Scalable Detection of Sentiment-Based Contradictions

Mikalai Tsytsarau
University of Trento
Trento, Italy
tsytsarau@disi.unitn.eu

Themis Palpanas
University of Trento
Trento, Italy
themis@disi.unitn.eu

Kerstin Denecke
L3S Research Center
Hannover, Germany
denecke@L3S.de

ABSTRACT

The analysis of user opinions expressed on the Web is becoming increasingly relevant to a variety of applications. It allows us to track the evolution of opinions or discussions in the blogosphere, or perform product surveys. The aggregation of sentiments and analysis of contradictions is another important application, which becomes effective since we are able to capture the diversity in sentiments on different topics with more precision and on a large scale. Though, there is still a need for a scalable way of sentiment aggregation with respect to the time dimension, which preserves enough information to capture contradictions.

In this paper, we are focusing on the problem of finding sentiment-based contradictions at a large scale. First, we define two types of contradictions, depending on the distributions of opposite sentiments over time. Second, we introduce a novel measure of contradiction based on the mean value and the variance of sentiments among different texts. Third, we propose a scalable method for identifying both types of contradictions at different time scales. We evaluate the performance of our method using synthetic and real-world datasets, as well as a user-study. The experiments demonstrate the effectiveness of the proposed method in capturing contradictions in a scalable manner.

1. INTRODUCTION

During the recent years we have been witnessing the Internet becoming an open platform, where people can express their opinions and can be heard. There are many services that allow people to publish information and opinions, such as blogs, wikis, forums, social networks and others. They all represent a rich source of opinionated information on different topics, which can be analyzed and exploited in various applications and contexts. Sentiment analysis can be used, for example, to learn about a customer's attitude to a product or its features, or to reveal people's reaction to some event. Such problems require a scalable analysis and some form of sentiments aggregation to produce a representative result.

The problem of contradictions, or sentiment diversity on some topic, has been studied in the context of different research areas, having a slightly varying notion in each case. For instance, in Information Retrieval opposite opinions and sentiments introduce noise to the fact-centric search and must be avoided [14]. In contrast, conflicting sentiments is one of the desired targets of mining of product reviews. Recently proposed methods can aggregate opinions expressed in customer reviews and extract a representative summary of sentiments on a feature-by-feature basis; or they can capture and aggregate sentiments on some topic among different texts [8].

Although aggregated sentiments do represent some information on contradiction, this information may be biased. For example, if two opposite sentiment values are averaged, the result may have a neutral polarity. The information about the contradiction is then lost. On the other hand, representative sentiments (which best describe opposite opinions) are likely to capture the meaning of contradiction, but not its level. Therefore, this problem essentially requires a consistent definition and new methods to deal with it.

In this paper, we introduce a framework[1] that defines the concepts of aggregated sentiment, sentiment variance and contradiction with respect to the time dimension, and formulates relevant problems of contradiction discovery. We say that we have a contradiction when there are conflicting opinions for a specific topic, which is a form of sentiment diversity. This kind of contradiction can occur at one specific point of time or throughout a certain time period. Furthermore, a contradiction can occur within one text when an author presents different opinions on the same topic, or across texts when different authors express different opinions on the same topic. We further extend this framework of contradiction detection by focusing on its performance and effectiveness for large-scale datasets.

Our method operates on sentence-level sentiments, which are represented in a continuous scale. This allows us to exploit different approaches for sentiment detection, which can be plugged in our framework. The use of mean and variance for contradiction detection allows our method to be fast and linearly scalable on the number of texts, which is an important feature for large-scale analysis. Tests on real datasets, as well as a user-study, demonstrate that our approach is able to efficiently and effectively identify contradictions.

The main contributions of this work can be summarized as follows.

- We formally define the problem of contradiction detection, and further describe two variations of the problem, namely, *synchronous* and *asynchronous* contradictions.
- We present an approach for contradiction detection, which is based on fine-grained sentiment extraction. Moreover, we describe techniques that enable this approach to scale to very large data collections.
- We experimentally evaluate the proposed approach using several synthetic and real datasets. The results show the effectiveness and scalability of our solution. In addition, we perform a user-study that demonstrates the usefulness of the proposed framework.

The remainder of this paper is structured as follows. In Section 2 we discuss the related work, and in Section 3 we formally define the problem. We present our approach for detecting and storing contradictions in Section 4 and Section 5, respectively, and the experimental evaluation in Section 6. We discuss our experiences in Section 7, and conclude in Section 8.

[1] Some preliminary ideas have appeared as a poster [16].

2. RELATED WORK

In the past few years, we have witnessed an increasing research interest in the area of blog analysis and specifically in opinion mining [13]. Contradiction analysis is a rather new research area. In particular, contradictions in opinions as considered here, have not been addressed before. Harabagiu et al. [6] present a framework for contradiction analysis that exploits linguistic information such as negation or antonymy as well as semantic information, such as types of verbs. De Marneffe et al. [3] introduce a classification of contradictions consisting of seven types that are distinguished by the features that contribute to a contradiction (e.g., antonymy, negation, numeric mismatches). They define contradictions as a situation where 'two sentences are extremely unlikely to be true', and describe a contradiction detection approach to their textual entailment application [12]. Ennals et al. [5] describe an approach that detects contradicting claims by checking whether some particular claim entails (i.e., has the same sense as) one of those that are known to be disputed. For this purpose, they have aggregated disputed claims from Snopes.com and Politifact.com into a database. Additionally, they populated this database by selecting explicit statements of contradiction or negation from web texts.

The above approaches are based on linguistic analysis and textual entailment. In contrast, our approach is based on statistical principles and intended for a large-scale operation, where pairwise comparisons of texts may not be computationally efficient. In addition, we are considering a time dimension for contradiction, which allows us to introduce such new types as, for example, change of opinion (asynchronous contradiction). To the best of our knowledge, this problem has not been studied so far.

Problems related to the identification and analysis of contradictions have also been studied in the context of social networks and blogs. A recent work by Liu et al. [10] introduces a system that allows to compare contrasting opinions of experienced blog users on some topic. In contrast, we take into account the opinions of all web users, regardless of their expertise. Clustering accuracy as an indicator of blogosphere topic convergence was proposed by [17]. By analyzing how accurate clustering is in different time intervals, one can estimate how correlated, or diverse, blog topics are. Such an approach can also be adapted to opinion contradictions as well, by replacing topic feature vectors by sentiment feature vectors. Our work goes beyond trend analysis by automatically recognizing contradictions regarding some topic within and across documents.

Analysis of product reviews is another opinion mining task that is close to contradiction analysis. A system for mining the reputation of products in the Web is described in [11]. A similar approach is proposed by the Opinion Observer system [9] that focuses on summarizing the strengths and weaknesses of a particular product. Even though the above studies consider both positive and negative opinions, they do not aggregate these two classes. In our approach, we describe an effective way for performing this aggregation, which leads to more insights on the user opinions.

Chen et al. [2] study precisely the problem of conflicting opinions on a corpus of book reviews, which they classify as positive and negative. Their main goal is to identify the most predictive terms for the above classification task, and visualize the results for manual inspection. However, the results are only used to visualize opposite opinions without further aggregation. It is up to the user to visually inspect the results and draw some conclusions. In contrast, we propose a systematic and automated way of performing sentiment aggregation, revealing contradictions, and analyzing the evolution of these contradictions over time.

3. PROBLEM DEFINITION

The problem we want to solve in this paper is the efficient detection of contradicting opinions[2] (on specific topics).

Usually, a particular source of information covers some general topic T (e.g., *health, politics*) and has a tendency to publish more texts about one topic than another. Yet, within a text, an author may discuss several topics. When using the term 'text' we refer either to the entire web document or its individual sentences. With the term sentence we assume a particular piece of text expressing an opinion about a certain topic, which can not be split into smaller parts without breaking its meaning. For each of the topics discussed in some text, we wish to identify the sentiment expressed towards it. In this study, we restrict ourselves to identifying and recording the intensity of these sentiments, which we represent as numbers. In the following, we refer to sentiment polarity simply as *sentiment*.

DEFNINITION 1 (SENTIMENT). *The sentiment S with respect to a topic T is a real number in the range $[-1, 1]$ that indicates the polarity of the author's opinion on T expressed in a text. Negative and positive values represent negative and positive opinions respectively, while the absolute value of sentiment represents the strength of the opinion.*

Apart from computing sentiments for individual texts, we also need to compute the polarity on some topic aggregated over multiple texts (that may span different authors, as well as time periods).

DEFNINITION 2 (AGGREGATED SENTIMENT). *The Aggregated Sentiment μ_S expressed in a collection of documents $\mathcal{D}$ on topic T, is defined as the mean value over all individual sentiments assigned in that collection. μ_S is defined on the same range of $[-1, 1]$ as sentiments and calculated as follows: $\mu_S = \frac{1}{n}\sum_{i=1}^{n} S_i$, where n is the cardinality of $\mathcal{D}$.*

By comparing the sentiment values of different collections of texts, contradictions are identified as follows.

DEFNINITION 3 (CONTRADICTION). *There is a contradiction on a topic, T, between two groups of documents, $\mathcal{D}_1, \mathcal{D}_2 \subset \mathcal{D}$ in a document collection $\mathcal{D}$, where $\mathcal{D}_1 \bigcap \mathcal{D}_2 = \varnothing$, when the information conveyed about T is considerably more different between $\mathcal{D}_1$ and $\mathcal{D}_2$ than within each one of them.*

In the above definition, we purposely not specify exactly what it means for a sentiment value to be very different from another one. We define contradiction on a *pairwise* basis, where we evaluate the disagreement between two groups of documents in a collection. In this case, the similarity of information within each group serves as a reference point, providing a basic disagreement level. This definition can lead to different implementations, and each one of those will have a slightly different interpretation of the notion of contradiction. We argue that our definition captures the essence of contradictions, without trying to impose any of the specific interpretations. Nevertheless, in Section 4, we propose a specific method for computing contradictions, which incorporates many desirable properties.

When identifying contradictions in a document collection, it is important to also take into account the time in which these documents were published. Let $\mathcal{D}_1$ be a group of documents containing some information on topic T, and all documents in $\mathcal{D}_1$ were published within some time interval t_1. Assume that t_1 is followed by time interval t_2, and the documents published in t_2, $\mathcal{D}_2$, contain a conflicting piece of information on T. In this case, we have a special

[2] For the rest of this document we will use the terms *sentiment* and *opinion* interchangeably.

type of contradiction, which we call *Asynchronous Contradiction*, since $\mathcal{D}_1$ and $\mathcal{D}_2$ correspond to two different time intervals. Following the same line of thought, we say that we have a *Synchronous Contradiction* when both $\mathcal{D}_1$ and $\mathcal{D}_2$ correspond to a single time interval, t.

In order to detect contradicting opinions in collections of texts, we first need to determine all the different topics and then calculate the corresponding sentiments.

PROBLEM 1 (SINGLE-TOPIC CONTRADICTION DETECTION). *For a given time interval τ, and topic T, identify the time regions of a predefined size w, where a contradiction level for T is exceeding some threshold ρ.*

The time interval, τ, is user-defined. As we will discuss later, the threshold, ρ, can either be user-defined, or automatically determined in an adaptive fashion based on the data under consideration. We can also determine all the topics in a dataset that are involved in contradictions, as follows.

PROBLEM 2 (ALL-TOPICS CONTRADICTION DETECTION). *For a given time interval τ, identify topics T, which have high contradiction level, or large number of contradicting regions above some threshold.*

The latter problem is interesting if we want to consider the popularity of certain web topics. Frequent contradictions may indicate "hot" topics, which attract the interest of the community. Due to space limitations, in this paper we only discuss a solution to the first problem, since a solution to the second one is its direct extension. Though, the approach we propose in this work is general, and can lead to solutions for several other variations of the above problem, such as detection of topics with periodically repeating contradictions or with the most frequently alternating *Aggregated Sentiment*.

4. CONTRADICTION DETECTION

Given the problems described before, we propose a three step approach to contradiction analysis, that includes:

- Detection of topics for each sentence,
- Detection of sentiments for each sentence-topic pair, and
- Analysis of sentiments for topic across multiple texts.

Steps one and two can be achieved using existing methods, or adaptations of existing methods. We will refer to these steps as 'preprocessing' and describe them briefly in the following. The focus of this paper is then the contradiction detection approach.

4.1 Preprocessing

For identifying topics per sentence, we apply the Latent Dirichlet Allocation (LDA) algorithm [1], which we extended to work on the sentence level [4]. So sentences are considered as input documents for the LDA and assigned with several most probable topics.

Then, for each sentence-topic pair we assign a continuous sentiment value in the range [-1;1] that indicates a polarity of the opinion expressed regarding the topic. For the sentiment assignment step, we use an existing tool for fine-grained opinion analysis [7]. Nevertheless, this tool can be replaced by any other suitable one that calculates continuous sentiment values at a sentence level. Then we average sentiments over text's sentences having the same topic, to get one sentiment value for each topic in a text.

Based on the analysis described so far, we can now describe our approach for contradiction detection with respect to different topics. In the following paragraphs, we first propose a novel contradiction measure, and then describe two simple approaches aiming at detecting contradictive periods in time.

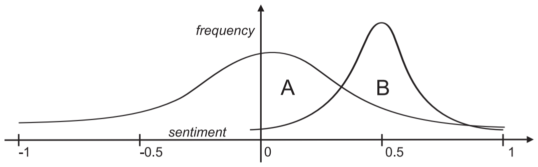


Figure 1: Example of two possible sentiment distributions.

4.2 Measuring Contradictions

In order to be able to identify contradicting opinions we need to define a measure of contradiction. Assume that we want to look for contradictions in a shifting time window[3] w. For a particular topic T, the set of documents $\mathcal{D}$, which we use for calculation, will be restricted to those, that were posted within the window w. We denote this set as $\mathcal{D}(w)$, and n as its cardinality, $n = |\mathcal{D}(w)|$.

In this example, a value of aggregated sentiment μ_S close to zero implies a high level of contradiction because positive and negative sentiments compensate each other. A problem with the above way of calculating topic sentiment arises when there exists a large number of documents with very low sentiment values (neutral documents). In this case, the value of μ_S will be drawn close to zero, without necessarily reflecting the true situation of the contradiction. Therefore, we suggest to additionally consider the variance of the sentiments along with their mean value. The sentiment variance σ_S^2 is defined as follows:

$$\sigma_S^2 = \frac{1}{n}\sum_{i=1}^{n}(S_i - \mu_S)^2 \quad (1)$$

According to the above definition, when there is a large uncertainty about the collective sentiment of a collection of documents on a particular topic, the topic sentiment variance is large as well.

Figure 1 shows two example sentiment distributions. Distribution A with μ_S close to zero and a high variance indicates a very contradictive topic. Distribution B shows a far less contradictive topic with sentiment mean μ_S in the positive range and low variance. For example, a group of documents with μ_S close to zero and a high variance (distribution A on the Figure 1) will be very contradictive, and another group with sentiment μ_S shifted to negative or positive with low variance is likely to be far less contradictive (distribution B on the Figure 1). We note that neither the mean nor the variance can be used independently to identify contradictions. For example, a fairly large variance among sentiments does not lead to a contradiction when only positive or negative sentiments are present. Moreover, a zero mean value may occur even when all posts are neutral, which once again does not indicate a contradiction. When assuming a large number of neutral sentiments in the collection, we have two opposite trends: the average sentiment moves towards zero and sentiment variance decreases. If these trends will compensate each other, the neutral documents would not affect the contradiction value much.

Evidently, we need to combine mean and variance of sentiments in a single formula for computing contradictions. Then, the contradiction value C can be computed as:

$$C = \frac{\sigma_S^2}{(\mu_S)^2} \quad (2)$$

where μ_S is squared so that its units are the same as of σ_S^2.

This formula captures the intuition that contradiction values should be higher for topics whose sentiment value is close to zero, and sentiment variance is large. Nevertheless, the contradiction values

[3] Without loss of generality, in this work we consider windows of days, weeks, months, and years.

generated by this formula are unbounded (i.e., they can grow arbitrarily high as μ_S approaches zero), and does not account for the number of documents n. This latter point is important, because in the extreme where $\mathcal{D}(w)$ contains only two documents with opposite values, C will be very high, and will compare unfavorably to the contradiction value of a different set of T documents with a much higher cardinality.

Incorporating to the contradiction formula the observations made above, we propose the following final formula for computing contradiction values:

$$C = \frac{\vartheta \cdot \sigma_S^2}{\vartheta + (\mu_S)^2} W \tag{3}$$

In the denominator, we add a small value, $\vartheta \neq 0$, which allows to limit the level of contradiction when $(\mu_S)^2$ is close to zero. The nominator is multiplied by ϑ to ensure that contradiction values fall within the interval $[0;1]$. Figure 2(c) shows how a contradiction value depends on ϑ in the denominator. Smaller ϑ values emphasize contradiction points with μ_S close to zero, for example changes of opinion. Larger ϑ values mask this difference, making levels of contradictions more equal. In this study, we used a value of ϑ set at 5% of the expected value of squared sentiment mean, which was effective for its purpose, exhibiting a stable behavior across datasets, without distorting the final results.

W is a weight function aiming to compensate the contradiction value for the varying number of documents that may be involved in the calculation of C. The weight function is defined as:

$$W = (1 + exp(\frac{\overline{n} - n}{\beta}))^{-1} \tag{4}$$

where the constant $\overline{n}$ reflects the average number of topic documents in the window, and β is a scaling factor. This weight function provides a multiplicative factor in the range $[0;1]$ Using W we can effectively limit C when there is a minor number of documents, as well as when this same number of documents increases significantly. What W achieves is essentially a normalization of the contradiction values across different sets of documents, allowing them to be meaningfully compared to each other.

Figure 2 shows the operation of the proposed contradiction function. To demonstrate this, we generated a time series of sentiments for a period of 8000 time units composed of 8000 normally distributed points, half of which follow a custom trend with dispersion 0.125 and another half with dispersion 0.25 and median 0 is acting like noise. Time stamps of all points followed the Poisson distribution with parameter $\lambda = 1$ time units. We have chosen these distributions because they are simple but still resemble the real data. The graph at the top (Figure 2(a)) shows generated sentiments. The bold line in this graph depicts the custom trend, showing an initial positive sentiment that later changes to negative (at time instance t_1), which represents a change of sentiment. There is also a point around time instance t_2, where the sentiments are divided between positive and negative, a situation representing a simultaneous contradiction. Using this dataset, we verify the ability of the C function to capture the planted contradictions.

As can be seen in Figure 2(b), μ_S closely captures the aggregate trend of the raw sentiments. The following two graphs in the figure show the contradiction value, calculated using a sliding window of size 500 and 1000 time units. When we use a window of small size (Figure 2(c)), C correctly identifies the two contradictions at points t_1 and t_2, where the values of C are the largest. Using a larger window has a smoothing effect in the values of C (Figure 2(d)). Nevertheless, we can still identify long-lasting contradictions: In this case, the largest value of C occurs at time instance t_1, corresponding to a change of sentiment that manifests itself across the entire dataset.

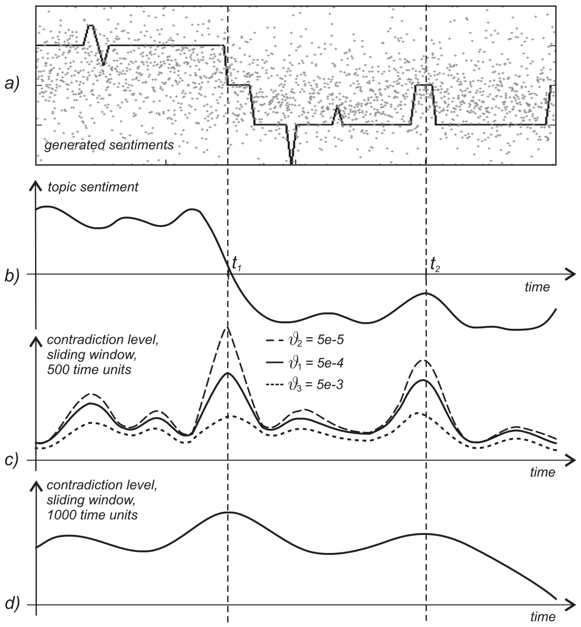


Figure 2: Example of contradiction values computed from a synthetic dataset with two planted contradictions.

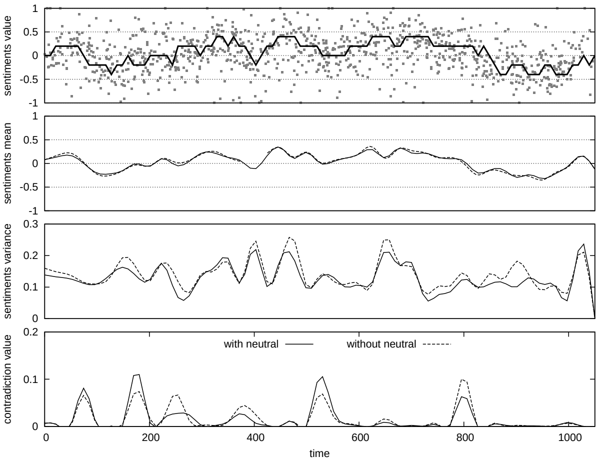


Figure 3: The effect of neutral sentiments on contradiction.

Subjective sentences take a considerably small part in the text when compared to objective statements. So neutral sentiments usually shift the aggregate sentiment towards zero, masking contradictions. Our contradiction formula is designed to compensate such effects by exploiting the sentiment variance.We demonstrate such behavior on another synthetic dataset shown in Figure 3. The bottom graph shows that the proposed formula can successfully identify the main contradicting regions, both with or without neutral sentiments.

5. STORING CONTRADICTIONS

So far we have described a technique for processing web documents to extract sentiments on various topics, and subsequently to use this information in order to identify contradictions. But our final goal is to identify contradictions in large collections of documents, what requires scalable methods. To this end, we demonstrated the need to analyze sentiment information on each topic across different time windows. Assuming this requirement, scalability may be achieved by storing pre-computed values for windows of different size. We now turn our attention to the problem of organizing all these data in a way that will allow the efficient detection of contradictions in large collections of data that span very long time intervals.

An important observation is that the Formula 3 that calculates the contradiction values is based on the mean and variance of the topic sentiment. Remember that aggregated sentiment and sentiment variance can be written as the following:

$$\mu_S = \frac{1}{n}\sum_{i=1}^{n} S_i; \qquad \sigma_S^2 = \frac{1}{n}\sum_{i=1}^{n}(S_i - \mu_S)^2 = \frac{1}{n}\sum_{i=1}^{n} S_i^2 - \mu_S^2$$

In the formula above, n is the number of documents published on topic T in a specific time window (see Definition 2).

We now define the first- and second-order moments of the topic sentiment as $M_1 = \sum_{i=1}^{n} S_i$ and $M_2 = \sum_{i=1}^{n} S_i^2$, respectively. Based on the above discussion, and using the sums M_1 and M_2, we can rewrite Formula 3 as follows:

$$C = \frac{nM_2 - M_1^2}{\vartheta n^2 + M_1^2} W \qquad (5)$$

The above form of the contradiction values formula gives us additional flexibility, since we can now compute the contradiction of a large time window by composing the corresponding values from the smaller windows contained in the large one. We can therefore build data structures that take advantage of this property.

In the next paragraphs, we describe such a data structure, and we show how it can be used to identify contradictions. We also demonstrate that it can be easily maintained in an incremental fashion when new documents are added in the system.

5.1 TimeTree for Contradictions

The need to analyze contradictions at different time granularities predicts a hierarchical structure for contradiction storage. There is a number of ways to organize contradiction values by time. The first solution is to store a time-tree structure for each topic separately. It allows to achieve a scalability on the number of topics, and has a good performance when looking for contradictions at a single topic, but also brings larger update costs, because for each text the storage needs to be parsed as many times as there are topics in that text. Also it makes all-topic queries extremely ineffective, because for each topic we need to navigate through a time structure to find the right interval. The second solution that we propose is to store contradiction values for different topics under the same time-tree structure.

We introduce the TimeTree for managing the information on sentiments and contradictions. The TimeTree is organized around the sentiment moments, M_1 and M_2, and a hierarchical segmentation of time, as outlined in Figure 4. In this example, the time windows are organized on days, weeks, months, and years (though, other hierarchical time decompositions are applicable as well). Using this kind of structure, we can answer queries on *adhoc* time intervals, by dynamically computing the contradiction values based on Formula 5. In the following, we will refer to the levels of the TimeTree as the different *granularities* of the time decomposition, the root node having granularity 0.

Each node in the TimeTree corresponds to a time window, and summarizes information for all documents, whose timestamp is contained in this time window.

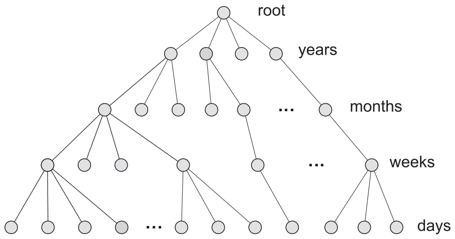


Figure 4: Logical representation of the TimeTree.

5.2 Querying for Contradictions

When trying to detect contradictions, we would like to identify those that have a contradiction value above some threshold. The intuition is that these contradictions are going to be more interesting than the rest in the same time interval. An obvious solution in this case is to define some fixed threshold, ρ, and only report the contradictions above this threshold. We refer to this solution as *fixed threshold*. However, by adopting the above solution, we cannot normalize the threshold to better fit the nature of the data within each time window (that may vary over time and across topics).

In order to address this problem, we propose an *adaptive threshold* technique, which computes a different threshold for each topic and time window as follows. The adaptive threshold ϱ_w for a topic T in time window w is based on the contradiction value C_{w_p} that has been calculated for T in the parent time window of w, w_p, and is defined for each time window and topic as $\varrho_w = p \cdot C_{w_p}$, $0 < p < 1$. In our experience with real datasets, p values between $0.5 - 0.7$ work well. In this work, we use $p = 0.6$.

Note that we cannot achieve the same result by using *top-k* queries (though, they can be complementary to our approach). The reason is that adaptive threshold does not impose a strict limit on the number of contradictions in the result, and can thus report the entire set of interesting contradictions within some time interval.

5.3 Updating the Contradictions

As discussed earlier, the nature of the contradiction function (Formula 5) and the TimeTree nodes allows us to incrementally maintain the TimeTree in the presence of updates. When new collections or individual documents are analyzed, their contribution to the contradiction of the corresponding topics and time windows in the TimeTree can be easily taken into account by updating the set of relevant $\{n, M_1, M_2\}$ values in the nodes of the tree.

In order to reduce update costs, we propose first to accumulate several updates and then submit them in a batch. When new documents arrive, as a preprocessing step, they are aggregated in time windows of the finest granularity of the TimeTree. Then, these aggregated values are used to update the counts and topic sentiment moments of all TimeTree nodes containing respective time windows.

The update cost for each batch of aggregated documents depends on the depth of the TimeTree, d, and the number of topics, $|T|$ (in the worst case), that participate in the time windows relevant to the update. Thus, the complexity can be expressed as $O(d \cdot |T|)$

6. EXPERIMENTAL EVALUATION

As mentioned earlier, the contradiction detection problem has not been considered before. Therefore, no annotated data set is available to measure the quality of the proposed approach in terms of accuracy. Anyway, we applied the algorithm to real world data sets and run several experiments with settings and results described in this section. The objectives of these experiments are to: Analyze the quality of the approach; Study its usefulness from a user perspective; Study the performance of the introduced approach.

6.1 Corpus Description

Our algorithms are applied to a data set of drug reviews collected from the DrugRatingz website[4], a data set of comments to YouTube videos from L3S [15] and a dataset with comments on postings from Slashdot, provided for the CAW2 workshop[5].

The first dataset contains 2701 positive, 352 neutral and 1616 negative reviews for 477 drugs. These reviews are provided by persons that took a specific drug. They describe their personal experience with the drug including contra-indications that occurred.

The second dataset contains approximately 6 million comments to YouTube videos, with an average number of comments per each video of five hundred. Unlike texts in review datasets which usually contain opinions specific to a topic, some of these comments contain information irrelevant to a topic, thus introducing extra noise to sentiment detection.

Our third dataset, Slashdot, is from a popular website for people interested in reading and discussing about technology and its ramifications. It publishes short story posts which often incite many readers to comment on them and provoke discussions that may trail for hours or even days. It contains about 140,000 comments under 496 articles, covering the time period from August 2005 to September 2006. Compared to usually brief comments on YouTube videos, comments from the latter dataset may span for several paragraphs and typically contain many objective statements.

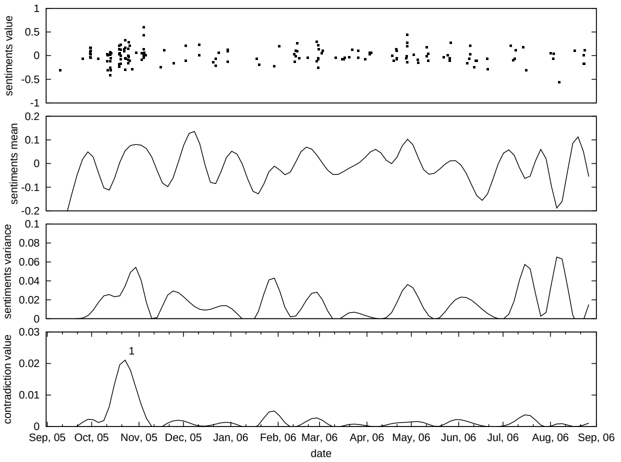
6.2 Evaluation of Contradictions

We now apply the introduced contradiction analysis approach to our datasets. In Figure 5, the top graph depicts the raw sentiment values for the topic "internet government control" taken from the Slashdot dataset, for the time interval September 2005 to September 2006. The following graphs show the aggregated sentiment and variance (two middle graphs), and contradiction values (bottom graph) for the above topic and time interval. Contradiction values have been calculated using a time window of ten days. Note that contradiction values are high for the time windows where topic sentiment is around zero and variance is high, which translates to a set of posts with highly diverse sentiments. These situations are not easy to identify either with a quick visual inspection of the raw sentiments, aggregated sentiments or sentiment variance.

The analysis shows that in this time interval there is one major contradiction (marked 1 in the bottom graph of Figure 5). This contradiction discusses the pros and cons of a law that would give the government more power in controlling the internet traffic, especially personal correspondence. Minor peaks in contradiction level here correspond to the discussion of a possible transfer of jurisdiction and control over top-level domains to United Nations. The table below shows extracts from several opposing posts that contributed to this contradiction. By taking a closer look at the corresponding weblog posts, we find out that the discussion is about restricted internet access and its advantages, while other contradictions contain a general discussion on the possibility of organizing the content by several top-level domains and restricting access to them.

Another example of contradicting posts may be observed in Figure 6, which illustrates conflicting opinions for the topic "Yaz"[6] for a selected time interval. In this case, there was an opinion disagreement on the effectiveness and possible side-effects of this drug.

Evidently, all the discovered contradictions correspond to discussions expressing different points of view on the same topic, and having an automated way of identifying them can be very useful.



PRO: It would be helpful for restricting the flow of information, which is a double edged sword.
PRO: I suppose we better wrap a firewall around our country and not let those damn foreigners access to our internet.
CONS: And what exactly does a neutral Internet do? It takes away the right of anyone who lays down the wires or installs the access points to control what goes through their network. My point: don't complain about taking rights away when you advocate to take rights away.
CONS: While it sounds like a decent idea, I'm really all for the whole uncensored and unregulated internet. I really like my internet the way it is.
CONS: Sure, they can ruin Internet inside USA, but the rest of the world couldn't care less.
CONS: We don't need the FCC regulating the Internet. Not for "neutrality" or any other excuse someone can think of.

Figure 5: Mean, variance and contradiction values of sentiments for the topic "Internet government control".

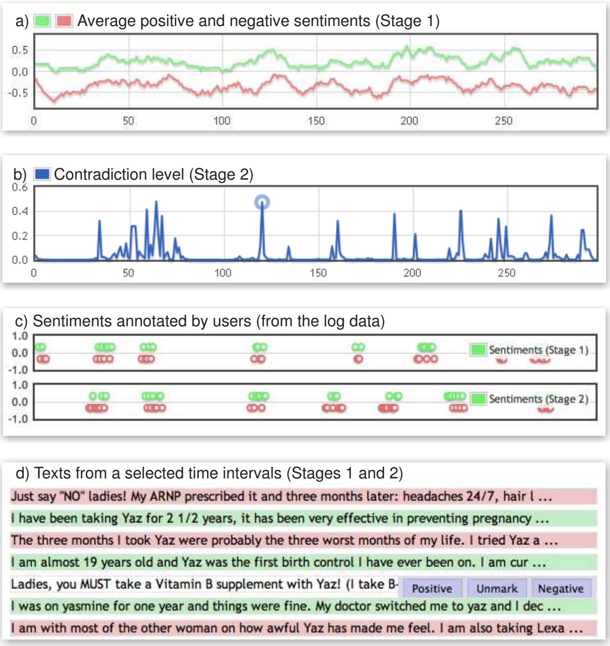
6.3 Evaluation of Usefulness

In the following paragraphs we describe a user study which we conducted in order to evaluate the effectiveness and usefulness of our approach for the task of contradiction discovery.

In our usefulness evaluation, we used four datasets corresponding to opinionated posts for four topics extracted from three diverse real datasets (refer to Table 1). For each topic, we selected a varying number of posts, spanning in time from one to almost three years. The shortest list contained 60 posts, and the largest about 480. Moreover, the quality of posts for topics also differed a lot. The drug review datasets contained primarily brief and concise opinions about drugs; Slashdot topics featured large and detailed comments, with an average size of several paragraphs; YouTube comments were, on the contrary, short and often off-topic.

The group of users consisted of eight persons (PhD students at the University of Trento), and the experiment was conducted as follows. Users were asked to detect groups of contradicting posts for each of the topics in the above datasets (and label the positive and negative posts). We provided users with a web application that featured two approaches to help them identify time-intervals with potentially contradicting posts (see Figure 6): The first approach (marked as "stage 1" in the figure), based on the visualization method proposed by Chen et al. [2], displays to users the intensity over time of the positive and negative sentiments expressed in the posts (Figure 6(a)). The second approach (marked as "stage 2" in the figure) is based on the method proposed in this study, and displays to users a graph that marks the time points at which contradictions were automatically detected (Figure 6(b)). Using our tool, the users could see the time intervals that our tool had identified as contradictory, and could therefore, focus their exploration in these

[4] http://drugratingz.com
[5] http://caw2.barcelonamedia.org/
[6] *Yaz* is a drug for contraception


a) 🟩 🟥 Average positive and negative sentiments (Stage 1)
0.5
0.0
-0.5
0
50
100
150
200
250
b) 🟦 Contradiction level (Stage 2)
0.6
0.4
0.2
0.0
0
50
100
150
200
250
c) Sentiments annotated by users (from the log data)
1.0
0.0
-1.0
🟩 Sentiments (Stage 1)
1.0
0.0
-1.0
🟩 Sentiments (Stage 2)
d) Texts from a selected time intervals (Stages 1 and 2)
Positive
Unmark
Negative

Just say "NO" ladies! My ARNP prescribed it and three months later: headaches 24/7, hair l ...

I have been taking Yaz for 2 1/2 years, it has been very effective in preventing pregnancy ...

The three months I took Yaz were probably the three worst months of my life. I tried Yaz a ...

I am almost 19 years old and Yaz was the first birth control I have ever been on. I am cur ...

Ladies, you MUST take a Vitamin B supplement with Yaz! (I take B-

I was on yasmine for one year and things were fine. My doctor switched me to yaz and I dec ...

I am with most of the other woman on how awful Yaz has made me feel. I am also taking Lexa ...

Figure 6: Annotation page for the dataset "Yaz" demonstrating opposite opinions.

regions. Figure 6(d) shows some posts in a time-interval, which have been marked with positive (green) and negative (red) sentiments. These sentiments values are also illustrated in the overall time-line, depicted in Figure 6(c). In order not to favor any of the two approaches, in our experiments we alternated the approach required to be completed first.

For both approaches, we measured the average time, T_1 and T_2, and the average number of time-intervals examined by the users during the search, N_1 and N_2, needed to identify a single contradiction. Additionally, we asked users to rate the overall difficulty, D_1 and D_2, of completing the task when using each one of the two approaches, according to the following scale: 1- very difficult; 2 - somewhat difficult; 3 - normal; 4 - somewhat easy; 5 - very easy.

The aggregated results (averaged over all the users) of our evaluation are reported in Table 1. We report the improvements[7] we measured when our approach was used (stage 2), compared to the alternative approach (stage 1), computed as follows: $\Delta D = D_2/D_1$, $\Delta T = T_2/T_1$, and $\Delta N = N_2/N_1$.

We observe that when users employed our approach in order to detect contradictions, they were able to identify contradictions faster, requiring 23% less time on average (ranging between 7% and 40%). The biggest improvement was for the topic "Ambien"[8] (ΔT = 0.60), which had a few contradicting posts visible using our approach, but otherwise hard to discover. Our approach also led to a reduction by 28% of the time-intervals examined in order to identify contradictions (ranging between 12% and 42%). The largest reductions were observed for the topics "Zune HD" and "Internet Control" (ΔN = 0.62 and 0.58, respectively), which contained several posts that did not take a position, or were off topic. The average difficulty ratings were also favorable for our approach, which was consistently being marked as more helpful. This difference was most pronounced for the "Zune HD" topic (ΔD = 2.07), which involved many posts. In this case, going through the posts was not easy, and our approach allowed users to focus their search and identify the contradicting posts.

Dataset	**Topic name**	**Size**	**ΔD**	**ΔT**	**ΔN**	P_1	P_2	**ΔP**
Drug Ratingz	Ambien	60	1.50	0.60	0.88	0.70	0.81	1.20
	Yaz	300	1.58	0.93	0.78	0.75	0.95	1.32
Slashdot	Int. control	159	1.17	0.89	0.58	0.37	0.63	2.14
YouTube	Zune HD	472	2.07	0.68	0.62	0.36	0.61	2.09
Average			**1.58**	**0.77**	**0.72**	**0.55**	**0.75**	**1.69**

Table 1: Evaluation results for different topics.

Finally, in Table 1 we report an additional measure of usefulness: since both approaches aim at guiding the users to the time-intervals that are most promising for containing contradictions, we computed the percentage, P_1 and P_2, of the examined time-intervals that led to the identification of a contradiction, as well as the improvement of our approach when compared to the alternative, $\Delta P = P_2/P_1$. Even though the approach by Chen et al. [2] (stage 1) was not designed with this measure in mind, in the case of our approach, this measure is indicative of its precision since it measures how many of the automatically identified contradictions were real ones (i.e., verified by the users). The results show that our approach was always more successful in suggesting to users time-intervals that contained contradictions, with an overall average success rate of 75%, and as high as 95% (topic "Yaz").

The above results demonstrate that our approach can successfully identify contradictions in an automated way, and quickly guide users to the relevant parts of the data.

6.4 Evaluation of Scalability

We evaluate the scalability of the TimeTree for solving Problems 1 and 2, using a relational database implementation, where information is stored in a single table that contains contradiction values for each topic with respect to time intervals of different granularities. This implementation leads to simple and efficient SQL queries for detecting interesting contradictions. Remember that in the topic contradiction problem (Problem 1) we want to identify the contradictions and corresponding time windows of a single topic within some time interval, while in the all topic contradictions problem (Problem 2) we are interested in doing the same for all topics.

During this study, parameters of the contradiction formula were at their default values as described in Section 4. Changing formula's parameters will enlarge or reduce the number of contradictions being detected, but the computational efficiency will be the same. Performance of our approach does not depend on the value of threshold because we are not storing pre-computed contradiction values, and so the database is unable to apply indices or filtering on this parameter. Fixed and adaptive threshold approaches, however, return slightly different sets of contradictions. The first one returns largest contradictions themselves, and the second returns contradictions that are greater than p-times values of their respective parent intervals. The value of p was empirically set at 0.6 to return a result set with an average size equal to the one when using a fixed threshold. This allows us to compare the relative performance of both methods.

To test the performance of our solutions, we generated sets of 25 single-topic and all-topics queries (corresponding to the Topic and Time Interval Contradictions problems, respectively), using granularities and topic ids drawn uniformly at random. In these experiments, we used 1,000 topics. We measured the time needed to execute these queries against the database as a function of the time

[7] We omit presenting the detailed results for all parameters measured and each approach due to lack of space.

[8] *Ambien* is a drug for treating insomnia

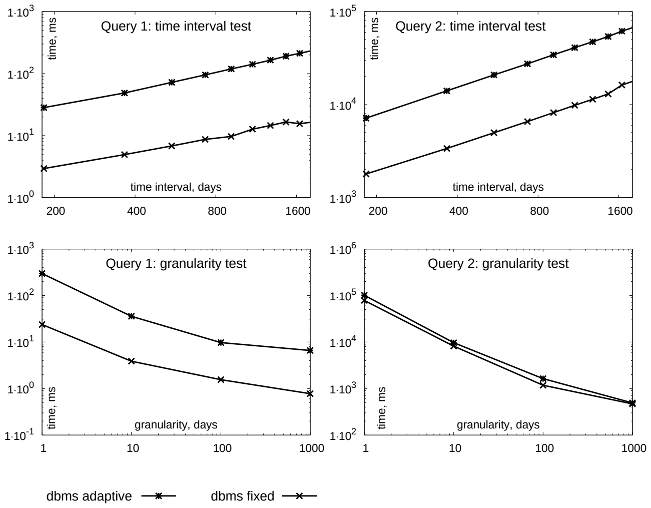


Figure 7: Scalability of single-topic and all-topics queries.

interval, τ, and the granularity of the time windows (Figure 7). We report results for both the fixed and the adaptive thresholds.

The adaptive threshold queries require in all cases more time since the threshold in this case has to be computed based on the contradiction value of the parent time window, which incurs more computation. This difference is pronounced for the database implementation, because it involves an extra join for obtaining the parent time window.

We observe that both single-topic and all-topics queries (see Figures 7(a-b)) scale linearly with the size of τ. This confirms our analytic results, and is explained by the fact that the queries have to return contradictions for all time windows (of a specific granularity) that are contained in τ. For single-topic queries with fixed threshold, the database is able to use all its indices (i.e., on topic id, time windows, and granularity) to answer the queries, therefore, achieving very fast response times.

Figures 7(c-d) depict the time results when we vary the granularity of the time windows specified by the queries. Increasing the granularity translates to larger time windows (i.e., moving up in the time hierarchy) and a smaller number of time windows for the same time interval. Thus, response times get lower.

7. DISCUSSION

The problem considered in this paper is new, in the sense that it considers contradictions on the large scale, while taking time into account (i.e., we consider the timestamps of the texts, as opposed to treating the text collections as sets). An approach that relies upon sentiment information and that exploits data engineering methods to detect such contradictions in texts at a large scale has been introduced and evaluated.

The evaluation of our approach on various datasets proved its ability of discriminating highly contradicting regions provided with a sequence of sentiments on some topic. Being scalable and computationally efficient, it can serve as a preliminary step for more sophisticated contradiction analysis, identifying the most interesting points for further processing.

An important feature of our contradiction detection method is its ability to operate on data with neutral sentiments. The contradiction formula we propose shows almost the same performance with or without neutral sentiments, allowing it to incorporate sentiment detection algorithms of different types.

As was mentioned previously, to build the contradiction formula we used such values as mean and variance. We believe that the effectiveness of our approach increases with the growing scale, relying on the fact that representativeness of statistical metrics also increases when larger number of samples is involved in computation. Moreover, tests on the synthetic data proved our formula's stable behavior in the presence of noise.

Finally, we note that we are aware that the evaluation of our (and related) approach to contradiction detection is still limited with respect to the precision and recall measures. The main reason for this is the absence of a benchmark dataset, and the difficulty in creating one. We are currently working toward such a dataset, suitable for testing different algorithms in this area.

8. CONCLUSIONS

In this paper, we proposed an approach to detect contradictions in documents, which is the first general and systematic solution to the problem. The experimental evaluation, with synthetic data and three diverse real-world datasets, as well we the user-study, demonstrate the applicability and usefulness of the proposed solution.

We are currently working on extending our approach so that it can work in an online mode. This will enable us to continuously monitor opinions in real-time.

9. REFERENCES

[1] D. M. Blei, A. Y. Ng, M. I. Jordan, and J. Lafferty. Latent dirichlet allocation. *JMLR*, 3, 2003.

[2] C. Chen, F. Ibekwe-SanJuan, E. SanJuan, and C. Weaver. Visual analysis of conflicting opinions. In *IEEE Symposium on Visual Analytics Science and Technology*, pages 59–66, 2006.

[3] M. C. de Marneffe, A. N. Rafferty, and C. D. Manning. Finding contradictions in text. In *ACL-08: HLT*, pages 1039–1047, 2008.

[4] K. Denecke and M. Brosowski. Topic detection in noisy data source. In *ICDIM*, pages 50–55, 2010.

[5] R. Ennals, B. Trushkowsky, and J. M. Agosta. Highlighting disputed claims on the web. In *WWW*, pages 341–350, 2010.

[6] S. Harabagiu, A. Hickl, and F. Lacatusu. Negation, contrast and contradiction in text processing. In *AAAI*, pages 755–762, 2006.

[7] R. Johansson and A. Moschitti. Reranking models in fine-grained opinion analysis. In *COLING*, pages 519–527. ACL, 2010.

[8] K. Lerman, S. Blair-Goldensohn, and R. Mcdonald. Sentiment summarization: Evaluating and learning user preferences. In *EACL*, pages 514–522, 2009.

[9] B. Liu, M. Hu, and J. Cheng. Opinion observer: analyzing and comparing opinions on the web. In *WWW*, pages 342–351. ACM, 2005.

[10] J. Liu, L. Birnbaum, and B. Pardo. Spectrum: Retrieving different points of view from the blogosphere. In *ICWSM*, pages 114–121, 2009.

[11] S. Morinaga, K. Yamanishi, K. Tateishi, and T. Fukushima. Mining product reputations on the web. In *KDD*, pages 341–349, 2002.

[12] S. Pado, M.-C. de Marneffe, B. MacCartney, A. N. Rafferty, E. Yeh, and C. D. Manning. Deciding entailment and contradiction with stochastic and edit distance-based alignment. In *TAC*, 2008.

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

[14] E. Riloff, J. Wiebe, and W. Phillips. Exploiting subjectivity classification to improve information extraction. In *AAAI*, pages 1106–1111, 2005.

[15] S. Siersdorfer, S. Chelaru, W. Nejdl, and J. San Pedro. How useful are your comments?: analyzing and predicting youtube comments and comment ratings. In *WWW*, pages 891–900. ACM, 2010.

[16] M. Tsytsarau, T. Palpanas, and K. Denecke. Scalable discovery of contradictions on the web. In *WWW*, pages 1195–1196, 2010.

[17] I. Varlamis, V. Vassalos, and A. Palaios. Monitoring the evolution of interests in the blogosphere. In *ICDE Workshops*, pages 513–518, 2008.