Topic Sentiment Change Analysis

Yu Jiang[1], Weiyi Meng[1], and Clement Yu[2]

[1] Department of Computer Science, Binghamton University,
Binghamton, NY 13902, USA
{yjiang5,meng}@cs.binghamton.edu
[2] Department of Computer Science, University of Illinois at Chicago,
Chicago, IL 60607, USA
yu@cs.uic.edu

Abstract. Public opinions on a topic may change over time. *Topic Sentiment change analysis* is a new research problem consisting of two main components: (a) mining opinions on a certain topic, and (b) detect significant changes of sentiment of the opinions on the topic and identify possible reasons causing each such change. In this paper, we discuss topic sentiment change analysis using data on the Web. We adopt probabilistic topic model and language grammar based sentiment analysis techniques, and integrate them together into a topic level sentiment analysis method. This method is capable of analyzing sentiment and identifying sentiment changes of a given topic from a set of documents covering this topic and possibly other topics. In addition, as the contents of relevant topics are differentiated, our method is also able to identify hot events which are possible causes of a sentiment change. Experimental results show that our method is very promising.

Keywords: sentiment change, opinion mining, topic model.

1 Introduction

Nowadays, many people express their opinions online, such as in product reviews, Web blogs, forums and online discussion groups. The large volume of individual opinions on diverse topics and the ease of access to these opinions make the Web a valuable source for opinion mining and analysis. Sentiment analysis, as a new notion of opinion mining, has been an active research area of the research community in recent years. Pang et al. [15], Dave et al. [3] and Gamon et al. [4] adopt machine learning classifiers to perform sentiment classification for documents. Liu et al.[10] consider the overall opinion about a product as a combination of opinions of its aspects and perform detailed sentiment analysis on product reviews. All these works consider documents as the basic interest unit of sentiment analysis. Recently, Mei et al. [13] introduce a probabilistic topic model into sentiment analysis, and their model is able to track opinions and even sentiment dynamics of a topic within many documents. It proposes **topic sentiment analysis (TSA)**, whose basic interest unit is topic. Given a document D, a topic's content is hidden in D, and its sentiment is not necessarily consistent

P. Perner (Ed.): MLDM 2011, LNAI 6871, pp. 443–457, 2011.

with the overall sentiment of D. Besides, a topic may appear in many documents. In the corpuses like web blogs and news articles, topic sentiment analysis usually is more important than document sentiment analysis. Take the text snippet in Figure 1 as an example, the author expresses an objection to offshore gas operation, because he/she opposes to potential pollutions such operations may cause. We can see that here the sentiment target is the topic of offshore gas operation. In general, a document is a mixture of sentiments of related topics; and there are interesting relations between the related topics. Investigating people's sentiment toward an interested topic and how related topics influence people's sentiment on the topic is an interesting and challenging research problem.

I would like to express my support for those in Barbados who are seeking redress from Shell Oil with regard to the pollution of your beautiful homeland. Currently on my own Island of Ireland , Shell are attempting to build an offshore gas operation despite the objections of the local population. Peaceful protesters were subjected to horrific police brutality for trying to protect their homes.

Fig. 1. A web blog text snippet

In this paper, we study Topic Sentiment Change Analysis (TSCA), a new problem whose goals are to detect the sentiment and its changes toward an interested topic (called *target topic*) over time, and further identify the possible causes of the changes. More specifically, to perform TSCA for a given topic, we first collect a sufficient number of related documents within a certain period of time; with these documents sorted by time, we then learn the sentiment distribution of the topic over time and identify the occurrences of changes of the sentiment; finally, we identify the related topics that influence each change of sentiment. There are several significant challenges in performing TSCA.

First, as far as we know, topic level sentiment analysis is still an open problem. Despite topic being a more interesting term compared with other terms like document for human being, it is not a good object for sentiment analysis for computer program. The content of a topic usually is sparse and "hidden" within documents, so some kind of approximation should be introduced to identify the contents of a topic. No matter what kind of topic representation (probabilistic topic model, a set of keywords, or other form) is used, we should guarantee that the contents of different topics in documents can be correctly separated. Also, many researchers [4,7,12,14] agree that the semantics of expression is critical in performing sentiment analysis. Take the sentence *"He is cute, still I do not like him."* as an example; there are two positive words *cute* and *like*, but the overall sentiment of the sentence is negative because of the existence of negation (word *not*) and contrast (word *still*) relations in it. This example shows that the polarity of a sentiment expression (positive, neutral, or negative) is not only determined by the opinionated words, but also by its grammatical structure. We believe a good TSA model ought to: (1) divide document content into different topics, and (2) use mature techniques (e.g., language grammar based rules, etc) to perform sentiment analysis of topics.

Second, how to discover sentiment changes, i.e., how to obtain the properly aggregated sentiment distribution over time is a new problem. Unlike stock prices which are continuous data streams, sentiments toward certain topic are only expressed by people with some degree of randomness. As a result, public opinions about a topic may appear sparsely sometimes and fluctuate heavily from time to time. Aggregation/regression is needed to reduce the impact of randomness to acceptable level.

Third, assuming all topics have been properly extracted and their sentiment distributions over time have also been properly calculated; given a corpus, how do we identify a sentiment change event? How do we identify the possible causal events of such change? The answers to these questions could help a company identify a design defect based on user comments, or help government adjust public policy in alignment with public opinion. Techniques need to be developed to tackle them.

In this paper we tackle the above challenges. Our contributions include (a) a TSA framework which integrates the probabilistic topic model [6] and language grammar based sentence level sentiment analysis technique together; (b) a simple but effective time partition method and some rules to identify sentiment change events as well as their causal events; and (c) some metrics for evaluating the ranking of candidate causal events for each sentiment change. As far as we know, this is the first study on finding events which cause sentiment change. Our experimental results indicate that our solution is effective.

The rest of this paper is organized as follows. In Section 2, related works are reviewed. In Section 3, we provide an overview of our approach. In Section 4, we describe our TSA framework, which consists of topic content division and topic sentiment evaluation. In Section 5, we discuss the time period partition and the discovery of sentiment changes and topic burst events and their relevance evaluation. Experimental results are presented in Section 6. Finally, we conclude the paper in Section 7.

2 Related Works

There has been a large amount of research on opinion mining of Web data in recent years. Some (e.g., [5,8,19]) study the semantic orientation (positive or negative) of English words, especially adjectives. Some (e.g., [12]) focus on sentence level sentiment analysis; they use language grammar based rules and adopt words' semantic orientation information to obtain the sentiment polarity of a sentence. A majority of the remaining works focus on document level sentiment analysis; some (e.g., [3,4,15,18]) use machine learning classifiers to determine the overall sentiment of a document, some [10] performs feature level sentiment analysis and offer sentiment summary of product reviews as commercial products usually contain several important features based on which users can make choice. These works offer the fundamentals of sentiment analysis; but they cannot be directly used to solve the topic level sentiment analysis problem.

Recently, some researchers propose some probabilistic topic models to perform sentiment analysis. Among them, Lin and He [9] use a joint sentiment topic model for document sentiment classification; Mei et al. [13] propose a new probabilistic model containing two special sentiment topics, which is able to perform topic sentiment analysis with some post processing. It offers the first serious study on TSA. Their work is the most relevant to our work, and it actually inspired our method on TSA reported in this paper. All these models can identify topics properly; but their sentiment analysis result is not convincing, as they use only the occurrences of opinionated words and the document level co-occurrence information of opinionated words and topical words, which may assign a sentiment expression of one topic to another topic just because they both occur in the same document. There are also some studies on topic's evolving trends over time. Wang and McCallum [20] integrate the time factor into probabilistic topic model and learn topic's distribution over time based on the topic model. Mei et al. [13] count the number of word occurrences of each topic to estimate topic's dynamics over time. As far as we know, there is no directly related work about sentiment change cause discovery so far.

3 Solution Overview

Our proposed solution to TSCA consists of two modules as shown in Figure 2. The first module performs topic-level sentiment analysis and it has two steps: (a) extract topics in the corpus and divide the contents of documents into these topics, while retaining the completeness of sentiment expressions; (b) perform TSA on each extracted topic. The second module performs the topic sentiment change analysis based on the results of the first module. This module also has two steps: (a) partition the corpus into subsets based on proper time periods and generate each topic's popularity and sentiment distributions over these time periods; (b) compute the sentiment change events and other related events of each topic based on the obtained distributions; and for each sentiment change event, rank the top relevant events as its potential causes. The system modules reflect our understanding to the main challenges mentioned in Section 1.

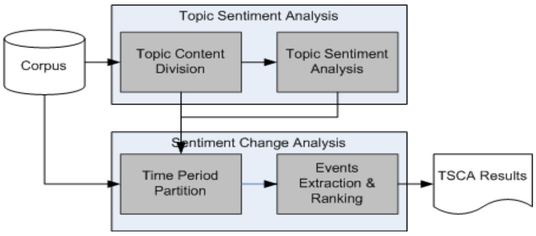

Fig. 2. TSCA system overview

4 Topic-Level Sentiment Analysis

4.1 Topic Content Division

The first step to perform TSCA is to learn the hidden topics in the corpus. In the text mining community, usually a topic is either modeled as a mixture of weighted words in probabilistic models [6,9,13], or modeled as a group of representative phrases/keywords [10]. The probabilistic models can represent each topic naturally as a cluster of words, but they are unable to determine the exact occurrences (like sentences) of a topic within a document. The phrase model is direct and easy for detecting a topic's content in documents, but usually it may miss some latent topic phrases/keywords.

Neither of the above techniques can be used directly for TSCA's topic division task. As we consider finding all contents of a topic to be important for TSCA, we choose to use a classic probabilistic model (i.e., the PLSI model [6]) to first extract the topics in corpus. The PLSI method assumes that each word in a document is sampled out as follows: first select a topic from the document specific multinomial distribution of the topics, then select a word from the topic specific multinomial distribution of words. Given a corpus *C*, the PLSI topic model can be learned using the Expectation-Maximization algorithm. In the learned topic model, the probability that an occurrence of word w in document d represents topic z is defined in Formula 1. Here $P(z|d)$ denotes topic z's probability in document *d*, $P(w|z)$ denotes word w's probability in topic *z*, and $P(d)$ denotes the probability of d in *C*.

$$P(w,d,z) = P(w|z)P(z|d)P(d) \tag{1}$$

We can use the learned topic model of PLSI to estimate a sentiment expression's belonging topic. Suppose a sentiment expression corresponds to a language unit u (e.g., a sentence), then its topic can be estimated as u's most likely topic as is determined by Formula 2. Here Z is the set of all topics in corpus *C*. We say the sentiment expression's sentiment target is $topic(u)$, or it belongs to $topic(u)$.

$$topic(u) = argmax_{z \in Z} P(z,u) = argmax_{z \in Z} \prod_{w \in u} P(w,d,z), u \in d \tag{2}$$

In reality, the size of a sentiment expression may vary; it could be a subsentence, a sentence, or a paragraph. Accurate sentiment calculation is possible only when the complete sentiment expressions are maintained. As how to extract accurate sentiment expressions is not our focus in this paper, in our current system, we choose sentence as the default language unit of sentiment expression, because it is the minimum language unit for a complete expression, in which a majority of sentiment expressions are kept.

Some sentences may have multiple topics. For example, the subject and the object of the sentence *"The high oil price leads support to offshore drilling."* are two different topics. In this case, Formula 2 may introduce errors by assigning such sentences to just one topic. To analyze how multi-topic sentences will impact TSA, we allow them to be assigned to multiple topics when certain condition is satisfied. Specifically, given a sentence u and its most likely topic *z*, we also consider another topic *z'* as u's topic if $log(P(z|u)/P(z'|u)) < \alpha$, where α is a threshold (we set α to 1.0 in our experiment, indicating that the two topics' probabilities are of the same order of magnitude). We also analyzed the our experimental corpuses to investigate the distribution of sentences having different number of topics. We found that about 92% of them have 1 topic, 7.3% have 2 topics, and only < 1% of them have 3 or more topics. Thus in our current implementation, we allow a sentence to be assigned to at most two topics.

4.2 Topic Sentiment Evaluation

After topic content division, all contents of a topic are collected together in units of sentences. We can either use the machine learning classifier method [4,15] or language rules based accumulative method [7,12] to compute the overall sentiment of an interested topic. In our un-supervised system, we use the second approach. We first compute each sentence's sentiment polarity value, then sum them up to get a topic's overall sentiment.

Our algorithm to determine the sentiment value of a sentence utilizes some existing techniques in [5,7,8,12,17]. Generally, we use two kinds of information to obtain the sentiment polarity of a sentence. One is the opinionated words in it; we build up two dictionaries of frequent positive and negative words respectively. The positive and negative tagged English word lists from General Inquirer [17] are used as the two seed sets, and then the synonym relation in WordNet [1] are used to expand them [8] to make the final dictionaries. The other is the grammatical relations; especially the two most frequent sentiment sensitive grammatical relations: negation and contrast. We developed several lists of indicating words to identify them; such as *not* for negation and *but* for contrast. Given a sentence, when there is no grammatical relation involved, we simply count the number of opinionated words in it, if there are more positive words, then it is of positive polarity, and vice versa. When negation and/or contrast are involved, we adopt some empirical rules to perform sentiment analysis. These rules consider the found grammatical relation indicating words together with the typed dependencies [11] on them, and determine the overall sentiment polarity of the sentence based on the semantics of the grammatical relations. Due to space limitation, the details of the rules are not included in this paper.

We use $s(u)$ to denote a sentence's sentiment polarity, the possible value of $s(u)$ can be 1 (positive), 0 (neutral) or -1 (negative). Each sentence contributes an equal score to its topic's sentiment. When a sentence has multiple topics, its sentiment will be counted in the computation of the sentiment of all its topics as in our current expedient solution.

The sentiment of a topic z within document d is formally represented as a percentage triple (Formula 3). Here u is a sentence of document d belonging to topic z, and X is the normalization factor, which is the number of sentences of topic z in document d.

$$S(z,d) = (\frac{|\{u|s(u) > 0\}|}{X}, \frac{|\{u|s(u) = 0\}|}{X}, \frac{|\{u|s(u) < 0\}|}{X}) \tag{3}$$

5 Sentiment Change Analysis

In this section, we first introduce our time period partition scheme. Then we discuss how to compute the sentiment distribution and the popularity distribution of topics based on the time period partition. At last, we discuss how to discover sentiment change events and their causal events based on these two distributions.

5.1 Time Period Partition

As we have discussed in Section 3, Time Period Partition (TPP) is important for effective TSCA. Formally, we define $P = (t_1, t_2, ..., t_n)$, a sequence of consecutive and non-overlapping time periods covering all documents in corpus C, as a TPP of C. A TPP P also partitions the whole documents set of C into a sequence of document sets.

The basic partition schemes are incapable of supporting TSCA. Given a corpus of limited size, if we adopt equal-length time period TPP, the time periods when the target topic is not hot would contain too few documents, making the post aggregated data analysis prone to the influence of noise. Alternatively, if we adopt a TPP scheme which ensures that each time period contain the same number of documents, each time period may contain enough documents; still the boundaries between the time periods may not match the start and/or end time of real sentiment changes, which may produce inaccurate aggregated results.

Based on our observation on the testing corpuses, we found that a sentiment change of a topic is usually accompanied by hot discussions related to the topic. This indicates that an increase of the popularity of a topic suggests a possible sentiment change to the topic. As far as the target topic is concerned, because all documents in the corpus are related to the target topic (a topic specific query is submitted to a search engine to collect the documents of the corpus), a sudden change of the number of documents over time suggests a possible sentiment change to the target topic. Our document cardinality based TPP scheme is developed based on this observation.

Given a timestamp ts_i in the whole time interval of corpus C (each timestamp is one day in our experiments), intuitively, we consider it as the start of an increasing period of document counts if 1) $|D(ts_i)|$ is much larger than $|D(ts_i-1)|$ (locally significant), and 2) $|D(ts_i)|$ itself is relatively large (globally significant). Here $|D(ts_i)|$ denotes the document count of given timestamp ts_i, and $ts_i - 1$ is the preceding timestamp of ts_i. In our algorithm, we use the product of the above two factors to determine whether or not ts_i should be considered as the start time (boundary) of a new time period (Formula 4).

$$BoundaryScore(ts_i) = \frac{|D(ts_i)|}{|D(ts_{i-1})|} log(|D(ts_i)|) \tag{4}$$

The pseudo code of the TPP algorithm is shown in Figure 3. The recursive program “Partition” defines a series of time period splitting operations in top-down fashion. In each turn, given the start timestamp (inclusive) and end timestamp (exclusive) of the whole time interval, for each timestamp ts in-between, if it is a qualified boundary, i.e., the document counts of both the split time intervals before and after it are greater than a predefined value *minDocCnt* (line 3), then we compute its boundary score (Formula 4) and add it into the candidate queue

Q (line 4). After all timestamps are tested, if the candidate queue Q is empty, no partition is needed and the given time interval is considered as a single time period in the final partition (line 9). Otherwise, we choose to split the given time interval at the timestamp ts with the highest *BoundaryScore* in Q(line 6), and make recursive calls of the program "Partition" to get the partition results of the split sub time intervals, finally union the returned results together with ts to form the final partition result of the given time interval. (Note that a partition result is represented as a series of time period boundaries.)

```
Partition(start, end)
1.  init empty priority queue Q;
2.  for each timestamp ts between timestamps start and end,
3.      if both DocCnt(start, ts) and DocCnt(ts, end) >= minDocCnt,
4.          add ts into Q with priority value BoundaryScore(ts);
5.  if Q is not empty,
6.      pick the timestamp ts with the highest BoundaryScore from Q;
7.      return Union(Partition(start, ts), {ts}, Partition(ts, end));
8.  else,
9.      return {};
```

Fig. 3. Time Period Partition Algorithm

In our experiment, we set *minDocCnt* to an empirical constant value 70, such that the aggregated topic sentiment in each time period is relatively accurate. Note that when the number of documents in the corpus is sufficiently large, there is no need to set the minDocCnt threshold. Instead, we can modify the above algorithm to just select the top N most significant boundaries where N is a user specified constant. Suppose P is a TPP. The popularity distribution H and sentiment distribution S of a topic z are defined as follows:

$$\{H(z,t)|t \in P\} \quad where \quad H(z,t) = \sum_{d \in D(t)} P(z|d) \ / \ |D(t)| \tag{5}$$

$$\{S(z,t)|t \in P\} \quad where \quad S(z,t) = \sum_{d \in D(t)} P(z|d)S(z,d) \ / \sum_{d \in D(t)} P(z|d) \tag{6}$$

Here $S(z,d)$ is the sentiment triple of topic z in document d calculated using Formula 3, and probability $P(z|d)$ is the contribution factor of document d to the sentiment of topic z. The sum operation on $S(z,d)$ in Formula 6 is a vector sum, so $S(z,t)$ is also a percentage-value triple. With the two distributions, we define two kinds of events.

Topic Burst Event (TBE): Let z represent a topic and $P_{i,j}$ represent the time interval composed of n=*j*-*i*+*1* time periods t_i, t_{i+1}, ..., t_j. A tuple $e = (z, P_{i,j})$ is called a TBE if the average popularity H of topic z in the time interval $P_{i,j}$ is larger than the average popularity of z in the n time periods immediately before $P_{i,j}$ as well as the average popularity of z in the n time periods immediately after $P_{i,j}$. Given a TBE e, we use $Topic(e)$ and $Period(e)$ to denote, respectively, its corresponding topic and time interval. Furthermore, we use $Strength(e)$ to denote the strength of e which is as the ratio of the average popularity of $Topic(e)$ in $Period(e)$ over the average popularity of the two n time periods before and after $Period(e)$. Based on this definition, some TBEs may overlap with each other; for example, $P_{i,j}$, $P_{i-1,j+1}$ and $P_{i,j-1}$ all contain t_i. In implementation, we only keep the TBE that has the largest Strength value. This ensures that the boundaries of real topic burst event be correctly identified, and there should be no overlapping TBEs of the same topic.

Sentiment Change Event (SCE): A triple $o = (z, pol, P_{i,j})$ is an SCE of topic z in time interval $P_{i,j}$ (here *pol* is the polarity of the sentiment change, either positive or negative) if (1) z's sentiment keeps on increasing or decreasing in $P_{i,j}$, (2) the absolute difference in sentiment between t_i and t_j is larger than a threshold value β, and (3) the length of $P_{i,j}$ should contain at least K time periods (in our experiment, $K = 3$ is used). Condition (2) is used to avoid counting random fluctuations of sentiment as SCEs. Empirically, in our experiment, we set β to be $1/10$ of the maximum difference in P, i.e., the difference between the highest and lowest sentiments in all time periods in P. Condition (3) is used to avoid counting random fluctuations on sentiment distribution as sentiment changes. We use $Topic(o)$, $Period(o)$ and $Polarity(o)$ to denote the SCE o's topic, time interval, and polarity, respectively.

5.2 Cause Identification

The causes of an SCE can be indicated by many kinds of information hidden in the corpus, such as causal sentences in documents, topic relevance, etc. In this paper, we use the coherence information of different topic contents. Particularly, we consider TBEs as the candidate causes of SCE. For each SCE o, we rank each TBE e according to its relevance to the SCE. The relevance considers the following factors: 1) The significance of e as measured by $Strength(e)$. 2) The time relevance between o and e. Intuitively, for e to have a chance to be relevant to o, e should occur around the same time as o. In this paper, we use $|Period(o) \cap Period(e)|/|Peroid(o) \cup Period(e)|$ to calculate the time relevance between o and e. 3) The content relevance between *Topic(e)* and *Topic(o)* during *Period(o)*, denoted as $CR(e, o)$, which is the relevance of the contents of $Topic(e)$ and the content of $Topic(o)$ with sentiment polarity $Polarity(o)$ during $Period(o)$.

Given a sentence u, we define the set of its adjacent sentences within distance γ as $Adjacent(u)$, and all sentences in $Adjacent(u)$ are considered relevant to u. Then for an SCE o, we define $Sent(o)$ to be the set of sentences which have sentiment polarity $Topic(o)$, belong to topic $Topic(z)$, and are within the document set of $Period(o)$. The numeric value of content relevance between a topic z and o can be calculated using Formula 7. It is the logarithm value of the total counts of the sentences of topic z which are adjacent to the sentiment expressions of o. In our experiment, we set γ to 1, ensuring that only closely adjacent contents are considered relevant.

Based on the discussion above, we can use the products of the three factors to compute the relevance value of a TBE e to an SCE o (Formula 8).

$$CR(z, o) = log(\sum_{u \in Sent(o)} |\{v | z \in topic(v), v \in Adjacent(u)\}|) \tag{7}$$

$$R(e, o) = Strength(e) \ \frac{|Period(o) \cap Period(e)|}{|Period(o) \cup Period(e)|} \ CR(Topic(e), o) \tag{8}$$

Given a sentiment change SCE o, we rank all TBEs in descending order of their relevance to o and consider the top-ranked TBEs to be the potential causes of o. To compute the CR values efficiently, we perform a one-time scan on the whole corpus in unit of sentence; for each topic, each time period, we maintain two counters of the number of sentences which are adjacent to the positive and negative sentiment expressions (also sentence) of the target topic, respectively. Then the CR value between any topic and SCE can be easily calculated based on these counters. Given a corpus of K topics and N time periods in its TPP, there are less than $K*N$ TBEs and less than N SCEs in total, and the time complexity of the whole ranking process is $O(K * N^2)$, a reasonably small number.

6 Experiments

6.1 Experiment Setup

We use Web blog documents from the famous blogging site wordpress.com as the corpus source. First, we use Google's Blog Search Engine to get the blog pages related to our interested topics using keyword queries. Next, we extract article content from each page using a wrapper, while keeping the grammatical structures of the article as complete as possible. We save the result as "plain-text" format corpus. After that, we adopt stemming (WordNet Stemmer) and stop-words removal [2] on each plain text document, and save this result as "word-sequence" format corpus. The "word-sequence" corpus is used for generating the probabilistic topic model and the "plain-text" corpus is used for producing the topic content division.

Two corpuses are used in our experiments. One (C1) has 600 documents retrieved using keyword query "offshore drilling" with time range from August 2008 to December 2008. Another (C2) has 1000 documents with keyword query "airport security", from December 2009 to early January 2010. Most documents in these corpuses have about 20 to 50 sentences.

We compare our proposed TSA method with some other methods. One is a baseline method called **DSM** (Document Sentiment Model); this method sums up sentiment scores of all the sentences in document to form the target topic's sentiment. Another method is **HMM** proposed by Mei et al. [13]; it introduces two sentiment topics (positive & negative, with prior information) into a modified PLSI model, then uses a Hidden Markov Model [16] induced from the modified PLSI model to compute the adjacencies of topical words and opinionated words, which is used to estimate each topic's sentiment dynamics over time. In our implementation of the HMM method, the in-corpus occurrence frequency information of words in the two opinionated word dictionaries are used to generate its sentiment topic priors; and each document is considered as a separate observation sequence for training (the concatenation of all documents, which is used in [13], is too long for our blog corpus). We implement two variations of our TSA method: **TSA-W** and **TSA-G**. While computing a sentence's sentiment polarity, TSA-W only utilizes information of opinionated word occurrences, while TSA-G also consider the influences of negation and contrast relations in sentence sentiment analysis.

6.2 Topic Identification Results

Table 1 shows some of the topics learned from the two corpuses using the PLSI model, among them target topics are shown in boldface. The representative words of each topic are the top words with the highest probabilities. The names of non-target topics are provided manually based on the top 20 words in each topic. Due to limited space, only a small number of topics learned are listed in the table. For C1, topic "drilling" denotes the target topic "offshore drilling"; topic "sea-warming" represents the global warming issue, topic "oil-price" denotes the long lasting oil price increase issue during 2008; and topic "election" represents "energy plan", which was an important issue during the US presidential election in 2008. For C2, topic "security" denotes the target topic "airport security"; topic "Newark" denotes the security violation incident at Newark airport on Jan 3, 2010 in the afternoon; topic "NW253" denotes the failed body-bomb attack on airline NW253 on Nov 25, 2009. We can see that all the identified topics are real events/issues and they are all related to the target topics. And the top 5 words of most of these topics represent the topics very well.

Table 1. Topics learned using PLSI

Corpus	C1				C2		
Topic	**drilling**	sea-warming	oil-price	election	**security**	Newark	NW253
Words	engineer marine construction material drilling	sea warming global ice level	oil energy drilling price gas	mccain obama tax people Palin	Scanner Body machine Image Privacy	passenger Flight Crew Muslim Man	Flight Passenger Abdulmutallab Plane Mutallab

6.3 Sentiment Change Analysis Results

TPP Result. Figure 4 depicts our document cardinality based TPP result of Corpus C2. The histogram bars denote the document count of each day from 12/01/2009 to 01/13/2010, the vertical lines in the figure denote the time period boundaries found using the Algorithm in Figure 3.

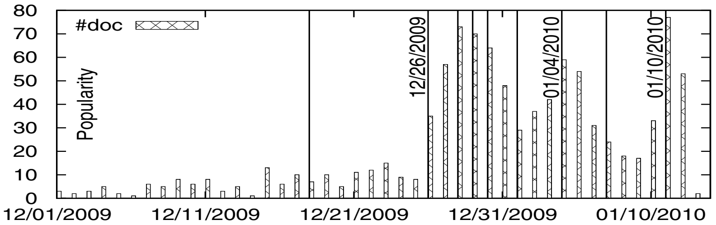

Fig. 4. Time Period Partition of C2

There are three boundaries (12/26/2009, 01/04/2010 and 01/10/2010) with high BoundaryScore (see Formula 4), they match three real events exactly: the first one is the failed body-bomb attack on Flight NW 253 on 12/25/2009; the second one is the security violation incident at Newark Airport on 01/03/2010; and the third one is the news that the suspect of the above Newark event was arrested by police on 01/09/2010. This result verifies that our TPP scheme is effective for TSCA.

SCE Identification. For corpus C1, we expect a positive sentiment change on offshore drilling in the summer of 2008, in correspondence to the historically high oil price that appeared at the end of July 2008 (called oil-price-event). For corpus C2, we expect a positive sentiment change on Airport Security (i.e., people expressed more support and stronger measures against terrorism) following the failed body-bomb attack on Flight NW 253 on Dec 25, 2009 (called NW253-event). We also expect a negative sentiment change reacting to the security violation incident at Newark airport on Jan 3, 2010 which caused the airport shutdown and many complaints on airport security (called Newark-event).

The sentiment distributions of the target topics of the two corpuses are depicted in Figure 5. We use the $\frac{pos-neg}{pos+neg}$ derived from the sentiment triple (pos, neu, neg) (see Formulas 3 and 6) as the normalized sentiment measure. From Figure 5(a) we can see that the bars for the DSM method do not fluctuate much compared to the other three methods. This confirms that document level sentiment analysis method is incapable of performing topic level sentiment analysis, especially for multi-topic long blog articles. The HMM method's sentiment distribution has big fluctuations, the sentiments of time period $t1_3$ and $t1_8$ are even opposite to those of the other methods. This is because that for many documents, the hidden markov model recognizes them to be of single topic word sequence, which causes many sentiment expressions to be assigned to wrong topics. Among the bars of the two TSA methods, there is a steady increase of positive sentiment from $t1_2$ (01/23/2008-06/12/2008) to $t1_4$ (07/01/2008-07/27/2008) during the first half year of 2008, which matches the support increase for offshore drilling caused by high oil prices. In Figure 5 (b) for C2, we omitted the distributions of the HMM and DSM methods as they are not convincing. We can see a long positive SCE from $t2_3$ (12/26/2009-12/27/2009) to $t2_7$ (01/01/2010-01/03/2010); this matches our expectation in response to the NW253-event. For

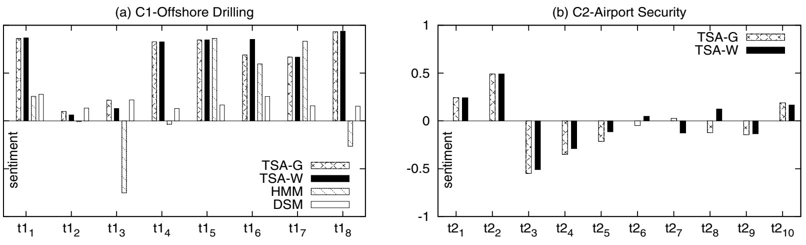

Fig. 5. Sentiment Distribution of Target Topics

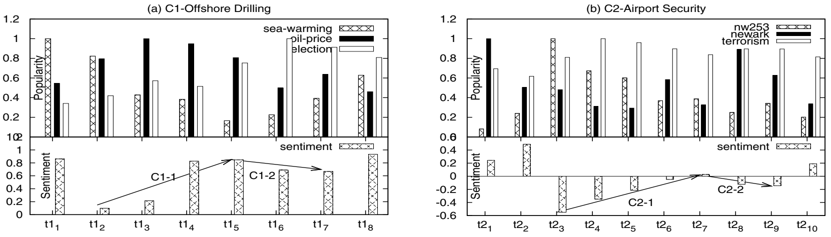

Fig. 6. Popularity Distribution of Related Topics and SCEs of Target Topics

the TSA-G method, there is another negative SCE from $t2_7$ to $t2_9$ (01/07/2010-01/10/2010) in response to the Newark-event.

Based on the results of the two corpuses, the TSA methods are able to identify most sentiment changes effectively while the others cannot. It indicates that keeping the complete sentiment expressions is important in TSA. Comparing the sentiment distributions of TSA-W and TSA-G, we can see in most cases their sentiment values are close; TSA-G outperforms TSA-W as it correctly identified the second sentiment change in corpus C2. It suggests that expression semantics analysis (negation and contrast as in our implementation) is helpful for accurate sentiment evaluation.

Causal TBE Identification & Ranking. In Figure 6, the normalized popularity distributions of the related topics (upper part) are put together with the SCEs of the target topics (lower part) for the two corpuses. There are four SCEs as being marked with arrows, in which the three sentiment changes we expected are all found (C1-1, C2-1 and C2-2) based on our SCE definition. One other SCE is also detected (C1-2); it might be caused by the popularity decrease of the "oil price" topic.

The topics of the top three TBEs for the three SCEs we expected are listed in Table 2. The topics that represent our expected causes are highlighted in boldface. Due to space limitation, we cannot provide the details about the TBEs. The overlap of the time periods of SCEs and their causal TBEs can be observed in Figure 6. These TBEs can be indicated by peak values on the topics' popularity distributions. In Figure 6(a), the time interval of the TBE of topic "oil-price" is exactly the same to the time interval of SCE C1-1 ($t1_2$ to $t1_5$, or 01/23/2008-08/09/2008). In Figure 6(b), the time interval of the TBE of topic "NW253" ($t2_3$ to $t2_5$) is within the time interval of SCE C2-1 ($t2_3$ to $t2_7$, or 12/26/2009-01/03/2010); and the time interval of the TBE of topic "Newark" ($t2_8$ to $t2_9$) is within the time interval of SCE C2-2 ($t2_7$ to $t2_9$, or 01/01/2010 to 01/10/2010). Go back to Table 2, we can see that for the three expected SCEs, our ranking method identified their right causes in the top three candidate TBEs (we mark both "Terrorism" and "NW253" as valid causes of this positive SCE of airport security C2-1 since their topics are tightly related).

Table 2. SCEs and their top relevant TBE topics

SCE	1st TBE topic	2nd TBE topic	3rd TBE topic
C1-1	**Oil-price**	Energy	Election
C2-1	**Terrorism**	TSA related	**NW253**
C2-2	**Newark**	Muslim	Obama

Overall, based on our experimental result on two corpuses of different time coverages, we conclude that 1) our TSA method is effective as topics in document are separately considered, and sentiment analysis is performed based on the unit of semantic expression (i.e., sentences); and 2) our proposed TSCA method is able to identify sentiment changes and their possible causal events.

7 Conclusions

In this paper, we proposed a solution to the Topic Sentiment Change Analysis (TSCA) problem which is a significant problem but has not been seriously studied before. Our solution tackled the main issues in TSCA. The first is topic sentiment analysis; our method uses probabilistic topic model PLSI for topic content division and perform the sentiment analysis on complete sentiment expressions. The second is the discovery of sentiment changes and their possible causes. Our method first divides documents into different time periods and detects steady topic sentiment changes in consecutive time periods, and then identifies significant TBEs related to each sentiment change.

In the future, we plan to improve our solution for TSCA along several directions: (1) try other topic models for topic content division; (2) integrate more mature language grammar based sentiment analysis techniques to evaluate a sentence's sentiment; (3) extend the TSA method by allowing sentiment expression unit of various granularities; and (4) incorporate sentence level NLP based causal relationship analysis to enhance the accuracy of identifying the causes of sentiment changes.

Acknowledgment. This work is supported in part by the following NSF grants: IIS-0842608, IIS-0842546 and CNS-0958501.

References

1. http://wordnet.princeton.edu/
2. ftp://ftp.cs.cornell.edu/pub/smart/english.stop
3. Dave, K., Lawrence, S., Pennock, D.M.: Mining the peanut gallery: opinion extraction and semantic classification of product reviews. In: WWW, pp. 519–528 (2003)
4. Gamon, M.: Sentiment classification on customer feedback data: noisy data, large feature vectors, and the role of linguistic analysis. In: ACL. COLING 2004. Association for Computational Linguistics, Stroudsburg, PA, USA (2004)
5. Hatzivassiloglou, V., McKeown, K.: Predicting the semantic orientation of adjectives. In: ACL, pp. 174–181 (1997)
6. Hofmann, T.: Probabilistic latent semantic indexing. In: SIGIR, pp. 50–57. ACM, New York (1999)
7. Jia, L., Yu, C.T., Meng, W.: The effect of negation on sentiment analysis and retrieval effectiveness. In: CIKM, pp. 1827–1830 (2009)
8. Kamps, J., Marx, M., Mokken, R., de Rijke, M.: Using wordnet to measure semantic orientation of adjectives, vol. IV, pp. 1115–1118 (2004)
9. Lin, C., He, Y.: Joint sentiment/topic model for sentiment analysis. In: CIKM, pp. 375–384 (2009)
10. Liu, B., Hu, M., Cheng, J.: Opinion observer: analyzing and comparing opinions on the web. In: WWW, pp. 342–351 (2005)
11. de Marneffe, M.C., Maccartney, B., Manning, C.D.: Generating typed dependency parses from phrase structure parses (2006)
12. Meena, A., Prabhakar, T.V.: Sentence level sentiment analysis in the presence of conjuncts using linguistic analysis. In: Amati, G., Carpineto, C., Romano, G. (eds.) ECiR 2007. LNCS, vol. 4425, pp. 573–580. Springer, Heidelberg (2007)
13. Mei, Q., Ling, X., Wondra, M., Su, H., Zhai, C.: Topic sentiment mixture: modeling facets and opinions in weblogs. In: WWW, pp. 171–180 (2007)
14. Pang, B., Lee, L.: Opinion mining and sentiment analysis. Foundations and Trends in Information Retrieval 2(1-2), 1–135 (2007)
15. Pang, B., Lee, L., Vaithyanathan, S.: Thumbs up? sentiment classification using machine learning techniques. CoRR cs.CL/0205070 (2002)
16. Rabiner, L.R.: A tutorial on hidden Markov models and selected applications in speech recognition. Proceedings of IEEE 77(2), 257–286 (1989)
17. Stone, P.J., Dunphy, D.C., Smith, M.S., Ogilvie, D.M.: The General Inquirer: A Computer Approach to Content Analysis. MIT Press, Cambridge (1966)
18. Thomas, M., Pang, B., Lee, L.: Get out the vote: Determining support or opposition from congressional floor-debate transcripts. In: EMNLP, pp. 327–335 (2006)
19. Turney, P.D., Littman, M.L.: Unsupervised learning of semantic orientation from a hundred-billion-word corpus. CoRR cs.LG/0212012 (2002)
20. Wang, X., McCallum, A.: Topics over time: a non-markov continuous-time model of topical trends. In: KDD, pp. 424–433 (2006)