Clustering sentences for discovering events in news articles

Martina Naughton, Nicholas Kushmerick, and Joe Carthy

School of Computer Science and Informatics, University College Dublin, Ireland
{martina.naughton, nick, joe.carthy}@ucd.ie

Abstract. We investigate the use of clustering methods for the task of grouping the text spans in a news article that refer to the same event. We provide evidence that the order in which events are described is structured in a way that can be exploited during clustering. We evaluate our approach on a corpus of news articles describing events that have occurred in the Iraqi War.

1 Introduction

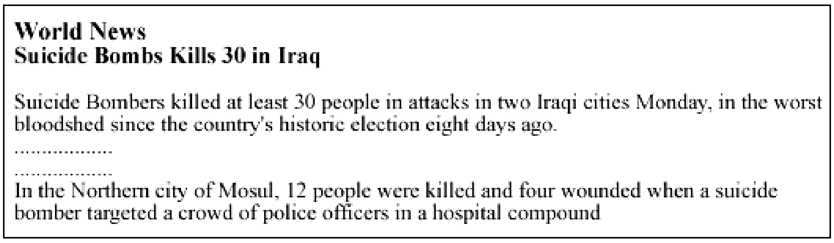
A news event is defined as a specific thing that happens at a specific time and place [1], which may be reported by one or more news sources. Multiple news articles often contain duplicate information concerning the same event, but differ in choice of language used. Specific details regarding the event may vary from source to source. For example, one article about a given bombing in Iraq may say *"at least 5 people were killed"* while a second may contain the phrase *"6 people were found dead"*.

Our research focuses on merging descriptions of events from multiple sources to provide a concise description that combines the information from each source. We decompose this problem into three sub-problems: (1) Annotation: identifying the spans of text in an article corresponding to the various events that it mentions; (2) Matching: identifying event descriptions from different articles that refer to the same event; and (3) Aggregation: converting the event descriptions into a structured form so that they can be merged into a coherent summary.

In this paper we focus on the first sub-problem. Specifically, we describe and evaluate methods for annotating each sentence in an article with a set of identifiers specifying which event(s) the sentence mentions. This set can be empty (if the sentence does not mention any event) or it can contain multiple identifiers.

Event annotation is challenging for several reasons. Most news articles refer to multiple events. Moreover, sentences that refer to the same event are usually scattered through the article with no simple sequential pattern. Fig. 1 shows a sample article that demonstrates these issues.

The task of clustering similar sentences is a problem that has been investigated particularly in the area of text summarization. In SimFinder [2], a flexible clustering tool for summarisation, the task is defined as grouping small paragraphs of text containing information about a specific subject. However, we examine the use of clustering at sentence level.



World News
Suicide Bombs Kills 30 in Iraq

Suicide Bombers killed at least 30 people in attacks in two Iraqi cities Monday, in the worst bloodshed since the country's historic election eight days ago.
....
....
In the Northern city of Mosul, 12 people were killed and four wounded when a suicide bomber targeted a crowd of police officers in a hospital compound

Fig. 1. Sample news article that describes multiple events.

2 Event extraction as sentence clustering

This paper investigates the use of clustering to automatically group sentences in terms of the event they describe. We generated sentence clusters using average link, complete link and single link agglomerative clustering. Hierarchical agglomerative clustering (HAC) initially assigns each data point to a singleton cluster, and then repeatedly merges clusters until a specified termination criteria is satisfied [3]. HAC clustering methods require a similarity metric between two sentences. We use the standard cosine metric over a bag-of-words encoding of each sentence. We removed stopwords, but did not employ term weighting.

We evaluated our clustering algorithms using a collection of 219 news stories describing events related to the recent war in Iraq. Excess HTML (image captions etc.) was removed, and sentence boundaries were identified. The corpus was then annotated by two volunteers. Within each article, events were uniquely identified by integers. Starting at the value 1, the annotators were asked to assign labels to each sentence representing the event(s) it describes. If a sentence did not refer to any event, it was assigned the label 0. Sentences may refer to multiple events. For example, consider the sentence *"These two bombings have claimed the lives of 23 Iraqi soldiers"*. This sentence would be annotated with two labels, one for each of the two bombings. Note that sentences from the same document that refer to the same event are assigned the same label.

To evaluate our clustering method, we define precision and recall as follows. We assign each pair of sentences into one of four categories: a, clustered together (and annotated as referring to the same event); b, not clustered together (but annotated as referring to the same event); c, incorrectly clustered together; d, correctly not clustered together. Precision and recall are thus found to be computed as $P = \frac{a}{a+c}$ and $R = \frac{a}{a+b}$, and $F1 = \frac{2PR}{P+R}$.

We also need to consider sentences annotated with multiple event labels. For each pair, where one or both of the sentences were annotated as referring to multiple events, we consider them as belonging in the same event cluster if the intersection between their labels is not empty. For example, we consider that a sentence pair with labels "1,2" and "1,3" respectively as belonging to the same cluster.

A fully-automated approach must use some termination criteria to decide when to stop clustering. In this preliminary work, we simply compare the results

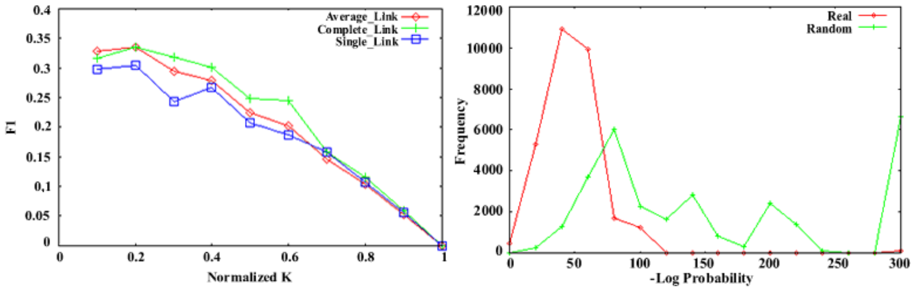


Fig. 2. Left: F1 at each value of normalized k, for complete link, single link and average link clustering algorithms. Right: Distribution in the probability that actual and random event sequences are generated by the tuned FSA.

emitted by the HAC algorithm for various values of k, where k is the number of remaining clusters.

As seen in Fig. 2, F1 increases slightly as k is increased, but then rapidly falls with increasing k. We also implemented a semi-supervised approach in which k is manually set to the annotated number of incidents in each article. We found that precision ranges between 0.58 and 0.39 and recall falls between 0.29 and 0.25 for all three algorithms. Interestingly, we observe that the "correct" value for k is not necessarily the value of k that maximizes accuracy.

3 Sequential event structure

Our clustering approach ignores an important constraint on the event associated with each sentence: the position of the sentence within the document. Intuitively, adjacent sentences are more likely refer to the same event, later sentences are likely to introduce new events, etc.

To confirm the intuition that such latent structure indeed exists, we treat each document as a sequence of event labels (namely, one label per sentence). We trained a finite state automaton (FSA) from the sequences, where states corresponded to event labels, and transitions corresponded to adjacent sentences that mention the pair of events. The automaton is stochastic: we counted the number of each transition across a set of training documents (as well as the fraction of documents whose first and last sentences are labeled with each event). We can calculate the probability that the trained automaton generated a given document as the product of the probability that the first sentence's event is an initial state, the probabilities of each transition in turn, and the probability that the last sentence's label is a final state. (This assumes that each sentence mentions at most one event. We deal with multi-event sentences in various ways, such as making a "copy" of each article for each permutation of its labels; for example, the article sequence "1, {1,2}, 2, {2,3}" is mapped to 4 sequences, "1 1 2 2", "1 2 2 2", "1 1 2 3" and "1 2 2 3".)

Finally, we estimate how much sequential structure exists in the sentence labels as follows. The document collection was split into training and test sets. The automaton parameters were learned from the training data, and the probability that each test sequence was generated by the automaton was calculated. These probabilities were compared with those of a set of random sequences (generated to have the same length distribution as the test data).

The probabilities of event sequences from our dataset and the randomly generated sequences are shown in Fig. 2. The test and random sequences are sorted by probability. The horizontal axis shows the rank in each sequence and the vertical axis shows the negative log probability of the sequence at each rank. The data suggest that the documents are indeed highly structured, as real document sequences tend to be much more likely under the trained FSA than randomly generated sequences.

4 Discussion

We have presented exploratory work on the use of clustering for event annotation in news articles. We are currently trying variations of our approach, such as using WordNet [4] to deal synonymy (eg, *"killed"* and *"died"*).

Although the precision of our approach is approximately 50%, we are encouraged since the similarity metric ignored the sequential structure demonstrated in Sec. 3. We have developed a revised distance metric that incorporates the sequential regularities demonstrated in Fig. 2. Preliminary experiments show that this enhancement provides a modest increase in F1.

Finally, our approach did not use term weighting. We have developed a TFIDF-like weighting scheme where we define a "document" to be the set of sentences which discuss a given event and then weight terms according to their frequency in the document compared to the entire corpus. Of course, these "documents" are precisely what the clustering algorithm is trying to discover. We therefore initialize the term weights uniformly, and then iterate the clustering process, re-calculating the term weights based on the previous output, stopping when the event labels converge. Preliminary results show that this approach converges rapidly and also produces a modest increase in F1.

References

1. Li, Z., Wang, B., Li, M., Ma, W.Y.: A probabilistic model for retrospective news event detection. In: Proceedings of the 28th annual international ACM SIGIR Conference on Research and Development in Information Retrieval, ACM Press (2005) 106–113
2. Hatzivassiloglou, V., Klavans, J., Holcombe, M., Barzilay, R., Kan, M.Y., McKeown, R.: Simfinder; a flexible clustering tool for summarisation. In: NAACL Workshop on Automatic Summarisation. (2001) 41–49
3. Manning, C.D., Schtze, H.: Foundations of Statistical Natural Language Processing. MIT Press (1999)
4. Miller, G. A., E.: Wordnet: An on-line lexical database. International Journal of Lexicography (1990) 235–312