, pp. 71–81, 2017.

[34] C. M. de Oliveira Rodrigues, F. L. G. De Freitas, and R. R. De Azevedo, "An ontology for property crime based on events from ufo-b foundational ontology," *Proceedings of the 5th Brazilian Conference on Intelligent Systems (BRACIS)*, pp. 331–336, 2016.

[35] J. Osorio, "The Contagion of Drug Violence: Spatiotemporal Dynamics of the Mexican War on Drugs," *Journal of Conflict Resolution*, vol. 59, no. 8, pp. 1403–1432, 2015.

[36] B. Büyüköz, A. Hürriyetoğlu, and A. Özgür, "Analyzing ELMo and DistilBERT on socio-political news classification," in *Proceedings of the Workshop on Automated Extraction of Socio-political Events from News 2020*. Marseille, France: European Language Resources Association (ELRA), May 2020, pp. 9–18.

[37] F. Olsson, M. Sahlgren, F. ben Abdesslem, A. Ekgren, and K. Eck, "Text categorization for conflict event annotation," in *Proceedings of the Workshop on Automated Extraction of Socio-political Events from News 2020*. Marseille, France: European Language Resources Association (ELRA), May 2020, pp. 19–25.

[38] F. K. Örs, S. Yeniterzi, and R. Yeniterzi, "Event clustering within news articles," in *Proceedings of the Workshop on Automated Extraction of Socio-political Events from News 2020*. Marseille, France: European Language Resources Association (ELRA), May 2020, pp. 63–68.

[39] B. Radford, "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*. Marseille, France: European Language Resources Association (ELRA), May 2020, pp. 35–41.

[40] Z. Zhu, S. Li, G. Zhou, and R. Xia, "Bilingual event extraction: a case study on trigger type determination," in *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 2014, pp. 842–847.

[41] D. Lu, A. Subburathinam, H. Ji, J. May, S.-F. Chang, A. Sil, and C. Voss, "Cross-lingual structure transfer for zero-resource event extraction," in *Proceedings of The 12th Language Resources and Evaluation Conference*, 2020, pp. 1976–1981.

[42] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Neural Information Processing Systems (NIPS)*, pp. 5998–6008, 2017.

[43] M. M'hamdi, M. Freedman, and J. May, "Contextualized cross-lingual event trigger extraction with minimal resources," in *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, 2019, pp. 656–665.

[44] J. Ni, T. Moon, P. Awasthy, and R. Florian, "Cross-lingual relation extraction with transformers," *arXiv preprint arXiv:2010.08652*, 2020, unpublished Manuscript.

[45] P. Verga, D. Belanger, E. Strubell, B. Roth, and A. McCallum, "Multilingual relation extraction using compositional universal schema," *In Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2015.

[46] Z. Chen and H. Ji, "Can one language bootstrap the other: a case study on event extraction," in *Proceedings of the NAACL HLT 2009 Workshop on Semi-Supervised Learning for Natural Language Processing*, 2009, pp. 66–74.

[47] A. Hsi, J. G. Carbonell, and Y. Yang, "Modeling event extraction via multilingual data sources." in *TAC*, 2015.

[48] A. Hsi, Y. Yang, J. G. Carbonell, and R. Xu, "Leveraging multilingual training for limited resource event extraction," in *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, 2016, pp. 1201–1210.

[49] X. Zhu and Z. Ghahramani, "Learning from labeled and unlabeled data with label propagation," 2002.

[50] S.-A. Rebuffi, H. Bilen, and A. Vedaldi, "Efficient parametrization of multi-domain deep neural networks," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 8119–8127, 2018.

[51] A. C. Stickland and I. Murray, "Bert and pals: Projected attention layers for efficient adaptation in multi-task learning," *Proceedings of the 36th International Conference on Machine Learning*, 2019.

[52] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 7132–7141.

[53] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2980–2988.

[54] M. S. Sorower, "A literature survey on algorithms for multi-label learning," *Oregon State University, Corvallis*, vol. 18, pp. 1–25, 2010.