COMPLEX EVENT PROCESSING IN SOCIAL MEDIA

Calin RAILEAN, Monica BORDA, Alexandra MORARU
Technical University of Cluj-Napoca, Memorandumului 28, Cluj-Napoca, Romania

Abstract: **In a world where virtual communities get ahead, social media, supported by web and mobile technologies, represents a valuable source of large amounts of data that can encompass the information shared between individuals related to the traffic conditions. Therefore, a metric that measures the logical link between traffic condition and tweets reported in social media, can be useful in estimating the traffic impact. Accordingly, responsible authorities are provided with a measure of the resources needed for handling them, which can contribute to improving facilities such as public transportation services or traffic estimations. This paper aims to analyze the traffic condition covered in social media, more specifically, in twitter data and to reveal the correlation between traffic conditions and twitter messages. We suggest an automatic method for this association in order to discover its impact and importance. The proposed method is tested on a real world dataset and the results show a positive correlation between the automatic method and a set of manually evaluated associations. The usefulness of this research consists of forecasting public transportation needs and supporting events' organizers to better arrange their planned activities. The results have real applicability and are filled with consistent interpretations.**

Keywords: *event processing, traffic correlation, twitter data*

I. INTRODUCTION

The interaction among people in virtual communities and networks becomes a more used way of creating and exchanging different contents. The open data movement has gained popularity in the last years from more and more platforms provide their data through open application programming interfaces (APIs) or from other initiatives such of open-data government or linked open data. These sources of data give data mining researchers an increasing number of problems to be discovered and solved. The challenges now come not from getting the data for testing hypotheses, but from finding the appropriate technologies that can handle such large volumes of data.

Social media channels are attractive due to the fact that they allow the user to create, share or modify different messages. They are recognized as highly interactive platforms that report on all kinds of events happening around the world at any moment in time. As a micro-blogging service, Twitter generates constantly a large number of short messages that give the pulse of the communities involved in using it. Research performed on Twitter messages (i.e. tweets) has been a very popular topic in the last years, with applications ranging from sentiment analysis, to opinion mining and from topic model summarization to event extraction [1][2]. The application we are proposing is that of discovering traffic conditions from tweets. Knowing the information about traffic can help in improving it prediction and management. We propose and implement a method that determines wheatear a tweet is associated to a traffic condition and we analyze the performance of the method proposed.

In our research, taking into consideration the streaming nature of twitter data, we considered appropriate to perform our experiments following the Complex Event Processing (CEP) principles. CEP is emerging as a new paradigm for continuous processing of streaming data in order to detect relevant information and provide support for timely reactions. The main role of a CEP engine is to detect the occurrence of event patterns on the incoming streaming data [4]. We implemented our application using a CEP platform that provides classic operators for real time processing of streaming data.

The rest of the paper is organized as follows. Section 2 describes in more detail the problem addressed and introduces the traffic processing concepts and methods used. Section 3 presents the results of our experiments and the evaluation of the method proposed, while section 4 briefly discusses some of the related work. Finally we conclude the paper.

II. CORRELATION OF TRAFFIC CONDITIONS AND TWETS

The problem that we address in this paper is to determine the traffic impact (i.e. roadwork, queuing traffic) based on their presence in social media (i.e. tweets). We propose an algorithm that computes the degree of association between traffic events and tweets.

II.1 Dataset Description

In order to test the hypothesis, we referred to a dataset consisting of traffic events and micro-blog posts. The first set contains data reported by Bing Microsoft concerning the traffic conditions in London. The second one has been drawn up by collecting data from tweets from the public stream of Twitter. The time period taken into consideration lays between March 6th and April 11th and comprises 55000 traffic conditions and over 4 million tweets in London.

The traffic events are separated in categories: roadwork, queuing traffic etc. Each traffic condition has 17 features, from which we were interested in: traffic description, ID, start and stop time of traffic, location. Some of traffic conditions are repeating with different ID's, therefore we

Manuscript received July 2, 2014; revised September 20, 2014

identify and filter them by time and description.

Tweets have 90 fields that include, besides the tweet text, information such as: hash tags, time of posting tweet and geographical coordinates, used to be associated with traffic.

II.1 Event Processing with NEsper

NEsper is the .NET version of Esper [4] that shares the same syntax; therefore, throughout the rest of the paper we will refer to it as Esper. The Esper system provides the functionalities of an Event Stream Processing (ESP) system, as well as those of a Complex Event Processing (CEP) system. The interaction with Esper is supported by the Event Processing Language (EPL) that defines the main operators for expressing queries that are run by the engine. It is designed for a high-volume of data where one cannot store all information in database and process it in real time by using classical database queries. It is used in several areas such as finance, fraud detection, medicine where decisions need to be made as fast as possible.

The principle of the Esper system is that it allows registering queries in the engine and creates a listener class that will be called if the incoming event matches one of the inserted queries. The query can contain timeline windows, filtering, aggregation and sorting operators. Another functionality of the Esper system is to generate a new stream as combinations between two or more input streams. The EPL statement used in our application for correlating the two data sources based on time is the following:

select *from pattern [every Traffic->every TweetSet[Traffic.Stop_time>tweets.StopTime)and (Traffic.Start_Time < tweets.StartTime)],

which is similar to an inner-join statement from classical database management systems, where the join condition is represented by the time constraint. A pattern-based event stream structure (on which we specify the time constraints we want to impose) was applied to obtain only the combinations that overlap over time. In this situation, the event stream generated will consist of traffic-tweet set pairs for which the timestamp for the set is between that start and stop time of the traffic condition. A tweet set is a window consisting of 1000 tweets and for each traffic condition we have an average of 3 tweet sets that fits its duration. For each traffic-tweet set we applied a method of measuring the degree of correlation between a tweet and a traffic condition. To determine how strong the relationship is, we use cosine similarity[5], because it is used to match the relevant correlations between terms. This method it is used when are more words that are uncommon between tweet and traffic condition[6]. First, we count how many times a traffic description term appears in a tweet text. After calculating the term frequency, we determine the inverse term frequency as below:

$$Idf\,(term) = 1 + \log(\frac{NT}{NTT}) \qquad (1)$$

where NT represents the total number of tweets and NTT-number of tweets with term game in it. Next step is for each term in the traffic description multiply its normalized term frequency with its IDF on each tweet. And finally, the cosine similarity:

$$CS(tr, tw) = \frac{tr[term].(TF * IDF) * tw[term].(TF * IDF)}{\sqrt{\sum_{i=1}^{n} tr[j].(TF * IDF)}^{2} * \sqrt{\sum_{i=1}^{n} tw[j].(TF * IDF)}^{2}} \qquad (2)$$

where tr represents traffic condition, tw is tweet, TF represents term frequency and IDF inverse term frequency (1). As an illustrative example, let us consider the traffic condition with the description “between LONDON HEATHROW AIRPORT and BRENTFORD Roadworks” and 3 tweets with following description:

Tweet1: “I'm at London Heathrow Airport (LHR) - (Hounslow, Middlesex)”
Tweet2: “I'm at London City Airport”
Tweet3:” I'm at Brentford now”

The result of cosine similarity for these associations is: 0.43, 0.27 and respectively 0.26. First value represents a strong correlation and it is determined by matching” LONDON HEATHROW AIRPORT”. The biggest score for this value is given by the “Heathrow” because it is appear just once in all tweets its IDF value is higher, that determines a non-linear lower score for second association. In third case, even if the traffic condition and tweet has a logical link, the cosine similarity is the smallest and it is explained by a lowest score for term frequency. For solving this problem we use for searching a correlation just address terms, in our case without “between, and, Roadworks”. This solution gives us a higher impact for addresses with less numbers of terms, in our case “Brentford”.

Another issue was that matching location doesn’t mean that user is physical at this place. After analyzing results we decided to consider only “check in” tweets that contain expression of presence at that location: “I’m at”.

III. RESULTS

The results of the processing performed with the Esper system comprise 61770 tweets correlated with 5843 traffic conditions having the cosine coefficient higher than 0.5 (with an average of 11 tweets per event. Figure 1 illustrates the influence of cosine similarity over the average number of tweets associated with a traffic condition. We can observe that if the cosine threshold is higher, the number of tweets per traffic condition decreases and therefore the best correlations are kept.

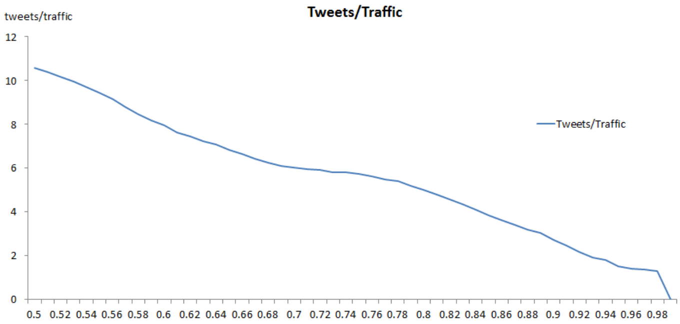

Figure 1. Tweets-Traffic ratio

Specific examples of event-tweet associations can be found in Table 1. We can observe that the association with traffic description “between LONDON HEATHROW AIRPORT and CRANFORD - Queuing traffic for 3 km.” with tweet text “I'm at London @Gatwick_Airport (LGW) (Crawley, West Sussex) w/ 13 others” is incorrect because the tweet refers another location. Similarly, two more examples are incorrect because the event titles have commonly used words and they are mistakenly associated with the tweet. However, the other examples show correct associations, indicating that the coefficient defined can yield positive results. In order to analyze the performance of cosine coefficient we manually evaluated a sample of event-tweet association, as it is explained in the next section.

Table 1. Examples of traffic-tweet associations

Traffic Description	Tweet	Cosine Coefficient
between CHISWICK and EALING - Queuing traffic for 2 km.	I'm at Ealing Broadway Shopping Centre - @ealingshopping (Ealing, Greater London)	0.919
between SOUTHWARK BRIDGE and WESTMINSTER BRIDGE - Queuing traffic for 4 km.	I'm at Platform 7 (Westminster, Greater London) http://t.co/5xKlziLCj5	0.928
between BLACKFRIERS BRIDGE and KINGS CROSS - Queuing traffic for 1 km.	I'm at London Kings Cross Railway Station (KGX) - @nationalrailenq (London, Greater London) w/ 3 others http://t.co/JZCwXYvVXA	0.951
between NORTH CIRCULAR ROAD and TOWER HAMLETS - Queuing traffic for 2 km.	I'm at The Nazrul (Tower Hamlets, Greater London)	0.936
between LONDON HEATHROW AIRPORT and CRANFORD - Queuing traffic for 3 km.	I'm at London Heathrow Airport (LHR) (Hounslow, Middlesex) w/ 22 others	0.970
between HAMMERSMITH FLYOVER and KEW - Queuing traffic for 3 km.	I'm at Apollo for Amon Tobin (Hammersmith, Greater London) w/ 23 others http://t.co/hDzgt4y0YU	0.913
between LONDON HEATHROW AIRPORT and CRANFORD - Queuing traffic for 3 km.	I'm at London @Gatwick_Airport (LGW) (Crawley, West Sussex) w/ 13 others	0.908
between TOWER HAMLETS and TOWER OF LONDON - Queuing traffic for 3 km.	I'm at Lloyds Banking Group (City of London, Greater London) http://t.co/hPUe7mOlV1	0.917
between NORTH CIRCULAR ROAD and WEMBLEY - Roadworks.	I'm at Abbey Road Studios (London, Greater London) w/ 2 others http://t.co/jLLIFfQqgX	0.900

IV. EVALUATIONS AND DISCUSSION

When the cosine coefficient takes low values, this means that there is a poor correlation between the tweet and the traffic condition analyzed. In order to assess the performance of cosine coefficient, we have manually tested a random set of associations of traffic with tweets. We randomly selected 100 associations of traffic and tweets for each interval with cosine coefficient between 0.5-0.6, 0.6-0.7, 0.7-0.8 and 0.8-0.9 and obtained 4, 5, 8 and 11% correct associations. For the associations between 0.9-1.0 two human annotators have analyzed the tweet and the traffic description and evaluated them as correct or incorrect. The inter-annotator agreement has been calculated in order to illustrate the utility of the annotator's results. The inter-annotator agreement, or Cohen's kappa coefficient, is described by the next equation [7]:

$$k = \frac{\Pr(a) - \Pr(e)}{1 - \Pr(e)} \qquad (3)$$

where Pr(a) is the relative observed agreement among annotator, and Pr(e) is the hypothetical probability of chance agreement. The Cohen's kappa coefficient is equal to 0.681 for the two annotators, which can be considered as a "substantial" level of agreement [8].

After determining the coefficient, there were analyzed only those annotated associations for which the annotators gave the same score. The analysis showed that the average value of cosine coefficient for the correct associations is higher than the average value of cosine for the incorrect ones: 0.94 and 0.90, indicating that the cosine defined is a fair metric for association. Figure 2 and 3 illustrates a more detailed analysis of the performance of cosine coefficient is. Figure 3 shows that the values of cosine for the correct associations are generally higher than the values of cosine for the incorrect ones.

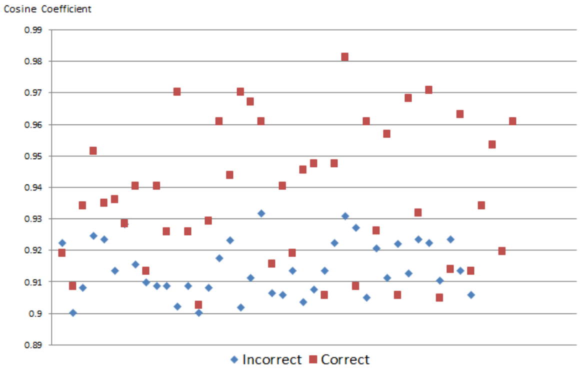

Figure 2.Values of cosine coefficient for the associations of tweets and traffic conditions evaluated

Taking a step further, we examined the accuracy of correct associations, calculated as the ratio between the number of correct associations and the total number of associations. As expected, the precision of tweet event associations increases together with the increase of cosine coefficient. A more detailed picture on the relation between the coefficient and the estimation accuracy can be noticed in Figure 3 and it is described by next equation:

$$Accuracy(cs) = \frac{TA(>= cs)}{TA(>= cs) + FA(>= cs)} \qquad (4)$$

where TA represents good associations and FA false ones, with cosine coefficient equal and higher then queried value.

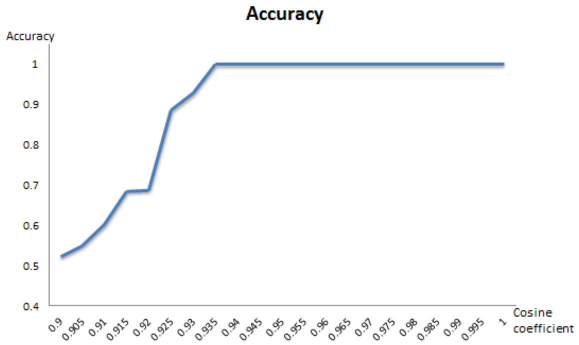

Figure 3. Accuracy performance for different values of cosine coefficient

We can observe that for coefficient > 0.92 the precision improves substantially. Although the recall performance would have been an interesting measure to analyze, it was considered too expensive to be done manually. A larger dataset fully annotated would be more appropriated for such an analysis. And finally, in figure 4 can be noticed the histogram of tweets:

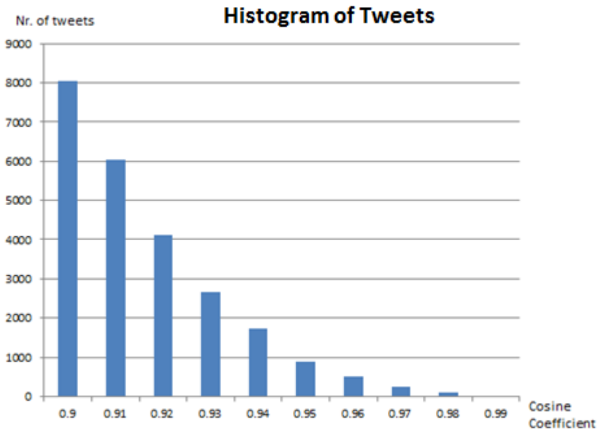

Figure 4. Histogram of tweets with cossine coefficent between 0.9 and 1.0

V. RELATED WORK

A large amount of research on Twitter messages has been performed and reported in literature and a full comparison of the present paper with other works is out of the scope of this paper. However, we would like to mention the work reported in [1] where a system for processing tweets in real time is introduced. The applications tackled refer to sentiments analysis and detection of term frequencies in real time. Although our method is not comparable in terms of complexity with the methods proposed in [1], the similarity can be found in the stream processing concept.

Another similar problem is reported in [2], where the problem addressed is that of linking tweets with news articles. The authors propose a graph based latent variable model for enriching the short text of tweets in order to create a larger context for it. In [3], another study on open data sources for London city presents the results of possible associations between social events, weather data and traffic.

We have proposed another method in [8] where the associations between social events and tweets were found. The most important improvement we made in this paper is the implementation of cosine coefficient algorithm. In comparison with accuracy graphic from previous paper we can observe that it has a greater slope and it becomes easier to find threshold value for coefficient.

VI. CONCLUSIONS

We have proposed and evaluated a method of measuring the popularity of social events taking place in the city of London, based on tweets. The results show that this method yields a positive outcome and is a valid solution for the problem addressed. The number of false positive associations of traffic and tweets can be reduced by setting a higher threshold for the cosine coefficient. Further contributions to the method can be made by including geo-location parameters in the cosine coefficient equation, as well as by improving the pre-processing of data in terms of extension of the stop-word list or by including natural language processing techniques.

REFERENCES

[1] Bifet, A., Holmes, G., Pfahringer, B. (2011). MOA-TweetReader. Real-Time Analysis of Twitter Streaming Data. Discovery Science. Springer Berlin Heidelberg, pp. 46-60.

[2] Guo, W., Li, H., Ji, H., Diab, M. (2013). Linking Tweets to News: A Framework to Enrich Short Text Data in Social Media. *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics.*

[3] Moraru, A., Mladenić, D. (2012). Complex Event Processing and Data Mining for Smart Cities. *Proceedings of the 15th International Multiconference Information Society, Ljubljana, Slovenia.*

[4] EsperTech Inc, Esper Reference, Version 4.8.0 (2012). http://esper.codehaus.org/esper-4.8.0/doc/reference/en-US/pdf/esper_reference.pdf (last access date: 01.03.2014).

[5] Jana, V., Tf-Idf and Cosine similarity, (2013) http://janav.wordpress.com/2013/10/27/tf-idf-and-cosine-similarity (last access date: 05.05.2014).

[6] Discussion of Similarity Metrics. http://mines.humanoriented.com/classes/2010/fall/csci568/portfolio_exports/sphilip/cos.html (last access date: 27.09.2014).

[7] Cohen's kappa. http://en.wikipedia.org/wiki/Cohen's_kappa (last access date: 01.03.2014).

[8] Railean, C., Moraru, A.(2013). Discovering popular events from tweets. *Proceedings of the 16th International Multiconference Information Society, Ljubljana, Slovenia.*