Automatic Extraction of Events from Open Source Text for Predictive Forecasting

Elizabeth Boschee, Premkumar Natarajan, and Ralph Weischedel

1 Introduction

Forecasting political instability has been a central task in computational social science for decades. Effective prediction of global and local events is essential to counter-terrorist planning: more accurate prediction will enable decision makers to allocate limited resources in a manner most likely to prove effective. Most recently, programs like DARPA's Integrated Conflict Early Warning Systems [2] have shown that statistically-driven quantitative models can produce out-of-sample predictive accuracy that exceeds 80% for certain events of interest, e.g. rebellion and insurgency. Research on a wide variety of predictive approaches continues to thrive.

At the center of forecasting research is the assumption that predicting the future requires some understanding (or model) of both the present and the past. Approaches vary widely in the methods used to construct these models. In some cases, the models are informed by human subject matter experts, who manually evaluate tendencies like an individual leader's propensity for repression, or a particular country's susceptibility to political unrest. Models may also rely on specific quantitative economic or political indicators, where available, or many other types of information. However, there is also a wealth of political, social, and economic information contained in open source textual resources available across the globe, ranging from news reports to political propaganda to social media. To make use of this rich data trove for predictive modeling, an automatic process for extracting meaningful information is required—the volume is too great to be processed by hand, especially in real-time. When this is done, however, the

E. Boschee (✉) • P. Natarajan • R. Weischedel
Raytheon BBN Technologies, Cambridge, MA, USA
e-mail: eboschee@bbn.com; pnataraj@bbn.com; weischedel@bbn.com

V.S. Subrahmanian (ed.), *Handbook of Computational Approaches to Counterterrorism*,
DOI 10.1007/978-1-4614-5311-6_3,
© Springer Science+Business Media New York 2013

predictive models gain access to a data source in which trends and possibilities can be revealed that would otherwise have gone unnoticed

One particular area of research for predictive models using open source text has been the incorporation of events involving actors of political interest; event data was used by several of the successful models developed or advanced in the ICEWS program. These events can cover a range of interactions that span the spectrum from cooperation (e.g. the United States promising aid to Burma) to conflict (e.g. al-Qaeda representatives blowing up an oil pipeline in Yemen). Taken as a whole, the fluctuation in the level and severity of such events (as reported in text) can provide important statistical fodder for predictive models.

These types of events were once coded by hand, but over the past 20 years, many have abandoned that time-consuming, expensive, and often inconsistent process and have moved to automated coding as a method to generate large-scale event data sets. One of the pioneers and leaders of this transition has been the KEDS/TABARI project, begun in the late 1980s. However, these automatically-generated event sets remain noisy: many incorrect events are mistakenly included, and many are missed. Because of this, there is still a wealth of event information not yet available to predictive models using existing methods.

Meanwhile, while computational social scientists have moved towards automatic event coding, new algorithms and processes have been separately developed in machine learning and natural language processing, resulting in effective and efficient algorithms that can extract entities, relations between entities, and events involving entities from text. The full breadth and depth of these techniques has yet to be fully applied to the task of event coding for the computational social sciences. In addition, the speed of computer processing continues to increase so significantly that the application of more sophisticated techniques to this task is now much more feasible than it was 10 years ago.

In this chapter we present the results of studies intended to apply these new, state-of-the-art natural language analysis algorithms to the field of event extraction for computational social science. Our initial results demonstrate that using a state-of-the-art natural language processing system like BBN's SERIF^{TM} engine [3] can yield significant gains in both accuracy and coverage of event coding. It is our hope that these results will (1) lead to greater acceptance of automated coding by creators and consumers of social science models that depend on event data and (2) provide a new way to improve the accuracy of those predictive models.

This chapter is organized as follows. We begin by describing the technical approach underlying the two systems we will compare, one based on existing computational social science techniques for event extraction and the other based on BBN SERIF's state-of-the-art statistical natural language processing techniques. We will then lay out the experimental design and present detailed results for each of five different dimensions of performance:

1. **Accuracy.** When an event is reported by the system, how often is it correct?
2. **Coverage.** How many events are correctly reported by the system?

3. **Historical events.** Can the system correctly filter historical events out of the current event data stream? For instance, a reference to World War II should not give downstream models evidence of new unrest in Europe.
4. **Event/document filtering.** How well do systems filter out "red herring" events based on document topics? For instance, sports documents in particular pose a problem for event extraction systems, since sporting "clashes" often read as violence.
5. **Domain shift.** How well do event extraction models perform on data from sources other than traditional newswire articles?

2 Task Description

The experiment described in this chapter focuses on event coding as performed under the CAMEO (Conflict and Mediation Event Observations) scheme. Details can be found in [4] and [5] as well as at http://web.ku.edu/~keds/data.dir/cameo.html. In CAMEO, each event is represented as a "triple" consisting of an event code, a Source actor, and a Target actor. For instance, in the sentence "*The U.S. Air Force bombed Taliban camps*", the appropriate event triple would be (195, USAMIL, AFGINSTAL) or (*Employ aerial weapons, United States military, Taliban*).

The systems evaluated here used the same manually-generated actor and agent lists specifying that, e.g., Hamid Karzai is affiliated with the government of Afghanistan.

The comparative evaluation for this project covers four of the twenty top-level CAMEO event codes: Provide Aid (07), Disapprove (11), Assault (18), and Fight (19). For purposes of this experiment, codes 18 (Assault) and 19 (Fight) were considered the same top-level event code ("Violence"), since humans could not reliably distinguish between the two. The Violence codes were chosen due to their importance to predictive models in social science. The Disapprove code was chosen due to its frequent confusion with Violence (e.g. "*The president attacked Congress*"). The Provide Aid code was chosen as a representative cooperative code to balance the other three non-cooperative codes.

3 System Descriptions

3.1 Tabari

TABARI, the pioneering and widely-used automatic event coder in the computational social science community, generates event data from text using a computational method called "sparse parsing"; the system uses pattern recognition and grammatical parsing to identify events using a large inventory of

human-generated rules, e.g. *(X) FIRE* MISSILE* AT (Y)*. The system does not use advanced natural language processing techniques but relies on shallow surface structure of sentences to detect events. A full description of the TABARI system can be found in [1].

3.2 BBN SERIF

BBN SERIF (Statistical Entity and Relation Information Finder) is a state-of-the-art, trainable, language-independent understanding system that produces rich linguistic analyses of natural language. The goal of BBN SERIF is to robustly handle widely varying inputs, ranging from formal text (e.g. newswire) to informal speech (e.g. meeting transcripts), and everything in between (e.g. blogs, newsgroups, and broadcast news). The system produces

- Propositional representations of text that capture "*who did what to whom*",
- Entity classification,
- Reference resolution (to whom does "*he*" refer?),
- The detection and characterization of relations between entities, and
- Event detection according to a pre-specified ontology.

All of these components are based on trained statistical models that have learned correct behavior from example data rather than being powered by manually-generated rules, which are often less accurate as well as more difficult to generate and maintain (requiring the use of linguistic experts to hand-craft rules, rather than simply the use of native speakers to highlight examples in text). BBN SERIF has been extensively applied and repeatedly tested in a wide variety of projects and government-sponsored evaluations, including Automatic Content Extraction[1] (ACE) for information extraction, Global Autonomous Language Exploitation (GALE) for query-response distillation and the *who, what, when, where, and why* of distillation [6], and AQUAINT for question answering [7]. BBN SERIF is also being used under an ongoing ONR-sponsored project that detects bias and sentiment with respect to topics of interest.

As the focus of this study is on the improvements in event coding enabled by BBN SERIF, we describe here in detail its primary operations. Significantly more technical detail on specific statistical models employed can be found in [3].

BBN SERIF's event coder consists of a series of natural language analysis components that operate on both the sentence level and the document level. These components are shown in Fig. 1, below.

As shown in the diagram, BBN SERIF first processes an input document's metadata (e.g. the document date and any text zoning information) and then breaks the document into sentences for processing. A set of components then produce

[1] http://www.nist.gov/speech/tests/ace/

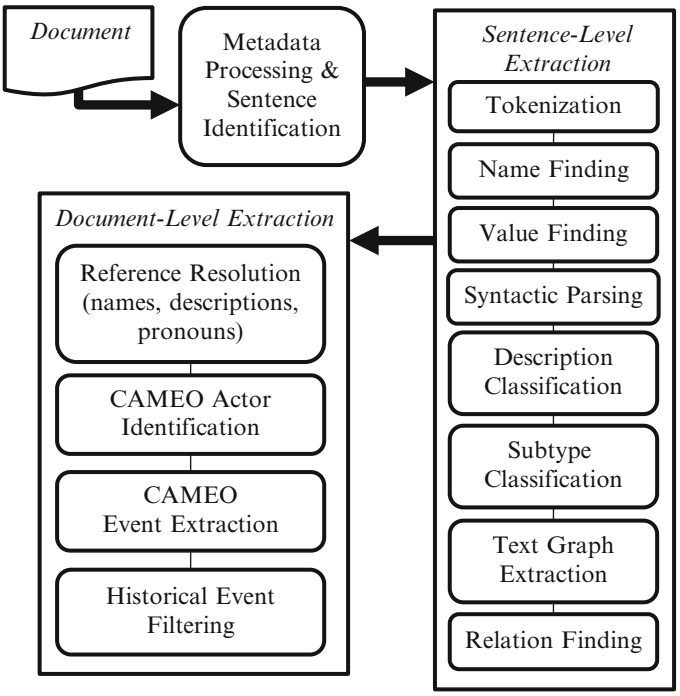


Fig. 1 BBN SERIF system diagram (as used for event coding)

linguistic analyses for each sentence, and then a separate set of document-level components are applied to the document as a whole, using the sentence-level analyses as inputs.

The sentence-level processing components are shown on the right in Fig. 1. Each sentence is first tokenized using a heuristic component that tries to match the tokenization used in downstream components (for English, we use the Penn Treebank tokenization). The tokenized string is then fed to components that use trained models to perform name finding, value finding, and syntactic parsing. ("Values" are abstract concepts such as dates, percentages, and monetary amounts.) Relative dates are resolved with respect to the document's publication date, e.g. by resolving "*Sunday*" to August 8, 2009 or "*last year*" to 2008. The resulting parse tree is then processed by the description classification model to identify nominal mentions of entities, e.g. "the president" or "the missile". "Structured mentions" like appositives and lists are also identified at this stage. A separate component predicts subtype values for each entity mention. The text graph analysis component translates the parse trees into a semantically normalized "propositional" form, which among other things resolves trace information and regularizes the treatment of passive verbs. Finally, another trained model detects and classifies relationships between entities, e.g. *works-for*(A,B) or *located-in*(A,B).

Throughout sentence level processing, SERIF supports beam search to track multiple interpretations. This reduces the chances that a suboptimal choice by an early stage model will prevent the system from finding the best overall interpretation. The beam search capability is limited to the sentence level by design, since it is rare for "garden path" alternative interpretations to remain ambiguous beyond sentence boundaries.

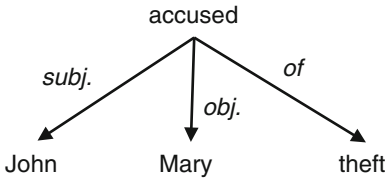
Once all of the sentences have been processed, SERIF executes the document level processing shown on the left side of Fig. 1. First, the system resolves entity coreference, using resolution models for names, descriptions, and pronouns to group together all mentions that refer to the same real-world entity. This component is particularly important for event coding, as it allows the system to identify events involving actors who are not named in the sentence where the event occurs. For instance, in the following sentence, BBN SERIF will determine that "*he*" refers to Ahmadinejad and correctly extract the (*Condemn, Iranian government, United States*) triple stated in the second sentence: "*Iranian President Mahmoud Ahmadinejad called nuclear weapons disgusting. However, he condemned the U.S. on a number of issues.*"

Next, the system applies patterns from an externally-provided actor dictionary to identify actor affiliations for each named entity, in accordance with the CAMEO codebook. For instance, the dictionary contains a pattern [HAMID_KARZAI → AFGGOV], which indicates that Hamid Karzai is a part of the Afghanistan government. This component further classifies certain entities as "agents" of other named actors. For instance, "Iraqi rebels" will be classified as IRQREB. TABARI only classifies agents in cases where the agent word (e.g. "rebels"), is immediately preceded or followed by a known named actor, or when it is followed by the word "in" or "of" and then a known named actor. In contrast, BBN SERIF relies on the linguistic analysis it has already generated to extend the range of its agent matching to include agents detected via *works-for* relations (e.g. "*Jim is employed by Microsoft*") or other syntactic constructions not allowed by TABARI (e.g. "*Jim is a representative for Microsoft*").

As a part of CAMEO actor identification, BBN SERIF propagates all detected actor labels to coreferent entity mentions, as shown above in the example with Mahmoud Ahmadinejad. This obviously significantly improves system actor coverage. It also has the side benefit of increasing system actor accuracy, because it allows the system to be more conservative with named actor patterns. For example, the actor dictionary contains the pattern [HASHIMOTO → JPNGOV]. Although Riyutaro Hashimoto was the prime minister of Japan at one point, there are many other persons named Hashimoto in the world, so this pattern is likely too generic. However, TABARI includes this pattern because it is likely that a sentence might only refer to this important person by only his last name, and without this pattern, any event in that sentence would be missed. In contrast, BBN SERIF can use a more specific pattern like [RIYUTARO_HASHIMOTO → JPNGOV] to label the first instance of Hashimoto mentioned in a document (full names are almost always used at least once in news articles), and then propagate the label JPNGOV to other,

abbreviated references to Hashimoto in the same document, using its automatically-generated coreference links.

After actor identification has been performed, BBN SERIF identifies and labels the actual CAMEO events. This component relies heavily on BBN SERIF's generation of "text graphs", which represent the propositional representations of document content. These graphs directly represent who did what to whom, for instance:



These representations of text are useful for event coding because they are both flexible and accurate. Their flexibility allows a single text graph to represent a wide variety of surface-level phrasings of an event. For instance, the same text graph could represent the core event in "*John accused Mary*", "*John, a friend of Sheila, accused Mary*", or "*Mary was accused by John*". This reduces the set of patterns needed to match various events; one text graph pattern can provide the same coverage as several surface-level patterns.

In addition, a text graph will not match sentences with similar words but a different meaning. For instance, a text graph that matches "*John accused Mary*" will not match the following sentences: "*Bob, a friend of John, accused Mary*", "*Mary was accused by critics of copying John*", or "*John accused someone other than Mary*".

For this study, text graph event patterns were generated semi-automatically from TABARI output on training data: text graphs were automatically extracted from TABARI output and were then pruned and modified by a human developer. In future work, the patterns could be generated by the same process from example data marked by humans, with less or no need for human intervention due to the lower level of noise in the input; the TABARI engine was used as a stand-in for noisy human annotation in this study.

Finally, BBN SERIF identifies and tags "historical" events (defined here as events that occurred more than a month prior to story publication). Historical events can provide misleading information to downstream models and are thus important for the system to identify. As previously described, BBN SERIF already resolves all relative dates with respect to the document's publication date. In this component, BBN SERIF identifies dates connected to extracted events via text graphs, as well as other historical indicators (e.g. "*the anniversary of*...") that modify extracted events. Events that are deemed likely historical are identified and marked for eventual removal, for instance: "*In 1839, the British invaded Afghanistan on the basis of dubious intelligence*" or "*The talks came two months after nine Indians were killed in a suicide attack in Kabul that officials blamed on LeT*".

4 Experiment Design

4.1 Evaluation Corpus

The evaluation corpus consisted of English newswire data collected via LexisNexis. It is standard practice in natural language processing communities that test material be drawn from a time period that does not overlap with the data used to train or tune the system. The most reliable evaluation will involve documents from a newer time period than those used in training/development, so that systems have not already seen (nor been tuned to) the same world events as in the test set. This separation of test and development data is important for both rule-based systems (e.g. TABARI) and statistical systems (e.g. BBN SERIF). To enforce this separation, the evaluation corpus was selected from the most recent data available, namely the 2009–2010 time period.[2] All BBN system development and analysis was performed on documents dated no later than December 31, 2008. TABARI dictionaries were improved by developers looking at events from data through March 2010.

4.2 Evaluation Procedure

For each top-level event-code, we randomly selected 500 triples from each system for evaluation. We then presented these triples in random order to the evaluators (triples produced by both systems were only presented once).

Event triples were selected for evaluation from the first five sentences of each document. This decision was made to ensure the most accurate evaluation: it is important for an evaluator to be able to see and understand the preceding context of any event triple, particularly when actors are referenced using pronouns (e.g. "*he criticized*") or other potentially ambiguous descriptors (e.g. "*the soldiers were attacked*"). Restricting evaluation to the first five sentences of each document ensured that evaluators could easily see and process all necessary context and thereby make accurate assessments about the correctness of system actor labeling.

Each event triple was displayed along with the sentence from which it was derived. The display also included up to four sentences immediately preceding the target sentence, if they existed in the document.

An event triple was considered correct if the sentence it originated in did in fact describe an event of the same top-level type, involving that Source actor and that Target actor. Future, hypothetical, and negated events were not considered correct: the event must actually have been reported as having occurred.

[2]The evaluation corpus included approximately 250,000 documents. Documents judged to be a near duplicate via BBN's semantic de-duplication filter were removed before evaluation.

The evaluator was asked to ignore the event subcode and merely judge the correctness of the top-level event code (e.g. 07). So, if the system said the event was 071 (Provide economic aid) and the correct subcode was actually 073 (Provide humanitarian aid), this would still be acceptable, because the top-level event code was correct. Error rates on subcode detection were separately evaluated and appeared similar across the two systems, but because humans disagreed about 20% of the time on subcode correctness, we looked only at top-level code correctness for our overall evaluation.

Similarly, some flexibility was allowed for actor codes. The evaluator was asked to consider an actor code correct if it was either a "too general" or "too specific" version of the fully correct actor code. An example of a "too general" actor code would be "IRQ" (Iraq) for "IRQMIL" (the Iraqi military). An example of a "too specific" actor code would be "IRQMIL" for "IRQ". However, an actor code could not be considered correct if it referred to a different "branch" of a code than the one suggested by the system. For instance, if the correct code was IRQMIL (the Iraqi military), and the system proposed IRQBUS (an Iraqi business), this is actually contradictory and should have been marked as incorrect.

All triples were seen by at least one evaluator. Most triples were seen by two evaluators, to ensure more stable results.[3] When evaluators disagreed on correctness (i.e. one said "yes" and the other said "no"), the system was given half-credit for that triple.

5 Evaluation Results

5.1 Overview

We evaluated accuracy (precision) directly, based on the number of the 500 triples that were judged to be correct. Table 1 shows these results.

We evaluated coverage, or recall, indirectly, by estimating the number of correct event triples in the corpus from the total number produced multiplied by the estimated precision. Table 2 shows the estimated number of correct triples produced for each top-level event code:

Table 1 Accuracy of extracted events

	Disapprove	Provide Aid	Violence	ALL
TABARI	39%	17%[4]	29%	28%
BBN SERIF	68%	69%	75%	70%

[3] Specifically, the percentages of triples seen by two annotators were 100% for Violence, 100% for Provide Aid, and 69% for Disapprove.

[4] The CAMEO guidelines explicitly prohibit future tense instances of Provide Aid from being marked as correct. This was a significant problem for TABARI; without these errors its performance on Provide Aid would be more comparable to its performance on the other codes.

Table 2 Density of extracted events

	Disapprove	Provide Aid	Violence	ALL
TABARI	1,020	170	620	1,810
BBN SERIF	1,240	270	1,800	3,310

As seen in these tables, the BBN SERIF system appears to both improve accuracy and increase coverage.

5.2 Comparison to Previous Studies

Previous studies have also estimated the accuracy of automated event coders in various ways. It is worth briefly addressing the differences in the metrics involved in these studies. In [8], the authors estimate that an automated event coder (in their case Virtual Research Associates Inc. Reader) could place an event in a correct top-level category with 55% accuracy, which rivaled human coders' accuracy at the same task.[5] This number is significantly higher than the accuracy number cited above for TABARI, but it does not require the coder to correctly extract (or code) the Source or Target of the event, so the numbers are not comparable. In fact, using this metric, our evaluation would have assigned TABARI an average accuracy of 75% and BBN SERIF an average accuracy of 93%. However, these numbers should also not be considered comparable to those cited by King and Lowe, since the data set, ontology, and even the task definition differ significantly; we give them as only an approximate frame of reference to demonstrate the important difference between the two metrics.

In addition, we note that the primary goal of this study is to assess the suitability of statistical natural language processing techniques for social science event coding tasks. For this purpose, BBN SERIF is a suitable representative of these techniques, as it has been separately judged to represent the state-of-the-art in event extraction (on different ontologies) in government-sponsored evaluations such as the Automatic Content Extraction (ACE) evaluation. Many other systems do

[5] King and Lowe report a suite of numbers for accuracy; this number assumes a constant weighting across event categories. In addition, King and Lowe report an 85% accuracy number for a very different metric: the probability of a correct event or non-event judgment on a given sentence. We did not compute this number. Given the high percentage of non-event sentences in our data, it would be meaninglessly high—the trivial baseline, where a system never returns an event, would achieve 96% accuracy on our data set. In contrast, the King and Lowe test set is specifically constructed to contain mostly sentences that have a valid event of some kind (and their raw data pool is also more event-heavy than ours), so that number has a very different meaning in their context.

exist which perform event extraction for other ontologies or, more generally, "triple" extraction. Some of these systems rely on manually-generated rules of various sorts, and some also employ statistically trained models. Some encode their output using Semantic Web technology, often in RDF (Resource Description Framework) triple stores; this data representation is agnostic as to the methods of the actual extraction of triples from text. Unfortunately it is difficult to project performance from one event extraction task to another, as seen in the discussion in the previous paragraph. We therefore focus on the specific comparison between TABARI and BBN SERIF as two exemplars of significantly contrasting techniques for event extraction.

5.3 Error Analysis

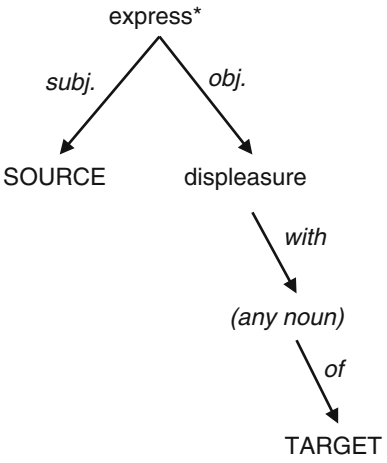
For the purposes of analysis, we identified four primary categories of system error: absolute false alarms (no event of the indicated type exists in the sentence); incorrect actors; actor role reversal; and future, hypothetical, or negated events reported as factual.

In every category, BBN SERIF showed significant reduction in error when compared with TABARI.

The most significant area of error reduction came in the reduction of **absolute false alarms**: cases where no event of the indicated type actually existed in the sentence. Here BBN SERIF reduced TABARI's false alarms from an estimated 1460 to an estimated 320. The constraints placed on the pattern recognition process by text graphs prohibit many spurious matches by requiring the Source and Target actor to be syntactically related to the event verb or noun. For instance, TABARI finds a violent event in the sentence "*Some 150 Japanese soldiers battling piracy are stationed in a US base in Djibouti*", with the United States as the Target. However, since "*US*" is not connected in a meaningful way to "*battling*", BBN SERIF does not find this as a violent event. This 80% reduction in complete noise will provide a significantly more accurate event set to downstream predictive models.

An additional source of error reduction came in sentences where an event of the appropriate type was present in the sentence, but **the actors selected by the system were incorrect**. For instance, the system reported that the perpetrator of a suicide bombing was a U.N. official rather than an Afghani guerrilla. This was the largest category of errors observed for both systems, accounting for an estimated 1740 incorrect TABARI triples and 700 incorrect BBN SERIF triples. Surface-level patterns (as in TABARI) are more vulnerable to this type of error because they have to rely on sentence positioning (rather than logical structure) to hypothesize actors for events. For instance, TABARI finds an event indicating the United States' disapproval of Mexico in the following sentence: "*President Obama, appearing on Wednesday with Felipe Calderon, the president of Mexico, denounced Arizona's new law on illegal immigration.*"

Text graphs help eliminate that guesswork by specifying logical or syntactic roles for both Source and Target actors. Still, text graph patterns can generate errors of this type, as in the following sentence, where BBN SERIF mistakenly believes that the Target of the following Disapprove event is "*Chinese navy*" rather than "*Japan*": "*The Chinese ambassador expressed strong displeasure with the recent monitoring of a Chinese navy fleet by Japan*". This is triggered by the text graph pattern shown below:



This text graph allows any noun to fill the role played by "monitoring" in this sentence. It could be imagined to fire correctly on a similar but meaningfully different sentence, e.g. "*The Chinese ambassador expressed strong displeasure with the recent movements of the Japanese fleet*".

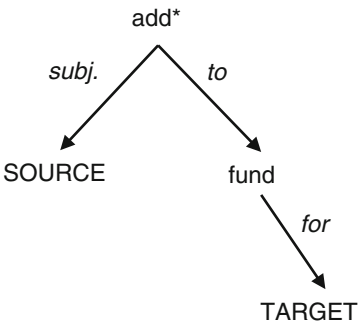
Another type of error reduction came in the situation where the event type was correct, but **the roles of the actors were reversed**. So, rather than reporting that China criticized the United States, the system reported that the United States criticized China. This category accounted for an estimated 320 TABARI errors and 70 BBN SERIF errors. Again, the constraints of the text graphs (particularly their normalization of predicate-argument structure which accounts for passive verb constructions) allow for better actor role assignment.

Finally, a fourth source of error reduction came in situations where the event triple was correct except for the fact that **the event was future tense, hypothetical, or false**; this accounted for an estimated 240 TABARI errors and 100 BBN SERIF errors. BBN SERIF avoids many of these errors by using "counter-factual" text graph patterns to identify and propagate modality to appropriate events (verbs and nouns). These patterns are not specific to any particular event code but can be applied effectively to any task of this nature as one of BBN SERIF's pre-existing capabilities, allowing BBN SERIF to easily discard false, hypothetical, and future-tense events for any event code.

5.4 System Overlap

We also estimated the overlap in system output by looking at the distribution of triples judged correct by evaluators. We estimate that 21% of correct BBN SERIF triples were also found by TABARI, and 38% of correct TABARI triples were also found by BBN. Stated differently, we estimate that TABARI found approximately 1,100 triples not found by BBN SERIF, and BBN SERIF found 2,600 triples not found by TABARI. The relatively low system overlap is indicative of the very different approaches taken by the two systems.

One focus of continued work on BBN SERIF would be the expansion of the text graph pattern set. This expansion could be done by extracting text graphs from a wider set of training data than was used for this pilot experiment. For instance, TABARI finds a Provide Aid triple in the following sentence: "*Chinese government Thursday added 30 million yuan to the snowstorm disaster relief fund for north China's Inner Mongolia Autonomous Region*." This triple is triggered by a high-recall/low-precision TABARI pattern that involves simply the word "fund". The version of BBN SERIF used for this experiment does not have a text graph pattern that matches this sentence, but if it had seen this example in training data, it could have automatically generated a simple text graph pattern that would cover this instance and others like it, with relatively low risk of false alarms:



5.5 Historical Events

News articles often discuss the historical context of current events. For instance, an article discussing the controversy over building a mosque in New York City will likely mention the 9/11 attacks. However, this should *not* give a forecasting model evidence of current terrorist attacks in New York City. To avoid this, historical events must be filtered out of the event data provided to models.

The TABARI system does not natively remove historical events. To add this capability, BBN implemented a historical filtering algorithm based on temporal resolution and syntactic parsing (described earlier in this chapter).

We judged system performance on historical event removal as a stand-alone task. Before filtering, 12.3% of the 1,038 correct triples produced by the system were historical. After filtering, only 1.8% were historical. (Of the 910 triples that were *not* historical, only 39 (4.2%) were filtered out incorrectly.)

In summary, BBN's historical filter significantly reduced the number of historical triples provided to downstream models, with only a slight loss in the number of correct non-historical triples.

5.6 Topic Filtering

Avoiding spurious off-topic event triples is crucial for automatic event coding. Both systems used a type of "document filtering" approach to improve performance on this task. The baseline system used for the TABARI event coding used a keyword filter developed by Lockheed Martin to remove documents likely to be about sports, weather, or financial topics. Sports documents in particular are important to remove, lest the system report an act of war in a sentence like "*Japan and Korea clashed during the qualifying World Cup match earlier today*". TABARI did not code any events in documents that were red-flagged by this keyword filter.

BBN SERIF improved significantly on this filter by using BBN's OnTopic™ classifier to identify potentially off-topic documents and to restrict (but not entirely quash) triple output for these stories. OnTopic is a statistical classifier that automatically learns relationships between topics and words in a passage [9] and has been shown to be robust to noisy data [10]; we trained this model using the output from the keyword filter on a development corpus and then ran it on the separate evaluation corpus to detect documents falling into these potentially off-topic categories.

Of the approximately 90,000 documents in the 2010 evaluation corpus, about 15,000 were marked as "off-topic" by either BBN or the baseline filter. The overlap between these two approaches was relatively low: the baseline filter identified 11,000 documents and the BBN filter identified 7,000 documents, but only 3,000 were shared. (Note however that the BBN filter did not consider finance stories off-topic, as the data indicated that events found in these documents were frequently still correct, e.g. the political violence event in "*Police fired water cannons at opposition party activists protesting in New Delhi against the oil price hike, which economists warned would fuel rising inflation.*")

We compared end-to-end system performance on these 15,000 documents. The first system used the baseline keyword filter and the TABARI event engine. The second system used a combination of BBN OnTopic and BBN SERIF to extract events, extracting only high-confidence events in documents considered to be potentially off-topic.

Using simply the keyword filter to remove off-topic documents, the baseline/TABARI system produced 126 incorrect event triples and no correct event triples. In contrast, using the statistical topic classifier combined with the less black-and-white approach to document filtering, the BBN system produced 18

Table 3 Comparison of system performance on different corpora

	TABARI		BBN SERIF	
	Lexis-Nexis	OSC	Lexis-Nexis	OSC
Disapprove	40%	34%	75%	64%
Provide Aid	17%	21%	69%	71%
Violence	28%	21%	66%	67%
TOTAL	31%	28%	69%	66%

incorrect event triples and 13 correct triples. It seems clear that the BBN system identifies and discards many more misleading documents than the baseline keyword filter. (Virtually all incorrect events produced by TABARI were sporting events mistaken for violence.) The BBN system also "recovered" 13 correct triples that were presumably incorrectly discarded by a keyword filter in the baseline system. For instance, the keyword filter discarded a document that contained the following disapproval event, because the document also mentioned the upcoming World Cup: "*Tension between the two Koreas in the wake of the Cheonan's sinking, an act the South blames on the North . . .* ". The statistical OnTopic engine recognized that although the document mentioned some words relating to sports, the document itself was not fundamentally *about* sports.

5.7 Adapting to New Corpora

A question that often comes up in the context of operational deployment is whether systems will still perform well if the same techniques are applied to new corpora made up of different types of data. Therefore, to test the flexibility of our approach, we performed a secondary evaluation on a corpus made up of data from the Open Source Center. Some of these documents were similar to the newswire available in the LexisNexis corpus, but others were quite different (e.g. radio transcripts). Both systems found approximately 75% as many events in OSC sources as they did in LexisNexis sources. Without human coding of the data, it is impossible to know the reason for this difference: it could be a difference in system performance or simply a difference in the natural event density of the new data source. However, differences in accuracy can be evaluated directly and we did so using the same human evaluators to judge performance on a randomly selected subset of event triples for both systems. Accuracy results were as follows (Table 3).

Both systems degraded only minimally over this new corpus, showing a 15% relative degradation for Disapprove and a slight improvement for Provide Aid. The only significant difference between systems was that the BBN SERIF system sustained its performance on the new corpus for the violence event codes, while the TABARI system suffered a 25% relative degradation in performance. We believe this points further to the robustness of statistical natural language processing when facing problems of domain shift.

6 Conclusion

The BBN SERIF event coding system was designed to build on the "best of both worlds": the knowledge and expertise encoded in the TABARI dictionaries and the most recent statistical learning algorithms for natural language processing. These algorithms perform tasks such as probabilistically assigning a syntactic role for every word and phrase in a sentence, determining who did what to whom in every clause, predicting the real-world entity to which pronouns and other definite descriptions refer, and disambiguating between confusable senses of important event words, such as a verbal *attack* and a military *attack*. Though such algorithms are machine intensive, without any optimization, the BBN SERIF event coding prototype already processes approximately 300,000 sentences an hour on a standard dual quad-core machine. Significant further optimization could be expected as part of a deployed system.

Given that the system capitalizes on the knowledge encoded in TABARI lexicons and sophisticated state-of-the-art statistical learning algorithms, it should not be surprising that the combination substantially improves both in accuracy of output (from 28% to 70%) and also discovers more events (estimated at an 83% increase). The capabilities supplied by BBN SERIF could also improve the scalability and portability of event coding to new genres, new domains, or even new languages, drastically shortening the time needed to generate a dictionary for a new type of event or different type of news source. Initial evidence also suggests that they could significantly automate the currently time-consuming process of actor dictionary maintenance, or actor dictionary creation for a new region.

In addition, the SERIF system

- Significantly reduced reporting of historical events as if current
- Effectively filtered out irrelevant stories, leading to fewer false alarms, and
- Performed robustly on genres outside of news.

The improved precision and coverage of event coding output demonstrated here could significantly help build trust in the output of automated event coding. This has become increasingly important as consumers of the output of forecasting models want to be supplied with evidence convincing them of their validity—to be able to drill down into the details that might support a particular forecast or prediction. In addition, we hope that improving the quality and coverage of the automated event coding data will also improve the quality of statistical models that consume it; we hope to have the opportunity to investigate this in the future.

References

1. Schrodt P (2001) Automated coding of international event data using sparse parsing techniques. Paper presented at the International Studies Association, Chicago
2. O'Brien S (2010) Crisis early warning and decision support: contemporary approaches and thoughts on future research. Int Stud Rev 12:87–104. doi: 10.1111/j.1468-2486.2009.00914.x

3. Ramshaw L, Boschee E, Freedman M, MacBride J, Weischedel R, Zamanian A (2011) SERIF language processing—effective trainable language understanding. Handbook of natural language processing and machine translation: DARPA Global Autonomous Language Exploitation. Springer, New York
4. Gerner D, Schrodt P, Yilmaz O, Abu-Jabr R (2002) Conflict and Mediation Event Observations (CAMEO): a new event data framework for the analysis of foreign policy interactions. Paper presented at the International Studies Association, New Orleans, and American Political Science Association, Boston
5. Schrodt P, Yilmaz O, Gerner D, Hermreck D (2008) The CAMEO (Conflict and Mediation Event Observations) actor coding framework. Paper presented at the International Studies Association, San Francisco
6. Olive J, Christianson C, McCary J (2011) Handbook of natural language processing and machine translation: DARPA Global Autonomous Language Exploitation. Springer, New York
7. Maybury M (2004) New directions in question answering. AAAI Press/The MIT Press, Menlo Park
8. King G, Lowe W (2003) An automated information extraction tool for international conflict data with performance as good as human coders: a rare events evaluation design. Int Organ 57:617–642
9. Schwartz R, Imai T, Kubala F, Nguyen L, and Makhoul J (1997) A maximum likelihood model for topic classification of broadcast news. Proceedings of Eurospeech, Greece
10. Prasad R, Natarajan P, Subramanian K, Saleem S, Schwartz R (2007) Finding structure in noisy text: topic classification and unsupervised clustering. Paper presented at IJCAI-2007 Workshop on Analytics for Noisy Unstructured Text Data, Hyderabad, India