The Impact of Task and Corpus on Event Extraction Systems

Ralph Grishman
New York University
Dept. of Computer Science
715 Broadway, 7th Floor
New York, NY 10003 U.S.A.
grishman@cs.nyu.edu

Abstract

The term "event extraction" covers a wide range of information extraction tasks, and methods developed and evaluated for one task may prove quite unsuitable for another. Understanding these task differences is essential to making broad progress in event extraction. We look back at the MUC and ACE tasks in terms of one characteristic, the breadth of the scenario – how wide a range of information is subsumed in a single extraction task. We examine how this affects strategies for collecting information and methods for semi-supervised training of new extractors. We also consider the heterogeneity of corpora – how varied the topics of documents in a corpus are. Extraction systems may be intended in principle for general news but are typically evaluated on topic-focused corpora, and this evaluation context may affect system design. As one case study, we examine the task of identifying physical *attack* events in news corpora, observing the effect on system performance of shifting from an *attack*-event-rich corpus to a more varied corpus and considering how the impact of this shift may be mitigated.

1. Event Extraction

Event extraction (also referred to as "scenario template" extraction) involves the identification in free text of instances of a particular type of event, and the identification of the arguments of each such event. There is now a considerable literature on event extraction, and in particular on supervised and semi-supervised methods for constructing such systems for new tasks. Implicit in the presentation of these methods is the assumption that they are suitable for a range of event extraction tasks. However, preparing the corpora to evaluate event extraction is an expensive undertaking. As a result, typically, each such method is evaluated on only one or two tasks and corpora. In consequence, it may be hard to judge how dependent the method is on the task and corpus.

Previous work has sought to compare event extraction tasks by measuring the complexity of the linguistic representation of the information to be extracted (Bagga 1998) and analyzing the distribution of information in the document (Huttenen et al. 2002).

In this paper we analyze a few characteristics of the task and corpus, and their influence on the extraction system and its evaluation. By studying some of the differences between tasks, and their implications for extraction system design and evaluation, we hope to achieve a better understanding of the requirements of a general event extraction system.

We consider in particular two issues: the amount of information subsumed by a single event frame (template), and the heterogeneity of the evaluation corpus. We focus primarily on three event extraction tasks that have been the topic of multiple studies: MUC-3/4 on Latin American terrrorist incidents (MUC 1991; MUC 1992), MUC-6 on executive succession (MUC 1995), and ACE 2005 (33 event types covering the most common events of national and international news) (ACE 2005).[1]

2. Breadth of Scenario

One significant variation is in the amount of information captured in a single event template. The ACE event templates are quite narrow – they generally express the core arguments plus place and time information of a single event. Examples of types of ACE events are *marry*, *attack*, *injure*, *kill/die* (considered a single event type), *start-position*, *arrest*, *try*, and *convict*. An *attack* event may take attacker, target, and instrument arguments; a *start-position* event employer, employee, and job title arguments. The goal was to achieve some degree of generality (in covering news stories) through an inventory of such elementary events.

The MUC templates were conceived as being more task specific and richer in their individual content and structure. The basic facts of the MUC-6 task are relatively simple … the person, company, post, and whether the person is entering or leaving the post … but the template structure is more complex, linking the events of one person leaving and another person starting a given job, as well as the prior and current organizational affiliation of a given person.

The MUC-3/4 event templates cover a wider range of information, including both the terrorist event itself (type of incident, date, location, perpetrators, instruments, and targets) and its impact on people and physical objects (people injured or killed; objects damaged or destroyed). It exemplifies the possibility of linking predications that frequently occur together into a larger pattern or *scenario*. We shall refer to

[1] We are not considering "implicit relation extraction" tasks such as the "seminar announcement" task, where each document conveys exactly one event. While such tasks have been intensively studied, they do not raise the same issues of event *identification* raised here.

these predications as the *sub-events* of a scenario. Note that injuries or deaths are only reportable if tied to a terrorist incident; we will refer to this incident (e.g., "bombing") as an *essential* sub-event. An ACE event is typically expressed by a single clause or nominalized NP[2], whereas the information in a MUC template is more likely to correspond to multiple clauses or NPs within a single sentence or several adjacent sentences.

These differences in task characteristic influence the effectiveness of procedures for learning event extraction patterns and for extraction itself.

One approach for the semi-supervised training of event extractors is based on distinguishing relevant and irrelevant documents and selecting predicate-argument patterns which occur significantly more often in relevant documents. This approach was introduced by Riloff (1996) using hand-classified documents and subsequently extended by (Sudo et al. 2003) using document retrieval techniques and by (Yangarber et al. 2000; Yangarber 2003) using a bootstrapping procedure. This procedure works well if there is a document topic (i.e., a partition of the documents into relevant and irrelevant ones) such that the events to be extracted, and only these events, are correlated with the document having that topic. This approach has been successful for MUC-3/4, where the topic is 'terrorism' and the events include both the terrorist act and its effects. It has also been successful for MUC-6, where the events include both starting and leaving a job.

On the other hand, this approach does not work well for some ACE events. For example, applying this procedure to collect *attack* events will also collect *injure* and *kill/die* events; we have applied Yangarber's procedure [3] to discover linguistic representations of *attack* events and found rapidly deteriorating precision for this reason (Altmeyer and Grishman 2009). This is not surprising in as much as *attack* and *kill/die* events are highly correlated in the corpus: 47% of documents in the training corpus with an *attack* event also have a *kill/die* event. This correlation extends to the sentence level: 12% of sentences with an *attack* event also have a *kill/die* event. Similar high correlations exist among ACE justice events (*arrest*, *indict*, *try*, *release*, …). Running Yangarber's bootstrapping with realizations of *arrest* events as seeds quickly expands to include other 'justice' phrases such as "enter plea", "sentence", and "extradite". We can observe that such co-occurring event groups form *natural scenarios* and that document-centric methods are not effective at learning individual events of such scenarios.

The distinction between minimal ACE events and MUC scenarios also affects the extraction process itself, once event patterns have been collected. ACE events can be captured in isolation. For scenarios with no essential sub-events (MUC-6) grouping of sub-events is required for template filling. [4] For scenarios with essential sub-events (MUC-3/4), we must condition subsidiary sub-events on the presence of the essential events. (Patwardhan and Riloff 2009) have described a 2-stage model to capture this dependency.

Studies of learning methods for event extraction have often treated event extraction as a single unified problem. As the brief discussion above suggests, event extraction is really a range of tasks with several dimensions of variation, and these dimensions must be taken into account in any evaluation of the learning methods. Only in that way can we move towards general, high-performance event extraction.

3. Heterogeneity of Corpora

All evaluations are of course affected by the evaluation corpora they employ. In general, an effort is made to have the test corpora be representative of the sort of texts to which the NLP process is intended to be applied. In the case of the extraction evaluations mentioned, this has generally been news sources such as newswires, broadcast news transcripts, and FBIS (Foreign Broadcast Information Service). The problem is that a particular event type is likely to occur infrequently in the news, so a typical evaluation corpus (a few hundred hand-annotated documents), if selected at random, would contain only a few events. Instead these corpora are artificially enriched through a combination of topic classification and manual review so that they contain a high concentration of the events of interest. In the MUC-3/4 test corpora, for example, about 60% of the documents include relevant events, and in the ACE 2005 training corpus 48% include *attack* events.[5]

If we view event extraction as consisting of event identification and argument extraction, and focus on event identification as a separate classification task, it is not surprising that a classifier developed on event-rich collections may overgenerate on more balanced collections. For example, in a small experiment (described in more detail in the next section), we applied our ACE event extractor (Grishman et al. 2005) to classify sentences as containing / not containing an *attack* event. On a small newswire sample from the ACE training corpus the spurious event rate (as a fraction of the true event rate) was below 10%, whereas on unfiltered *New York Times* newswire (from the same epoch as the ACE corpus) it exceeded 100%.

Event identification can be seen as a problem of word sense discrimination (WSD). Particular words are indications of a given event type, but only in some of their senses. For example, "his departure" may

[2] Although there may be several separate, co-referential *mentions* of the event in a document.

[3] This is a re-implementation based closely on the description in (Yangarber 2003).

[4] Grouping is not needed if the evaluation is in terms of slot fillers found, without regard to their grouping into events, as is often the case in evaluations of extraction pattern discovery procedures.

[5] In contrast about 17% of articles from our contemporaneous sample of *The New York Times* newswire contained *attack* events.

represent a resignation (an ACE *end-job* event) or leaving a location (a *movement* event); "shoot" generally indicates an *attack*, but not in "photographers who shoot wildlife". If the corpus is skewed to favor the word senses associated with event instances, a classifier developed using this corpus may not be well-matched to WSD on a more balanced corpus, or may simply ignore WSD altogether.

In the next section we give a small example of the effect of applying an event extractor to a more balanced corpus, and how these effects may be reduced by incorporating into the extractor additional features which provide better word sense discrimination.

4. *Attack* Event Extraction

As our task we selected the identification of *attack* events, part of the ACE 2005 inventory of event types. An *attack* is "a violent physical act causing harm or damage" (ACE 2005). *Attack* is particularly relevant to the issues raised here. Many *attack* event "triggers" (words which are indicative of events) have alternate senses which do not satisfy the criteria for an *attack* event – either they are not *physical* acts or they are not intended to cause harm or damage. The contexts in which these alternate senses appear include

sports:
… fired a high shot towards the goal …

business:
… engaged in a bidding war …

computing:
… conflict in the systems folder …

This is a reasonably frequent event in national and international news, which facilitates annotation and evaluation. Furthermore, some effort was made to insure the presence of many *attack* events in the ACE 2005 corpus – as noted above, nearly half the documents in the collection have an attack event.

As our baseline system we used an event extraction engine originally developed for the ACE 2005 evaluation (Grishman et al. 2005), based on supervised learning methods. The system begins by identifying potential event triggers (words which triggered an event in the training corpus[6]). It then seeks to identify arguments of the event, using both pattern matching (for arguments bearing the same syntactic relation to the trigger as an instance in the training corpus and having the same semantic type) and a maximum-entropy classifier (using as features the trigger word, head and semantic type of the possible argument, syntactic relation to the trigger, etc.). Finally, another maximum-entropy classifier, an *event classifier*, decides whether this is a reportable ACE event, based on the trigger, its direct object or adverbial particle (for verbal triggers), and the arguments identified for the potential event (and the confidence with which they were identified).

Two characteristics of the event extractor are relevant to the investigation. First, like most supervised and semi-supervised event extractors, it relies on local evidence in forming events – evidence of arguments of the appropriate semantic types appearing in the same sentence and, for the most part, in the immediate syntactic neighborhood of the trigger,. Second, it produces some estimate of the probability of an event based on the local evidence, *P(reportable_event)*.

To complement the local evidence used by the event classifier, we created a very simple *document classifier* which predicts whether a document includes one or more *attack* events. We used a maximum-entropy model whose features are the set of words in the document; the model computes *P(relevant_document)*. The intuition is that this will be > 0.5 for topics frequently associated with physical attacks, such as warfare, and < 0.5 for topics such as sports. In other words, it will be performing binary word sense disambiguation over a class of *attack*-related triggers.

The baseline system will report an event if *P(reportable_event)* > τ, where a threshold $\tau = 0.5$ was used for the baseline system.. The version incorporating topic modeling will report an event if

$$\sqrt{P(reportable\_event) \times P(relevant\_document)} > \tau$$

To train both systems, we used 90% of the ACE 2005 training corpus, containing a variety of genres, including newswire, blogs, and transcripts of news broadcasts, talk shows, and other conversations. For evaluation we used two test corpora: first, the remaining 10% of the ACE training documents (55 documents); second, a set of 75 consecutive New York Times news service articles from the same time period (June 2003) which were annotated for the presence of *attack* events. The ACE test corpus has an average of about 3 *attack* event mentions per article; the New York Times corpus about 0.7 event mentions per article. Systems were evaluated on their ability to detect sentences with *attack* events. The results are shown in Table 1, which also reports separately on the 10 newswire articles in the ACE test corpus. For the baseline system we used a threshold of 0.5; for the system with document-level features we show three alternative thresholds.

We can see that, for the baseline system, the spurious rate – the ratio of spurious events extracted by the system to true events -- is much higher for *The New York Times* corpus, reflecting the wider variety of topics. At the same threshold τ=0.5, the document-level model reduces this sharply for the *Times* corpus (and to a lesser extent for the mixed-genre ACE corpus), although our simple model does not do a perfect job of topic classification. The document model also leads to some loss of recall on the ACE and Times corpora, although the net effect is to leave F1 unchanged for ACE and considerably improved for the *Times*.

[6] Each event mention in the training corpus is annotated with its trigger – the word which most clearly expresses the event occurrence. For example, the most common triggers of *attack* events in the training corpus are "war", "attack", "fighting", "fire", and "bombing".

To better compare the two models, we also report the effect of lower thresholds using the document features. For the *Times* corpus, the table shows that document-level features can achieve *both* improved recall and a (more modest) reduction in the spurious rate.

We should not expect similarly dramatic effects for all event types. The results here reflect a rough balance between reports of physical and non-physical attacks. A similar investigation of the *Die* event showed much less ambiguity and hence fewer spurious events in either corpus. So this should rather be taken as a cautionary tale of what *may* go wrong in changing corpora, and one possible measure for addressing the problem.

5. Conclusion

"Event extraction" represents a broad range of tasks. The differences among these tasks reflect themselves in differences in the applicable extraction procedures and differences in the effectiveness of particular learning methods. These factors must be recognized if we are to make continued progress in event extraction.

We must also maintain an awareness of the characteristics of the corpora, particularly if the corpora for development or evaluation differ in some way from the final target application. Understanding the characteristics of the corpus is an inherent part of understanding the extraction task.

References

(MUC 1991) *Proc. Third Message Understanding Conference.* http://aclweb.org/anthology-new/M/M91/

(MUC 1992) *Proc. Fourth Message Understanding Conference.* http://aclweb.org/anthology-new/M/M92/

(MUC 1995) *Proc. Sixth Message Understanding Conference.* http://aclweb.org/anthology-new/M/M95/

(MUC 1998) *Proc. Seventh Message Understanding Conference.* http://aclweb.org/anthology-new/M/M98/

(ACE 2005) *ACE (Automatic Content Extraction) English Annotation Guidelines for Events,* http://projects.ldc.upenn.edu/ace/docs/English-Events-Guidelines_v5.4.3.pdf

Randolf Altmeyer and Ralph Grishman (2009). Active Learning of Event Detection Patterns. Proteus Project Technical Report 09-014, Dept. of Computer Science, New York University. http://nlp.cs.nyu.edu/publication/

Amit Bagga (1998), Analyzing the complexity of a domain with respect to an information extraction task. In (MUC 1998).

Ralph Grishman, David Westbrook, and Adam Meyers (2005). NYU's English ACE 2005 System Description. *Proc. ACE 2005 Evaluation Workshop.*

Silja Huttunen, Roman Yangarber, and Ralph Grishman (2002). Diversity of scenarios in information extraction. *Proc. LREC 2002*.

Siddharth Patwardhan and Ellen Riloff (2009). A unified model of phrasal and sentential evidence for information extraction. *Proc. Conf. Empirical Methods in Natural Language Processing (EMNLP 2009),* Singapore.

Ellen Riloff (1996) Automatically Generating Extraction Patterns from Untagged Text. *Proc. Thirteenth National Conference on Artificial Intelligence*.

Kiyoshi Sudo, Satoshi Sekine and Ralph Grishman (2003). An Improved Extraction Pattern Representation Model for Automatic IE Pattern Acquisition. *Proc. 41st Annual Meeting Assn. Computational Linguistics (ACL 2003)*, Sapporo, Japan.

Roman Yangarber, Ralph Grishman, Pasi Tapanainen and Silja Huttunen (2000). Automatic Acquisition of Domain Knowledge for Information Extraction. *Proc. 18th Int'l Conf. Computational Linguistics (COLING 2000)*, Saarbrücken, Germany.

Roman Yangarber (2003). Counter-Training in Discovery of Semantic Patterns. *Proc. 41st Annual Meeting Assn. Computational Linguistics* (ACL 2003), Sapporo, Japan.

Corpus	System	τ	Correct Events	Missing Events	Spurious Events	Spurious Rate	F1
ACE nw	Baseline	0.5	59	30	7	8%	76%
	DocModel	0.5	59	30	6	7%	77%
	DocModel	0.4	62	27	8	9%	78%
	DocModel	0.3	62	27	8	9%	78%
ACE mixed	Baseline	0.5	103	57	31	19%	70%
	DocModel	0.5	97	63	20	13%	70%
	DocModel	0.4	102	58	27	17%	71%
	DocModel	0.3	106	54	32	20%	71%
NYT	Baseline	0.5	21	25	51	111%	36%
	DocModel	0.5	15	31	11	24%	42%
	DocModel	0.4	18	28	22	48%	42%
	DocModel	0.3	23	23	38	83%	43%

Table 1. Effect of document-level model on extraction of *attack* events.