INT J COMPUT COMMUN, ISSN 1841-9836
7(5):857-864, December, 2012.

Using Opinion Mining Techniques for Early Crisis Detection

A. Iftene, A.L. Ginsca

Adrian Iftene, Alexandru-Lucian Ginsca
"Alexandru Ioan Cuza" University of Iasi,
Faculty of Computer Science
E-mail: adiftene@infoiasi.ro,
lucian.ginsca@infoiasi.ro

Abstract: The goal of our research is to investigate the use of internet monitoring in crisis management using linguistic processing and text mining techniques. We present a system that detects and classifies events on topics and, using an altered opinion mining workflow, detects geographical entities related to these events and the sentiments expressed towards them. The results are displayed in customized GoogleMaps views, indicating areas with a potential risk, such as natural disasters, unfavorable weather or threatening protests. All the processing is done in real time and, depending on the monitored sources, our work could be of used as a population warning system, but it could also be useful for regional or local authorities in managing intervention time and resources by prioritizing the situations for which they have to act.
Keywords: Opinion mining, Event detection, Crisis management.

1 Introduction

In recent years, an increasing trend regarding Internet monitoring for multiple types of crises (political, weather, terrorist or health related) can be observed. An example of the use of web mining in conflict detection that fits in such a trend is described in [5], in which the authors focus on the 2011 African protests. The main differences between their approach and ours are that we propose a system that can be easily adapted to different types of crises, identifies threats at a more localized level (districts, streets) and that we use a purely automatic approach, whereas they combine web mining with human reports.

The main components of our system allow us to monitor a collection of newspapers and to save on our computers all the news. After they are processed locally, we detect the main topics and we find the most relevant topic(s) for a particular crisis scenario. In next step, named entities and users opinions are identified, and based on them the risks are identified. Accordingly to the values attached to locations, a Google Map is created and a set of "islands", some with potential risks and some without risks are generated on the map. In the last step a user receives an alternative path for a pair of (start location, end location), which avoids as much as possible the islands with negative scores (those drawn in red, with potential risks) and that approaches the "islands" with positive scores (those drawn in green, without potential risks).

In the following chapters, we will present the main components of our system. Given the fact that the main purpose of the research described in this paper is to incorporate opinion mining elements in crisis detection systems, we will insist more on those components that are used for this task such as event detection and a proper graphical representation of the results and less on those concerning strictly opinion mining. Also, we present the new resources especially created for this task, such as those used for the identification of streets and the detection of opinions related to crisis management.

2 System Description

Below, we present the most important components of our system. We will also show how these are used for monitoring protests that took place in Romania between 13 and 26 January, 2012.

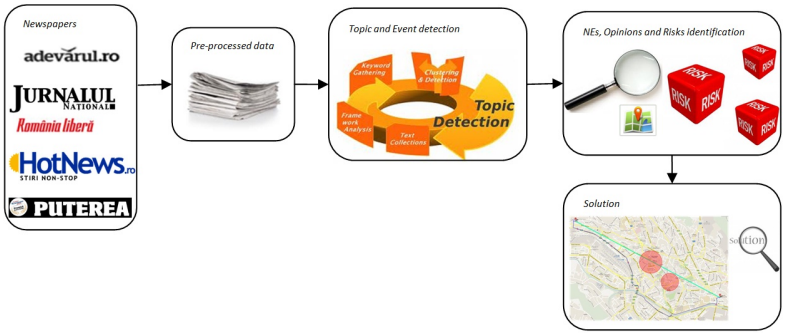

Figure 1: *System architecture*

2.1 Newspaper Monitoring

A number of newspapers are monitored using RSS feeds and the articles are gathered using a crawling component. For our case study, we monitored five newspapers Adevarul, Hotnews, Jurnalul, Puterea and Romania Libera (see Figure 1). This data was stored locally and it was preprocessed (from html pages were removed links, photos, menus, special characters).

2.2 Identification of potential risk events

After the initial processing step, we identify news articles containing mentions of potential risk events. For this task, we propose a novel approach that uses topic models for event classification and semantic similarities for event recognition. On a collection of news articles, after a couple a preprocessing steps, such as lemmatization and stop word elimination, a topic model is applied in order to identify a predefined number of topics present in that collection. In general, topic models describe the topics by a list of relevant words for each topic [1]. This raises a problem, because we want to be able to identify a particular event with minimum human intervention. In order to solve this issue, we use semantic similarities between the words that describe the topic, given by the topic model, and a small manually built vocabulary for an event. Such a similarity measure will be able to find the topic most related to the followed event. In the next sections, we will detail each of these components.

2.3 Topic Detection

To identify groups of topics we have used the Latent Dirichlet Allocation (LDA) topic model. LDA represents documents as mixtures of topics that generate words with certain probabilities [2]. We have applied LDA on our protests corpus. Although the news articles were taken so that they correspond to this scenario, we wanted to evaluate the results of LDA over this corpus. For our experiment, we assumed that we track 3 topics.

In the word clouds from Figure 2, we have put the first 10 words for each topic in the descending order of their relevance. The words have been translated into English and their size is directly proportional to the LDA weight. As it can be observed from Figure 2, although 3 topic clusters were formed, all of them have words related to the street protests. This result indicates that LDA performs well even if the number of topics in not known in advance.

Figure 2: *Topics terms word clouds*

Semantic similarity

LDA offers a set of terms for each detected topic in the descending order of their relevance for the topic. For our event detection task, we want to establish which of the detected topics is the most related to our scenario. We address this problem by using semantic similarity measures between the first n LDA words for a topic, and a small vocabulary describing the scenario. This vocabulary can be entirely manually built or it can be automatically extended, although, as it can be seen from our experiments, a cardinality of 5 for the vocabulary is sufficient. For computing the semantic similarity between two terms, we have tested three WordNet semantic similarity algorithms, Wu, Resnik and Lin. Next, we give more details about these measures.

Wu and Palmer measure. The Wu & Palmer measure calculates semantic similarity by considering the depths of the two synsets in the WordNet taxonomies, along with the depth of the least common subsumer [10]. The formula is as follows:

$wuPalmerScore(t_1, t_2) = \frac{2 \times depth(lcs(s_1, s_2))}{depth(s_1) + depth(s_2)}$, where:

s_1: the synset of the first term, s_2:the synset of the second term, $lcs(s_1, s_2)$: the synset of the least common subsumer. This means that $0 < wuPalmerScore \leqslant 1$. The score can never be zero because the depth of the least common subsumer is never zero. The depth of the root of a taxonomy is one. The score is one if the two input synsets are the same.

Resnik measure. This measure also relies on the idea of a least common subsumer (LCS), the most specific concept that is a shared ancestor of the two concepts. The Resnik [9] measure simply uses the Information Content of the LCS as the similarity value:

$resScore(t_1, t_2) = IC(lcs(t_1, t_2))$, where $lcs(t_1, t_2)$: the least common subsumer.

$IC(t) = -log\left(\frac{freq(t)}{maxFreq}\right)$, where:

$freq(t)$: the freaquecy of term t in a corpus, $maxFreq$: the maximum frequency of a term from the same corpus. The Resnik measure is considered somewhat coarse, since many different pairs of concepts may share the same LCS. However, it is less likely to suffer from zero counts (and resulting undefined values) since in general the LCS of two concepts will not be a very specific concept.

Lin measure. The Lin measure augments the information content of the LCS with the sum of the information content of concepts A and B themselves [7]. The Lin measure scales the information content of the LCS by this sum.

$linScore(t_1, t_2) = \frac{2 \times resScore(t_1, t_2)}{IC(t_1) + IC(t_2)}$

Topic set similarity. For computing the semantic similarity between two sets of words using one of the three measures described above, we use the lemma of each term. We propose two different methods for computing the global similarity. In the first case, the final similarity score is obtained using a weighted average over the maximum score obtained by applying a semantic similarity measure on each combination of a term from the first set and one from the second set. In second one, we simply add all the similarity values between each combination of terms. This is suitable for situations where there are a consistent number of similar words with scores less,

but close to the maximum and that would have been ignored by the first formula:

$$globalMaxSim(T_1, T_2) = \frac{\sum_{t_1 \in T_1} max(sim(t_1,t_2) \in T_2))}{|T_1|}$$

$globalAddSim(T_1, T_2) = \frac{\sum_{t_1 \in T_1} \sum_{t_2 \in T_2} sim(t_1,t_2)}{|T_1|}$, where T_1: first set, T_2: second set, $sim(t_1, t_2)$: one of the Wu and Palmer, Resnik or Lin similarity measures.

Event detection evaluation

For evaluation, we have used 10 of the 20 topics from the "The 20 Newsgroups", a widely used corpus for text classification [6]. We have included the following topics: "rec.sport.baseball", "rec.motorcycles", "talk.politics.guns", "alt.atheism", "comp.graphics", "sci.electronics", "sci.med", "sci.space", "soc.religion.christian", "talk.politics.mideast".

In order to evaluate the similarity measures, we have observed which measure captures the similarity between 2 sets of words describing the same topic, while lowering the similarity between sets describing different topics. For the experiments, we have chosen the "baseball", "guns" and "motorcycle" topics. We compare the first $n(5 \leqslant n \leqslant 50)$ relevant words for each topic as identified by the LDA topic model and a set containing the following words: "bike", "tire", "motorcycle", "helmet", "drive". In a first series of experiments in which we compared the Resnik, WuPalmer and Lin similarity measures, Resnik was the single one that found a higher similarity between the sample vocabulary and the "motorcycle" topic words for every instance of n and disregarding the global similarity measure that was used.

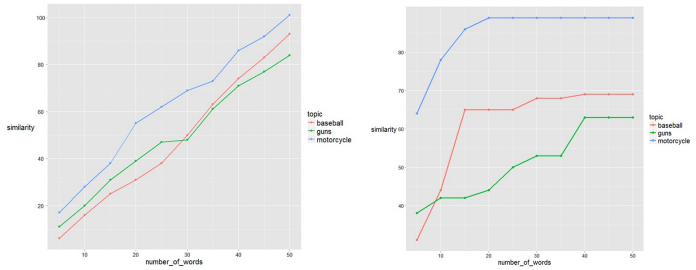

Figure 3: *(a) Progress of globalAddSim (b) Progress of globalMaxSim*

In the next series of tests, we wanted to establish which of the two global similarity measures provides the best results when using Resnik as a similarity between two words. In Figure 3 (a), we present the evolution of the *globalAddSim* similarity between the sample vabulary and the LDA words for each of the 3 topics. In Figure 3 (b), we track the evolution of the $globalMaxSim$ similarity measure. As it can be seen from the two figures, the $globalMaxSim$ provides the best separation between the correct similarity (represented in blue) and the others. Based on the previous experiments, we have chosen the globalMaxSim measure with the Resnik similarity and we keep the first 20 LDA relevant words.

For the experiments, we have used 80% of the data to train the topic model and 20% to evaluate it. Due to the fact that the results of LDA depend on the random initialization of the initial topic distributions, in Figure 4 we have tracked the average accuracy over 3 runs when using a number of topics varying from 2 to 10. The sudden drop in accuracy when using 9 and 10 topics appears due to the inclusion of the "christian" topic, which shares a high number of terms with the "atheism" topic.

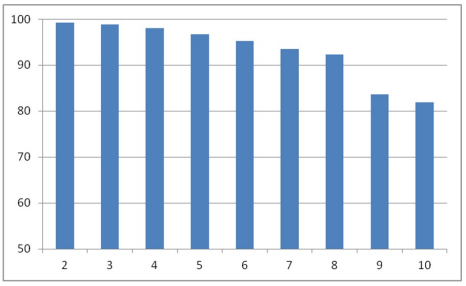

Figure 4: *LDA Accuracy*

2.4 Data Processing

Identifying locations, regions, keywords: Named entity identification is a crucial component of our application. The correct identification of "islands" with potential risks on a map depends on the accuracy of this component. For this, we use the Romanian language specific resources [4] that contain cities (Iasi, Bucuresti, Ploiesti, etc.), regions (Bucovina, Moldova, Transilvania, etc.). Additionally, we have added a new type of named entity, "street", for which we have created specific resources (containing the major streets of big cities "Iasi, Bulevardul Independentei", "Bucuresti, Calea Victoriei", etc.) and specific rules to identify streets (Street + *entity*, Boulevard + *entity*, etc.). To refine the localization to smaller inner city regions, we have added a new category, "area" that captures locations such as Pacurari district, center of Iasi, Arch of Triumph Square, etc. Using rules designed for this specific type of entities, our system is able to capture location related expressions, such as "the area between street A and street B" or "the area of the building A".

The quality of the module responsible with NE identification and with NE classification remains the same, after the adding of a new type of named entity "street". Thus our evaluation on 538 files with 2,806 entities of "street" type shows that the quality of NE identification component is around 92% and the quality of NE classification component is around 67%. Problems in NE identification: incorrect spelling (Pieta Universitatii), anaphora resolution (only Piata or only Universitate are not identified), other problems (Cotroceni, Primaria, Prefectura, sediul PDL, in fata simbolului Iasiului, Palatul Culturii, Piata Romana). The most frequent problems in NE classification are related to ambiguity situations when from the context we cannot conclude that the NE is a person name or a street name.

2.5 Identification of Opinions

While the majority of the opinion mining systems have in common the use of a sentiment lexicon, a distinction can be made between rule based and statistical approaches [8]. Due to the fact that we propose a general architecture that needs to be easily adapted to different crisis situations, we use the first type of approach. In this case, switching from a crisis scenario to another will require only the changing of the lexicon, whereas in the statistical approach, a significant training corpus would be required for each scenario. We use manually built resources to identify opinion keywords that signal the (good, bad, etc.), amplifiers (most, more, etc.), diminishers (less, etc.), Negation (not, never, etc.) [3]. Additionally, for our "street protests" test case we have added 21 specific words for conflict monitoring, such as "protest", "conflict", "fight".

The application described in [3] allows us to calculate the valences for groups of feelings and pairing named entities with scores based on the distance, punctuation and context. Based on

these values we are able to classify named entities previously identified based on the opinion expressed towards them. Although obtaining a general opinion, as defined by the opposites positive/negative still can provide valuable clues concerning a potential threat, by adapting the context to a specific issue (protests, weather etc.) and introducing a relevant seed vocabulary, we can shift the semantics of the opinion towards the problem in hand. For example, we will be able to present the results in terms of degrees of danger.

2.6 Building a Customized GoogleMaps Map for Events

The purpose of this component is to create a map based on GoogleMaps, in which the locations and critical values calculated for them will be placed. Depending on these "islands", we will inform concerned people of the potential risks that appear and we find a solution which can be adopted. In order to build the GoogleMap we use JavaScript and accordingly with sentiment values associated to locations we create "red islands" (when the values are negative) with potential risks and "green islands" (when the values are positive) without potential risks.

3 Results

For the street protests scenario, our application has identified 698 news articles in this topic from the 13 to 26 January 2012 time span a total of 21,156 named entities, out of which 1,166 locations. In Figure 5 (a) we can see how cumulated sentiment values were greater with negative values, between 15 January and 19 January, and similar the numbers of mentions per days and per entities are higher in the same period (Figure 5 (b)). After analyzing the newspapers we see how between 15 January and 19 January were fights and confrontations between guardians and football teams supporters and a part of protesters.

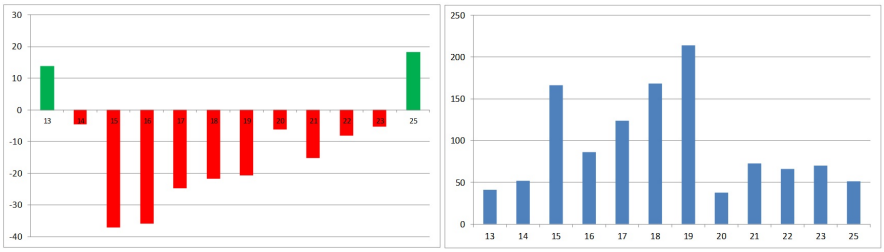

Figure 5: *(a) Cumulated sentiment values by days (b) Location type entities mentions by day*

After aggregation of 1,166 values on location entities we obtained 198 unique entities. From 198 location entities, 61 represent countries, 5 represent cities outside Romania, in 12 cases entities are marked as cities, but because we didn't perform anaphora resolution we didn't know at which cites they are referring to, and in 12 cases we classify wrong identified these entities in ambiguous cases. In the end we have 101 cities, with 51 with negative values associated, 23 with value 0 associated, and with 27 with positive values. The cities with lowest values are Bucuresti (-38.8), Cluj-Napoca (-19.41), Tulcea (-10.13), Sibiu (-9.09), Slobozia (-8.57). For these cities we can see in the next Figure the associated red circles. The cities with highest values are Mangalia (2.61), Medgidia (2.61), Bacau (2.48), Vaslui (2.05), Brasov (1.45) and we can see in next Figure 6 the associated green circles. Although a part of these entities are without diacritics, GoogleMaps API is able to identify correct the entity and to put it on the map.

Optimal path between cities

In the above scenario after we put red and green circles, we want to find an optimum path

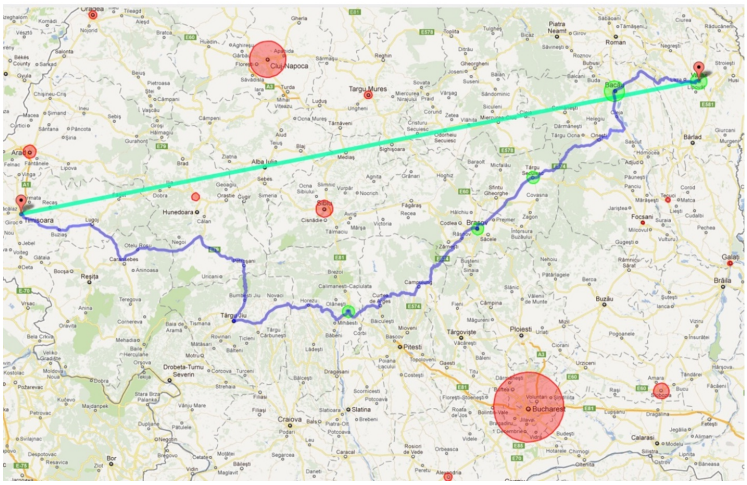

Figure 6: *The "optimum" path which passes near red islands and pass through green islands between cities*

from Vaslui to Timisoara. In this scenario, the shortest way uses a part of the cities which have associated red circles. Our algorithm is able to find another path (longer) which passes near the "red islands" and prefers the ways near the "green islands". The algorithm uses a graph structure corresponding to the streets network, in which the nodes correspond to cities and edges to streets. In this structure we associate sentiment values to nodes and length values to edges. When all nodes have associated values equals with zero, the solution identified by our algorithm corresponds to the shortest path. The graph structure was built by us starting from the main streets from Romania. On this structure we apply our algorithm which built step-by-step partial solutions starting from start node, until the partial solution reach the end node and the partial solution become the final solution. At every step is possible to insert penalties when the partial solution crosses red islands (with potential risks) and add bonuses when the partial solution crosses green islands (without potential risk).

In the above case we can see how our solution doesn't prefer red islands (Sibiu, Deva, Cluj-Napoca, Tirgu-Mures) which are on the shortest way and prefer green islands (Vaslui, Bacau, Tirgu Secuiesc, Brasov, Rimnicu Vilcea) from a longer way. A first type of problems with our algorithm is related to the following situation: even if some cities have negative values close to zero our algorithm prefers to pass near them. For these situations we must identify a threshold value, and under this value we will ignore the negative value. Other problems are with some cities which have false positive values and in a wrong manner attract routes, and other cities have false negative values and influence the final solution.

Optimal path between streets locations in the same city

When we want to find a solution for a path between two locations which are in the same city the things is different. At street level we identify 2,806 entities (172 streets, 150 boulevards, 2,299 squares, 185 areas). If in the cases of streets and boulevards the GoogleMaps API is able to put these entities on the map, for specific squares and areas it is not able to do this. In such cases we built an additional resource which specifies the GIS coordinates for them. In this way we can generate red islands at city level.

4 Conclusions

We have described in this paper a system that, starting from an opinion mining architecture, can be used to detect and localize different types of threats and which offers an expressive visualization for a rapid and targeted intervention.

We have proposed a global system architecture that can be easily adapted for the detection of multiple crisis situations with minimum human intervention. An important aspect of our approach is the detection of crisis events. Due to the fact that the opinions expressed towards geographical entities are strongly related with that context, by correctly identifying the context, even without using a dedicated vocabulary for a particular situation, and we can use any opinion mining configuration with the results being relevant for the detected context. We have proposed a novel use of topic models with semantic similarities to indentify and classify the main topics from a news collection. Our results, both for English and for Romanian, have shown that by using Latent Dirichlet Allocation we can obtain an accurate language independent topic distribution and by using a WordNet based semantic similarity, we can successfully correlate a discovered topic with any given topic. More than that, we identify various inner city location and we offer a clear visualization suggesting alternative routes to bypass potential dangerous areas.

Bibliography

[1] D. Blei, J. Lafferty, Topic models, *Text Mining:Theory and Applications*, Taylor and Francis, London, UK, 2009.

[2] D. Blei, A. Ng, M. Jordan, Latent Dirichlet allocation, *Journal of Machine Learning Research*, 3:993-1022, 2003.

[3] A. L. Ginsca, et al., Sentimatrix - Multilingual Sentiment Analysis Service, *In Proceedings of the 2nd Workshop ACL-WASSA*, 2011.

[4] A. Iftene, D. Trandabat, M. Toader, M. Corici, Named Entity Recognition for Romanian. *In Knowledge Engineering, Principles and Techniques. Selected Papers*, 49-60, 2011.

[5] F. Johansson, et al., Detecting Emergent Conflicts through Web Mining and Visualization, *In Proceedings of the European Intelligence and Security Informatics Conference*, 2011.

[6] K. Lang, Newsweeder: Learning to filter netnews, Proceedings of the Twelfth International Conference on Machine Learning, pages 331-339, 1995

[7] D. Lin, An information-theoretic definition of similarity, *In Proceedings of the International Conference on Machine Learning*, Madison, August, 1998.

[8] B. Pang, L. Lee, Opinion mining and sentiment analysis, *Foundations and Trends in Information Retrieval*, 2(1-2), 1-135, 2008.

[9] P. Resnik, Using information content to evaluate semantic similarity, *In Proceedings of the 14th International Joint Conference*, 1995.

[10] Z. Wu, M. Palmer, Verb semantics and lexical selection, *In 32nd Annual Meeting of the Association for Computational Linguistics*, pages 133-138, Las Cruces, New Mexico, 1994.