Media REVEALr: A Social Multimedia Monitoring and Intelligence System for Web Multimedia Verification

Katerina Andreadou[1], Symeon Papadopoulos[1(✉)], Lazaros Apostolidis[1], Anastasia Krithara[2], and Yiannis Kompatsiaris[1]

[1] Information Technologies Institute, CERTH, Thessaloniki, Greece
papadop@iti.gr
[2] NCSR Demokritos, Aghia, Paraskevi, Greece

Abstract. Modern online social networks, such as Twitter and Instagram, are nowadays important sources for publishing information and content around breaking news stories and incidents related to public safety, ranging from natural disasters and aeroplane accidents to terrorist attacks and industrial accidents. A crucial issue regarding such information and content is the extent that they can be relied upon and used for improving the situational awareness and operational capabilities of decision makers. Given the proliferation of noisy, irrelevant and fake content posted to such platforms, two important requirements for systems supporting the information access needs in incidents, such as the ones described above, include the support for understanding the "big picture" around the incident and the verification of particular pieces of posted content. To this end, we propose Media REVEALr, a scalable and efficient content-based media crawling and indexing framework featuring a novel and resilient near-duplicate detection approach and intelligent content- and context-based aggregation capabilities (e.g. clustering, named entity extraction). We evaluate the system using both reference benchmark datasets as well as datasets collected around real-world incidents, and we describe the ways it contributes to the improvement of the situational awareness and journalistic verification in breaking news situations, like natural disasters.

Keywords: Social media monitoring · Event mining · Situational awareness · Multimedia verification · Breaking news reporting

1 Introduction

We propose a framework for the real-time collection, indexing and search of multimedia elements from several social media sources. Breaking events, for instance a natural catastrophe (e.g., Hurricane Sandy), a terrorist attack (e.g., Boston Marathon bombings), an aeroplane crash or a massive protest, naturally attract the attention of locals and Internet users, who in turn flood the social networks with personal comments, stories, images and videos. Popular and widespread

M. Chau et al. (Eds.): PAISI 2015, LNCS 9074, pp. 1–20, 2015.
DOI: 10.1007/978-3-319-18455-5_1

Fig. 1. Hurricane Sandy tweet with a more than one-year-old real thunderstorm photo

subjects, however, tend to also cause an abundance of fake media content. An image might be photoshopped in order to convey a certain message or opinion concerning the subject in question, and it might often be maliciously manipulated in order to trigger the public opinion and provoke a specific reaction. It might even be the case that a real picture of a past event is retweeted and presented as depicting current events. An example of this can be seen in Figure 1, where a picture of a thunderstorm over New York dating from 2011 during a tornado alert was massively retweeted as "a picture of Hurrican Sandy descending in New York" in late 2012.

A framework for the real-time mining of online multimedia content can become an analysis tool of utmost importance for journalists, analysts as well as users, who intend to be well-informed. To the best of our knowledge, Media REVEALr is the first real-time social multimedia indexing and mining system that is designed for supporting the content and information verification needs in the context of breaking news stories and incidents such as the aforementioned ones.

Although a variety of text and social media data analysis approaches have been previously proposed in similar settings with the goal of mining information out of big sets of social network posts (e.g. topic detection and social graph analysis), they lack support for improved situational awareness and content verification. In particular, Media REVEALr offers the following unique capabilities:

- it enables the precise and resilient identification of near-duplicate images and videos (based on selected keyframes) in a stream of social media content even in the presence of overlay graphics and fonts;
- it supports the identification and comparative view of multiple independent sources of content that discuss the same incident;

– it extracts and aggregates the named entities extracted from the collected social media messages and presents them through an intuitive and visually appealing interface with the goal of improving the situational awareness over the incident of interest.

We evaluated the main components of the proposed framework in a number of reference archived datasets and in the context of datasets collected in vivo around real-world incidents. The obtained results validate the performance of the framework components and demonstrate that, overall, Media REVEALr offers a powerful tool that information analysts, reporters and decision makers can use in real-time settings for gaining better understanding over the evolving incident and over the veracity of posted content.

2 Related Work

Mining social media content has arisen as an important problem in the last decade due to the increasing tendency of users to produce and share content, in the form of text messages, images and videos. This problem has been extensively examined from various perspectives. Gradually the focus has shifted to real-time mining approaches, because the need for real-time analysis becomes very challenging as the amount of data increases and existing systems fail to cope with the rate and scale of the incoming content. Moreover, the dynamic nature of social network streams such as Twitter, calls for specially designed approaches, which take this dynamic nature and the massive size of the data into account.

Marcus et al., in [8], presented a novel system for visualizing and summarizing events on Twitter with the goal of offering to the end users a comprehensive chronological representation, highlighting the relevant peaks of activity. This way, the users can pick an interest point on this timeline and explore the Twitter stream further by geolocation, sentiment and popularity.

Mathioudakis and Koudas proposed TwitterMonitor, a system for real-time trend detection over the Twitter stream [9]. This automatically identifies emerging topics and provides users with meaningful and comprehensive statistics and related information. TwitterMonitor performs trend detection, by first identifying emerging new keywords, then grouping them together based on their co-occurrences and then mining the tweets that belong to the identified trend in order to disclose its hidden aspects.

Some other related approaches attempt to identify the virality of content based on community detection and graph analysis methods. For instance, in [14], Weng et al. propose a system for predicting successful memes based on their early spreading patterns. They present two different approaches: The first one is based on time series analysis of the spreading patterns of a meme in its early stages of diffusion. The second employs a supervised learning approach using a variety of features, which represent the meme popularity (network topology, community diversity and growth rate).

Visual memes, images or video segments that are replicated amongst social network users, and corresponding to breaking news stories and events, are also the subject of other relevant fields of research. For instance, in [15], Xie et al. attempt to comprehend the diffusion of videos containing one or more memes and to predict the lifespan of such a video and hence, the conveyed message. In [11], Petkos et al. propose a semi-supervised method for clustering multimedia items in order to predict social events.

An additional application of such methods, which is based on the visual content of social media items, is the identification of the history of an image, once a certain number of image copies or near-duplicates has been gathered. In [7], Kennedy and Chang propose a system for tracing the manipulations that an image has undergone through the course of time in order to highlight the perspectives of the authors and draw interesting conclusions on the diffusion patterns (e.g. images that are closer to the original tend to have more descendants).

Although previous systems and approaches were developed to address the individual problems of event summarization, content diffusion and verification, there has not been an overall system that offers support for all aforementioned problems. To this end, Media REVEALr aspires to provide a comprehensive solution to problem settings where large streams of social multimedia content need to be mined and verified.

3 Overview

3.1 Architecture

Figure 2 illustrates the basic elements of the proposed framework and their interconnections. The crawler is responsible for collecting text messages, images and their respective metadata from a number of social networks (in this work we focus on Twitter) and the Web (e.g. specific news sites). The collected HTML content is then processed by the HTML semantic segmentation module, which extracts the clean text as well as the main article and accompanying images from the Web pages. In the case of social network posts, this step is not necessary as the content is collected by querying the respective social network APIs and eventually all the desired information and fields are available in structured format. Finally, the Named Entity Recognition (NER) module extracts the named entities from the collected text elements, and the respective texts are also indexed in Solr. Further details on this step are provided in section 3.2.

Regarding the processing of visual content, the main component of the system is the visual feature extractor, which performs the extraction of a feature vector from every incoming image (or video frame). Further details on this step are given in sections 3.3 and 3.4. Subsequently, for every processed image, the metadata is saved in a mongo database and the feature vector is provided to the visual indexer, which constitutes a highly optimized search structure that supports near-duplicate image retrieval. As a last step, the clustering component organizes the collected images into groups based on their visual similarity; the information about the resulting clusters is again stored in the mongo database.

The proposed architecture enables the provision of numerous capabilities through a single API: advanced keyword-based queries (implemented on top

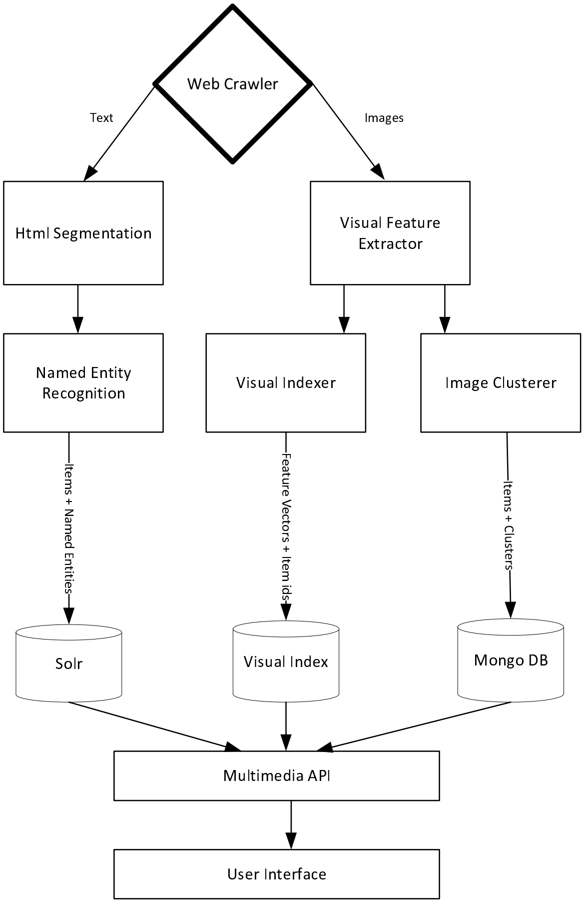

Fig. 2. Media REVEALr architecture

of the Solr text indexing engine), search queries based on the image metadata stored in mongoDB (e.g. width, height, publication date, name of the publisher) and most importantly visual similarity search queries, which, given an image, attempt to retrieve near-duplicates. The interaction of the user with the system services takes place through a web-based user interface that is described in detail in section 3.6.

Our choice of Solr for full-text indexing and mongoDB as a datastore is due to their cross-platform portability, robustness, scalability, and their widespread use in similar projects. Furthermore, the fact that mongoDB is schemaless and natively supports the storage and querying of JSON documents, simplifies many operations (e.g exposing the data through a web service).

3.2 NLP: Named Entity Detection

Named Entity Recognition (NER), is a well-known problem that has been around for decades (e.g. [4]), while entity linking and tweet tagging are newer problems that emerged in the past few years [3]. Nevertheless, because of their importance for a large variety of text-centric applications, these problems have received significant and increasing attention. Despite this attention, few solutions exist today to solve these problems for social media text content, and these solutions are limited in several important ways. First, the solutions often recycle techniques developed for well-formed English texts. However, a significant portion of the text content found in social media contains misspelled ungrammatical short sentence fragments, thereby proving very challenging for conventional techniques. Second, the solutions often employ computation-intensive techniques that do not scale to high-speed streams, which could reach rates up to thousands of items per second (in the case of Twitter). Third, existing solutions typically do not exploit context information, such as the topics discussed in users' messages. In the past years, several systems have been in place to extract, link, classify and tag Web text data, such as OpenCalais[1] and the Stanford Named Entity Recognizer[2].

However, to our knowledge, none of these deployed systems has been specifically tailored for social media. In this paper, we adapt an existing approach to the nature of social media texts, by taking into account their specific characteristics, such as hashtags, mentions, etc. In particular, for every item, the named entities are extracted from the text (e.g., the tweet) using the Stanford NER. Additionally, some pre-processing steps allow to also identify named entities that are contained in mentions and hashtags, a typical phenomenon in tweet texts.

The pre-processing steps include tokenization, user mention resolution and further text cleaning. Tokenization is used to identify the tokens that will be replaced or removed from the text, such as URLs, `@user` mentions, etc. First, we exploit tweet metadata to resolve user mentions to their canonical names. In particular, each tweet that contains a `@user` mention, carries a list of the corresponding full user names. Thus, we substitute the user mentions in the tweet text with the corresponding full names using the tweet metadata. We then use regular expressions to remove the remaining special symbols, such as `@` and `#`, URLs and non-ASCII characters.

3.3 Visual Indexing

Similarity-based image search, also known as Content-based image retrieval (CBIR), is the problem of retrieving similar images to a given query image, based solely on the content (pixels) of the given image and no additional textual (tags) or geographic metadata (e.g. GPS coordinates). In the context of Media REVEALr, we are interested in retrieving similar images in the sense that they have the same original content (one of them has been post-processed,

[1] http://opencalais.com
[2] http://nlp.stanford.edu/software/CRF-NER.shtml

e.g. cropped, scaled, color-adjusted) but also images that depict the same object or scene even when viewed from a different viewpoint.

The design and deployment of an efficient image search system calls for the suitable and optimized combination of several multimedia analysis components. To this end, for our implementation we selected SURF descriptors for the feature representation in combination with VLAD aggregation and Product Quantization (PQ)-based indexing. This setup was demonstrated in [13] to outperform several other state-of-the-art image search systems. Below, we provide a quick summary of the individual components of the image processing pipeline.

Speeded-Up Robust Features (SURF) [1] include a high-performing scale- and rotation invariant interest point detector and descriptor. SURF is much faster to compute compared to SIFT, which is why it has emerged as a popular choice for real-time image search application in the past few years. Vector of Locally Aggregated Descriptors (VLAD) provides an extension of the BoW aggregation mechanism and proposes a way of aggregating local image descriptors into a vector of low dimensionality. Empirical results in the domain of image search indicate that VLAD significantly outperforms BoW for the same length, while being equally fast to compute [6]. Despite being relatively compact compared to BoW, the "raw" dimensionality of VLAD is still prohibitive for large-scale search applications. For this reason, Jégou et al. [6] propose applying PCA to reduce the dimensionality of the vectors by an order of magnitude (e.g. from 4096 to 128) with negligible impact on accuracy.

3.4 Overlay Detection

The main problem we identified with the previous image retrieval system was that the algorithm did not perform well for images with overlays, for instance quotes, fonts and banners, as in the case of popular Internet memes. This is in fact a very interesting use case as it could allow to trace the history of the different ways in which an image has been manipulated and republished.

Our approach to solving this problem is to train a suitable classifier, which is inserted immediately before the VLAD aggregation in Figure 3 in order to filter out overlay SURF descriptors, which belong to areas of the image with fonts, banners, etc. This way, the remaining SURF descriptors will correspond to pure image content. The proposed solution is illustrated in Figure 3. From now on, we will refer to this method as NDS+ as opposed to the simple near duplicate search method described in section 3.3, to which we will refer as NDS.

The descriptor filtering step was implemented based on a supervised learning approach. To this end, we carried out several experiments with different classifiers and different configurations and we concluded to one that outperforms the rest, using Random Forest as base classifier and a Cost Sensitive meta-classifier on top of that, to penalize the misclassification of true positives.

The most difficult challenge during the supervised training stems from the *class imbalance problem*. One class (the font descriptors) is represented by only a few examples in comparison to the other class (the non-font descriptors). Furthermore, the proportion of images with fonts in our training set is much higher than the one

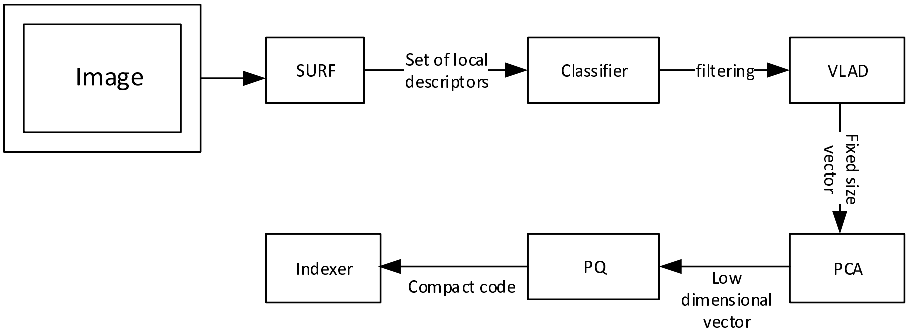

Fig. 3. NDS+ scalable similarity-based image search based on SURF, overlay classifier, VLAD, PCA and PQ

Fig. 4. Examples of filtered-out SURF descriptors

in a real-world scenario, in the whole web for example. When class sizes are skewed, it is quite easy to achieve good accuracy just by overlooking small classes and adjusting the classifier to yield good results for the over-represented class. So, it might be quite simple to create a classifier with a 99% accuracy when the majority class represents 99% of the samples, but it is hard to train a classifier evenly in order to have almost equal true and false positive error rates. This is why we measured the precision and recall for every class and tuned the classifier to achieve the best performance for both classes. Several methods have been proposed to tackle the class imbalance problem. Most of them are based on random over- or under-sampling and some attempt to remove noisy examples before training the classifier. The selected Cost Sensitive meta-classifier performs over-sampling of the minority class in order to balance the dataset.

To train the overlay descriptor classifier, we used example images with fonts, quotes, banners, etc. that we manually collected with the help of image search engines. Obviously, none of those images were used for evaluating the performance of the classifier to avoid misleading results (typically over-optimistic). The positive class annotation was carried out by manually defining the regions of each image that contain fonts and then training the classifier using those as positive examples and the descriptors contained in the rest of the regions as negative examples. For the manual annotation of the regions, we used a slightly altered version of ImageJ. Figure 4 depicts two example images highlighting the SURF descriptors that were detected by the classifier to be overlaid on the image.

3.5 Mining: Clustering and Aggregation

An added value of Media REVEALr lies in the ways the data is aggregated to facilitate the user in identifying the context of use of a particular media item within a large collection of seemingly unrelated images, and in revealing hidden relations and dependencies among the multimedia items. Two types of aggregation are offered:

– visual aggregation by creating clusters of images based on visual similarity;
– entity aggregation by extracting named entities from the accompanying text and then grouping images together based on the entity occurrences.

For visual aggregation, the clusters are created in batch mode after the collection of the images has been completed using the DBSCAN algorithm [2] and consequently each cluster is represented by its most representative image. This is defined as the image with the largest amount of occurring keywords in the accompanying text. This helps the user to easily grasp the visual context of a specific story. The clustering algorithm at the moment is based solely on visual features, in particular the Euclidean distance between the PCA-reduced SURF-VLAD vectors.

Entity aggregation is performed using the Named Entity Recognition process described in subsection 3.2 and subsequently computing the frequency of every named entity in the corpus of collected content items. Eventually, entities are ranked according to the frequency of their appearance. This way, the prominent aspects of an incident (persons, events, locations etc.) stand out and offer a concise semantic view of the incident of interest.

3.6 User Interface

The user interface consists of different views, which, combined with the real-time analysis of the collected data, facilitate the exploration of all available data aspects and the evaluation of the produced results.

In the first view (Figure 5), all available crawled collections are presented, accompanied by some statistics, such as the creation date, the time the last item was inserted, the total duration of the crawl, the total number of images and videos, the keywords for the crawl and the current state. The state is signified by a coloured bar which is grey for waiting, green for running and red for stopped or finished. Every collection is represented by a card and the user can interact with it in the following ways: stop a running crawl, delete a collection, or click on it for further exploration, which navigates the user to the next three views.

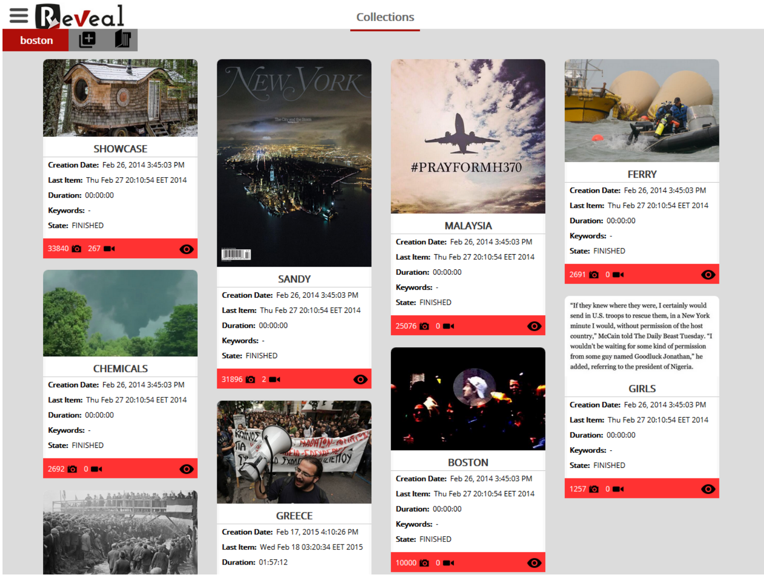

Fig. 5. Collections View

In the second view (Figure 6), one can observe all items of the collection of interest. For every item, a multitude of related information is available: the date it was published, the related text (e.g. the tweet), how many times it was shared, the name of the author and a thumbnail of their profile picture. One can click on the user thumbnail to visit the users profile page or on the image to see a bigger version and the full text. Additionally, this view offers a variety of search options. The user can drag and drop an image or video in the search box to search for similar images. He/she can also use the search dialog to filter the images by several criteria: image dimensions, the date it was published, the username of the publisher or some text search terms.

The third and fourth views (Figures 7, 8) are the mining and aggregation views. The cluster view presents the clusters that were created as a result of the DBSCAN algorithm. For every cluster there is a representative image and some additional information, such as the number of items in the cluster. By clicking on the cluster item, one can navigate in a detailed view where all the items of the cluster are available. In the named entities view one can explore the extracted named entities that can additionally be grouped by appearance (often, occasionally, seldom) or by type (person, location, organization, other). By clicking on a named entity bubble, the user can again navigate to a detailed view where all items containing this entity are presented.

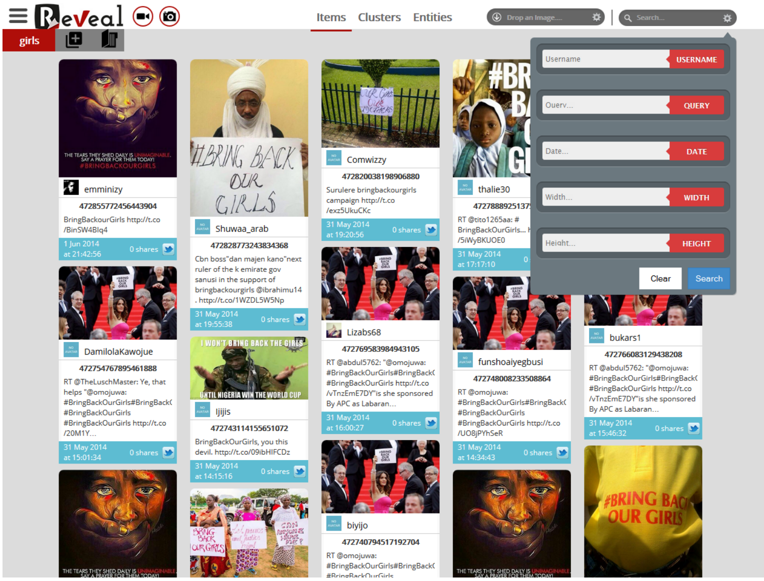

Fig. 6. Items View

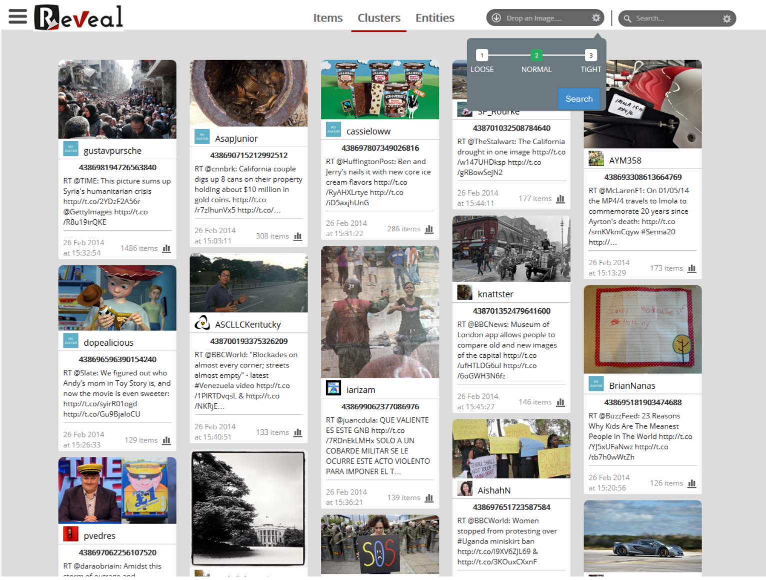

Fig. 7. Clusters View

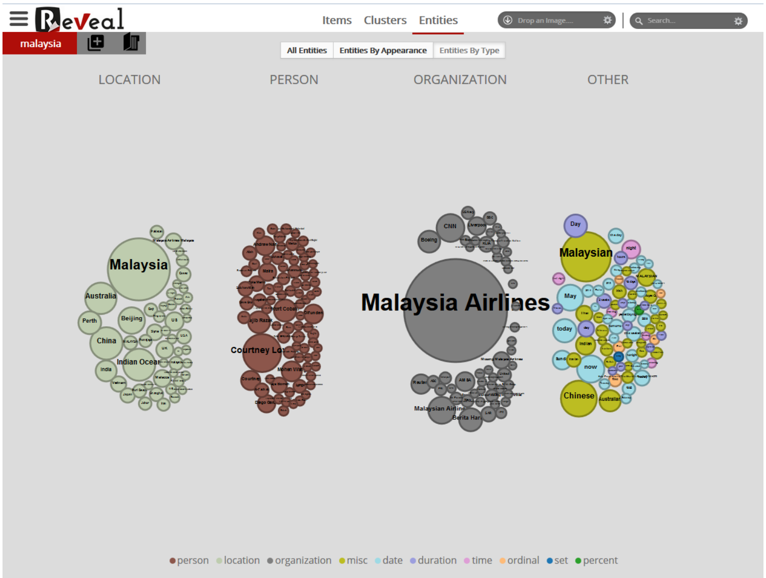

Fig. 8. Entities View

4 Evaluation

4.1 NLP

For the evaluation of the Named Entity Recognition approach, we manually annotated 400 tweets from the SNOW Data Challenge dataset [10]. The tweets were randomly selected out of approximately 1M tweets of the dataset, making sure that they cover different topics. As a quality metric we use accuracy. As we are interested in the identification of both the entity and its type, we consider a result correct if both the entity and the type are correctly recognized. We compared our approach (S-NER+) with Stanford NER without any pre-processing (S-NER) and Ellogon NER[3]. As we can see in Table 1, the extended NER approach (S-NER+) managed to achieve better accuracy than the rest of the approaches. The reason is that it manages to identify cases where the name entity is "hidden" behind mentions or hashtags. For example, with the help of the pre-processing steps the proposed approach managed to find the name entity `Michael Owen` when it appears as `@themichaelowen` in a tweet.

Table 1. Comparison between accuracy of tested NER approaches

	S-NER+	S-NER	Ellogon
accuracy	**0.852**	0.827	0.653

[3] http://www.ellogon.org/

4.2 Near-Duplicate Detection with Overlays

For the evaluation of our overlay detection system, we used the following publicly available benchmark datasets:

– The Holidays dataset [5] contains 1491 holiday images, 500 of which are used as queries.
– The Oxford dataset [12] consists of 5063 images collected from Flickr by searching for particular Oxford landmarks. 55 of the images are used as queries.
– The Paris dataset [12] consists of 6412 images collected from Flickr by searching for particular Paris landmarks. 55 of the images are used as queries.
– The SNOW Data Challenge dataset [10] consists of approximately 1M tweets, from which the images are extracted.

Given a reference dataset, where for each test query, the "correct" results are known, there are several ways of assessing the retrieval performance of a system. The Average Precision (AP) is a measure that rewards a system's ability to retrieve relevant documents in the first positions. If in the list of ranked images, all the relevant images come first, then AP will be 1. For multiple queries, we compute the mean Average Precision (mAP) by averaging the AP scores over all queries. It is important to note that mAP depends on the dataset: scores achieved on different datasets are not easily comparable. The Equal Error Rate (EER) or Crossover Error Rate (CER) indicates the rate at which the proportion of false acceptance is equal to the proportion of false rejections. It is possible to have a relatively poor EER score with a relatively good AP score.

Which evaluation metric is the best depends really on the type of task at hand. In a ranked retrieval case, where a human is going to look at the first few images, it makes sense to try and maximize the AP in order to improve the user experience. On the other hand, maximizing the EER may improve the performance of completely automated scenarios. We decide to evaluate accuracy using the mAP score, because the user friendliness of the results seems to be more important according to the system requirements.

In the first experiment, of which the results are presented in Table 2, we are interested in finding out the impact of adding the overlay classifier to the base near-duplicate search pipeline. The results demonstrate that there is a marginal deterioration of the performance, ranging from 0.39% in the case of the Holidays dataset to 2.67% in the case of Oxford buildings. Such small deterioration is expected since none of the test images contain any overlay descriptors; hence, any descriptor filtered by the overlay classifier would be a false positive, thus affecting the overall accuracy of near-duplicate search.

In the second experiment, we created a new set of images by adding fonts and banners to 55 of the query images of the Holidays dataset and calculated the mAP score for this smaller dataset with and without the use of the classifier. Moreover, we gradually added more irrelevant images (distractors) to the collection from a common Flickr dataset in order to evaluate the robustness of our method in the presence of noise. Table 3 presents the obtained results.

Table 2. Comparison of mean Average Precision (mAP) between NDS and NDS+

Dataset	NDS	NDS+
Holidays	0.7076	0.7037
Oxford buildings	0.4832	0.4565
Paris buildings	0.4824	0.4666

Table 3. Comparison of mean Average Precision (mAP) between NDS and NDS+ (queries with overlays) in the presence of noise (distractor images)

Noise	NDS	NDS+
0	0.8668	0.9049
1K	0.6320	0.7216
5K	0.6197	0.7216
20K	0.5611	0.7455

Something that might strike the reader as surprising is the fact that the mAP in the first two rows of Table 3 is very high, even when not using the classifier. This is due to the fact that the mixed dataset is very small in comparison (79 images for 55 queries). As we can see, by gradually adding more distractor images, the mAP of the NDS+ system stays almost constant while the NDS mAP greatly diminishes, leading to a difference in mAP of 18.44% when 20K distractor images are added to the test collection.

To also give a qualitative impression of the impact of the supplementary classification step, Figure 9 illustrates an example where the classifier greatly improves the performance of the regular system. We indexed the first image in a set of approximately 20K images, and we attempted to retrieve the nearest neighbour of the second one (image with overlay fonts). With the use of the classifier the image on the left came as the first result, while without the classifier it was 27th, ranked lower than 26 irrelevant results.

Fig. 9. Example image showcasing the resilience of the system

Table 4. Average execution time per operation and total processing per image (msec)

Operation	**1K**	**5K**	**20K**
Feature extraction	80	80	79
Classification	18	18	17
VLAD & PCA	54	54	52
Indexing	124	119	110
Total	276	271	258

To evaluate the impact of the use of the classifier on the system efficiency, we timed the different steps of Figure 3, when indexing a collection of 1K, 5K and 20K images. The measured times represent the average duration of the specified processing step per image and are presented in Table 4.

One can see that the average processing time is about 250ms on an Intel Core i7-4770K CPU @ 3.50 GHz processor with 16GB of RAM and the classification delay is minimal (18ms or 6.5 %). Another critical observation is that the whole processing operation per image is independent of size of the image collection. This is of great importance for the system as we aim to deploy our framework for large scale collections, so it is vital that the indexing time is not influenced by the collection size.

4.3 Use Cases

Here, we go through some case studies based on Twitter datasets that were collected around a number of recent incidents of interest with respect to public safety: a) the Malaysia MH370 missing flight incident that took place on March 8, 2014, b) the Boston Marathon bombings taking place on April 15, 2013. As will be seen in the next sections, the proposed system offers valuable insights into hidden aspects of the collected content, which would not be straightforward to gain unless one went through all the items trying to find correlations and underlying relations. The aforementioned two datasets along with five more datasets (Hurricane Sandy, #BringBackOurGirls crisis in Nigeria, MV Sewol sinking, Columbian Chemicals hoax) and the SNOW dataset [10], were inserted and explored through a locally deployed version of the tool. The tweet IDs and image URLs for all seven datasets are publicly available[4].

Incident Summary View (`malaysia`). As a first step, grouping the named entities extracted from the set of collected tweets by type, we can get a quick grasp of all the involved parties and the main information about the event (Figure 10). We can directly deduce that the main location of the event is Malaysia, with a couple of secondary locations being involved, namely China and Beijing (where the flight was heading) as well as places where the search effort extended, for instance Australia, India, Perth, Indian Ocean; in addition, the principal organizations involved are Malaysia Airlines, CNN, Reuters and Boeing.

[4] http://mklab.iti.gr/files/media_revealr_data.zip

Table 5. Twitter event datasets. The clusters were created using DBSCAN, using parameters $\epsilon = 1$ and `minPts`=3.

Collection	Description	Images	Users	Clusters
`snow`	One day of Twitter data [10]	33840	26877	184
`sandy`	Hurricane Sandy	25765	25110	28
`malaysia`	Malaysian Airlines flight MH370	24784	22175	940
`boston`	Boston Marathon bombings	5487	4683	181
`ferry`	South Korea ferry sinking	2524	1921	94
`girls`	Girls kidnapped in Nigeria	1250	1163	56
`chemicals`	Chemicals plant explosion hoax	377	307	13

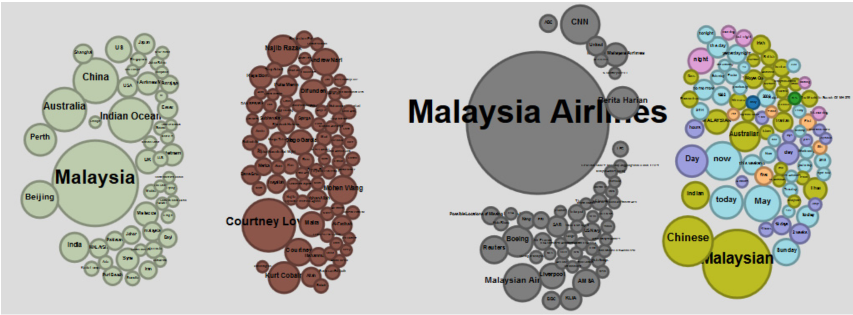

Fig. 10. Named entities for the missing Malaysia Airlines flight MH370

What at first glance strikes us as odd is the fact that the main persons related to the story are Courtney Love and Kurt Cobain, followed by the names of passengers, crew members as well as politicians and officials, who often made statements about the developments of the ongoing search for the missing aircraft. For instance Najib Razak, the Malaysian Prime Minister, Mohen Wang, a 2-year old passenger onboard, Rajah Bomoh, a kind of shaman who performed rituals to locate the missing plane. The reason why Courtney Love appears as the most prominent one among the person named entities is that she posted a picture on her Facebook page claiming that she had located the plane, this picture went subsequently viral on social media and eventually Love's and Cobain's names were connected to the MH370 story through this mishap. By clicking on the respective items on the UI, we can see some of the tweets referencing the original post from Courtney Love (Figure 11).

Near-Duplicate Search (`boston`). A further analysis step in breaking news situations, such as the Boston Marathon bombings, is the near-duplicate search functionality, which is a valuable assisting tool when looking for repostings and retweets of the same multimedia content. In Figure 12, we used the image in the center as a query and configured the search filter to be loose (lower similarity threshold). It is evident from the results that, apart from retweets of the same

Fig. 11. Tweet referencing Courtey Love's Facebook post

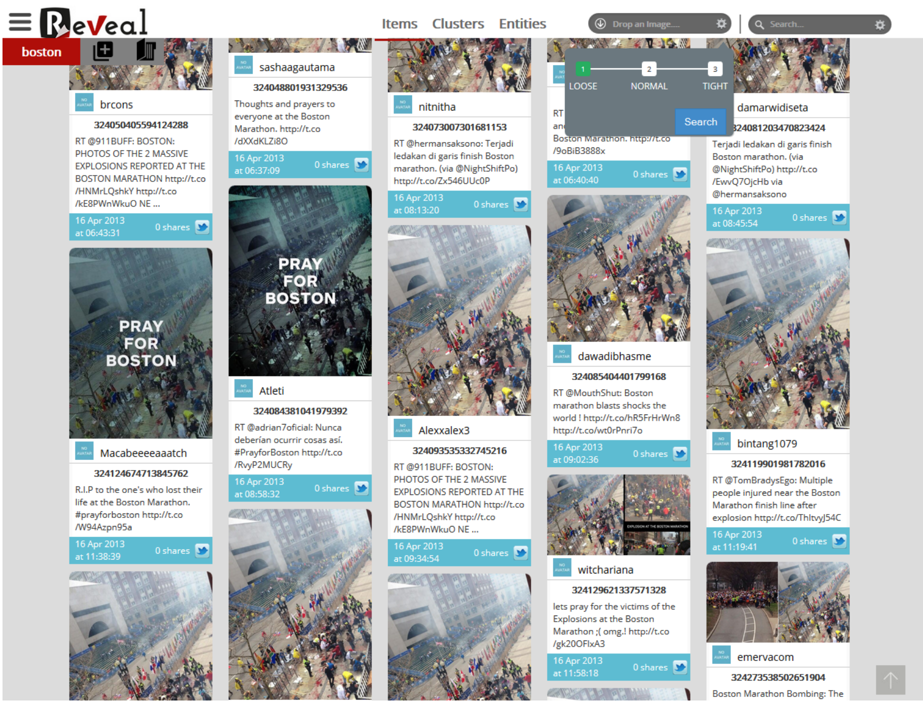

Fig. 12. Near-duplicate search with "loose" threshold

image, the system is also able to find images which have been color adjusted, images with overlays as well as images that contain the image in question as their part (splicing).

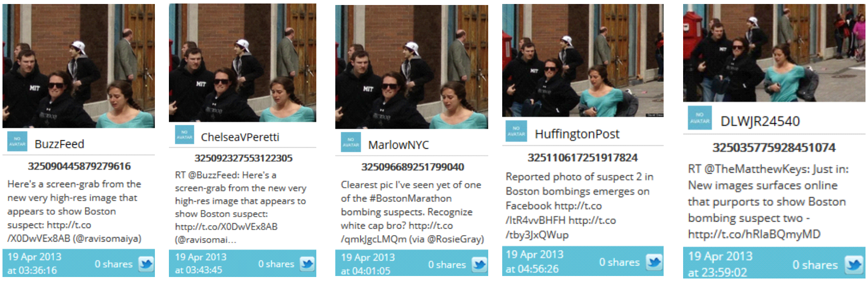

Fig. 13. Cluster items for the Boston marathon bombings

Image Clustering and Message Comparison (`boston`). On the other hand, if we take a closer look at the image clusters, which are created based on the visual similarity of the images, we can observe how different people comment on the same image when retweeting and which tweets have the highest penetration (i.e. are retweeted more often). An interesting observation concerns the way in which the degree of confidence expressed by the tweet text increases as the time goes by and the information gets cross-checked from other sources. For instance, after the Boston marathon bombing, there was intense discussion on Twitter about a high-resolution photo of one of the suspects (Figure 13). The first tweet reports *"Clearest pic Ive seen yet of one of the #BostonMarathon bombing suspects. Recognize white cap bro?"*. Fifty minutes later there is another one: *"Reported photo of suspect 2 in Boston bombings emerges on Facebook"* And then some hours later there is another, which is the most retweeted one: *"Just in. New images surfaces online that purports to show Boston bombings suspect two"*.

5 Future Work

In the future, we plan to extend the system with additional components that will reinforce its analysis capabilities. Our efforts will be concentrated on developing an efficient algorithm for dynamic clustering and then applying it to a growing set of dynamic data, e.g., from long-running crawling jobs. Moreover, we plan to add a temporal analysis component and respective view to the web user interface in order to allow for a more detailed observation of the multimedia items of interest along time.

Finally, we also aspire to develop a multimedia forensics toolbox that will provide insight into the diffusion patterns of fake multimedia items in social networks and will facilitate image and authorship verification. In breaking news situations, there is a flood of pictures uploaded by locals or witnesses at the spot; simultaneously, however, there is a growing spread of fake images, which are published with the purpose of causing a certain reaction or of becoming viral by being provocative or sensational. In the case of the Hurricane Sandy for instance, a popular picture was circulated of a shark swimming in the streets of

Brigantine, which was later exposed as a hoax, but was however earlier shared by tens of thousands of people in social networks. The multimedia forensics toolbox might be a great tool for journalists straggling to verify whether a photograph is authentic or the person who claims to have captured it is indeed the creator. Once a fake has been identified, the user could explore the publishing timeline to gain insight into the diffusion patterns of forgeries and their relation to the initial source.

6 Conclusions

We demonstrated the operation and usage of Media REVEALr on the problem of real-time collection, indexing and search of multimedia-enriched social network items. We have further demonstrated the advantage of our image similarity search algorithm, which combines fast and efficient indexing and search functions with an addition to exclude overlays from the feature vectors in order to diminish the influence of such manipulations on the findability of the original image. Last but not least, we have showed the qualities of the user interface, which allows to aggregate multimedia items in an effort to support the user in grasping the context of the story in a faster and more meaningful way.

We evaluated our system with a variety of benchmark datasets and demonstrating the robustness and effectiveness of the proposed extensions in comparison to the base approaches (18.44% improvement for NDS+ and 1.1% for S-NER+). Additionally we demonstrated the ways an analyst researching a specific subject can gain insights by means of the system user interface, and we illustrated specific real-world cases, where the context of the social media activity comes to light thanks to the detailed information that was extracted from the analysis performed by the system.

Acknowledgement. This work was supported by the REVEAL project, partially funded by the European Commission (contract no. FP7-610928).

References

1. Bay, H., Ess, A., Tuytelaars, T., Van Gool, L.: Speeded-up robust features (surf). Comput. Vis. Image Underst. 110(3), 346–359 (2008)
2. Ester, M., Kriegel, H., Sander, J., Xu, X.: A density-based algorithm for discovering clusters in large spatial databases with noise. In: Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD 1996), Portland, Oregon, USA, pp. 226–231 (1996)
3. Gattani, A., Lamba, D.S., Garera, N., Tiwari, M., Chai, X., Das, S., Subramaniam, S., Rajaraman, A., Harinarayan, V., Doan, A.: Entity extraction, linking, classification, and tagging for social media: A wikipedia-based approach. Proc. VLDB Endow. 6(11), 1126–1137 (2013)
4. Hobbs, J.R.: Fastus: A system for extracting information from natural-language text. Technical Report 519, AI Center, SRI International, 333 Ravenswood Ave., Menlo Park, CA 94025 (November 1992)

5. Jegou, H., Douze, M., Schmid, C.: Hamming embedding and weak geometric consistency for large scale image search. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part I. LNCS, vol. 5302, pp. 304–317. Springer, Heidelberg (2008)
6. Jégou, H., Perronnin, F., Douze, M., Sanchez, J., Perez, P., Schmid, C.: Aggregating local image descriptors into compact codes. IEEE Transactions on Pattern Analysis and Machine Intelligence 34(9), 1704–1716 (2012)
7. Kennedy, L., Chang, S.-F.: Internet image archaeology: Automatically tracing the manipulation history of photographs on the web. In: Proceedings of the 16th ACM International Conference on Multimedia, MM 2008, pp. 349–358. ACM, New York (2008)
8. Marcus, A., Bernstein, M.S., Badar, O., Karger, D.R., Madden, S., Miller, R.C.: Twitinfo: Aggregating and visualizing microblogs for event exploration. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI 2011, pp. 227–236. ACM, New York (2011)
9. Mathioudakis, M., Koudas, N.: Twittermonitor: Trend detection over the twitter stream. In: Proceedings of the 2010 ACM SIGMOD International Conference on Management of Data, SIGMOD 2010, pp. 1155–1158. ACM, New York (2010)
10. Papadopoulos, S., Corney, D., Aiello, L.M.: SNOW 2014 Data Challenge: Assessing the performance of news topic detection methods in social media. In: Proceedings of the SNOW 2014 Data Challenge Workshop co-located with 23rd International World Wide Web Conference (WWW 2014), Seoul, Korea, April 8, pp. 1–8 (2014)
11. Petkos, G., Papadopoulos, S., Kompatsiaris, Y.: Social event detection using multimodal clustering and integrating supervisory signals. In: Proceedings of the 2nd ACM International Conference on Multimedia Retrieval, ICMR 2012, pp. 23:1–23:8. ACM, New York (2012)
12. Philbin, J., Chum, O., Isard, M., Sivic, J., Zisserman, A.: Lost in quantization: Improving particular object retrieval in large scale image databases. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2008), pp. 1–8 (June 2008)
13. Spyromitros-Xioufis, E., Papadopoulos, S., Kompatsiaris, Y., Tsoumakas, G., Vlahavas, I.: A comprehensive study over VLAD and Product Quantization in large-scale image retrieval. IEEE Transactions on Multimedia 16(6), 1713–1728 (2014)
14. Weng, L., Menczer, F., Ahn, Y.: Predicting successful memes using network and community structure. CoRR, abs/1403.6199 (2014)
15. Xie, L., Natsev, A., Kender, J.R., Hill, M., Smith, J.R.: Visual memes in social media: Tracking real-world news in youtube videos. In: Proceedings of the 19th ACM International Conference on Multimedia, MM 2011, pp. 53–62. ACM, New York (2011)