Tracking Terrorism News Threads by Extracting Event Signatures

Syed Toufeeq Ahmed, Ruchi Bhindwale, and Hasan Davulcu
School of Computing and Informatics
Arizona State University
Tempe, Arizona
{toufeeq, rbhindwa, hdavulcu}@asu.edu

***Abstract*- With the humongous amount of news stories published daily and the range of ways (RSS feeds, blogs etc) to disseminate them, even an expert at tracking new developing stories can feel the information overload. At most times, when a user is reading a news story, she would like to know "*what happened before this?*" or "*how things progressed after this incident?*". In this paper, we present a novel real-time yet simple method to detect and track new events related to violence and terrorism in news streams through their life over a time line. We do this by first extracting signature of the event, at microscopic level rather than topic or macroscopic level, and then tracking and linking this event with mentions of same event signature in other incoming news articles. There by forming a thread that links all the news articles that describe this specific event, with no training data used or machine learning algorithms employed. We also present our experimental evaluations conducted with Document Understand Conference (DUC) datasets that validate our observations and methodology.**

Keywords*- *Named Entity Recognition; Event Detection; News Threads Extraction; First Story Detection;

I. INTRODUCTION

The ever-growing amount of electronically disseminated news stories available for daily peruse is almost overwhelming. To pay detailed human attention to every developing story and to track these stories over a time period is an insurmountable task. Imagine an intelligence expert who needs answers for the question "What happened during Greece Student Riots and how the story unfolded?" she has to search news paper websites and feeds to find the articles she could read about this story and arrange, organize and link these stories mentally in the fashion they would have unfolded, in order to understand what happened and to get the bigger picture.

As defined in Topic Detection and Tracking (TDT) domain, we present some definitions [1] and also define *Event Signature (ES)*. A "*story*" is a news article delivering some information to the users. A *"topic"* is a set of news stories strongly connected by a seminal event, whereas an "*event"* is something that happens at some specific time and place [2]. For example "Chinese airplane crash in Korea in April 2002" is an event. Topic is more general, for example news stories about "Mars Probe Phoenix" or "Hurricane Katrina". TDT research is focused on three main tracks [3]: 1) *First Story Detection (FSD)* - identify if a news article is talking about a new story or it belongs to known topic. 2) *Story Segmentation* – segment a news stream into topically cohesive stories, this task mainly applies to audio or TV news (e.g., transcribed speech), since web news articles/feeds are supplied in segmented form. 3) *Topic Tracking* – track events of interest based on sample news stories, and associate incoming news stories with the related stories, which were already discussed before.

II. RELATED WORK

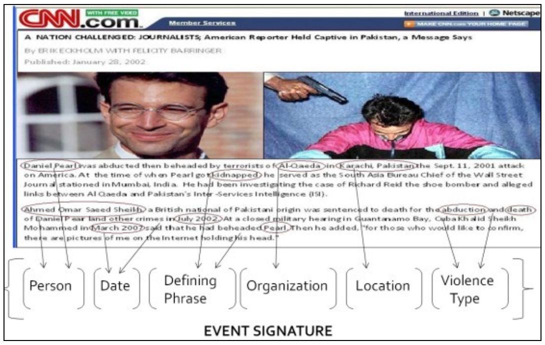
One of main problem tackled in text data mining research is to extract meaningful structure from document streams that arrive continuously over time, [6] presents an approach to model "burst of activity" in a such document stream. Seminal work [7, 8] in event detection and tracking explored both *retrospective detection* and *online detection* approaches and clustering algorithms like agglomerative clustering, augmented Group Average Clustering [8] were used. Well-known *idf-weighted* cosine coefficient metric method [11] was also used to detect and track, with good results in tracking but not encouraging results in detection task.

Retrospective detection is defined as the discovery of previously unidentified events in historical news corpus [10]. In their work, [10] used both contents and time information of news articles. *Online (real-time) detection* strives to identity the onset of new events from live news feeds in real-time [8]. A real-time news event extraction system [9] extracts violence and disaster events by processing the news article using extraction grammars on each document in the cluster.

III. EXTRACTING EVENT SIGNATURES TO TRACK EVENT THREADS

A. Event Signature

We like to capture all the pertinent information that describes an event (like people involved, organizations mentioned, locations, dates, event describing phrases etc.) We extract this information from the text of the news articles and also try to capture the uniqueness about the event (example shown in figure 1). Figure 2 shows an example of an Event Thread, a chain/thread of news articles about a particular event sorted based on the date published. We remove duplicate articles which have exact same text, as it is a common occurrence to get the story from a news wire service (like Reuters, or AFP).



A NATION CHALLENGED: JOURNALISTS; American Reporter Held Captive in Pakistan, a Message Says

By ERIK ECKHOLM WITH FELICITY BARRINGER

Published: January 28, 2002

Daniel Pearl was abducted then beheaded by terrorists of Al-Qaeda in Karachi, Pakistan the Sept. 11, 2001 attack on America. At the time of when Pearl got kidnapped he served as the South Asia Bureau Chief of the Wall Street Journal stationed in Mumbai, India. He had been investigating the case of Richard Reid the shoe bomber and alleged links between Al Qaeda and Pakistan's Inter-Services Intelligence (ISI).

Ahmed Omar Saeed Sheikh a British national of Pakistani origin was sentenced to death for the abduction and death of Daniel Pearl and other crimes in July 2002. At a closed military hearing in Guantanamo Bay, Cuba Khalid Sheikh Mohammed in March 2007 said that he had beheaded Pearl. Then he added, "for those who would like to confirm, there are pictures of me on the Internet holding his head."

Figure 1. Extracting an Event Signature from a news article. Example article shown is describing "Daniel Pearl's Kidnapping and Beheading". (Source: CNN.com)

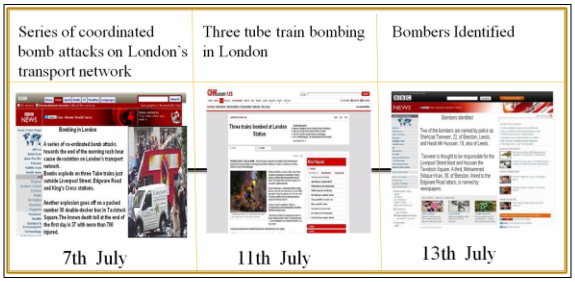

Series of coordinated bomb attacks on London's transport network
Three tube train bombing in London
Bombers Identified
7th July
11th July
13th July

Figure 2. An Event Thread unfolding over a timeline. Example articles shown are about "July 7th bombing in London". (Source: CNN.com/BBC.co.uk)

B. Extracting Named Entities

The field of Named Entity Recognition (NER) has matured considerably over last few years and systems based on Conditional Random Fields [4] have shown very good results. We used Stanford NER[1] system for labeling three entity class types: PERSON, ORGANIZATION, and LOCATION. To recognize DATE mentions in the text, we used regular expressions to recognize different date formats and also calendar months, days and years.

C. Extracting Violence Type Words (Action Words)

Recognizing the action words that describe a violent activity (like *bombing, kidnapping, blast, shot, killed, burnt*) in the text, is first step to classify whether the document (or paragraph) may be describing a violent event that happened at some location at a certain time. We recognize violence type words using a hash table built as ontology of these words, which we extracted using an initial (around 240) seed root words describing violence (e.g., kill, shot, burn, bomb) and recursively extracting synonyms from WordNet [2] using synonym sets (synsets). To match words, say *"burnt"* and "*burned"* as same words in the given context, we stem these words using Porter Stemmer [5] before matching process.

D. Extracting Event Defining Phrases

To fully capture the signature of the event, we also need to extract phrases that usually end up defining or describing the event or incident in few words (like "Daniel Pearl's Kidnapping", "9/11 attacks" or "Greece Student Riots"). We first extract candidate phrases from the news articles by dropping stop words and collecting remaining words as phrases. Next step is to filter these phrases through heuristics rules that keep only those phrases that match these rules (e.g. "… *[Named-Entity] ….. <Violence-Type word> …. ").* Then, we select the defining phrase that best matches the whole article using vector cosine similarity measure.

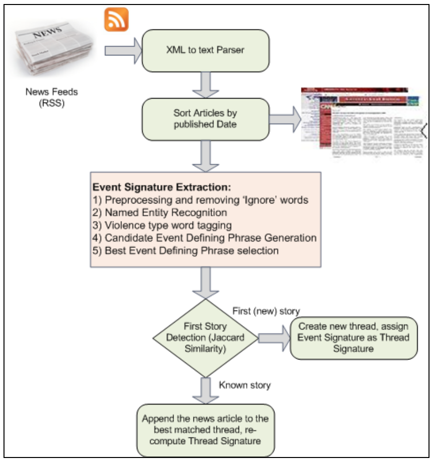

NEWS
News Feeds (RSS)
XML to text Parser
Sort Articles by published Date
Event Signature Extraction:
1) Preprocessing and removing 'Ignore' words
2) Named Entity Recognition
3) Violence type word tagging
4) Candidate Event Defining Phrase Generation
5) Best Event Defining Phrase selection
First Story Detection (Jaccard Similarity)
First (new) story
Create new thread, assign Event Signature as Thread Signature
Known story
Append the news article to the best matched thread, re-compute Thread Signature

Figure 3. Extracting Event Threads from RSS news feeds.

E. Thread Signature

In order to speed up event tracking, instead of comparing the new incoming article with every article previously seen, we maintain thread signatures for every thread and efficiently compare thread signatures to new article event signature.

Input: (RSS) Stream of News Articles
Output**:** Event Threads

1. For each input article, extract its Event Signature (ES_k)
2. Compare ES_k similarity with current list of Thread Signatures (TS_k).
3. Add the article to the thread with highest Similarity score, Jaccard-*Sim* (ES_k ,TS_k) > ζ, where ζ is *min. first story threshold.* Recalculate $TS_k = ES_k$ φ TS_k
4. If article's *JaccardSim*(ES_k ,TS_k) ≤ ζ, then create a new thread, and assign $TS_k = ES_k$

Figure 4. Algorithm to compute Event Signature similarity with exisiting threads.

F. Online Extraction of Event Threads

Algorithm (in figure 4) describes Event Signature similarity computation with Thread Signatures. Figure 3 shows the thread extraction process. Event Threads are extracted directly from live RSS feeds.

IV. Evaluation and Experiments

An evaluation of event thread extraction performance was carried on Document Understanding Conference (DUC) [3]

[1] http://nlp.stanford.edu/software/CRF-NER.shtml

[2] http://wordnet.princeton.edu/

[3] http://duc.nist.gov/

datasets, from year 2004 to 2006. Datasets (DUC2004, DUC2005 and DUC2006) each consists of 50 folders (clusters) of news articles from Associated Press and New York Times. Each cluster contains 25 news articles about a particular topic or event. For our experiments we take each cluster as a thread about that event or topic. We observed that in most articles, the primary event talked about is mentioned in the title and first two paragraphs. If we leverage this observation, we can save valuable real-time while doing expensive NER and also reduce noise in the signature extracted. To validate this, we conducted experiments on both *long* (full) articles and *short* (title and first two paragraphs) articles. To study the effects of *length* of the thread (number of articles in the thread), we conducted experiments for thread length of *6* and *10* articles in each cluster respectively. And to study the effects of increasing number of threads, we conducted experiments with datasets with *10* threads and *25* thread collections. So, a dataset (see Figure 5 and 6) labeled *"25-10-S"* means, *25 total threads, with 10 short articles in each thread cluster.* For each dataset, we collected all the articles from all clusters into one cluster and sorted them randomly, and tried to extract threads back from this collection. We evaluated event thread extraction performance by these criteria.

Number of threads extracted: (How many threads were extracted from initial dataset of 25 threads?). Figure 5 and figure 6 show results for 10 thread and 25 thread datasets respectively. We got almost comparable results for both short and long articles, short performing little better. We got better results for DUC2006 dataset as it has cleaner clusters than other two datasets.

F-measure: (The weighted harmonic mean of precision and recall), F-measure scores for each dataset are shown in figure 7. We see that short articles out performs long articles by ***6.5%*** and again DUC2006 performing best in three datasets.

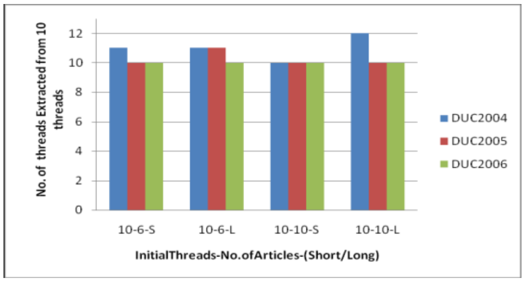


Figure 5. Number of event threads extracted from 10 thread datasets.

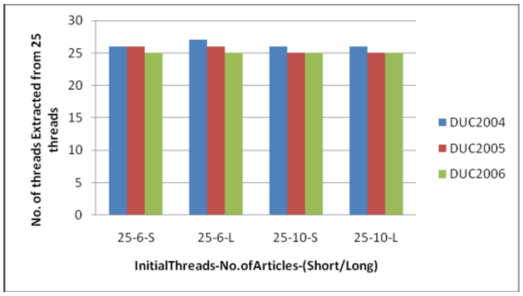


Figure 6. Number of event threads extracted from 25 thread datasets.

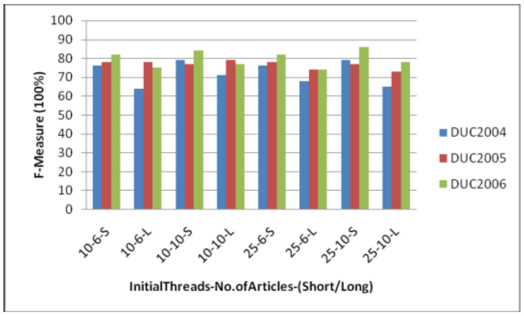


Figure 7. F-measure for different datasets as mentioned in figure 5 and 6.

V. CONCLUSIONS

In this paper, we presented a novel real-time yet simple method to detect and track new events related to violence and terrorism in news streams with no training data used or machine learning algorithms employed. We see that event thread extraction using only title and first two paragraphs of the article performed better than full articles. In future work, we need to handle splitting threads into multiple threads as the stories diverge into multiple stories.

REFERENCES

[1] Nallapati, R., Feng, A., Peng, F., and Allan, J. 2004. Event threading within news topics. *In Proceedings of the Thirteenth ACM international Conference on information and Knowledge Management,* CIKM '04. ACM, New York, NY, 446-453.

[2] Yang, Y., Carbonell, J. G., Brown, R. D., Pierce, T., Archibald, B. T., and Liu, X. 1999. Learning Approaches for Detecting and Tracking News Events. *IEEE Intelligent Systems* 14, 4 (Jul. 1999), 32-43.

[3] Chung, S., Jun, J., and McLeod, D. *Incremental Mining from News Streams.* Encyclopedia of Data Warehousing and Mining, Idea Group Inc. 2004.

[4] J. R. Finkel, T. Grenager, and C. Manning. 2005. Incorporating Non-local Information into Information Extraction Systems by Gibbs Sampling. *Proceedings of the 43nd Annual Meeting of the Association for Computational Linguistics (ACL 2005),* pp. 363-370.

[5] Porter, M. F. (1997). "An algorithm for suffix stripping." Progam, vol. 14, no. 3, July 1980: 313--316.

[6] Kleinberg, J. 2002. Bursty and hierarchical structure in streams. In *Proceedings of the Eighth ACM SIGKDD international Conference on Knowledge Discovery and Data Mining*, KDD '02. ACM, New York, NY, 91-101.

[7] Allan, J., Papka, R., and Lavrenko, V. 1998. On-line new event detection and tracking. In *Proceedings of the 21st Annual international ACM SIGIR Conference on Research and Development in information Retrieval* , SIGIR '98. ACM, New York, NY, 37-45.

[8] Yang, Y., Pierce, T., and Carbonell, J. 1998. A study of retrospective and on-line event detection. In *Proceedings of the 21st Annual international ACM SIGIR Conference*, SIGIR '98. ACM, New York, NY, 28-36.

[9] Tanev, H., Piskorski, J., and Atkinson, M. 2008. Real-Time News Event Extraction for Global Crisis Monitoring. Lecture Notes In Computer Science, vol. 5039. Springer-Verlag, Berlin, Heidelberg, 207-218.

[10] Li, Z., Wang, B., Li, M., and Ma, W. 2005. 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*, SIGIR '05. ACM, New York, NY, 106-113.

[11] J. Michael Schultz, Mark Liberman, "Topic Detection and Tracking using idf-weighted Cosine Coefficient," DARPA Broadcast News Workshop Proceedings, 1999.