# Extracting Political Events from Text Using Syntax and Semantics

Andy Halterman*

5 October 2020

**Abstract**

Many questions in empirical political science concern the relations between political actors. Researchers have long used text as a source of data on political actors behaviors and relationships. Manually extracting this information from text is slow and expensive, while existing automated methods are inaccurate or limited to a small set of pre-defined actors and actions. This paper formalizes the process of event extraction and introduces a method for identifying the words in text that report who is doing what to whom, along with where, when, why, and as reported by whom. To do so, it draws on natural language processing tools that provide information about the syntactic information of a sentence and neural networks trained on a diverse set of hand-labeled text. The extracted actors and events can be analyzed with hand-constructed dictionaries or classifiers, or can be clustered to inductively find types of actors or behaviors. I compare the performance of the model with existing techniques on a corpus of text from the *Times of India*. I then apply it to State Department reporting on human rights, extracting 1 million events from the corpus and apply a clustering algorithm to learn categories of human rights abuses.

## Introduction

Many questions in empirical political science concern relations between political actors: *Who* lobbied *whom* in Congress (Kim 2017)? *How*, or with what tactics, do resistance movements oppose the state (Chenoweth and Stephan 2011)? Against *whom* are laws enforced (Holland 2015)? *Where* does political violence occur (Kalyvas 2006)? Text is a rich source of data for answering questions such as these, but most existing automated text analysis techniques, such as topic models, are not well suited to extracting this kind of information. The techniques that do exist for extracting information on "who did what to whom" from text rely on enormous investment in creating dictionaries (e.g. P. A. Schrodt and Gerner 1994; Raytheon BBN Technologies 2015; Norris, Schrodt, and Beieler 2017) or focus on linguistics research rather than applied social science (Gildea and Jurafsky 2002; Carreras and Màrquez 2005; Palmer, Gildea, and Xue 2010).

*Ph.D. Candidate, Department of Political Science, Massachusetts Institute of Technology. ahalt@mit.edu, https://andrewhalterman.com. Thank you to Fotini Christia, In Song Kim, Rich Nielsen, Blair Read, and seminar participants at MIT and UMass for comments on this paper. The ideas in this paper greatly benefited from conversations with Brendan O'Connor, Katie Keith, Sheikh Muhammad Sarwar, John Beieler, and Phil Schrodt. I gratefully acknowledge the support of a National Science Foundation Graduate Research Fellowship.

This paper has three contributions to aid applied researchers in extracting political events from text. First, it formalizes the process of event extraction. Second, it introduces a schema of event components that are shared across political events. Third, it introduces a new automated technique for extracting events from text that can be applied to new domains without retraining.

I propose a new set of standard properties, or pieces of information, that generalize across event types, consisting of actors who do an action, the action itself, the political entity receiving the action, and the means or instrument involved in the action, along with properties for the reported cause or reason for the event, any reporter or source attribution in the text, and the date and location of the event. To fill these event properties in practice, I introduce a technique that combines a rule-based system that uses the grammatical information of the sentence to identify spans of text that potentially correspond to each event property, and machine learning models trained on labeled spans to determine the correct event property for each span of words. This automated technique does not require dictionaries, can inductively learn event types from text, and can be applied to a wide range of documents. The method makes it possible for researchers to extract event information and then usefully aggregate similar events from their own text corpora with an easy to use software package.

The first step is to identify the words that correspond to various attributes of the event, such as the actor performing an action, the event's location, and who is receiving the action. I describe a new algorithm below that can perform this task and is available as an easily used software package. The technique draws on recent advances in computational linguistics to identify the grammatical components of each sentence, in a process closely related to diagramming sentences. The grammar of the sentence offers important clues about who is doing what to whom, but a system that uses only rules and grammatical information will not correctly handle complex sentences. Therefore, the event extraction also draws on machine learning models trained on newly labeled data to resolve ambiguous words into their correct event properties.

Finally, I demonstrate the utility of the new method to answer substantive questions in political science by returning to the ongoing debate on whether respect for human rights has improved over time. I produce new, disaggregated data on the specific acts of human rights abuses reported by the State Department in their monitoring documents over time. I offer clear evidence that the contents of reporting are changing over time, and suggestive evidence that the threshold for inclusion are changing as well. This application demonstrates the importance of creating new, tailored data to answer open questions in political science.

## Extracting events from text

Political *events* are actions undertaken by political actors at a particular time and place (Davenport and Ball 2002, 438). Producing data on political events thus consists of creating a structured representation of information on actions, the involved actors, and the manner, place, and time of the actions. A large number of datasets in political science consist of events that are hand extracted from text (see Table 1). A large literature in political science has concerned automated extracting political events from text (King and Lowe 2003; Schrodt and Gerner 2004; Hanna 2014; Beieler et al. 2016).

Automatically generating structured event data from text can be conceptually decomposed into three separate steps: retrieving relevant documents, identifiying event properties from the text, and aggregating or resolving

| Dataset | Citation | Text Sources |
|---|---|---|
| MIDS | Jones, Bremer, and Singer (1996) | diplomatic sources, histories, newspapers |
| CIRI human rights | Cingranelli and Richards (2004) | State Dept. reports |
| GTD | LaFree and Dugan (2007) | newswire, newspaper, gov. documents |
| Archigos | Goemans, Gleditsch, and Chiozza (2009) | Encyclopedias, newspapers |
| ACLED | Raleigh et al. (2010) | news text and humanitarian reporting |
| coups | Powell and Thyne (2011) | NYT and other text sources |
| SCAD | Salehyan et al. (2012) | AP and AFP |
| SIPRI arms transfers | | commercial publications, newspapers, gov. publications |
| UCDP GED | Sundberg and Melander (2013) | newspapers |
| SPEED "civil strife" | Nardulli, Althaus, and Hayes (2015) | NYT, BBC Monitoring, FBIS |
| regime type | Geddes, Wright, and Frantz (2014) | News reports, published literature |

**Table 1:** Many standard datasets in comparative politics and international relations are derived from text sources. Producing and updating them is often a multi-year undertaking that demands major resources. For instance, the first version of MIDS, the most highly cited of these datasets, cost around a quarter million dollars to produce.

event properties to useful categories. For example, an event extraction system could begin by subsetting a corpus to the documents containing the term "Syria", then identify in a particular sentence that an action is being undertaken by "Assad's forces", and finally resolve "Assad's forces" to a category of "Syrian military forces".

This three step conception of event extraction brings together tasks that people saw as separate (for example, keyword-based retrieval of documents and event coding) and splits up tasks that some existing systems treated together (for example, recognizing that words describe an action vs. classifying that action as a particular type.) I describe what belongs in each of these steps, along with the existing work on how to do each.

## Identifying documents of interest

The first task in event coding is identifying the set of relevant documents or sentences. In many cases, this task consists of using metadata, such as publication date and venue, to select the documents to be analyzed.

Some approaches to finding relevant documents are more sophisticated than simple metadata approaches. For example, Hillard, Purpura, and Wilkerson (2007) suggest using active learning to retrieve relevant documents for later coding. Sometimes, simply identifying the sentences or documents that contain an event is sufficient. For instance, Nielsen (2013) measures human rights abuses by counting the documents matching human rights-related search terms for each country, Beieler (2016) trains a neural set classifier to identify cooperative and conflictual events in sentences, and Halterman, Irvine, and Jabr (2019) use a classifier on multilingual word embeddings to identify protests in English and Arabic text.

## Identifying event properties

The second step in my categorization of event extraction is to identify the words in a document that provide information on the "properties" of an event, such as who did the event, to whom an action was done, or where an event occurred.

Formally, a document can contain multiple events, each of which as a set of properties, such the "agent" who did an action or the "recipient" that an action was done to:

- A corpus $\mathcal{X}$ is comprised of $D$ documents $X_1 ... X_D$.
- Each document $X_d$ is comprised of words: $X_d = \{x_1, ...x_{nd}\}$
- A document $d$ contains a set of $e_{jd} \in E_d$ events.
- $A\left(e_d, S = s\right)$ is the set of words within $X$ that correspond to event property $s$ for each event in the document $e_{jd}$.

In the example sentence "Trump fired missiles at Syria.",

$$\begin{aligned} A(e_d = \text{"fired"}, S = \textsc{agent}) &= \text{"Trump"} \\ A(e_d = \text{"fired"}, S = \textsc{recipient}) &= \text{"Syria"} \end{aligned} \tag{1}$$

In political science, the dominant approach to event coding relies on dictionary methods. The words that matching a list of actions are assigned to action property and words matching entries in an actor dictionary were coded as actors (P. A. Schrodt, Davis, and Weddle 1994; Schrodt 2009; Boschee et al. 2015; Norris, Schrodt, and Beieler 2017; Osorio and Reyes 2017; Brathwaite and Park 2018). In later systems, some grammatical information was used to constrain which parts of the sentence were compared to the action dictionary and which to the actor dictionary, and to identify the direction of the action.

Dictionary-based methods have several major limitations. They require enormous up-front investment, have very low recall between 5% and 35% (Makarov 2018; Althaus, Peyton, and Shalmon 2018), and are difficult to extend to new event types. Moreover, systems that depend on dictionaries cannot be used to learn new event types inductively from text, as the systems depend on dictionaries to identify which words correspond to which event properties.

A promising hybrid approach comes from O'Connor, Stewart, and Smith (2013), which uses dictionaries to identify actors (countries, in their case) and grammatical information from the dependency tree to fill the "action" property linking the two actors. A modified topic model that accounts for temporal dependency in dyadic relationships learns events inductively. This approach still depends on pre-built dictionaries, however.

A wide body of literature on slot filling and "semantic role labeling" exists in computer science and natural language processing, attempting to create systems that can faithfully reflect the tremendous variety of human language and human behavior. Early semantic role labeling approaches have highly variable "slots" or "frame elements" that differ by the recognized event type. FrameNet (Baker, Fillmore, and Lowe 1998), for instance, specifies around 1,000 linguistic "frames." Many of these slots are specific to the event type: a "cook food" event, for instance, might have a slot for "source of heat". Many event types, however, have slots that are roughly comparable: a "crime" event's "victim" slot is roughly comparable to a "hire" event's "worker". Slots can only be filled once the type of event has been recognized, making automated approaches to slot filling difficult. For political scientists, some of these frames involve potentially political actions such as a "revenge" frame, specifying the injured party, the victim, and the manner of revenge. Other frames are less interesting to political scientists: a "clothing" frame includes roles for garment, material, color descriptors, and wearer.

Palmer, Gildea, and Kingsbury (2005) developed a much more general approach to the task in the form of

PropBank, which replaces specific frame elements with more general, numbered arguments. These numbered arguments often correspond to the "agents" committing an action, "instruments" used in committing the act, and "patients" receiving the action, but the meaning of each numbered argument varies by the specific verb. This makes models trained on PropBank difficult to use in an applied political science setting as not all verbs are defined in PropBank.

This approach suffers from several drawbacks for applied information extraction work. First, these themes must be laboriously constructed by expert linguists, and their great level of specificity is aimed more at linguistic correctness than at practical usefulness (for example, great care is taken to distinguish bank deposits from alluvial silt deposits, or a "killer" role in murder from the "perpetrator" role in a kidnapping). Second, as with all hand constructed dictionary methods, it faces problems of low recall (Pavlick et al. 2015): if the precise term used the sentence has not been manually added to the dictionary, the dictionary method will miss it. Finally, the specificity of the slots makes the system difficult to train. A "victim" of a crime and a "beneficiary" of a gift both receive an action in some sense but FrameNet treats them as completely different entities.

## Aggregating

The third step in producing events consists of *aggregation*, putting like entities and actions together. In many cases, researchers have theoretically motivated categories in mind: researchers studying mobilization might be interested in protest events and would like to identify all instances of public demonstrations, regardless of if the demonstration is described as a "demonstration", "rally", "protest", or "political gathering". For political actors, researchers might interested in the actions of Kurdish armed forces in Syria, regardless of whether they are referred to in text as "the YPG", "Syrian Democratic Forces", or "a Kurdish miltia".

In existing work, the aggregation step is often performed with hand-built dictionaries (Norris, Schrodt, and Beieler 2017; Boschee 2016). For the event extraction system to correctly categorize a phrase from the text, it has to be listed in the dictionaries with the code or label it should resolve to. If the term is not listed in the dictionaries, for instance "SDF" instead of "Syrian Democratic Forces", the event coder will not correctly resolve the actor to its labels.

When researchers know how they would like to aggregate actors and actions, several other techniques are available. For instance, researchers can use supervised learning techniques to assign labels or categories to actions and actors based on labeled training data. For resolving named entities specifically, a set of techniques exist for resolving mentions of people or organizations to a fixed identifier like a Wikipedia page (Broscheit 2020; Hulst et al. 2020).

But researchers often wish to explore their texts inductively learn their contents.[1] Techniques that rely on dictionaries to recognize event properties in text cannot be used to inductively learn events from text, because they can only extract spans of text that have already been added to the dictionaries, rather than new actors or actions.[2]

[1] Credit to Molly Roberts for emphasizing the importance of exploring text and the need to have methods that allow researchers to do so.

[2] Petrarch2 has some limited ability to identify spans of text that appear grammatically in the sentence where actors would be but that do not match the actor dictionaries.

To enable exploration of extracted events and to let researchers quickly analyze events without the need to build dictionaries or statistical classifiers, I also introduce a short-document clustering algorithm that can learn categories of event types, described below. The clustering algorithm I propose uses pretrained embeddings to provide prior knowledge on word meaning and is designed to handle very short texts.

## A New Technique for Extracting Events from Text

My new technique for extracting events from text addresses several of the shortcomings of previous work. First, it introduces a new schema of event properties that balances between existing work in linguistics and the specific needs of political scientists. Second, it introduces a specific algorithm for extracting political events from text using the grammar of the sentence along with machine learning classifiers.

### A schema of political event properties

The properties that make up an event are not obvious. Existing event data methods in political science assume a small number of property that all events are expected to have: a "source" actor, an action, and a "target" actor (Gerner et al. 2002).[3] Some event schemata in computational linguistics are much more detailed and event-specific (see the crime example above, with its "victim" and "perpetrator" slots).

Building on theoretical insights by Dowty (1991) on "proto-agents" and "proto-patients", I propose an schema of event properities that consists of eight possible properities comprising an event.

1. An "agent" property that contains the actor doing the action. Grammatically, this slot will usually consist of subject nouns. In computational linguistics, this slot is usually referred to as the "sender" or "agent".
2. An "action" property that contains a description of the action that took place. Grammatically, this slot will contain at least one verb, but they also contain adverbs and other modifiers of the verb.
3. A "recipient" property that contains information about the actor that the action is being done to. Grammatically, this lot will involve direct objects object of prepositions or indirect objects. In computational linguistics this is referred to as the "receiver" or "patient" and in some political science approaches (e.g. Gerner et al. (2002)), the "target".
4. An "instrument" property, comprising the objects used by the actor in performing the action or the means by which an action is carried out. For instance, the italicized objects in the following sentences are instruments or means: deliver *aid*, fire *mortars*, disperse using *tear gas*. Grammatically, these are reported in direct objects, prepositional phrases, and indirect objects. These grammatical roles are the same as where the action's recipient is alo reported,
5. A "reason" property for the purported reason or cause of a reported event in political text. For instance, the italicized span in "arrested two people *for participating in last week's protests*" does not provide information about the event itself, but rather context for the event.
6. A "time" property, with information on when the events took place.
7. A "location" property, with information on where the events took place.

[3]Many systems conceptually include a location property, but techniques for properly filling location properties are still under development. See Halterman (2019).

8. A "reporter" property, with information on what source reported the occurrence of the event, such as "*according to local sources*."

At a minimum, an event must have an action and at least one actor or recipient, but other properties are optional and sentences reporting all eight pieces of information are uncommon. Conceptualizing events in this way has several advantages over existing approaches in political science. First, it decomposes the previous "event" property into more granular "action" and "instrument" properties. Actions and instruments are grammatically quite distinct, and splitting them up will help automated systems to fill these properties from real sentences. Providing both an "instrument" and "recipient" property also clarifies the challenge of distinguishing (in)direct objects that receive actions and those that are involved in the commission of actions. This "direct object" problem is discussed at length below.

Finally, "reason" and "reporter" properties are useful for separating out parts of the sentence that provide important contextual information for the event, rather than themselves be coded as separate events. In this sense, they can be treated as "nuisance" properties that may not be of direct interest, but need to be accounted for to ensure that only interested or relevant events are extracted.

## Finding event properties in text

Given a definition of event properties, the next task becomes creating a system to fill these properties from real sentences. I make two simplifying assumptions about textual descriptions events in order to make my event extraction system feasible. First, I assume that all of the information about an event is contained within a single sentence. This assumption is made by existing dictionary-based event coders in political science, though recent work in natural language processing is developing techniques for cross-sentence event extraction (e.g. Ebner et al. 2019). This assumption makes it possible to use the grammatical structure of the sentence to simplify event extraction, as I describe below.

Second, I assume that all events are anchored on a verb and that each verb is connected to at most one event. (Verbs without events can be nominalizations like "is" or in sentences not describing political actors doing actions: "the legislation *concerns* food packaging regulation").

This implies the following setup for a corpus, sentences, events, verbs, and properties:

- A corpus $\mathcal{X}$ is comprised of $D$ sentences $X_1 ... X_D$.
- Each sentence $X_d$ is comprised of words: $X_d = \{x_1, ...x_{nd}\}$
- Each of $e_{jd} \in E_d$ events in sentence $d$ is anchored on one verb $v_{jd} \in X_d$, where the verb is not connected to any other event.
- $A\left(v_d, S = s\right)$ is the set of words within $X$ that correspond to event property $s$ for verb $v \in V_d$.

Thus, in an example sentence, "Trump fired missiles at Syria.",

$$\begin{aligned} A(v_d = \text{"fired"}, S = \textsc{agent}) &= \text{"Trump"} \\ A(v_d = \text{"fired"}, S = \textsc{recipient}) &= \text{"Syria"} \end{aligned} \tag{2}$$

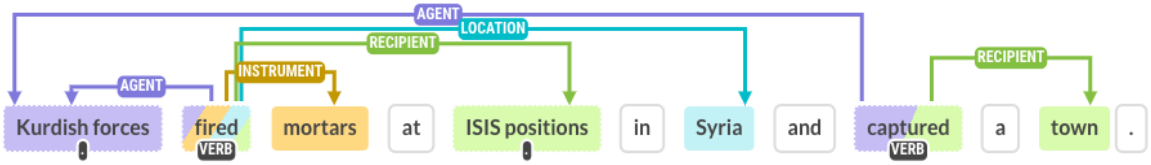


**Figure 1:** Example sentence annotated with event properties. Note that the event properties are anchored on a specific verb: the recipient for "captured" is not the same as for "fired".

Figure 8 shows a complete annotation for a single sentence. Note that the sentence contains two events, each anchored on a verb, with different recipients for the two events.

In practice, to identify the words that correspond to each event property, I draw on two insights. The first insight is that much of the information needed to identify which words correspond to each event property is encoded in the dependency parse of the sentence, which is a grammatical representation of the sentence that encodes not only the part of speech information for a word, e.g. whether it is a noun, verb, adjective, and so on, but also the relationship between words, where a noun could be either the subject noun or the direct object of a verb. Figure 2 shows an example of dependency parses, along with an example of their limitations.

Previous event extraction systems in computer science (Rudinger and Van Durme 2014; White et al. 2016) and political science (Van Atteveldt et al. 2017) have used rule-based systems and dependency parses to extract events from text. For instance, Van Atteveldt et al. (2017) identify subject nouns as the agents of an event and the entire collection of verb, direct objects, indirect objects, and prepositional phrases as a single "predicate".

Using a rule-based system is appealing, as it does not require the large amounts of expensive training data that machine learning systems require and produce easily explained coding decisions for each sentence. However, using dependency parses on their own has a major limitation, namely that they cannot reliably distinguish between the instruments and recipients of an action, as both can appear as direct objects to an event's verb. Van Atteveldt et al. (2017) use as an example sentence, "Hospital officials in Gaza said that 390 people were killed by Israeli fighter planes." Their method returns "390 people killed" as a single predicate span, rather than separating out "killed" as an action and "390 people" as a recipient of that action.

However, distinguishing between political actors and other objects is crucial for political science applications. Identifying which nouns are recipients and which nouns are instruments requires substantive knowledge to distinguish them. This requirement for substantive knowledge is part of the explanation for why existing event extraction systems from computer science have not been useful for political scientists. A purely syntactic representation of a sentence cannot distinguish between, for instance, a direct object being an instrument of an action ("missiles") and a direct object being an actor receiving the action ("Tillerson"). In contrast, semantic analysis of words provides information about whether words are likely to describe people, actions, weapons, locations, and so on, but cannot link these words together into the meaningful relations encoded in text. My model uses both syntactic and semantic information to fill an event's properties.

Concretely, the event extraction algorithm proceeds in three steps (Figure 3 presents an overview of the steps). It begins with a pre-processing step, creating a grammatical dependency parse of each sentence using existing natural language processing software (see Figure 2 for an explanation of dependency parsing). Next, it uses

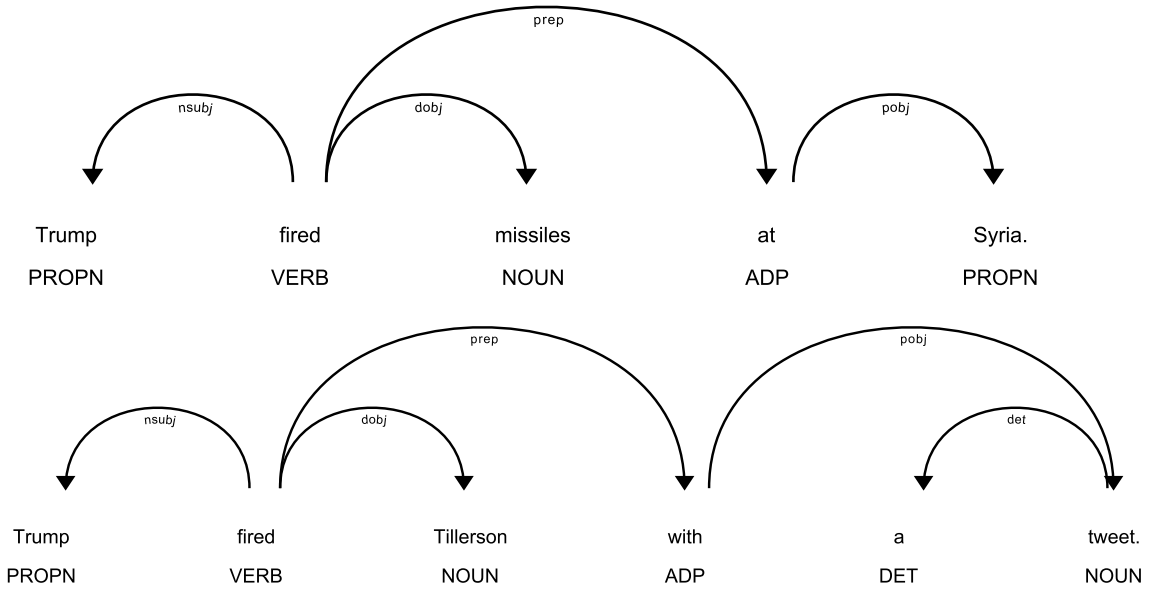


**Figure 2:** A dependency parse representation of two sentences. Dependency parses are a particular way of representing the grammar of a sentence, roughly analogously to a sentence diagram. Dependency parses encode not only the part-of-speech of a word (reported in all-caps below each word) but also the grammatical relationships between words (”Trump” is a noun, but is specifically the noun that fills the ”subject noun” role for the verb ”fired”). (Note the error made by the automated parsing system in labeling ”Tillerson” as a noun instead of a proper noun.) Part-of-speech tags are invariant to the grammar of the sentence: ”missiles” is a noun, but across sentences it could play the role of subject noun, direct object, dative object, or object of a preposition. The sentences are nearly identical in their grammatical structure, but the grammatical parts of the sentences correspond to different event properties, illustrating the need for semantic information about the words as well. In the first example, the direct object in the sentence plays the role of an ”instrument” of the action, while in the second sentence, the direct object plays the role of the recipient of the action.

hand-specified rules and the dependency parse to segment the sentence into rough spans. Finally it uses a machine learning model to determine whether objects are instruments/means or recipients of the action and separate models to locate reporter and reason spans.

It uses the automatically-recognized dependency structure of a sentence (Honnibal and Montani 2017). Dependency parses encode the grammatical relationships between words in a sentence in a directed tree. For example, a verb (“fired”) could be connected to its subject noun (“Trump”) and its direct object (“Tillerson”). I generate a set of deterministic rules on this tree to produce candidate spans for the actor, action, recipient, instrument, location, date, and reporter properties for each event.[4]

To resolve the ambiguity in instruments and recipients, I train a convolutional neural network (CNN) classifier on a new set of labeled data to classify noun phrases either as recipients or instruments of actions. Rather than treating words as unique tokens, I represent them using pretrained embeddings, as is standard in text classification in computer science. Word embeddings represent words as points in low-dimensional vector space, such that words used in similar context will be nearby in vector space. Replacing words with their embeddings meaning that the model can easily classify new words it did not see during training. CNNs preserve the order

[4]The model to detect “reason” spans is not yet included in the code. More labeled data is required to produce a model with good accuracy.

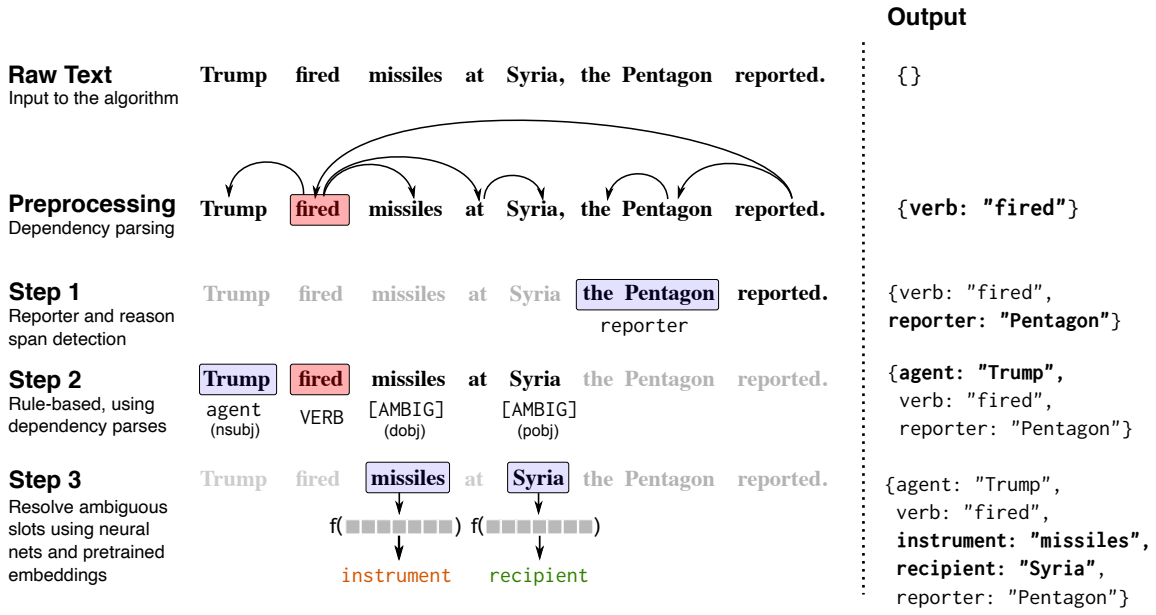


**Figure 3:** Algorithm for extracting politically relevant event spans from text. The output from each step is reported in the right column. Sentences are preprocessed with a dependency parser, then "reporter" and "reason" spans are detected with a word-level convolutional neural network (Step 1). "Actors" and "actions" are detected using rules applied to the dependency parse of the remaining text, along with spans that could either be instruments or recipients of the action (Step 2). A neural net classifier is applied to these ambiguous spans' embeddings to determine their role (Step 3). Dates and locations are detected using NER (not shown in this example).

of the words in the input, and thus can learn which pairs or sequences of words are especially informative (Kim 2014).[5]

To train the classifier, I create a dataset of "candidate" actors, consisting of spans of text that syntactically may be actors, but semantically may be instruments of actions. I manually labeled 2,000 of these spans, drawn from newspaper, newswire, Wikipedia, and government reports to give good cross-domain performance. The convolutional neural network that I fit uses pretrained embeddings as inputs. Each convolution in the network is applied to a window of three words at once, meaning that the model can learn trigram information. The model stacks several convolutional layers to learn wider relationships between words. The model achieves 81% accuracy and 83% F1 (see Table 2). In production, phrases that are recognized as recipients are then removed from the predicate and placed in the recipient property. See Figure 3.

I also train a "reporter" model that recognizes phrases with a sentence that provide a source attribution for the event, such as "…, Amnesty International reported." These phrases are then removed from the sentence, preventing them from being coded as extra events, and allowing them to be added as metadata to the extracted events. The reporter recognition task is similar to named entity recognition tasks, so I use a multilayer convolutional neural network that uses pretrained word embeddings and that performs well on named entity recognition tasks (Honnibal and Montani 2017). The reporter model achieves 78% accuracy.

[5]Moreover, because the model needs to learn only short sequences of words, rather than long-range dependencies, a CNN can suffice, without the need for a recurrent neural network and the computational cost in imposes.

| Classifier type | N | F1 | Accuracy |
|---|---|---|---|
| Recipient vs. instrument | 1,965 | 0.83 | 0.81 |
| Reporter | 912 | - | 0.78 |
| Contains event | 749 | 0.88 | 0.85 |

**Table 2:** Out-of-sample accuracy results for the three machine learning models used in the event extraction model. The recipient vs. instrument model is a convolutional neural network applied to short spans of text, represented as pre-trained word embeddings. The reporter model is a different convolutional neural network that takes in an entire sentence treats the task as named entity recognition task. The event detection model also operates on an entire sentence and returns a prediction for whether a sentence contains an event. All three models were trained on a wide diversity of English language sources, drawn from AP, Syrian paper, Wikipedia, State Department human rights reports, and Political Instability aatrocity reports. The models should generalize well to similar kinds of text, meaning that researchers will not need to retrain these models unless their text comes from a very different domain.

Van Atteveldt et al. (2017) develop a model for recognizing "sources" [reporters] that uses a set of hand-specified rules.[6] The advantage of using a machine learning model over a rule-based system is that machine learning models often higher recall on actual production text. Information on dates and locations is easily extracted using off-the-shelf named entity recognition. A more sophisticated approach to linking actions and the most specific locations where they are reported to occur is described in Halterman (2019) and could be easily incorporated into the algorithm.

Finally, not all text describes events, but the event extraction algorithm will attempt to fill event properties for each piece of text it recieves. Often, researchers have text they expect will contain events, such as the lead sentences of newswire reports and can ignore the non-event sentences that happen to come through. A better approach, however, is to pre-filter the sentences to restrict the input to sentences that are likely to have events. To filter out non-event containing sentences, I provide a sentence-based classifier that removes sentences that are unlikely to have events. The model uses a convolutional neural network trained on around 500 sentences and achieves accuracy of around 0.85 (see Table 2).

## Aggregating actions

Once spans are extracted from text, they still require more work before they can be analyzed. Specifically, spans of text that describe the same action need to grouped together.[7] In some cases, researchers will only be interested in a small, pre-identified set of behaviors, in which case they can use supervised learning techniques to identify the subset of actions they would like to label. In other cases, however, researchers may want to inductively learn clusters of actions from the text, either as exploratory research or to test hypotheses about the behaviors that actors engage in.[8]

Unsupervised text analysis, specifically latent Dirichlet allocation (Blei, Ng, and Jordan 2003) and its variants (Blei and Lafferty 2007; Roberts et al. 2013) is a mainstay of empirical political science. To cluster extremely

[6] I use the term "reporter" instead of "source" to avoid confusion with the terminology used in the standard political science ontology, CAMEO, where "source" is often used where I use the term "actor". (Gerner et al. 2002)

[7] I focus here on actions, rather than actors or recievers because the appropriate groupings of political actors are generally easy to specific a priori than the best groupings of actions, and because in practice, grouping actors is fairly straightforward using dictionary methods.

[8] A hybrid approach combines unsupervised clustering with the small number of human analyst decisions. For example, Ritter et al. (2015) propose a weakly supervised model for recognizing events, that require analysts to only specify small number of positive documents of interest. A semi-supervised approach to learning event categories is promising but is left for future work.

short spans of text, including those as short as a single word, I introduce a method to perform very short document clustering that draws on word embeddings to provide prior information about word meaning and a latent variable interpretation of a document embedding technique.

In situations such as this one, where "documents" are in fact short spans that can be as short as a single word, the matrix of word–document counts will be extremely sparse. In situations where documents have only one word, their representations in the document–word count matrix will be completely orthogonal, making it difficult to learn a low-rank approximation using LDA. More heuristically, LDA uses the co-occurance of related words in long documents to learn high-quality topics. In very short spans, synonyms are very unlikely to co-occur: the use of a word almost precludes the use of a close synonym in a span of 1–10 words.

An alternative technique is to use pretrained word embeddings to provide prior information on the similarity of words. Word embeddings are a technique for learning dense, low-dimensional vector representations of words based on their context in large corpora or text.

word2vec is already commonly used as a replacement for LDA is political science applications (see, e.g. Kornilova, Argyle, and Eidelman 2018; Spirling and Rodriguez 2019; Rheault and Cochrane 2019; Lauretig 2019 or recent conference programs from Text as Data or PolMeth.) Words that are used in similar contexts in a corpus will have similar vector representations, allowing researchers to learn how words are being used, for example, by different parties or in a way that changes over time. The technique I propose here does not learn new embeddings. Instead, it uses embeddings that have been pretrained on a large corpus of text to provide prior information about the (similar) meanings of words: very roughly, pretrained embeddings provide an approximation of $\beta$, the word–topic distribution. The model, like a human reader, thus comes to a corpus with a sense that "arrest" is more similar to "detain" than to "France."[9]] The next step is to learn an equivalent to $\theta_d$, the topic present in each document.

## Representing documents

Word embeddings provide a representation of *words*, but do not provide an obvious technique for representing *documents*. A standard technique is to produce a document vector by averaging the embeddings of all words in a document, and sometimes by appending the elementwise maximum to that vector (Goldberg 2017). Simple averaging treats all words as equally informative and causes documents to appear more alike as they increase in length. A more sophisticated approach is to learn a document embedding alongside word embeddings ("doc2vec") (Le and Mikolov 2014), but this approach requires a separate step to learn paragraph embeddings for new documents at test time and the resulting document vectors are not easily interpretable.

Instead, I adopt a sentence embedding model proposed by Arora, Liang, and Ma (2017). Their model is theoretically motivated, simple to implement, and achieves very good performance on sentence classification tasks, beating even sophisticated supervised sentence classification models. In previous work Arora et al. (2016) show that embeddings models such as word2vec and GloVe (Pennington, Socher, and Manning 2014) can be interpreted in a generative model: words in a span of text are emitted with a probability given by the word's

[9]In this sense, word embeddings are being used here as a form of transfer learning, in which a representation learned on one corpus or task is applied to another to improve performance. Recent improvements in transfer learning for natural language processing are producing rapid improvements in the field. See Howard and Ruder (2018); Peters et al. (2018); Devlin et al. (2018). Ruder (2018) provides a non-technical overview.

distance from a latent "discourse vector", which conducts a random walk through embedding space as words are emitted. They offer a simple interpretation of this discourse vector: "Its coordinates represent what is being talked about." (Arora et al. 2016, 387).

Arora, Liang, and Ma (2017) propose a sentence embedding technique the approximates the maximum likelihood estimates of this discourse vector. Specifically, each sentence embedding is initially represented using a smoothed, weighted elementwise mean of its constituent words' embeddings:

$$\tilde{v}_i = \frac{1}{|d|} \sum_{w \in d} \frac{a}{a + p(w)} v_w, \tag{3}$$

where $v_w$ is the pretrained word embedding of word $w$, $p(w)$ is the empirical frequency of word $w$ in a large corpus, and $a = 0.0001$ is a smoothing hyperparameter. This weighting approximates the standard tf-idf weighting scheme in traditional text analysis and information retrieval, hence the name "smooth inverse frequency" (SIF). Next, the sentence vectors then have a "common component" removed, in which the first singular vector of all the vectors in the corpus are removed from each: $v_i = \tilde{v}_i - uu^T \tilde{v}_i$, where $u$ is the first singular vector of the matrix $X$ of all $\tilde{v}_i$. This sentence embedding techniques has two nice properties: it generates a fixed size embedding for a short document, in a way that is theoretically motivated and preserves information in the document as well as more sophisticated task-specific representations.

I modify the original SIF sentence embedding model to improve its applicability to this specific domain. Specifically, I vary word weights by their part-of-speech, in addition to by their word frequency. Because I focus on the specific domain of actions in the clustering algorithm, and because verbs are generally the most important component, I give their embeddings full weight regardless of their empirical frequency. Auxiliary words, digits, and proper nouns, in contrast, are reduced in importance to have the embedding over-weight rare but uninteresting words and to avoid overfitting downstream.

To infer clusters of actors and actions, I then apply a simple k-means clustering algorithm to the SIF embedding. Because the spans being clustered are so short, it is reasonable to treat them as belonging to a single cluster, rather than the mixture of clusters that a model like LDA or a Gaussian mixture model assumes, though researchers could use a clustering algorithm of their choice.

## Reusability of the method

The techniques I introduce for filling event properties and aggregating actions are applicable across a wide range of domains and types of text and my software package can be easily applied to new domains. The training data used to train the machine learning components of the event extraction model is drawn from a number of sources and should work without the need for re-training if researchers are using text that looks similar to standard newspaper or encyclopedia articles. If the types of text that researchers are using are substantially different in domain, they may need to add some training data to the machine learning classifier.

In the abstract, the method is agnostic with respect to language. Dependency parses are desiged to apply a universal grammatical structure to languages, meaning that the rule-based components of the model will

work as-is on languages that have accurate dependency parsers available.[10] Pre-trained word embeddings are available for around 200 languages. The machine learning classifiers would need to be retrained for new languages, but this entails only several hours of labeling.

The end-to-end pipeline to produce events is available as easy to use Python code. The event extraction model does not require users to change parameters, so documents can converted to structured events in a single command. The aggregation step, if performed using the clustering technique I propose, only requires users to specify the number of clusters.

## Changing respect and changing reporting for global human rights

An ongoing debate in international relations and comparative politics concerns whether respect for human rights has changed over time. Many observers expect, on anecdotal or qualitative grounds, that the global human rights situation has improved since the 1970s. In contrast, the major datasets of respect for human rights, including the CIRI Human Rights Dataset (Cingranelli and Richards 2004) and the Political Terror Scale (Wood and Gibney 2010) dataset show a fairly constant level of human rights violations over the past four decades.

Fariss (2014, 2018) argues that this counterinuitive finding is the product of changes in how human rights violations are reported. As NGOs gain greater access and human rights observers have better information, a greater proportion of human rights violations will be recorded than in the past. If the probability of detecting human rights violations is increasing faster than the overall rate of actual violations is decreasing, we will observe an apparent increase in human rights violations. Similarly, as the human rights record improves in different countries, human rights activists are likely to change the focus of their activism to other, less egregious violations.

Fariss (2014) models this change using a dynamic IRT model, using incidents of genocide as a perfectly observed anchoring observation to estimate the probability of incident reporting. He distinctions between what he calls "event" and "standards"-based reporting, with "events" like genocide being more accurately measured than the "standards" that the State Department and Amnesty International measure because the definition of events changes less than the definition of standards and because data on events is updated retrospectively as better information becomes available.

Fariss' paper makes an important contribution to the debate in positing the existence and mechanisms of changing reporting standards. The model that it uses, however, rests on several major assumptions, the greatest of which is that all state repression, from arbitrary arrest to genocide, exists along a single latent space, meaning that values can be compared across them. Instead, we might believe that genocide is simply different from other violations of human rights, violating the assumption of the unidimensional latent variable. D. Cingranelli and Filippov (2018a) and D. Cingranelli and Filippov (2018b) dispute this finding, largely on objections to Fariss's IRT model.

Rather than relying on the same limited set of country-year ratings to measure human rights respect and

[10] In general, these are languages from rich or populous countries (e.g. French or Chinese) or languages of particular national security interest to the United States government (e.g. Arabic, Farsi).

the changing standard of human rights violations, I instead generate new data on respect for human rights by returning to the original State department text used to create the country year ratings. Other researchers (Greene, Park, and Colaresi 2019) have also begun looking directly at the text, but in ways that do not preserve the relationships between actors and actions in the text. This allows us to produce fine-grained data on actions and the ability to link those actions to government actors.

I applied both steps of my new method to the State Department's annual country human rights reports from 1977 until 1999, when the format of the documents changed. These reports are mandated by Congress and draw on information collected by US embassies and officials in Washington. The text of each report is narrative, but divided into sections that cover a range of human rights practices. From this text, the event extraction model produced 1.02 million events. Because this debate is over government respect for human rights, I then subset the events to only those in which the extracted actor span contained terms in a list of terms that I specified. This list included all country names and demonyms, along with terms describing government officials, such as "soldier", "authorities", "police", or "government". Approximately one quarter of the total events, 243,449, had actor spans that included these words. The data I produce is thus a compromise between between what Fariss calls "standards-based" reporting and event reporting. Rather than producing a single country or score as in the standard approach I produce a set of disaggregated events. Unlike codings of genocide, however, these machine extracted events are not updated retroactively as better data becomes available.

I then fit the SIF/$k$-means clustering algorithm to these extracted spans. I fit the model using $k = 60$ clusters, after experimenting with several values of $k$. Many of the topics are quite specific and contain only a small number of spans. A small number of topics together contain the majority of spans, which may be better modeled by an even larger number of topics.[11]

## Empirical Results

As Fariss observes, the total amount of reporting, measured by the number of words, has increased over time. Figure 4 shows that the number of reported events is gone up as well, from approximately 2,500 per year to around 25,000 per year. On its own, this figure offers some suggestive evidence that the standard of reporting has changed. We may believe that human rights practices are stagnant or perhaps even slightly worsening, but we do not believe that human rights violations have become an order of magnitude more common.

More interestingly, the data suggests that the nature of reporting is changing, becoming more focused on specific events over time. After normalizing by the length of documents, the number of events reported as increased. From 1979 to 1999, the number of events has gone from 10 to 16 events per 1,000 words. This indicates at least higher specificity in the content of the reports. Thee proportion of events with government actors, however, remains steady between 22% to 25% over the period.

Examining the clustered events by country offers further validation of the method. Figure 5 shows the count of events for a selection of countries within each human-intepretable cluster. Only clusters with clear unifying themes in a set of randomly selected documents were labeled. Labels were applied to the clusters without

[11]This follows guidance made by Brandon Stewart when presenting comments as a discussant at PolMeth 2019. He argues that most topic models are run with too few topics and that a good approach to topic modeling is to fit it with many topics and combine similar topics afterward.

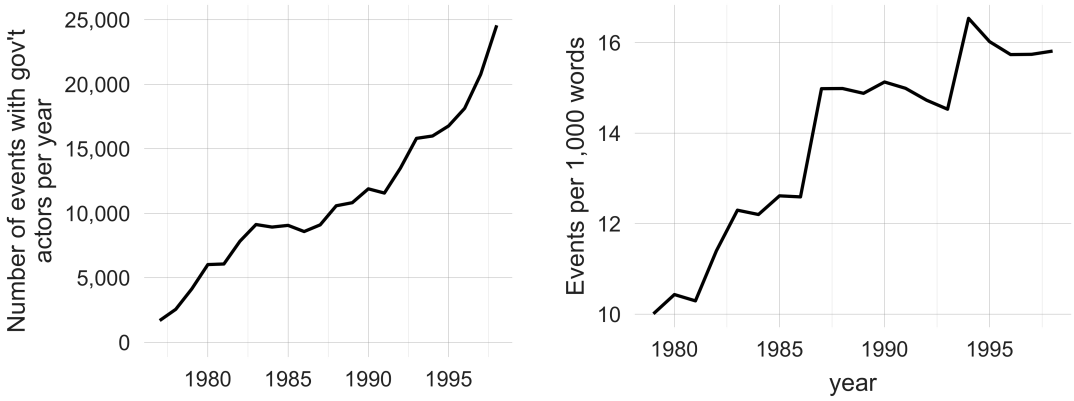


**Figure 4:** The number of extracted events from the State Department annual human rights reports with government actors (left) and the number of events per 1000 words per year. Events are extracted using the method introduced in the paper. Events are limited to those with an extacted actor that matches a government keyword (e.g. "police" or the name of a country). The results indicate both an overall increase in reporting and an increasing density and specificity of reporting.

reference to country information to prevent contamination. The results match our qualitative sense of human rights in the 1980s and 1990s: Colombia, Egypt, India, and Zimbabwe were quite poor, East Germany (GDR) had a mixed record, and Iceland and Norway were exemplery, free from state-sponsored violence and killing and torture.

Figure 6 provides a further validation of the approach. The figure shows the coefficients from a linear regression of the CIRI physical integrity score for a country-year on the count of extracted events of each type for that country-year. The cluster labels were assigned by examining randomly selected documents from each cluster and were not modified after comparing them to the CIRI human rights scores. The "NA" label corresponds to clusters that did not appear to have a distinct meaning. "Positive" topics are associated with higher (better) scores on the human-coded CIRI scale, while reports of events in the negative cluster, especially the cluster consisting of descriptions of torture and extrajudicial killing, are associated with much lower (worse) human rights scores. This provides some validation that the automated technique is recovering expert human judgement.

Examining the human rights reporting corpus directly demonstrates some of the advantages of this method. First, we can directly measure what actors are *doing*, as opposed to just the words used in the documents, as previous methods are limited to (Greene, Park, and Colaresi 2019). This allows us to more precisely measure the outcome of interest: human rights abuses. Second, because the new technique does not rely on dictionaries or pre-specified categories of behavior, we can inductively learn the types of behavior that government actors are reportedly engaged in. Doing so allows us to find types of reports that we might not have expected beforehand, such as the set of actions involving economic reforms.

Modeling the relationship between extracted events and the top-line CIRI scores lets us directly test whether the process by which State Department reports are converted to human rights scores in changing. Figure 7 shows the results of a regression of CIRI score on year, the count of all events, and an interaction between year and a specific event type of interest. On the left is the time-varying effect of a marginal event from the torture

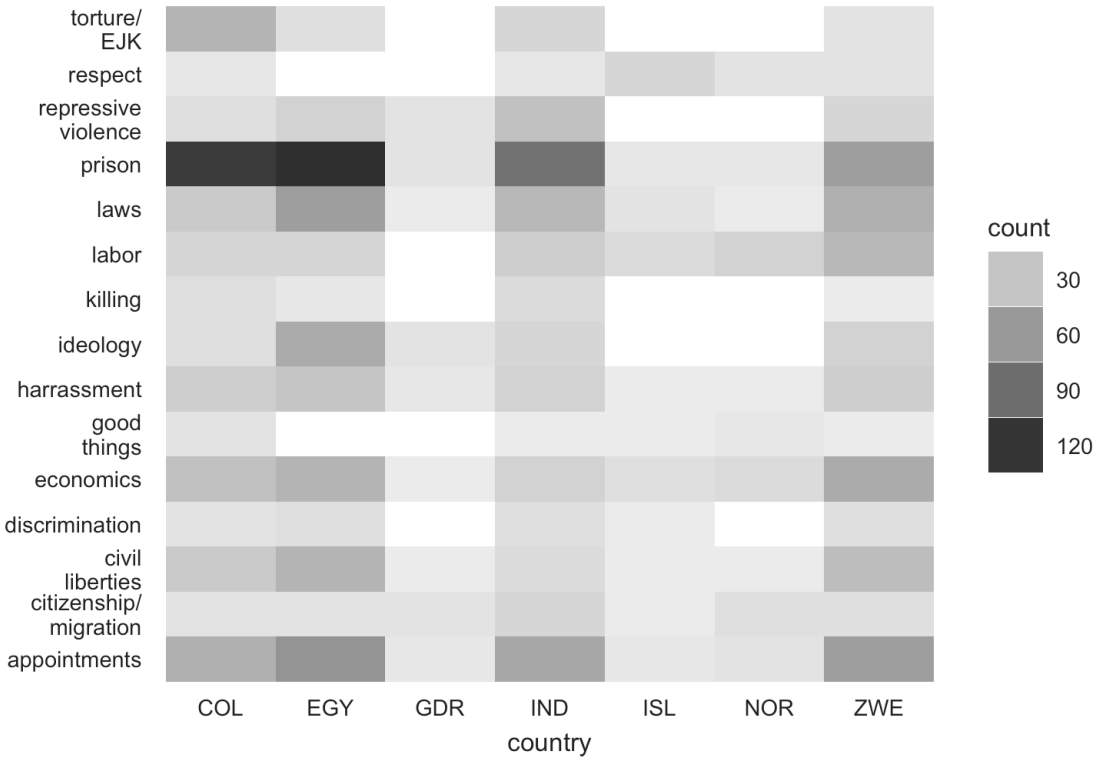


**Figure 5:** Counts of events per human-interpretable cluster per country.

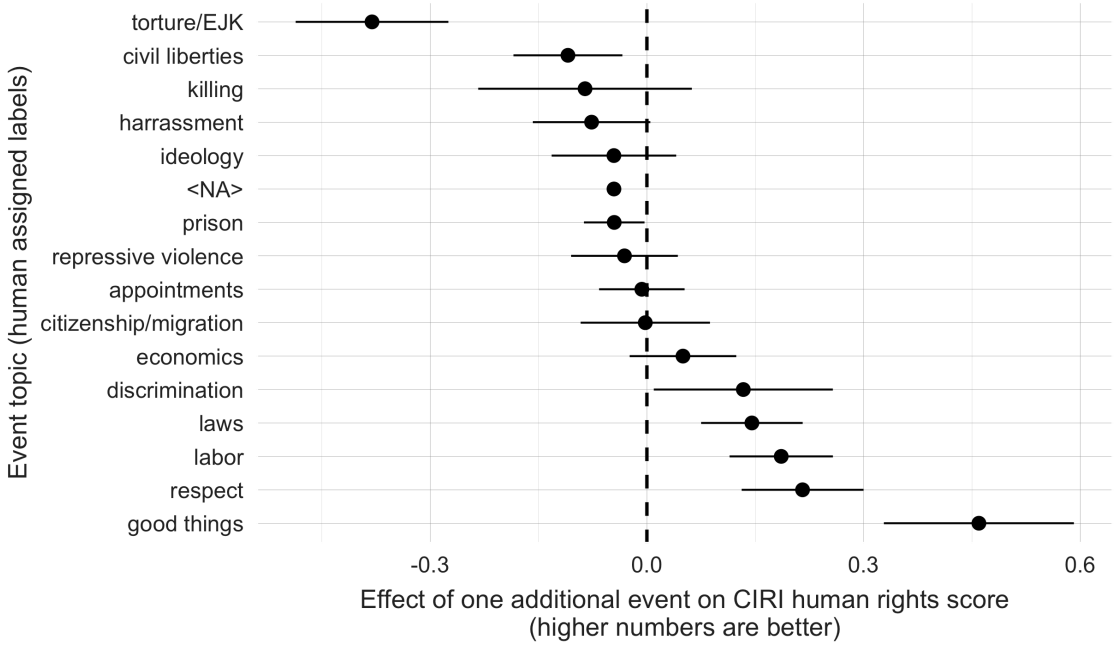


**Figure 6:** Regression of country-year CIRI scores on counts of extracted events by clustered action type. The CIRI physical integrity score ranges from 0 (no respect) to 8 (full respect). Positive coefficients for an event type thus indicate that that observing an event from that type increases the predicted level of respect for human rights. `<NA>` is all clusters that were not easily labeled. "Civil liberties" and "discrimination/harrassment" are amalgams of both respect and violations. "EJK"=extrajudicial killing.

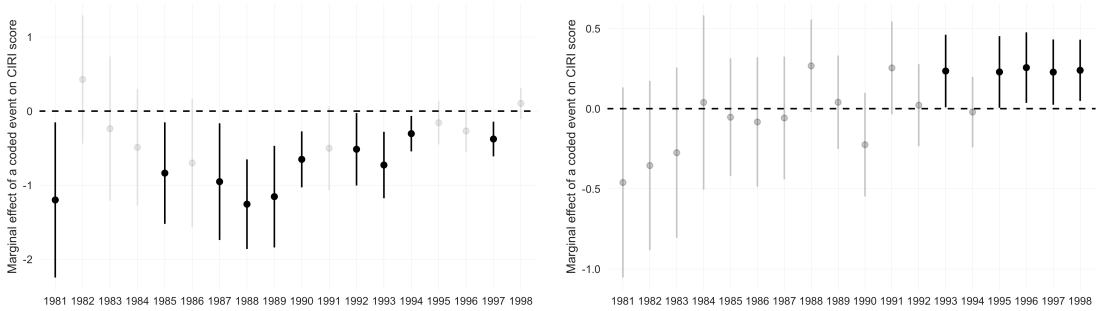


**Figure 7:** Regression of country-year CIRI scores on year, the count of all events, and an interaction between year and a specific event type of interest. On the left is the per-year effect of a marginal torture or extrajudicial killing event on CIRI score, while on the right is the effect of an event from the "labor" cluster. Higher scores indicate greater respect for human rights.

or extrajudicial killing cluster, while on the right is the time-varying effect of an event from the "labor" cluster. The figure on the left shows that the effect of a torture or extrajudicial killing event is stable over time: the interaction term is negative and consistently significant, revealing that, as expected, events of this type predict worse human rights scores. On the right, however, we see that the effect of events from the "labor" cluster change in importance over time. Beginning in the 1990s, reports from this clsuter improve the CIRI physical integrity score, providing evidence that the process of coding human rights respect from State Department documents is changing over time. Altogether, this application shows how extracting events from documents, rather than modeling documents as a whole, can provide a more detailed picture of the respect for human rights, and can provide support to the argument that how coders rate human rights respect is changing over time.

## Communal Violence and Police Response in India[12]

Wilkinson (2006) argues that *whether* police respond to communal violence in India determines how deadly it becomes. He draws on hand coded event data on Hindu-Muslim violence in India from the *Times of India* (Varshney and Wilkinson 2006) that reports whether police responded to an instance of communal violence. Because Varshney and Wilkinson (2006) were limited by their manual process in how much data they could collect, they do not report details on the actions that police forces took.

As a second application of my method, I create new data from the *Times of India* on *how* security forces respond to communal violence in Gujarat, India in 2002. I scraped the *Times of India* archive from 2002 and limited the corpus to a set of 8,600 articles that contained communal violence keywords. From these 8,600 documents, the event extraction model produces 222,000 events, averaging around 25 events per document. Because I was specifically interested in actions undertaken by police, I filtered the extracted events to only consider events with agents that matched a police keyword. This produced events like the ones below:

```
\{agent: "the task force, rapid action force, and the local police",
verb: "have increased",
instrument: "the patrolling"\}
\smallskip

\{agent: "the small posse of policemen",
   verb: "failed"
 instrument: "utterly to prevent the violence"\}

 \{agent: "the police, which had remained inactive initially,"
 verb: "beat up",
 recipient: "journalists and others"\}
```

Clustering the extracted actions and instruments using SIF embeddings (Arora et al. 2017) and k-means pro-

[12] This section draws on joint work with Katie Keith and Sheikh Muhammad Sarwar, UMass Computer Science.

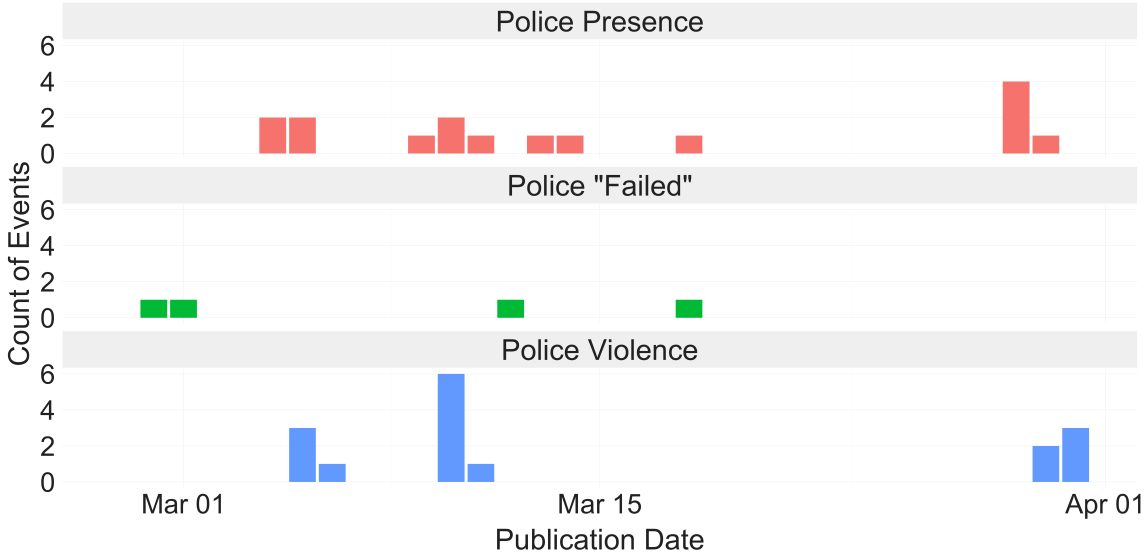


**Figure 8:** Example sentence annotated with event properties. Note that the event properties are anchored on a specific verb: the recipient for "captured" is not the same as for "fired".

duces the results seen in Figure 8. The findings reveal some heterogeneity in how police respond to communal violence in Gujarat, India beginning on 27 February. Initial police events consist of police *arresting* or *"failing to" act*. A week later, police engage in much more *patrolling*, *shooting* and other forms of violence. Moreover, the picture is very different from the one that emerges from the standard existing tool for automated event extraction, Petrarch2 (Norris, Schrodt, and Beieler 2017; Beieler et al. 2016). Petrarch2 is a dictionary-based coder, meaning that it is only able to recognize events in text if the descriptions of those events match a pre-specified set of phrases for actors and actions. Out of the set of documents from 2002 matching communal violence keywords, Petrarch2 identifies only 369 events with police actors, in contrast to the 1,900 identified by the method I propose here. More worryingly, it only includes one event done by police from the crucial period in March, and that event does not relate to the violence in Gujarat.[13]

By finding a wider set of actions undertaken by police during this period, we are better able to study how they responded to communal violence than if we had used existing methods. This application also illustrates key difference between existing dictionary-based methods and the proposed approach: dictionary methods only identify pre-specified phrases in text, while this method allows researchers to find event types and actors they might not have expected to find. By producing disaggregated data on specific kinds of police behavior, not just whether police were present, we can better understand the changing nature of the police response and the differential effects of different police responses.

[13]"19 held over beach clashes," https://timesofindia.indiatimes.com/city/thiruvananthapuram/19-held-over-beach-clashes/articleshowprint/4390684.cms

## Conclusion

This paper introduces a new method for researchers to extract political events from text. An event extraction model uses grammatical information and new machine learning models to identify the parts of a sentence corresponding to the different properties of an event. It does so with much finer resolution than previous grammar-based event extraction models, and with far greater coverage than dictionary based methods. A clustering algorithm takes these short spans and aggregates them into useful categories for further analysis. It overcomes the short document problem by using prior information in the form of word embeddings, a theoretically motivated document embedding scheme, and $k$-means clustering to learn useful aggregations of events.

I apply the model to an open question in international politics, about whether respect for human rights as improved overtime. I produce new disaggregated data on human rights related events with government actors and offer some evidence for the arguments that the standard of reporting has changed over time. While the volume of human rights reporting has increased greatly over time, specific kinds of rights violations have changed in their overall proportion of reporting. Second, I illustrate the technique's ability to disaggregate data on police behavior in 2002 India, learning types of behavior that police were engaged in, rather than a simple binary indicator for whether they were present. Because the model is general it can be applied to a wide range of questions in political science, anywhere information on the behavior of actors is important.

As in the rest of science, the availability of new data is often the precipitating cause of new research and improved understanding. Automating some production of structured data from text would allow more project-specific creation of data, leading to better measurement strategies that use better data that is customized to the question at hand, and ultimately, improved understanding of the world.

## References


Althaus, Scott L, Buddy Peyton, and Dan A Shalmon. 2018. "Spatial and Temporal Dynamics of Boko Haram Activity in 6 Event Data Pipelines." *APSA Mini Conference on Modern Event Data Development and Analysis*.

Arora, Sanjeev, Yuanzhi Li, Yingyu Liang, Tengyu Ma, and Andrej Risteski. 2016. "A Latent Variable Model Approach to PMI-Based Word Embeddings." *Transactions of the Association for Computational Linguistics* 4: 385–99. https://doi.org/10.1162/tacl_a_00106.

Arora, Sanjeev, Yingyu Liang, and Tengyu Ma. 2017. "A Simple but Tough-to-Beat Baseline for Sentence Embeddings." *ICLR*.

Baker, Collin F, Charles J Fillmore, and John B Lowe. 1998. "The Berkeley FrameNet Project." In *Proceedings of the 17th International Conference on Computational Linguistics-Volume 1*, 86–90. Association for Computational Linguistics.

Beieler, John. 2016. "Generating Politically-Relevant Event Data." *CoRR*. http://arxiv.org/abs/1609.06239.

Beieler, John, Patrick T Brandt, Andrew Halterman, Erin Simpson, and Philip A Schrodt. 2016. "Generating Political Event Data in Near Real Time: Opportunities and Challenges." In *Computational Social Science*, edited by R. Michael Alvarez. Cambridge University Press.

Blei, David M, and John D Lafferty. 2007. "A Correlated Topic Model of Science." *The Annals of Applied Statistics* 1 (1): 17–35. https://doi.org/10.1214/07-AOAS114.

Blei, David M, Andrew Y Ng, and Michael I Jordan. 2003. "Latent Dirichlet Allocation." *Journal of Machine Learning Research* 3 (Jan): 993–1022.

Boschee, Elizabeth. 2016. "Solutions for Coding Societal Events." Raytheon BBN Technologies Corp. Cambridge United States.

Boschee, Elizabeth, Jennifer Lautenschlager, Sean O'Brien, Steve Shellman, James Starz, and Michael Ward. 2015. "ICEWS Coded Event Data." *Harvard Dataverse* 12.

Brathwaite, Robert, and Baekkwan Park. 2018. "Measurement and Conceptual Approaches to Religious Violence: The Use of Natural Language Processing to Generate Religious Violence Event-Data." *Politics and Religion*, 1–42.

Broscheit, Samuel. 2020. "Investigating Entity Knowledge in Bert with Simple Neural End-to-End Entity Linking." *arXiv Preprint arXiv:2003.05473*.

Carreras, Xavier, and Lluı̈s Màrquez. 2005. "Introduction to the CoNLL-2005 Shared Task: Semantic Role Labeling." In *Proceedings of the Ninth Conference on Computational Natural Language Learning*, 152–64. Association for Computational Linguistics.

Cingranelli, David, and Mikhail Filippov. 2018a. "Are Human Rights Practices Improving?" *American Political Science Review* 112 (4): 1083–9.

———. 2018b. "Problems of Model Specification and Improper Data Extrapolation." *British Journal of Political Science* 48 (1): 273–74.

Cingranelli, David L, and David L Richards. 2004. "CIRI Human Rights Dataset." http://www.humanrightsdata.com.

Davenport, Christian, and Patrick Ball. 2002. "Views to a Kill: Exploring the Implications of Source Selection in the Case of Guatemalan State Terror, 1977-1995." *Journal of Conflict Resolution* 46 (3): 427–50.

Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. "BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding." *arXiv Preprint arXiv:1810.04805*.

Dowty, David. 1991. "Thematic Proto-Roles and Argument Selection." *Language* 67 (3): 547–619.

Ebner, Seth, Patrick Xia, Ryan Culkin, Kyle Rawlins, and Benjamin Van Durme. 2019. "Multi-Sentence Argument Linking." *arXiv Preprint arXiv:1911.03766*.

Fariss, Christopher J. 2014. "Respect for Human Rights Has Improved over Time: Modeling the Changing Standard of Accountability." *American Political Science Review* 108 (2): 297–318.

———. 2018. "Are Things Really Getting Better? How to Validate Latent Variable Models of Human Rights." *British Journal of Political Science* 48 (1): 275–82.

Gerner, Deborah J., 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*.

Gildea, Daniel, and Daniel Jurafsky. 2002. "Automatic Labeling of Semantic Roles." *Computational Linguistics* 28 (3): 245–88.

Goemans, Henk E, Kristian Skrede Gleditsch, and Giacomo Chiozza. 2009. "Introducing Archigos: A Dataset of Political Leaders." *Journal of Peace Research* 46 (2): 269–83.

Goldberg, Yoav. 2017. *Neural Network Methods for Natural Language Processing*. Synthesis Lectures on Human Language Technologies. Morgan & Claypool Publishers.

Greene, Kevin T, Baekkwan Park, and Michael Colaresi. 2019. "Machine Learning Human Rights and Wrongs: How the Successes and Failures of Supervised Learning Algorithms Can Inform the Debate About Information Effects." *Political Analysis* 27 (2): 223–30.

Halterman, Andrew. 2019. "Geolocating Political Events in Text." In *Proceedings of the Third Workshop on Natural Language Processing and Computational Social Science, 17th Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, 29–39.

Halterman, Andrew, Jill A Irvine, and Khaled Jabr. 2019. "Do the Answers You Get Depend on the News You Read? Protests and Violence in Syria." In *APSA 2019 Washington Meeting Paper*.

Hanna, Alex. 2014. "Developing a System for the Automated Coding of Protest Event Data." *Available at SSRN: Http://Ssrn.com/Abstract=2425232*.

Hillard, Dustin, Stephen Purpura, and John Wilkerson. 2007. "An Active Learning Framework for Classifying Political Text." In *Annual Meeting of the Midwest Political Science Association, Chicago*.

Honnibal, Matthew, and Ines Montani. 2017. "SpaCy 2: Natural Language Understanding with Bloom Embeddings, Convolutional Neural Networks and Incremental Parsing." *To Appear*.

Howard, Jeremy, and Sebastian Ruder. 2018. "Universal Language Model Fine-Tuning for Text Classification." *arXiv Preprint arXiv:1801.06146v2*.

Hulst, Johannes M van, Faegheh Hasibi, Koen Dercksen, Krisztian Balog, and Arjen P de Vries. 2020. "REL: An Entity Linker Standing on the Shoulders of Giants." *arXiv Preprint arXiv:2006.01969*.

Jones, Daniel M, Stuart A Bremer, and J David Singer. 1996. "Militarized Interstate Disputes, 1816–1992: Rationale, Coding Rules, and Empirical Patterns." *Conflict Management and Peace Science* 15 (2): 163–213.

Kim, Yoon. 2014. "Convolutional Neural Networks for Sentence Classification." *arXiv Preprint arXiv:1408.5882*.

King, Gary, and Will Lowe. 2003. "An Automated Information Extraction Tool for International Conflict Data with Performance as Good as Human Coders: A Rare Events Evaluation Design." *International Organization* 57 (03): 617–42. https://doi.org/10.1017/S0020818303573064.

Kornilova, Anastassia, Daniel Argyle, and Vlad Eidelman. 2018. "Party Matters: Enhancing Legislative Embeddings with Author Attributes for Vote Prediction." *arXiv Preprint arXiv:1805.08182*.

LaFree, Gary, and Laura Dugan. 2007. "Introducing the Global Terrorism Database." *Terrorism and Political Violence* 19 (2): 181–204.

Lauretig, Adam M. 2019. "Identification, Interpretability, and Bayesian Word Embeddings." *arXiv Preprint arXiv:1904.01628*.

Le, Quoc, and Tomas Mikolov. 2014. "Distributed Representations of Sentences and Documents." In *International Conference on Machine Learning*, 1188–96.

Makarov, Peter. 2018. "Automated Acquisition of Patterns for Coding Political Event Data: Two Case Studies." In *Proceedings of the Second Joint Sighum Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature*, 103–12.

Nardulli, Peter F, Scott L Althaus, and Matthew Hayes. 2015. "A Progressive Supervised-Learning Approach to Generating Rich Civil Strife Data." *Sociological Methodology* 45 (1): 148–83.

Nielsen, Richard A. 2013. "Rewarding Human Rights? Selective Aid Sanctions Against Repressive States." *International Studies Quarterly* 57 (4): 791–803.

Norris, Clayton, Philip Schrodt, and John Beieler. 2017. "PETRARCH2: Another Event Coding Program." *The Journal of Open Source Software* 2 (9).

O'Connor, Brendan, Brandon 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 (Volume 1: Long Papers)* Vol. 1.

Osorio, Javier, and Alejandro Reyes. 2017. "Supervised Event Coding from Text Written in Spanish: Introducing Eventus ID." *Social Science Computer Review* 35 (3): 406–16.

Palmer, Martha, Daniel Gildea, and Paul Kingsbury. 2005. "The Proposition Bank: An Annotated Corpus of Semantic Roles." *Computational Linguistics* 31 (1): 71–106.

Palmer, Martha, Daniel Gildea, and Nianwen Xue. 2010. *Semantic Role Labeling*. Edited by Graeme Hirst. Synthesis Lectures on Human Language Technologies. Morgan & Claypool Publishers.

Pavlick, Ellie, Travis Wolfe, Pushpendre Rastogi, Chris Callison-Burch, Mark Dredze, and Benjamin Van Durme. 2015. "FrameNet+: Fast Paraphrastic Tripling of Framenet." In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, 2:408–13.

Pennington, Jeffrey, Richard Socher, and Christopher D. Manning. 2014. "GloVe: Global Vectors for Word Representation." In *Empirical Methods in Natural Language Processing (Emnlp)*, 1532–43. http://www.aclweb.org/anthology/D14-1162.

Peters, Matthew E, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. "Deep Contextualized Word Representations." *arXiv Preprint arXiv:1802.05365*.

Powell, Jonathan, and Clayton Thyne. 2011. "Global Instances of Coups from 1950 to 2010: A New Dataset." *Journal of Peace Research* 48 (2): 249–59.

Raleigh, Clionadh, Andrew Linke, Håvard Hegre, and Joakim Karlsen. 2010. "Introducing ACLED: An Armed Conflict Location and Event Dataset: Special Data Feature." *Journal of Peace Research* 47 (5): 651–60.

Raytheon BBN Technologies. 2015. "BBN Accent Event Coding Evaluation." Technical report.

Rheault, Ludovic, and Christopher Cochrane. 2019. "Word Embeddings for the Analysis of Ideological Placement in Parliamentary Corpora." *Political Analysis*, 1–22.

Ritter, Alan, Evan Wright, William Casey, and Tom Mitchell. 2015. "Weakly Supervised Extraction of Computer Security Events from Twitter." In *Proceedings of the 24th International Conference on World Wide Web*, 896–905. International World Wide Web Conferences Steering Committee.

Roberts, Margaret E, Brandon M Stewart, Dustin Tingley, Edoardo M Airoldi, and others. 2013. "The Structural Topic Model and Applied Social Science." In *Advances in Neural Information Processing Systems Workshop on Topic Models: Computation, Application, and Evaluation*.

Ruder, Sebastian. 2018. "NLP's ImageNet Moment Has Arrived." *The Gradient* https://thegradient.pub/nlp-imagenet/.

Rudinger, Rachel, and Benjamin Van Durme. 2014. "Is the Stanford Dependency Representation Semantic?" In *Proceedings of the Second Workshop on Events: Definition, Detection, Coreference, and Representation*, 54–58.

Salehyan, Idean, Cullen S Hendrix, Jesse Hamner, Christina Case, Christopher Linebarger, Emily Stull, and Jennifer Williams. 2012. "Social Conflict in Africa: A New Database." *International Interactions* 38 (4): 503–11.

Schrodt, Philip A. 2009. "TABARI: Textual Analysis by Augmented Replacement Instructions." *Dept. Of Political Science, University of Kansas, Blake Hall, Version 0.7. 3B3*, 1–137.

Schrodt, Philip A, Shannon G Davis, and Judith L Weddle. 1994. "Political Science: KEDS—a Program for the Machine Coding of Event Data." *Social Science Computer Review* 12 (4): 561–87.

Schrodt, Philip A, and Deborah J Gerner. 1994. "Validity Assessment of a Machine-Coded Event Data Set for the Middle East, 1982-92." *American Journal of Political Science*, 825–54.

Schrodt, Philip A, and Deborah J. Gerner. 2004. "An Event Data Analysis of Third-Party Mediation in the Middle East and Balkans." *Journal of Conflict Resolution* 48 (3): 310–30.

Spirling, Arthur, and Pedro Rodriguez. 2019. "Word Embeddings: What Works, What Doesn't, and How to Tell the Difference for Applied Research." Working paper.

Sundberg, Ralph, and Erik Melander. 2013. "Introducing the UCDP Georeferenced Event Dataset." *Journal of Peace Research* 50 (4): 523–32.

Van Atteveldt, Wouter, Tamir Sheafer, Shaul R Shenhav, and Yair Fogel-Dror. 2017. "Clause Analysis: Using Syntactic Information to Automatically Extract Source, Subject, and Predicate from Texts with an Application to the 2008–2009 Gaza War." *Political Analysis* 25 (2): 207–22. https://doi.org/10.1017/pan.2016.12.

Varshney, Ashutosh, and Steven Wilkinson. 2006. "Varshney-Wilkinson Dataset on Hindu-Muslim Violence in India, 1950-1995, Version 2." *Ann Arbor, MI: Inter-University Consortium for Political and Social Research*, 02–17.

White, Aaron Steven, Drew Reisinger, Keisuke Sakaguchi, Tim Vieira, Sheng Zhang, Rachel Rudinger, Kyle Rawlins, and Benjamin Van Durme. 2016. "Universal Decompositional Semantics on Universal Dependencies." In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 1713–23. Austin, Texas: Association for Computational Linguistics.

Wilkinson, Steven I. 2006. *Votes and Violence: Electoral Competition and Ethnic Riots in India*. Cambridge University Press.

Wood, Reed M, and Mark Gibney. 2010. "The Political Terror Scale (PTS): A Re-Introduction and a Comparison to CIRI." *Human Rights Quarterly* 32 (2): 367–400.