Digging into Human Rights Violations
Phrase mining and trigram visualization

Ben Miller
Depts. of English and Communication
Georgia State University
Atlanta, GA
miller@gsu.edu

Fuxin Li
College of Computing
Georgia Institute of Technology
Atlanta, GA
fli@cc.gatech.edu

Ayush Shrestha
Dept. of Computer Science
Georgia State University
Atlanta, GA
ashrestha2@student.gsu.edu

Karthikeyan Umapathy
School of Computing
University of North Florida
Jacksonville, FL
k.umapathy@unf.edu
Abstract— Unidentified victims, perpetrators, and details of human rights violations are camouflaged by the scale of collections of witness reports. This project responds to that problem by developing an approach for identifying the appearance of often unnamed individuals across a corpus by integrating semi-supervised machine-learning based phrase mining, network visualization, and an event model from human rights reporting known as “Who Did What to Whom.” This natural language processing method facilitates the retrieval of cross-document narratives of victims -- and perpetrators -- of population level events.

Keywords: text mining; natural language processing; human rights data analysis; data visualization; September 11, 2001.

I. INTRODUCTION

Digging into Human Rights Violations (DHRV) is developing a computational reader for text corpora of human rights abuses to discover the stories of hidden victims and unidentified perpetrators only apparent when reading across large numbers of related documents. In part, this project began with an observation drawn from Ball, Tabeau, and Verwimp’s report on the Bosnian Book of Dead [Ball 2007]. That report on the tabulation of fatalities resulting from ethnic cleansing undertaken by the Milosevic regime was highly concerned with the de-duplication of entries. This over-reporting of individual victims within human rights corpora is endemic, and represents an opportunity for a system that can read across a corpus. For example, in the 511 interviews comprising our test corpus, one named individual appears more than 60 times. How many times might unnamed individuals reoccur? Automated readers exist that classify documents, produce summaries [Nenkova 2011], extract significant information [Strassel 2008], and highlight sentiment [Pang 2008]. This type of analysis works best with well-defined figures, such as occur in newspaper or government documents. That partially describes reports of human rights violations, as each generally describes a victim’s perspective of one event. Currently, systems have difficulty parsing peripheral entities, indeterminate language, or references that go beyond the boundaries of one document and are only significant when traced across documents; many reports peripherally describe the fates of other victims. These implicit links and duplicates amongst records enable horizontally reading across collections, rather than vertical reading through one record.

The technical goal of DHRV is an NLP system facilitating cross-document coreference within the domain of rights violations. Our approach to resolving the task [Kibble and Deemeter 2000] describe as “whether or not two mentions of entities refer to the same person,” begins by considering the subtype of anaphora (indicative language within a document) known as exophora (indicative language across documents), and relies on placing pronominal entities within a high-order Event Trigraph of location, time, and name. Because temporal information is often referential and ambiguous, and therefore difficult to extract and correlate [Northwood 2000] our approach uses a phrase-based establishment of semantic context to support identifying the temporal context. This noun- and verb- phrase extraction, collocation detection, and semi-automated matching, feeds a visualization reminiscent of network graphs. Uncertainties in the document and information retrieval processes are visualized to allow researchers to confirm whether entity occurrences should be conflated. Because human rights documentation contains sensitive, private information, this project is prototyping with another historically significant corpus that shares structural features with our primary data: the World Trade Center Task Force interviews with first responders to the attacks of September 11, 2001.

II. EVENT SUMMARIZATION BASED ON MATCHING PHRASES

Our main stratagem is to situate entities in the series of events that define their appearances. Phrases useful for this process accord to a “journalist template,” and are guided by the human rights violations reporting schema named “Who did What to Whom” [Chang 2012, Ball 1996]. Extracting these important entities as phrases happens via the system shown in Fig. 1.

DHRV is a recipient of a 2011 Digging into Data Challenge Grant and gratefully acknowledges the support of the NSF and SSHRC.
In the system, noun phrases and verb phrases are extracted by a parser. Then, a phrase classifier determines which fall into important entity and event categories. After categorizing these phrases, a Collocation Detector connects contextually similar phrases. This collocation of phrases differs from the collocation of words typical to NLP, in that it captures instances instead of global probabilities. After a set of collocated phrases have been detected, they are placed into the event template and fed to a visualization engine. The engine computes scores to suggest which events and entities are coreferent.

III. PHRASE EXTRACTION AND CLASSIFICATION

After parsing, noun phrases and verb phrases are extracted from the parse tree. A shallow parser (chunker) is inappropriate because it has lower recall than a full parser (may not capture all the desired phrases), so we use the Stanford parser [Klein 2003(1), Klein 2003(2)]. From the extracted phrases, we formulate a classification task for labeling important phrases for event extraction. Of the eight categories, Organization, Person, Title, Location, Date, Time, Event, Miscellaneous, and Unimportant, some are traditional NER or TimeML categories. Event and Miscellaneous labels are new, and indicate phrases that might not be readily interpreted as named entities. Phrases such as “the pedestrian bridge,” “the ferry,” or “the second tower” which are not identifiable as a particular named entity, are crucial in localizing the event.

To maximally utilize manual labeling, an unsupervised selection mechanism selects target phrases. In this mechanism, phrases are ranked similarly to a frequency or N-gram model, discounting the probability of a phrase if it is very common in a background corpus:

$$Sc(\text{phrase}) = \log P(\text{phrase}) - \max(\log P_{bg}(\text{phrase}) - \log P(\text{phrase}), 0)$$

(1)

where $\log P(\text{phrase})$ is computed by an N-gram language model trained on the current corpus, and $\log P_{bg}(\text{phrase})$ is based on a N-gram language model trained on a heterogeneous background corpus. Under this model, the probability of a phrase is only discounted if $P_{bg}(\text{phrase}) > P(\text{phrase})$. This method of TF-IDF [Cohen 2002] helps us to find frequent phrases idiosyncratic to the corpus for manual labeling. Our N-gram training uses the modified Kneser-Ney smoothing [Chen 1999] from the MitLM package [Hsu 2008]. The background language model is obtained from Microsoft Web N-gram Services. Given a set of human-labeled phrases, we then train two levels of classifiers on these phrases: a binary Important versus Unimportant phrase classifier, then a one-against-all multi-class classifier for each of the phrase categories described above, except Miscellaneous, which serves as the
background category for Important phrases. The classifiers use NER [Zhang 2003, Ratinov 2009] plus bag-of-words features. For the Date and Time phrases, we make use of the rule-based SUTime library [Chang 2012].

For collocation we use a Gaussian kernel on the distance between mentions of different phrases. Formally, the collocation probability of one occurrence of a phrase, given a set of other phrases is defined as:

\[
P(p_1 | p_2, p_3, \ldots, p_k) = \exp(-\beta \sum (S(p_i) - S(p_j))^2)
\]

where \(S(p_i)\) is the sentence number where \(p_i\) occurred. Given the defined conditionals, one can compute the joint probability \(P(p_1, p_2, p_3, \ldots, p_k)\) and use a threshold to determine which phrase set goes to an event template and ultimately feeds the visualization.

IV. VISUALIZING UNCERTAINTY IN EVENT TRIGRAPHS

Uncertainty can refer to statistical uncertainty, ranged values or missing data [Pang 1997] and can be introduced during acquisition, processing or visualization. In this paper, since the underlying data has been extracted from unstructured text narratives, uncertainty begins at the data acquisition phase. Linguistic uncertainties include the temporal, “By this time, it had to be 11:00 o’clock at night,” locative, “I guess that would be North End Avenue,” and entity, “At this point I had my five guys” [WTCTF 2001]. One of the major goals of this project is to visualize the triadic relationship amongst entity, time, and location while reflecting these various uncertainties. To convey the time-location-entity data and the associated uncertainty, we introduce a trigram based visualization called an Event Trigraph. Our approach is novel, as most research in this area addresses geospatial or scientific visualization [Skeels 2010, MacEachren 2005, Lodha 1996, Grigoryan 2004, Wittenbrink 1996], and according to [Skeels 2010] focuses on adding glyphs [Wittenbrink 1996, Lodha 1996], geometry, animation [Lodha 1996, Gershon 1992] or sonification [Lodha, “Sonic” 1996].

An Event Trigraph builds from individual 3-tuples of time, location, and entity as derived from the phrase mining method above. Events are represented as triangles with the aforementioned elements as vertices connected with weighted edges. The weights represent the confidence value in the relation as obtained via statistical text mining methods. These confidence values describe the connection both between elements and amongst the trigram. Figure 2 shows a trigram with confidence values and elements.

![Figure 2. Event trigraph with uncertainty values and voids](image)

Confidence values peak at 1, denoting a certain relationship between elements. To reduce visual clutter, we employ details on demand [Shneiderman 1996] and filtering capabilities on the dataset. Users can also drag-and-drop events over other events and manually revise confidence values. Since the users loads multiple documents at a time, this feature enables users to visually associate events and entities in different documents -- the main forte and a unique aspect of our visualization. Furthermore, since the resulting data structure is a weighted network graph, graph theory algorithms can be applied, making it scalable beyond our test corpus of 511 interviews, each of which contain on the order of 100 triadic relations.

V. CONCLUSION

As reports are processed, the diagram builds in complexity and supplements individual event trigraphs with ones that potentially correlate. In the figure above, John-Café-12:00 pm is correlated to Jill-Playground-12:00pm across the time element, and Jill-Playground-12:00 pm correlates to Eleanor-Playground-12:00 pm across both time and location elements. This aggregated event trigraph builds to represents a corpus, and offers a method for correlating the coreference of entities across documents, and potentially the identification of unnamed entities. For the WTC corpus, this system may let the peripheral stories of bystanders come to light as strongly as the stories of the witnesses themselves.
REFERENCES


