EventSense: Capturing the Pulse of Large-scale Events by Mining Social Media Streams

Emmanouil Schinas,
Symeon Papadopoulos,
Sotiris Diplaris,
Yiannis Kompatsiaris
CERTH-ITI
Thessaloniki, Greece
{manosetro, papadop, diplaris, ikom}@iti.gr

Yosi Mass,
Jonathan Herzig
IBM Haifa Research Lab
Haifa, Israel
{yosimass, hjon}@il.ibm.com

Lazaros Boudakidis
IT Department
Thessaloniki International Film Festival
Thessaloniki, Greece
it@filmfestival.gr

ABSTRACT

Social media platforms such as Twitter and Facebook have seen increasing adoption by people worldwide. Coupled with the habit of people to use social media for sharing their daily activities and experiences, it is not surprising that a substantial part of real-world events are well described by the online streams of status updates, posts and media content. In fact, in the case of large events, such as festivals, the number of online messages and shared content can be so high that it is very hard to get an objective view of the event. To this end, this paper presents EventSense, a social media sensing framework that can help event organizers and enthusiasts capture the pulse of large events and gain valuable insights into their impact on visitors. More specifically, EventSense enables the automatic association of online messages to entities of interest (e.g. films in the case of a film festival), the automatic discovery of topics discussed online, and the detection of sentiment (positive/negative/neutral) both at an entity level (e.g. per film) and on aggregate. In addition, the framework produces an informative social media summary of the event of interest by automatically selecting and putting together its highlights, e.g. the most discussed entities and topics, the most influential users, the evolution of the discussions' sentiment, and the most shared media and news content. A real-world case study is presented by applying EventSense on a rich dataset collected around the 53rd Thessaloniki International Film Festival.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Miscellaneous

Keywords

social media, topic and event detection, sentiment analysis

1. INTRODUCTION

With the recent growth of popularity of Online Social Networks (OSNs), such as Twitter and Facebook, it is not surprising that a substantial part of the status updates, posts and shared content generated by users, is related to real-world events. Especially for large-scale social events such as festivals, attended by large crowds of people, the amount of user generated content is constantly increasing as progressively more people use social media to express their opinion and sentiment, or share information about their participation to the event of interest.

With such increasing popularity, however, a major challenge arises: The vast amount of content and its lack of structure make it difficult to gain an accurate overview of the event. For example, since several sub-events occur within large-scale events (e.g. film screenings in the case of film festivals), it would be more informative for the end user to have the content organized on the basis of these sub-events. Similarly, the prevalence of redundancy among online comments and status updates makes it valuable to group together status updates that discuss the same topic. Moreover, as a consequence of the controversial nature of many event-related entities (e.g. films), a wide set of opinions and sentiments is expressed online by event participants. For the reasons above, the online representation of an event as a sequential list of posts and status updates is ineffective for conveying an objective view of the event to interested users. A more effective means of event representation would employ facets, such as entities, topics and sentiment to enable more effective information presentation and access.

To this end, this paper presents EventSense, a social media sensing framework that can help event organizers and event enthusiasts capture the pulse of a large event and gain valuable insights into the impact of the event on its visitors. Online messages about the event are organized around entities of interest (e.g. films) and topics, and sentiment scores are extracted for each of those, by aggregating the sentiment expressed by individual messages. This kind of aggregation enables the ranking of entities, topics and online users based on social interest and disposition, and thus conveys a succinct and informative view of the event highlights. In addition, through a real-world evaluation on the 53rd Thessaloniki International Film Festival (TIFF53), the paper provides evidence that real-world event variables, such as film ratings, are correlated with aggregate statistics mined from the stream of online messages.

The paper is organized as follows. Section 2 contains a brief survey of related work in pertinent research fields. Section 3 describes in detail the components of EventSense. Section 4 presents an experimental case study on TIFF53. We conclude this paper in Section 5.

2. BACKGROUND AND RELATED WORK

Numerous systems and applications aim to provide structured exploration, search and summarization of social media content. Although EventSense is related to the following works in some aspects, we could not find a single work that encompasses all aspects of the proposed framework.

For instance, Tweet Motif [9] is a faceted search system that indexes tweets by significant terms to provide exploratory search for Twitter. TwitInfo [7] is a system for visualizing and summarizing events on Twitter. Given a search query related to an event, TwitInfo creates an event-related timeline of tweets, identifies and labels event peaks and provides an aggregate view of user sentiment about the event. Tweetgeist [12] is a similar system that detects, summarizes and visualizes broadcast events by mining Twitter messages. The work in [4] uses Hidden Markov Models to create representative summaries of topic-specific Twitter streams. The work of [11] uses an approximate method to detect the first message in a stream of messages that discuss a new story.

SportSense [14] is a system that extracts US National Football League (NFL) game-related messages from the Twitter stream. The system employs team names as keywords for the Streaming API provided by Twitter. Then, signal processing techniques, such as matched filtering, are applied on the timeseries generated by the game-related tweets in order to detect specific types of events during games.

Regarding sentiment analysis in social media content, the authors of [5] employ a variety of features (i.e. uni-grams, bi-grams and Part Of Speech features) and classifiers (Naive Bayes, Support Vector Machine and Maximum Entropy) to study the problem of sentiment analysis in Twitter. Moreover, the authors of [2] use SVM and Multinomial Naive Bayes classifiers to examine the hypothesis that it is easier to classify the sentiment in short documents such as tweets, compared to longer form documents. In [10] the authors use the same method as in [6] to extract positive and negative training data and they extended it with neutral data from Twitter accounts of popular newspapers and magazines, such as New York Times, Washington Posts etc, assuming that those messages are objective and do not carry any sentiment. Than, they create a three classes classifier to classify tweets as positive, negative or neutral.

The authors in [1] use Twitter to predict future real-world trends. They conclude that there is strong positive correlation between the tweet rate (number of tweets per hour) of a particular movie and box-office revenues. Also, they apply sentiment analysis to movie-related tweets and calculate per movie aggregate values of subjectivity (i.e the fraction of negative and positive tweets to neutral ones) and polarity (i.e the ratio of positive to negative tweets). They find that subjectivity increases after the release of a movie and that polarity, although not strongly correlated with box-office revenues, can be used in conjunction with tweet rate to improve the prediction results. In [8] the authors attempt to correlate sentiment scores from Twitter to real world topics and compare against opinion poll results.

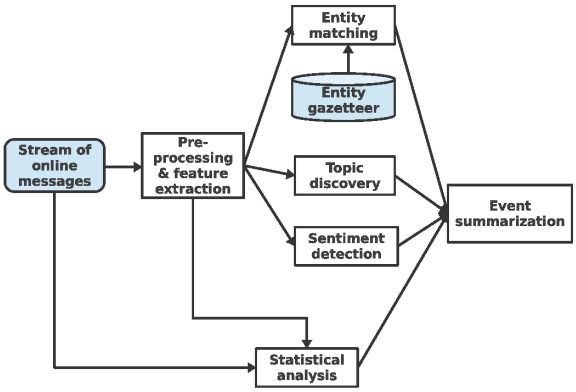

Figure 1: Main components of EventSense.

3. EVENTSENSE FRAMEWORK

An overview of the proposed framework is illustrated in Figure 1. The system processes a stream of online messages around an event with the goal of mining useful information and extracting informative summaries.

3.1 Preprocessing and feature extraction

At a first step, the messages are pre-processed and appropriate text features are extracted to feed the subsequent processing steps. First, each incoming message is cleaned by removing punctuation marks and other social media specific strings (e.g. URLs or retweets and mentions in Twitter). Thereafter, language detection [13] is applied in order to select an appropriate text tokenizer. We used the set of language-specific tokenizers provided by the Lucene library, and in cases that language detection is inconclusive (e.g. mixed language message) the StandardTokenizer of Lucene is used. After tokenization, n-grams ($n = \{1, 2\}$) are extracted from each message to represent it as a weighted feature vector using the standard $tf * idf$ scheme. For a given message m over a set of messages M the tf and idf components of a feature $f \in m$ are computed according to Equations 1 and 2 respectively.

$$tf(f, m) = \frac{count(f \in m)}{|m|} \tag{1}$$

$$idf(f) = 1 + \log \frac{|M|}{|\{m \in M : f \in m\}|} \tag{2}$$

To improve the score of important features, we use a boost factor as shown in Equation 3.

$$tf * idf(f, m) = tf(f, m) * idf(f) * boost(f) \tag{3}$$

Although seemingly ad-hoc, the boosting technique can be considered as an extension of the $tf * idf$ scheme, wherein more weight (in the similarity calculation) is given to terms that are expected to be particularly relevant for the domain of interest. For instance, in the case of a film festival the terms contained in film titles are boosted. We used several variants of the above feature representation (e.g uni-

/bi-grams, language-specific tokenization, stemming, stopword elimination, etc.) that lead to a diverse set of results as shown in Section 4. Table 1 illustrates three examples of messages from Twitter and the computed unigrams and bigrams with their corresponding weights.

3.2 Entity detection

EventSense provides support for the detection of entities of interest. An entity e is defined as a tuple $(a_1, a_2, ...a_n)$ where a_i are its canonical properties. According to this definition, the entities of interest are lists of properties. For example in case of a film festival, the main entities of interest are the films screened in the festival and their properties include their title and description, and possibly the names of directors and actors. Another domain could be a specific football league over a time period. In that case, possible entities of interest would be the games between teams and their properties would be the team, stadium and player names.

For each message m, the entity detection component of EventSense checks whether it contains a reference to one or more entities and then associates it to them. To represent entities as vectors, we rely on the same vector space model as online messages. We select a subset of entity properties, we detect their language and then tokenize it with the appropriate language-specific tokenizer of Lucene, and we merge all resulting vectors into a single vector. We measure the similarity between a message and an entity by using the cosine similarity between the two vectors. We associate a message to every entity for which their pairwise similarity exceeds a predefined threshold θ_1.

3.3 Topic Detection

To detect topics in a set of online messages, we use a similar approach to [11]. There, the appearance of new stories in streams of online messages is implemented by means of an approximate method based on Locality Sensitive Hashing for computing the nearest neighbour (NN) of each incoming message. If the similarity between them exceeds a predefined threshold θ_{2a}, the incoming message is assigned to the same cluster as its NN, else it forms a new cluster (story). In EventSense, we use the same approach to create clusters of similar messages. At the end of this step, the clusters that contain only one message are considered as outliers and discarded from the set of discovered topics.

The topics produced by the above NN clustering algorithm are typically groups of near-duplicate messages (e.g retweets on Twitter or shared posts on Facebook). A frequent problem stems from *cluster fragmentation*, i.e. a lot of messages that refer to the same topic are assigned to different clusters. To avoid this over-segmentation of topics, and to create larger clusters with more diverse messages, we introduce a post-processing step to fuse different clusters by measuring their pairwise similarities and merging those pairs that exceed a predefined threshold θ_{2b}. Each cluster is represented with a centroid vector and the similarities among clusters are measured using the cosine formula. We summarize each cluster in two ways: (a) a set of the most frequent terms in the aggregate text of the cluster, and (b) a representative title extracted by finding the most frequent sequence of terms accross all the messages of the cluster.

3.4 Sentiment Detection

It was shown in [3] that sentiment analysis can be formulated as a machine learning classification problem. As such, it is necessary to have labeled data for each class to train a sentiment classifier. Obtaining labeled data for positive and negative classes can be done automatically, by extracting tweets with emoticons [5, 10]. Messages that contain happy emoticons (":)", ":-)", ":D", etc.) form the positive training set, while messages that contain sad emoticons (":(", ":-(", etc.) form the negative training set. However, a large portion of messages are neutral, so it is necessary to detect such messages as well.

We start by building a Naive Bayes (NB) classifier for positive and negative data. Such classifier was found effective in sentiment classification [3] and other text classification applications. We use the labeled data and extract for each message two types of features[1], after removing stop words and emoticons, and trimming repeated letters. The first type is n-grams, more specifically uni- and bi-grams. Negation term presence is also exploited: Negation terms (e.g. "not", "isn't", "aren't") are attached to the subsequent terms to form a unigram (e.g. "isn't happy" → "nothappy"). Mentions and links are not counted as uni-grams. Furthermore, all uni- and bi-grams that occur only once are removed. The second type of features includes the following: user mentions, URLs, punctuation (question and exclamation marks), repeated letters (presence of words like "looove" and "noooo") that usually indicate sentiment expression, and all-caps words (e.g. "I REALLY want to go there").

Assuming a set of classes C, for a given message m and a class $c \in C$, a NB classifier is used to estimate the probabilities $P(c|m)$. Assuming a uniform prior for all classes, independence between features, and using the Bayes rule, we get:

$$P(c|m) \propto P(m|c) = \prod_{f \in m} P(f|c) \tag{4}$$

Estimating $P(f|c)$ for n-gram features can be done using maximum likelihood and Laplace correction [6] as follows.

$$P(f|c) = \frac{tf(f,c) + 1}{\sum_{f' \in V} tf(f',c) + |V|} \tag{5}$$

where $tf(f,c)$ is the frequency of feature f in class c, and $|V|$ is the number of n-grams in the vocabulary.

The probability $P(f|c)$ for the other five features can be estimated from the labeled data as follows. For each such feature and for each message, we consider a binary value that gets a value of 1 if the feature exists in the message, or 0 otherwise. Therefore, the probability for the presence of those features in each class, using the Bernoulli model and Laplace correction is defined as:

$$P(f = 1|c) = \frac{df(f,c) + 1}{|L_c| + 2} \tag{6}$$

where $df(f,c)$ is the number of documents of class c that contain feature f. The complementary probability is then $P(f = 0|c) = 1 - P(f = 1|c)$.

Given message m, we classify it as belonging to class c^* using

[1]Feature extraction for sentiment detection is carried out independently of that for entity and topic detection.

Table 1: Feature vector examples for three TIFF53 tweets.

Message	Unigrams	Bigrams
Georges @Corraface is here for his film premiere 'Papadopoulos & Sons'! Tomorrow, 19.30 at Warehouse D! #tiff53 http://t.co/NlHAW40C	(sons:0.416), (papadopoulos=0.415), (19.30=0.352), (tomorrow=0.327), (premiere=0.318), (his=0.303), (georges=0.303), (here=0.282), (warehouse:0.195), (film:0.147)	(papadopoulos sons:0.464), (premiere papadopoulos:0.232), (his film:0.232), (tomorrow 19.30:0.232), (sons tomorrow:0.232), (film premiere:0.232), (sons:0.229), (papadopoulos:0.229), (19.30:0.194), (warehouse d:0.180), (tomorrow:0.180), (premiere:0.175), (georges:0.167), (his:0.167), (here:0.155), (warehouse:0.107), (film:0.081),
Αργυρός Αλέξανδρος στην ταινία 'Μούχλα'! #tiff53	(μουχλ:0.806), (αργυρ:0.456), (αλεξανδρ:0.336), (ταιν:0.171)	(μουχλ:0.616), (ταιν μουχλ:0.418), (αργυρ αλεξανδρ:0.348), (αργυρ:0.348), (αλεξανδρ:0.289), (ταιν:0.191)
#tiff53 "Νεκρή Ευρώπη' Βαρύγδουπη σαχλαμάρα με συμβολισμούς της πλάκας και αλλοπρόσαλλους χαρακτήρες. Πλούσια παραγωγή.	(ευρωπ:0.458),(νεκρ:0.446), (σαχλαμαρ:0.303), (αλλοπροσαλλ:0.303), (πλους:0.303), (βαρυγδουπ:0.266), (πλακ:0.266), (συμβολισμ:0.248), (παραγωγ:0.239), (χαρακτηρ:0.232)	(νεκρ ευρωπ:0.315), (ευρωπ:0.310), (νεκρ:0.303), (αλλοπροσαλλ χαρακτηρ:0.206), (ευρωπ βαρυγδουπ:0.206), (αλλοπροσαλλ:0.206), (βαρυγδουπ σαχλαμαρ:0.205), (πλους παραγωγ:0.205), (σαχλαμαρ:0.205), (πλους:0.205), (πλακ:0.195), (βαρυγδουπ:0.181), (συμβολισμ:0.168), (παραγωγ:0.162), (χαρακτηρ:0.157)

the maximum log-likelihood

$$c^* = \arg\max_c \sum_{f \in m} log(P(f|c)) \quad (7)$$

Using the positive and negative labeled data, we can build a NB classifier and use Equation 7 to classify a message as either positive or negative.

To detect neutral messages (messages that carry no sentiment), we use Mutual Information (MI), which is defined between two random variables X and Y as follows:

$$I(X,Y) = \sum_{x \in X} \sum_{y \in Y} P(x,y) log \frac{P(x,y)}{P(x)P(y)} \quad (8)$$

where $P(x,y)$ is the joint probability of X and Y and $P(x)$ and $P(y)$ are the marginal probabilities of X and Y respectively. In our case, we want to measure the MI between the presence or absence of a feature f and the positive and negative classes. The motivation is that messages that contain only n-grams with low MI with the positive and negative classes can be classified as neutral, since the presence and absence of their terms do not contribute much to the information of a message being positive or negative. This is defined as

$$I(f) = \sum_{c \in \{pos,neg\}} \sum_{f \in \{0,1\}} P(c,f) log \frac{P(c,f)}{P(c)P(f)} \quad (9)$$

where $P(f=1)$ is the number of documents in the training data that contain the feature f divided by $|M|$ (the total number of training documents), $P(c)$ is the number of documents of class c divided by $|M|$, $P(c, f=1)$ is the number of documents that contain the feature f and were labeled as class c divided by $|M|$, and $P(c, f=0)$ is the number of documents that do not contain the feature f and were labeled as class c divided by $|M|$.

The sentiment intensity of a message m is then defined as:

$$I(m) = \arg\max_{f \in m} I(f) \quad (10)$$

The classification of a message m is then done as follows. If $I(m) \leq \theta_3$, for some threshold θ_3 that is learned, we classify the message as neutral; otherwise, we assign to it to the positive or negative class based on Equation 7.

3.5 Event summarization

The last component of EventSense applies statistical analysis to the set of incoming messages and the outputs of the previous components to produce an informative overview of the event. We start by aggregating messages by time to create a timeline of messages that can be used to detect peaks of high activity. Thereafter, we calculate the most shared messages and most shared media items contained in messages (e.g URLs pointing to pictures). We also find a set of most influential user accounts by measuring the diffusion of the information generated from each user. We do this by aggregating the times that a message created by a user was shared by other users (e.g retweets in Twitter or shares in Facebook). By taking into account the results of entity detection in messages and sentiment analysis, we aggregate the sentiment per entity. Namely for each entity we retrieve the set of associated messages and calculate the mean value of sentiment (i.e a value in the range [-1, 1]). Also we calculate the values of Polarity and Subjectivity per entity, $pol(e)$ and $subj(e)$, as defined in equations 11 and 12 respectively. Finally, we calculate the same sentiment statistics per day and per user (for the most influential users).

$$pol(e) = \frac{|\{m \in (e \cap C_{pos})\}| - |\{m \in (e \cap C_{neg})\}|}{|\{m \in (e \cap C_{pos})\} \cup \{m \in (e \cap C_{neg})\}|} \quad (11)$$

$$subj(e) = \frac{|\{m \in (e \cap C_{pos})\} \cup \{m \in (e \cap C_{neg})\}|}{|\{m \in (e \cap C_{neut})\}|} \quad (12)$$

4. EVALUATION

4.1 Event and dataset description

We conducted an evaluation of EventSense on a dataset related to the 53rd International Film Festival of Thessaloniki (TIFF53) that took place between November 2nd and 11th, 2012. The organization of the festival provided us with a detailed set of 168 films included in the official festival program. For each film, information is available about its title, description, director(s) and actors in two languages, Greek and English. Next, we collected tweets that contain the official hashtag of the festival (#tiff53) for the period between November 1st and 13th by using the filter method of the Twitter Streaming API. In total, 3974 tweets were collected and then manually annotated: First, each tweet was associated (if applicable) to one or more films of the festival, as well as classified with respect to the conveyed sentiment,

Table 2: Selection of properties from film entities.

Film Model	Film Properties
BowEn	Title in English
BowGr	Title in Greek
BowMl	Title in both languages
BowMlRich	Title and directors in both languages
BowMlAll	All properties in both languages

be it positive, negative or neutral. For the ground truth to be more reliable, a two-round annotation was conducted, wherein the second-round annotator checked the validity of the annotations produced by the first-round annotators. In addition, we had access to film rating and bookmarking data created by the ThessFest mobile app (available both for iPhone[2] and Android[3]). More specifically, for each screened film, we obtained data on the number of ratings and average rating, as well as number of times the film was added to the list of favorites (created by the app).

4.2 Tweet-film matching

Despite the fact that the technique of Section 3.2 is generic, for the experiments we considered the specific case where messages are tweets and entities are films screened during the festival. Each film was defined as a tuple of the form: <title, description, directors, actors>. To make the technique take advantage of the availability of film data in two languages, multiple entity representations were instantiated. For each film, we extracted the corresponding feature vector, as described in Section 3.2. In our experiments we used two types of features, uni- and bi-grams. To compute the feature vectors of tweets and films, we built the appropriate vocabulary using two types of documents: the text of all tweets in the dataset and the concatenated text of all film properties (i.e title, descritpion, etc.) in both languages.

To handle the problem of multilingual messages and films, we use two vectors for each film, as mentioned above. Regarding the representation of films as feature vectors, we followed and evaluated several combinations of properties and languages. The basic approaches are shown in Table 2. For example, in case of BowMl, we first create a feature vector for each film in each of the two languages by using the title property, and then we merge these into a single vector. In a similar manner, we create BowMlRich, where for each language-specific instance we use the concatenated text of title and directors.

As expected, the choice of similarity threshold (θ_1) affects the accuracy of the entity detection. A low threshold may incorrectly associate messages with entities leading to increased false positive rates, while a high threshold may fail to match entities with messages resulting in increased false negative rates. To locate the optimal threshold value, we randomly select a subset of the messages (20% of the dataset) and use it for tuning. For each combination of the entity models of Table 2 and feature types (uni- and bi-grams) we find the threshold that maximizes the F-score and use it to evaluate each combination. We illustrate the variation of F-score for the random subset in Figure 2 and present the results achieved by the best values in Table 3.

[2] itunes.apple.com/kr/app/thessfest/id504913309

[3] play.google.com/store/apps/details?id=com.mk4droid.FF_pack

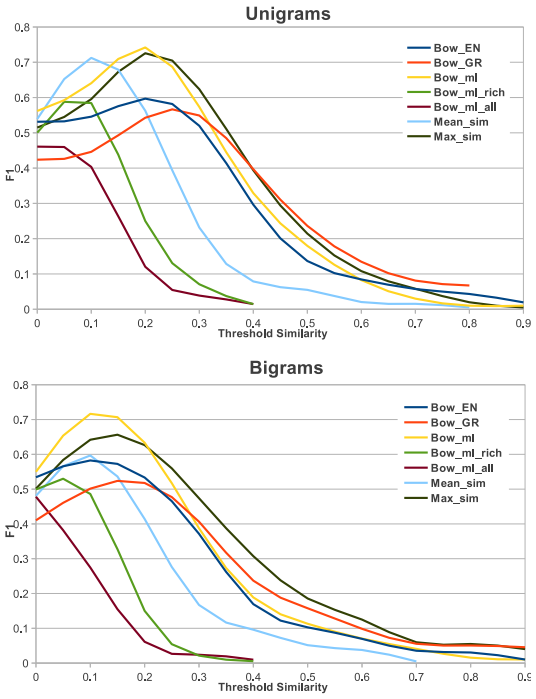

Figure 2: Sensitivity of entity detection accuracy vs. threshold θ_1 and feature selection.

Table 3: Precision-Recall of tweet-film matching.

Features	Precision	Recall	F1	Threshold (θ_1)
uni + BowEn	0.774	0.467	0.582	0.2
bi + BowEn	0.735	0.463	0.568	0.1
uni + BowGr	0.817	0.379	0.517	0.3
bi + BowGr	**0.847**	0.385	0.529	0.2
uni + BowMl	0.805	0.651	0.720	0.2
bi + BowMl	0.720	0.680	0.699	0.1
uni + BowMlRich	0.540	0.667	0.597	0.05
bi + BowMlRich	0.724	0.549	0.624	0.05
uni + BowMlAll	0.524	0.505	0.514	0.05
bi + BowMlAll	0.733	0.389	0.508	0.05
uni + MeanSim	0.734	0.687	0.710	0.1
uni + MaxSim	0.774	**0.697**	**0.734**	0.2

For the same set of film properties used to represent it as a feature vector, unigram features outperform the results of bigrams. Also, by checking the results for the different variants of properties and languages we conclude that where a combination fails, another combination performs well. With this in mind, we use the following strategy to impove the performance of our approach. For each pair of tweets and films we calculate the similary among the tweet and all the variants cited above. To caclulate an overall similarity, we use either the maximum or the mean value. As shown in the last two rows of Table 3 maximum similarity outperforms the mean similarity and all the other variants as well.

4.3 Topic analysis

A list of topics was produced by applying the method of Section 3.3 on the set of tweets. Both thresholds θ_{2a} and θ_{2b}

Table 4: Top 10 topics. Descriptions were manually translated in most cases (original in Greek).

#	Description	#Tweets
1	Papadopoulos & Sons wins audience award	29
2	The official spot of Tiff53	28
3	PapaSonsFilm thanking audience	20
4	Tweets about the opening day	17
5	ThessFest Application	16
6	Screening Schedule	15
7	Festival awards	14
8	Students protest on awards ceremony	13
9	Sunset at Tiff53	13
10	Photo contest	13

were empirically set to 0.25[4]. The top 10 topics (ranked by number of associated tweets) are presented in Table 4. The list includes "official" festival highlights, such as the audience award, the official festival spot and the screening schedule, but also "user-generated" topics such as a student protest and an attractive photo from a sunset in the vicinity of the festival. A qualitative evaluation of the topics (not only of the top 10) was conducted by event organizers resulting in positive feedback regarding the interest that these present from an organizer's perspective. In addition, we examined the full cluster list and for each one of them, we performed the following two quality checks: (a) whether the automatically generated cluster title sufficiently conveys the actual topic, and (b) whether there are tweets associated with this cluster that are irrelevant to the cluster topic. In total, 53,8% of the topic titles were considered sufficiently informative, and 98.5% of the clusters were found to be *pure*, i.e. to contain no irrelevant tweets.

4.4 Sentiment Analysis

We use a separate classifier for the English and the Greek tweets. The training data for both English and Greek was collected similar to [5] by using the Twitter API[5] to fetch tweets with positive and negative emoticons. In total we got 800,000 tweets with positive emoticons and 800,000 tweets with negative emoticons in English. For the Greek training data we got only 12,000 tweets for each class. We use the LanguageWare[6] library to divide the test data into tweets in English and in Greek. The test data was manually labeled and resulted in 324 positive, 33 negative and 724 neutral tweets in English and 901 positive, 315 negative and 1667 neutral tweets in Greek.

We learned threshold θ_3 as follows. For both classifiers, the threshold is learned using a separate dataset which contains tweets from the Thessaloniki Documentary Festival (tdf14). The dataset was manually labeled with sentiment information. After applying language identification, the dataset resulted in 325 positive, 73 negative and 553 neutral tweets in English and 781 positive, 216 negative and 781 neutral tweets in Greek. We obtained an accuracy (precision of correctly classified tweets) of 0.7382 and 0.6185 for the English and Greek test data respectively and 0.6511 for the whole test data. The threshold used is $7.5 \cdot 10^{-4}$ for the Greek classifier and $6.5 \cdot 10^{-4}$ for the English one. Figure 3 shows the accuracy of the English and Greek classifiers for varying threshold θ_3. We can see that the best threshold for both datasets is similar, which could hint that it is language independent. The results for the English test data are considerably better, which can be explained by the much larger training data that was available for English, and at the same time demonstrates the challenges involved in localizing sentiment detection for less popular languages.

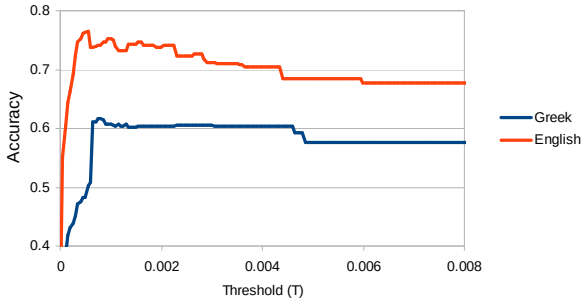

Figure 3: Accuracy vs. threshold θ_3.

4.5 Aggregation and summarization

We applied the analysis of Section 3.5 on the collected set of tweets. Figure 4 shows the hourly Twitter activity from 1 to 13 November. In addition to this, we compute the aggregate sentiment statistics (polarity, subjectivity) per day. Particularly positive days include the day before the TIFF53 beginning, the opening day and the day when the awards ceremony took place. In contrast, Tuesday 6 November was the day with the lowest polarity due to many negative tweets on screened films. The next most negative day was the last day of TIFF53, when many attendants expressed their sad feelings about the festival coming to an end.

Back to the activity timeline, we observe several peaks that correspond to highlights of the event: opening day, screening of Rhino Season, interviews with directors Ghobadi, Gavras and Yannakakis, awards ceremony and closing day. We also observe that several of the peaks are caused by mixed online discussions made for different films (denoted as "several films" in the figure) and other topics. This makes clear the need for tweet-film matching and topic detection for delving further into the online discussions around the event.

For each film screened in TIFF53, we calculate the num-

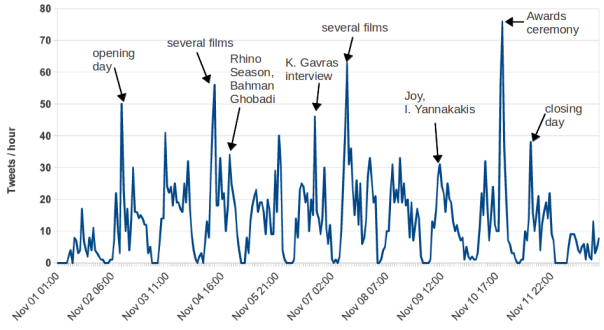

Figure 4: Tweet rate and event highlights.

[4]In the future, we plan to investigate more principled approaches, e.g. training a classifier to detect whether a new tweet belongs to the most similar topic.

[5]dev.twitter.com/docs/api

[6]www-01.ibm.com/software/globalization/topics/languageware/

Table 5: Polarity (Pol), subjectivity (Subj), average rate (R), number of rates (#R) and number of favorites (#F) for the most discussed films.

Film Title	#T	Pol	Subj	R	#R	#F
Papadopoulos & Sons	89	0.74	2.178	3.8	5	10
Holy Motors	53	0.0	0.709	3.6	13	17
Rhino Season	49	0.87	1.579	3.65	27	18
After Lucia	34	1.0	0.545	4.25	10	19
The Capsule	33	0.33	0.833	3.28	14	24
Boy Eating the Bird's...	31	0.23	0.722	2.0	7	5
Le Havre	31	1.0	0.292	4.1	9	17
Dead Europe	29	0.18	3.143	2.4	10	18
A Hijacking	28	1.0	1.154	3.9	16	9
Night of Silence	28	1.0	0.037	3.5	10	10
Beyond the Hills	28	0.78	1.8	3.5	12	12
Tabu	26	0.29	1.889	3.65	9	12
Capital	25	0.5	0.923	3.56	8	8
Higuita	25	-0.71	1.273	1.85	20	23
In Another Country	24	1.0	0.091	3.5	2	6

Table 6: Pearson's correlation across film statistics.

	AVG(Rate)	#Rates	#B
#Tweets	0.101	0.588	0.557
Polarity	0.512	0.032	-0.068
Subjectivity	0.087	0.133	0.004

ber of related tweets (#T) and the actual values of Polarity and Subjectivity. We also get the average rate of each film (R), the number of rates (#R) and the times that a film has been added to a user's schedule (#B), by aggregating usage logs of the ThessFest mobile app. Table 5 illustrates these attributes for the most discussed films (i.e films with large number of tweets). We try to investigate the presence of dependencies between social media information (i.e tweet rate and sentiment) and real world facts (e.g ratings, favourites, etc.). To this end, we compute the Pearson product-moment correlation coefficient between number of tweets, polarity and subjectivity and average rate (Table 6). The correlation coefficient of 0.588 between the number of tweets and the number of rates indicates that there is a profound dependence between these values. The same holds for the #Tweets-#B pair. Regarding the polarity of sentiment about a film, we found that it is correlated with the average film rating. Finally, we could not find any remarkable correlation between the subjectivity around a film and the other attributes. Overall, these results appear very promising, since they reveal that film festival organizers and enthusiasts can infer the impact of films to the festival audience just by looking at the results of EventSense. A limitation of this approach is that it requires a significant number of tweets to be matched to a film in order for the estimate to be reliable.

In Figure 5 we illustrate a scatter plot of average ratings versus (a) the actual polarity computed by the manual sentiment annotations, and (b) the detected polarity. In (a) there is a dependence with a Pearson's coefficient of 0.512, while in (b) there is not. This is probably due to the low accuracy of sentiment detection in Greek.

Other festival highlights are revealed by looking into the most retweeted messages within the #tiff53 stream. Table 7 presents the list of top 10 such messages (some of them

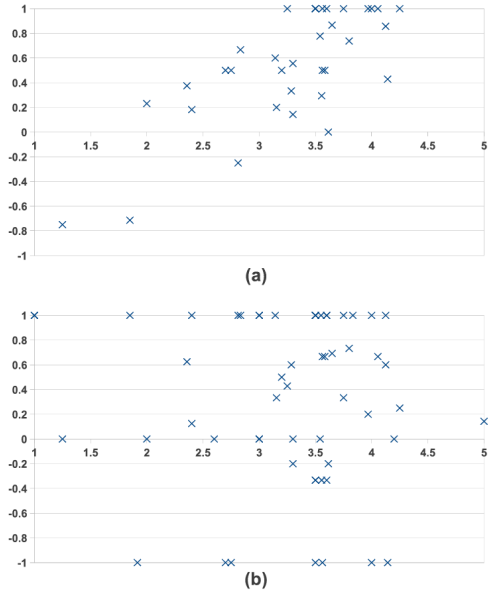

Figure 5: Scatter plot of average rating versus (a) actual polarity and (b) detected polarity.

Table 7: Most retweeted messages

#	Tweets Text	#rt
1	Papadopoulos & Sons wins audience award at Thessaloniki Film Festival. It's the only one that counts! #tiff53	24
2	Amazing reaction from Greek audiences in Thessaloniki! Many said it made them feel proud. Dream come true at #tiff53	19
3	Καλημέρα, ξεκινάμε! :) #tiff53	10
4	Zapas "Right now, being Greek in the film industry is a curse, ppl never pick up the phone, treat you like the devil" #tiff53	10
5	Kalimera!I'm in town and a guest@ #tiff53. Bringing "Papadopoulos & Sons" to you.2morrow 19:30 at Tonia Marketaki th. http://t.co/71yxkIEb	9
6	ThessFest: Το φεστιβαλ ξεκινάει με την ανανεωμένη εφαρμογή για iPhone. To update για το #tiff53 τώρα διαθέσιμο στο https://t.co/dTu59uTg	9
7	53rd Thessaloniki International Film Festival - The official spot: http://t.co/q7rNWjv5 #tiff53	9
8	This is the first day of #tiff53 people! #Thessaloniki #filmfestival #seafront http://t.co/fuZKg8iu http://t.co/YD5Rp8Ki	8
9	#tiff53 - 53ο Διεθνές Φεστιβάλ Κινηματογράφου Θεσσαλονίκης: Το απόλυτο σκονάκι για μηδαμινές απώλειες.: Κάθε χρό... http://t.co/aVy2x6rM	8
10	Thessaloniki sunset today. A big part of why I don't want to leave RT @Bezesteni Can't get enough #tiff53 #thessfest http://t.co/pSYUa1Tz	8

in Greek). Looking into them, we can observe significant overlap with the most discussed topics of Table 4. We can also note that already from this very short list, there is some redundancy, for instance tweets #1, #2, and #5, and tweets #3 and #8, refer to the same topics. This observation further demonstrates (in addition to the experiments of subsection 4.3) the value of clustering tweets around topics.

At a user level, Table 8 presents the most active users based on the number of tweets and the most influential users based

Table 8: Most active and influential users in #tiff53

Activity			Influence		
Author	**#tweets**	**P**	**Author**	**#rt**	**P**
baphometx	310	0.679	filmfestivalgr	123	0.938
asteris	290	0.546	asteris	95	0.546
StellaKarag	212	0.692	baphometx	85	0.679
CinephiliaGr	140	1.000	StellaKarag	78	0.692
manolis	132	0.070	PapaSonsFilm	76	1.000
montagdarko	115	0.353	Bezesteni	46	0.617
filmfestivalgr	115	0.938	manolis	41	0.070
Bezesteni	105	0.617	cinePANikOS	33	0.292
chriszlatis	97	0.667	filmandfestmag	32	1.000
cinePANikOS	90	0.292	kompats	30	0.857

on how many times a user is retweeted. In addition, for each user we also provide the aggregate sentiment polarity of his/her tweets. Such information can be valuable for festival organizers to identify influential users with positive or negative polarity and engage with them in appropriate ways (e.g. encourage/reward the positive influencers, and examine possible complaints from negative influencers).

Finally, mining #tiff53 tweets enabled us to surface the most interesting media content shared during the event. Figure 6 summarizes the most retweeted pictures in the #tiff53 stream. This provides a visual summary of the event highlights, including a party poster, an interview snapshot, a panel discussion, and glimpses over different sights of Thessaloniki. In the case of larger events, we expect that such visual summaries will play an increasingly important role for conveying the event experience and pulse to the audience.

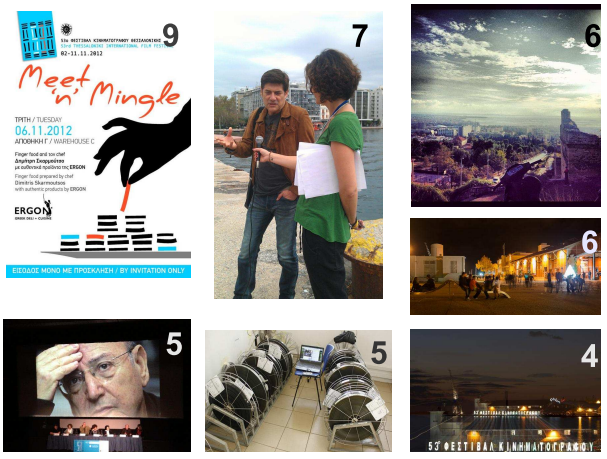

Figure 6: Most popular pictures in #tiff53 stream (numbers in each one of them refer to retweets).

5. CONCLUSIONS

In this paper, we presented EventSense, a framework for the extraction of insights from large events by mining large amounts of online messages shared through OSNs. The framework includes components for the matching of tweets to entities of interest, topic detection by use of clustering, sentiment detection, as well as aggregation and summarization techniques. A case study for the 53rd Thessaloniki International Film Festival (tiff53), with manually created ground truth, demonstrated that EventSense can perform those tasks with sufficient accuracy, to be valuable to event organizers and enthusiasts for gaining insights into the impact that large events have on the audience.

In the future, we plan to apply the proposed framework to larger-scale events (in terms of number of online messages), both film festivals and events of different nature (e.g. music festivals, sports events). In addition, we will consider monitoring and processing more OSN sources (e.g. Facebook, Instagram). Furthermore, we plan to further refine the proposed methods with the goal of improving accuracy and robustness over different datasets. Finally, we intend to experiment with techniques for automatically creating visual informative summaries (infographics) based on the results of the automatic analysis.

Acknowledgements: This work is supported by the SocialSensor FP7 project, partially funded by the EC under contract number 287975.

6. REFERENCES

[1] S. Asur and B. A. Huberman. Predicting the future with social media. *CoRR*, abs/1003.5699, 2010.

[2] A. Bermingham and A. F. Smeaton. Classifying sentiment in microblogs: is brevity an advantage? In *Proceedings of the 19th ACM international conference on Information and knowledge management*, pages 1833–1836. ACM, 2010.

[3] L. L. Bo Pang and S. Vaithyanathan. Thumbs up? sentiment classification using machine learning techniques. In *Conf. on Empirical Methods in Natural Language Processing (EMNLP)*, 2002.

[4] D. Chakrabarti and K. Punera. Event summarization using tweets. In *Proceedings of the Fifth Inter. AAAI Conference on Weblogs and Social Media*, pages 66–73, 2011.

[5] A. Go, R. Bhayani, and L. Huang. Twitter sentiment classification using distant supervision. 2009.

[6] C. D. Manning, P. Raghavan, and H. Schütze. *Introduction to information retrieval*, volume 1. Cambridge University Press Cambridge, 2008.

[7] A. Marcus, M. S. Bernstein, O. Badar, D. R. Karger, S. Madden, and R. C. Miller. Twitinfo: aggregating and visualizing microblogs for event exploration. In *Proceedings of the SIGCHI Conference*, CHI '11, pages 227–236, New York, NY, USA, 2011. ACM.

[8] B. O'Connor, R. Balasubramanyan, B. Routledge, and N. Smith. From tweets to polls: Linking text sentiment to public opinion time series. In *Proceedings of ICWSM*, pages 122–129, 2010.

[9] B. O'Connor, M. Krieger, and D. Ahn. Tweetmotif: Exploratory search and topic summarization for twitter. *Proceedings of ICWSM*, pages 2–3, 2010.

[10] A. Pak and P. Paroubek. Twitter as a corpus for sentiment analysis and opinion mining. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, 2010.

[11] S. Petrović, M. Osborne, and V. Lavrenko. Streaming first story detection with application to twitter. In *Human Language Technologies*, HLT '10, pages 181–189, Stroudsburg, PA, USA, 2010. ACL.

[12] D. Shamma, L. Kennedy, and E. Churchill. Tweetgeist: Can the twitter timeline reveal the structure of broadcast events. *CSCW Horizons*, 2010.

[13] N. Shuyo. Language detection library for java, 2010.

[14] S. Zhao, L. Zhong, J. Wickramasuriya, V. Vasudevan, R. LiKamWa, and A. Rahmati. Sportsense: Real-time detection of nfl game events from twitter. *arXiv preprint arXiv:1205.3212*, 2012.