Visual Event Summarization on Social Media using Topic Modelling and Graph-based Ranking Algorithms

Conference Paper · June 2015

DOI: 10.1145/2671188.2749407

4 authors:

Manos Schinas
Information Technologies Institute (ITI)
17 PUBLICATIONS **143** CITATIONS
SEE PROFILE

Symeon Papadopoulos
The Centre for Research and Technology, Hellas
256 PUBLICATIONS **4,720** CITATIONS
SEE PROFILE

Ioannis (Yiannis) Kompatsiaris
The Centre for Research and Technology, Hellas
1,023 PUBLICATIONS **14,035** CITATIONS
SEE PROFILE

Pericles A. Mitkas
Aristotle University of Thessaloniki
320 PUBLICATIONS **3,022** CITATIONS
SEE PROFILE

Visual Event Summarization on Social Media using Topic Modelling and Graph-based Ranking Algorithms

Manos Schinas
Information Technologies Institute, CERTH
57001, Thessaloniki, Greece
manosetro@iti.gr

Symeon Papadopoulos
Information Technologies Institute, CERTH
57001, Thessaloniki, Greece
papadop@iti.gr

Yiannis Kompatsiaris
Information Technologies Institute, CERTH
57001, Thessaloniki, Greece
ikom@iti.gr

Pericles A. Mitkas
ECE Department
Aristotle University of Thessaloniki, Greece
mitkas@eng.auth.gr

ABSTRACT

Due to the increasing popularity of microblogging platforms, the amount of messages (posts) related to public events, especially posts encompassing multimedia content, is steadily increasing. The inclusion of images can convey much more information about the event, compared to their text, which is typically very short (e.g., tweets). Although such messages can be quite informative regarding different aspects of the event, there is a lot of spam and redundancy making it challenging to extract pertinent insights. In this work, we describe a summarization framework that, given a set of social media messages about an event, aims to select a subset of images derived from them, that, at the same time, maximizes the relevance of the selected images and minimizes their redundancy. To this end, we propose a topic modelling technique to capture the relevance of messages to event topics and a graph-based algorithm to produce a diverse ranking of the selected high-relevance images. A user-centred evaluation on a large Twitter dataset around several real-world events demonstrates that the proposed method considerably outperforms a number of state-of-the-art summarization algorithms in terms of result relevance, while at the same time it is also highly competitive in terms of diversity. Namely, we get an improvement of 25% in terms of precision compared to the second best result, and 7% in terms of diversity.

Categories and Subject Descriptors

H.3 [**Information Search and Retrieval**]: Information Storage and Retrieval

Keywords

event summarization, social media, multimedia ranking

ICMR'15, June 23–26, 2015, Shanghai, China.

ACM 978-1-4503-3274-3/15/06 ...$15.00.
http://dx.doi.org/10.1145/2671188.2749407.

1. INTRODUCTION

Due to their increasing popularity, microblogging platforms, and especially Twitter, have evolved into a powerful means for getting connected with large scale public events. In such events, ranging from sports, such as football matches, to political events and festivals, users typically use social media to share their experiences and engage in discussions. Thus, not surprisingly, the amount of event-related messages has reached impressive levels [1]. Also for such type of events, a growing number of microblogging messages carry multimedia content that provides additional insights into the event. Without doubt, the existence of an image in a micro-post can convey a much better impression for the specific moment of the ongoing event compared to the limited textual content of the micro-post.

However, a significant percentage of microblogging messages can be considered as non-informative. This fact combined with the huge number of messages, makes it very challenging for interested users to monitor the evolution of the event and understand its important moments. In case we consider messages with visual content, this becomes even more difficult due to the existence of images that carry little information about the event, e.g., Internet memes and screenshots. In addition, the sharing mechanisms provided by social media, result in considerable amounts of duplication in terms of textual and visual content. In other words, an event-related stream of messages with images is quite diverse and noisy, with different associated topics and conversations among users, and a high degree of redundancy. Thus, there is a profound need for event-based summarization methods that can produce concise visual summaries, covering its main aspects.

To this end, we propose MGraph, a framework that aims to create visual summaries of real-world events by post-hoc analysis of the stream of event-related messages that leverages multiple modalities and signals of the messages. First, we calculate the significance of each message, based on the social attention it receives (i.e. the number of reposts). Then, we apply topic modelling to discover the underlying aspects of the event and assign messages to the detected topics. Next, we calculate the relevance of the message to the topic it belongs to. Finally, we use DivRank, a graph-based ranking algorithm, to obtain a set of relevant and signifi-

cant messages that at the same time maximize the coverage of the event by selecting the maximum possible number of topics and minimize redundancy across selected messages.

The proposed approach captures multiple aspects of the summarization problem in a single framework. Through the multi-graph representation, it captures different notions of similarity (textual, visual, temporal, social), while the use of sophisticated graph-based methods, Clique Percolation for near-duplicate removal [17], SCAN [22] for topic detection, and DivRank [15] for diversity-oriented ranking, enables the extraction of high-quality visual summaries of events. To demonstrate the effectiveness of the proposed approach, we present a comprehensive evaluation on a reference dataset [13] and demonstrate that it leads to superior summarization performance in terms of precision and diversity compared to a number of state-of-the-art methods.

2. RELATED WORK

2.1 Text-based Event Summarization

A substantial body of work exists in literature on the problem of textual summarization of microblogs, which is a special case of the multi-document summarization (MDS) problem. One of the first MDS approaches relies on the computation of centroids, based on textual content. Namely, the summary of a set of documents, represented by $tf \cdot idf$ vectors, consists of those documents that are closest to the centroid of the set [18]. Graph-based approaches have also been proposed to detect salient sentences from multiple documents, with LexRank [8] being the most notable among them. First, a graph of sentences is constructed, with the textual similarity between two sentences serving as the connection between them. Then, the saliency of each sentence is calculated using some centrality measure, such as the Eigenvector Centrality or the PageRank algorithm.

However, the text brevity, the existence of noisy documents, and the diversity of the underlying topics in a set of microblog documents make the summarization problem much more challenging compared to the traditional MDS. In addition, the temporal dimension, that arises from the timestamped micro-posts and the social interaction between users in these platforms, are totally ignored by the aforementioned methods. To this end, a lot of methods have been proposed in the literature, that incorporate not only the textual information of the documents, but also their temporal dimension and their social features. The core idea of the majority of previous works is the segmentation of documents set into coherent topics or sub-events and the selection of the most "representative" documents in each segment.

Nichols et al. [16] describe an algorithm that generates a summary of sports events. They use a peak detection algorithm to detect important moments in the timeline of tweets and then apply a graph-based method to extract sentences from the tweets around these moments. In [6], the authors propose a probabilistic model for topic detection in Twitter that handles the short length of tweets and considers time as well. Instead of relying only on the co-occurrences of words (as the majority of traditional probabilistic text models do), the proposed model uses the temporal correlation of Twitter content to make denser the co-occurrences of terms. For each detected topic, a set of tweets with the highest correlation with respect to the topic word distribution is considered as representative. Shen et al. [20] present a participant-based approach for event summarization. First, the participants of the event are detected and then a mixture model is applied to detect sub-events at participant level. Finally, the $tf \cdot idf$ centroid approach is used to select a tweet for each detected sub-event. Similarly, Chakrabarti and Punera [4] propose the use of a Hidden Markov Model to obtain a time-based segmentation of the stream that captures the underlying sub-events.

Recent works have focused on the creation of visualizations that summarize the key concepts of events, as presented through social media. TwitInfo ([12]) is a system for summarizing events on Twitter by using a timeline-based display that highlights peaks of high activity. Alonso and Shiells [2] create timelines for football games, annotated with the key aspects of the event, in the form of popular tags and keywords. Dork et al. [7] propose an interface for large events that employs several visualizations, e.g., image and tag clouds, for interactive presentation of the event. However, these methods use only textual and social features to create visualizations and ignore the visual content of the embedded multimedia items.

2.2 Multimedia Event Summarization

Taking into account the increasing use of multimedia content in microblog platforms, there have been many studies that consider visual information along with the textual content of microblog messages. Bian et al. [3] proposed a multimodal extension of LDA that detects topics by capturing the correlations between textual and visual features of microblogs with embedded images. The output of this method is a set of representative images that describe, in a visual way, the underlying event. A slightly different problem is tackled by Lin et al. [10]. Unlike other methods that generate summaries as sets of messages or images, that method aims to create a storyline from a set of event-related multimedia objects. A multi-view graph of objects is constructed, where two types of edges capture the content similarity, visual and textual, along with the temporal proximity among objects. Then a time-ordered sequence of important objects is obtained via graph optimization.

The authors of [14] propose a method to select and rank a diverse set of images with high degree of relevance to the event. An interesting part of their work, is the use of external websites as sources of multimedia content, when the amount of embedded images is insufficient for the creation of a meaningful visual summary. They use visual features first to discard irrelevant images and images of low quality, and then to detect near duplicates among them to increase diversity. Then, they apply several ranking methods to select a small number of images that describe the event.

To our knowledge, there is a lot of space for improvements on the problem of multimedia summarization, as most of the related methods are mainly based in the textual and temporal information and ignore the richness of visual and social signals in social media. To this end, our proposed framework incorporates textual, visual, temporal and social features to support the generation of visual summaries from event-focused social media content. The proposed framework extends our previous work, namely the StreamGrid framework [19], improving it in a number of ways: a) not requiring explicit temporal intervals to be defined, b) more careful handling of visual duplicates, c) using SCAN [22] instead of LDA for topic detection, d) incorporating social

interactions (mentions) in the message graph construction, e) using a more effective significance score, f) using DivRank [15] for diversity-oriented ranking.

3. APPROACH DESCRIPTION

3.1 Overview

Our intention is to use an event-related set of social media items, to create a visual summary that describes the main moments of the event. As visual summary we define a set of images that are highly relevant to the event and contain visually, the key aspects of the event. As a first step we apply a set of filters in the social media items to keep only the informative ones among them. Then, we create a multi-graph that captures the similarity of items across different modalities. Using this graph, we first detect and remove visual duplicates and we then apply topic modelling to detect the main topics of the events. Based on the detected topic models, we calculate a selection score for each message that captures the social attention that a message receives over time and the coverage of the corresponding topic. Finally, we use a graph-based ranking algorithm to diversify the top ranked social media items. An overview of the proposed method is depicted in Figure 3. Note that, although the goal is to select a subset of images to form a visual summary, the proposed framework makes use of all the available social media items, even those that do not have any associated multimedia content.

3.2 Representation of Social Media Items

We represent each message m as a tuple $\{id, t_s, C, E, u, p\}$, where id is a unique identifier of the message, t_s its publication time, C the content, E is the set of detected named entities and mentions contained in this message, u an identifier of the posting user, and p the number of times that this message has been reposted. Content C consists of two parts: textual and visual. The textual part of the content (C_{text}) is represented as a $tf \cdot idf$ vector v_m, where the tf part is the frequency of a term in the message normalized by the maximum frequency in the message. Due to the short length of the documents in microblogging platforms, this component often equals to one. The inverse document frequency (idf) of each term is calculated over the whole set of messages. Note that we extend $tf \cdot idf$ by using a constant boosting factor b to give more weight to terms that are expected to be particularly relevant for the sub-event, i.e. named entities and mentions. In other words, if a term w is a recognized named entity, its weight is given by $b \cdot tf_w \cdot idf_w$. The intuition is that two messages that share the same set of named entities or mention the same user, have a higher probability of belonging to the same topic. The visual part (C_{visual}) is optional, as not all items are associated with multimedia. In case they are, we represent them using the combination of Speeded Up Robust Features (SURF) with the VLAD scheme as implemented in [21].

3.3 Aggressive filtering

Content quality plays a key role in the generation of informative, but concise summaries. To this end, we first apply a set of heuristic rules to discard a significant amount of the initial set of event-related messages that are considered noisy and of low quality. More specifically, we apply two types of filters on the messages. The first is based on the textual content and is applied on items that do not contain any embedded image. The second one, is based on visual features and is applied only to messages with embedded images.

Regarding text-based filtering, we discard a message if it has very short text (e.g., less than six terms) and mentions more than three users, or contains more than three URLs or hashtags. The core idea behind the aforementioned filtering rules is that messages of that type do not carry enough textual content to be usable in a summary. Also the co-existence of a URL with many popular hashtags or mentions, is a strong indication that the corresponding message is spam that aims to redirect the user to the website pointed by the URL. Also, in order to discard messages that have an incorrect or incomplete syntactic structure, we apply Part-Of-Speech tagging and keep only messages that match the regular expression of Equation 1. Namely, we keep only items that contain at least one sentence that consists of at least one noun followed by one verb. Determiners and adjectives are optional. Finally we keep only original messages and discard all the reposts. However, for each original message we keep the number of times it has been reposted by other users, and we use it as a signal of the social attention it receives over time.

$$\texttt{regex} = (\texttt{determiner}?\ \texttt{adjective}^*\ \texttt{noun}+\ \texttt{verb})+ \quad (1)$$

Regarding visual filtering, first we discard small images, i.e. images having width or height less than 200px. To discard memes, screenshots and images having heavy text we use the semi-supervised method presented in [11] to build a model that detects these types. Typically, using a set of labelled and unlabelled images, represented as normalized VLAD vectors, we consider a similarity graph and we construct the Approximate Laplacian Eigenmaps (ALE) of this graph. More precisely, each ALE vector is a low dimensional representation of an image that captures in a compact way the position of the image in the manifold of the similarity graph. Intuitively, images of the same type will share the same neighbours and subsequently will have similar ALE representations. Finally, we use the set of labelled images and their ALEs to train an SVM classifier that classifies images into four types: memes, screenshots, images with overlay text, and real photos.

3.4 Multigraph generation

Given a set of message $M = \{m_1, m_2, ..., m_n\}$ we construct a multi-graph $\mathcal{G}_M = \{\mathcal{V}, \mathcal{E}_{textual}, \mathcal{E}_{visual}, \mathcal{E}_{social}, \mathcal{E}_{time}\}$, where vertex $v_i \in \mathcal{V}$ corresponds to message m_i. $\mathcal{E}_{textual}$ is a set of undirected edges expressing the textual similarity between nodes. For textual similarity we used the well known cosine similarity between the corresponding $tf \cdot idf$ vectors. $\mathcal{E}_{visual}$ is a set of undirected edges that represent the visual similarity between messages with images. Visual similarity is based on the L_2 distance between the corresponding SURF+VLAD vectors. Note that we add an edge in $\mathcal{E}_{textual}$ or $\mathcal{E}_{visual}$, only if the textual or visual similarity between the corresponding nodes is higher than $th_{textual}$ or th_{visual} respectively. Introduction of textual and visual thresholds at this step aims to prune the graph, make it more sparse, and avoid the addition of "noisy" associations between nodes. The directed unweighted edges of $\mathcal{E}_{social}$ are based on the social interactions between users: we connect two messages m_i and m_j, with a directed edge from m_j to m_i, if message m_j is a direct reply to m_i. Finally, the

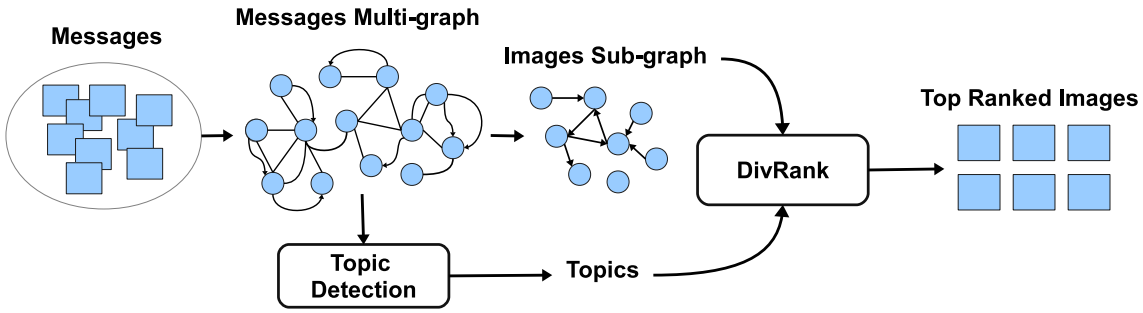

Figure 1: Overview of the proposed approach.

directed edges $\mathcal{E}_{time}$ are based on the temporal proximity between messages. Temporal proximity (TS) between two messages m_i, m_j, published with difference $\Delta t = |t_i - t_j|$, is modelled by using the Gaussian kernel function of Equation 2. Parameter σ controls the spread of the sub-events within the main event. In general, the optimal value of σ depends on the type of the event, because sub-events are wider in events of certain types, thus requiring a higher value for σ and vice versa. The direction of an edge is from m_j to m_i, meaning that message m_j is posted after m_i.

$$TS(\Delta t) = \exp(\frac{-\Delta t^2}{2\sigma^2}) \tag{2}$$

The multi-graph generation step requires the calculation of visual, textual and temporal similarity between all pairs of messages. The complexity of this step is $O(n^2)$, which would make it inapplicable to very big events. To reduce the complexity, for each message we efficiently retrieve its k nearest messages in terms of time, textual and visual content and calculate contextual similarities only to the union of these three sets. Temporal similarity retrieval is speeded up using range queries on a B-Tree index. For text, we used an efficient retrieval scheme based on Locality Sensitive Hashing [5]. For visual-based retrieval we used Product Quantization, an indexing scheme for visual features described in [9].

3.5 Visual deduplication

As mentioned before, we handle de-duplication of messages by keeping only original messages and discarding explicit reposts. However, there are duplicates for which there is no explicit connection. This is more obvious in case of visual content, as users can post the same image or near duplicates found in different sources, e.g., different news websites. To handle this high degree of visual redundancy we use the Clique Percolation Method, presented in [17], to find sets of messages that are visual duplicates. In particular, we use CPM on sub-graph $\mathcal{G}_{visual} = \{\mathcal{V}, \mathcal{E}_{visual}, \}$ to generate cliques of visual duplicates. We represent message cliques in a similar manner as single messages. More specifically, message clique mc is a tuple $\{id, M_{mc}, t_s, C, E, p\}$, where id is a unique identifier of the clique, M_{mc} is the messages of the clique, t_s is the mean value of publication time of the messages in M_{mc}, and p is the aggregated value of reposts of each message. Regarding the textual part of the content we use a merged $tf \cdot idf$ vector V_{mc} (Equation 3).

$$V_{mc} = \sum_{m \in M_{mc}} v_m \tag{3}$$

To create a single visual representation for clique mc, we use the SURF descriptors of all images in the clique and aggregate them in a single VLAD vector. In this way, we take into account small variations between images (e.g., cropping, rotation, brightness adjustment, etc.). After the detection of message cliques, we replace clustered messages in $\mathcal{G}_M$ with cliques and re-calculate the corresponding edges of $\mathcal{E}_{textual}$, $\mathcal{E}_{visual}$, $\mathcal{E}_{social}$ and $\mathcal{E}_{time}$.

3.6 Topic Detection

To detect the topics on a main event we opted for a graph clustering algorithm, namely the Structural Clustering Algorithm for Networks (SCAN) [22]. SCAN is applied on a graph $G = \{\mathcal{V}, \mathcal{E}\}$, where nodes correspond to the filtered set of event-related messages and message cliques and edges $\mathcal{E}$ represent the content-based similarities between adjacent nodes. Apart from content similarity, we also use social interactions to add edges that enhance the density of inter-topic links. Namely, we connect two messages if the one is a reply to the other, as the probability that these messages belong to the same topic is very high. We apply SCAN on the message graph, to identify dense sub-graphs of messages. These sub-graphs represent the topics that exist in the stream of messages. Hence, each topic is represented as a set of highly connected messages in the graph. Once topics T are detected, we use the messages M_i associated with each $topic_i \in T$ to calculate a merged $tf \cdot idf$ vector V_i that describe its content, in a similar manner to how we calculate merged vectors for cliques.

However, a substantial amount of messages is kept outside of the detected clusters. These messages are divided into two categories, hubs and outliers. Hubs are bridges to more than one clusters, while outliers are messages that are not related to any of the clusters. Some of these messages can be considered as non-informative messages that cannot contribute valuable information to the summary. However this is not the case for all of them, as some messages, despite not belonging to any cluster, may include valuable information that attracts a lot of social attention. This is more obvious in case of messages with images. Such messages could have little textual information, therefore very low textual similarity to the other messages. Moreover, visual content could be different, even between messages of the same topic. Therefore, it is likely that important images could be left unclustered. To this end, we do not discard the unassigned messages, but we form single item clusters and use them in the ranking process, as will be described in the next section.

3.7 Message Selection and Ranking

Our goal is to calculate an overall importance score for each of the messages or message cliques of the filtered set, and rank them according to it. The importance score of a message m or clique mc is a combination of two factors: a) the social attention it receives over time, and b) the significance of the topic it may belong to.

Social Attention. The popularity of a message or clique, i.e. the number of the reposts is receives over time, can be considered as a measure of the social attention it receives. A high value of social attention, indicates implicitly an important and hence representative message regarding the event. We measure social attention using Equation 4, where p is the number of reposts and λ a smoothing parameter. We opted for the use of a logarithmic function due to the fact that the number of reposts in social media follows a power law distribution.

$$S_{att}(m) = \log_2(p + \lambda) \tag{4}$$

Topic Coverage. The association of a message with a detected topic is a strong indication of its importance. Namely, a message that is part of a topic contributes valuable information about an aspect of the event and should get a high importance score. However some messages of a topic are more representative than others. Also some topics are more significant than others, hence messages from these topics should receive higher scores. To this end, we quantify the topic coverage of a message using Equation 6. Its first part captures the relevance to the topic and is calculated as the textual similarity of m to the topic centroid V_i. Its second part captures the significance S of the underlying topic, so that messages from largest clusters get higher scores.

$$S(topic_i) = \exp(\frac{|M_i|}{\max_{k \in T}|M_k|}) \tag{5}$$

$$S_{cov}(m) = \cos(u_m, V_i) \cdot S(topic_i) \tag{6}$$

The overall significance score of a message or clique is the product between its social attention and the respective topic relevance (Equation 7).

$$S_{sig}(m) = S_{att}(m) * S_{cov}(m) \tag{7}$$

3.8 Image Ranking and Diversification

The motivation behind computing the importance score of Equation 7 is to generate a diverse set of images in the top ranked positions of the summary. However, there are images that are considered relevant to an event and extremely popular, but they are not specific to the event of interest. For example, an image depicting the flag of Ukraine could be considered to be relevant for an event about the Ukraine crisis, but it does not provide important information about the event. To tackle this, we introduce a specificity factor that penalizes such images. Image specificity is a measure of how much information a specific message provides for a specific event. In other words, whether the message is common or rare across all topics of the event. In a similar manner as [14], we use the deduplication technique presented in section 3.6 to measure the number of topics $|T_I|$ that contain an image I. Then we calculate an *idf*-like score for each image using Equation 8.

$$S_{spec}(I) = \log(\frac{|T|}{|T_I|}), \tag{8}$$

where $|T|$ is the number of topics in the event and $|T_I|$ is the number of topics containing image I. Finally, the image selection score $S(I)$ of image I is the product of the importance score (Equation 7) and the image specificity score.

To incorporate diversity into the score calculation, we employ DivRank [15], a variant of PageRank that aims at diversity. We use the multi-graph $\mathcal{G}_M$ that was created initially, to get a directed sub-graph $\mathcal{G}_V = \{\mathcal{V}_V, \mathcal{E}_V\}$. Vertices $\mathcal{V}_V \subset \mathcal{V}$ are the subset of messages that contain an embedded image and will be used in the generation of a visual summary. For the creation of the set $\mathcal{E}_V$, we combine the two sets $\mathcal{E}_{visual}$ and $\mathcal{E}_{time}$. In particular, for each pair of vertices $v_i, v_j \in \mathcal{V}_V$, we create a weighted directed edge $e \in \mathcal{E}_V$ with the same direction as the corresponding edge in $\mathcal{E}_{time}$. The weight of this edges is the product of visual similarity and time proximity between the adjacent vertices. To ensure convergence of DivRank, we normalize the weights of the edges, such that the sum of the adjacent out-edges of each message equals to one.

To calculate the new selection score, we apply DivRank using the iterative scheme of Equations 9 and 10.

$$\mathbf{r} = dW^{-1}\mathbf{r} + (1-d)\mathbf{h} \tag{9}$$

$$W = dW\mathbf{r} + (1-d)\mathbf{h} \tag{10}$$

Vector $\mathbf{r}$ holds the DivRank scores and d is a dumping factor that controls the impact of the initial score to the re-ranking procedure. The initial value of matrix W is the adjacency matrix derived from the directed graph $\mathcal{G}_V$. Also, instead of using a uniform value for priors $\mathbf{h}$, we use the value of the calculated score of each image in the graph. Specifically, the prior $\mathbf{h}[i]$ of the i^{th} node in the graph that corresponds to image I_i is $\mathbf{h}[i] = S(I_i)$.

4. EXPERIMENTS AND RESULTS

4.1 Dataset and experimental setting

To evaluate the proposed framework, we conducted a set of experiments in the dataset of McMinn et al. [13] that contains more than 500 events of different domains. We used the 50 largest events in terms of tweets, as in the work of McParlane et al. [14]. These events range from sports events, e.g., the Sochi winter Olympics, to political events such as the Ukraine crisis and Venezuelan protests. The dataset contains 364,005 tweets in total, while each event is associated with 4730 tweets on average. However, due to suspended accounts and deleted messages we managed to get only 296,160 of these tweets. About 3,51% of these, i.e. 12,772 tweets, contain an embedded image.

In [14], the authors used CrowdFlower[1] to create relevance judgements for the top five images selected for summarization for each of the 50 events. This resulted in the generation of judgements for a very small percentage of the images in the dataset. To this end, we follow the same approach as [14] to create relevance judgements for the union

[1] `http://www.crowdflower.com/`

of images selected as summaries by all the methods used in the evaluation. In order to have a better insight into the performance of the methods we selected 20 images for each event. We asked from a group of human annotators to evaluate how relevant and representative are the selected images to the corresponding event. We ensured that each pair received three judgements at least, from different users. The group of annotators comprised 20 persons 24-32 years old, educated in the field of computer science, having some experience in the use of Twitter and social media. The task given to annotators was the following:

Task Description

You are presented with an image and an event title (describing a "trending" topic in Twitter). For each image and event title, you are asked to answer the following question:

Question: Is this image relevant to the event?

Possible Answers:

1. *The image is clearly not relevant to the event.*
2. *The image is probably not relevant to the event, but I am not entirely sure.*
3. *The image is somewhat relevant to the event, but I have my doubts on whether I would like to see it in a photo coverage of the event.*
4. *The image is clearly relevant to the event, and I would like to see it in a photo coverage of the event.*

For the text representation, we used several open source projects to analyse the text of the tweets. For tokenization we opted for the `StandardAnalyser` provided by Lucene, which performs well in English text. For named entity detection we used the Stanford NER library with the default 3-class model. For part of speech tagging we used the Stanford POS Tagger, but we opted for the Twitter-specific POS model from the ARK research group[2]. For visual features, we extracted Speeded Up Robust Features (SURF) from each image of the dataset. Then we used four codebooks of 128 visual words (in total 512) to quantize each descriptor and used the VLAD scheme to aggregate the descriptors of each image into a single vector of $64 \cdot 512 = 32,768$ dimensions. Finally, we used PCA to create a 1024-dimensional $L2$-normalized reduced vector that represents the visual content of the image.

For the generation of multi-graph $\mathcal{G}_M$, we retrieve the $k = 500$ nearest neighbours of each message in terms of textual, visual and temporal similarity. The visual and textual similarity thresholds were empirically set to 0.5 and 0.6 respectively. Parameter σ^2 of the temporal kernel was empirically set to 24 hours as most of the important sub-events in the dataset last less than a day. In other words, the temporal proximity between tweets in the same day is more than 0.6. In the topic detection step, we set the parameters of SCAN to $\mu = 2$ and $\epsilon = 0.65$. Finally, in the ranking step with DivRank we set $d = 0.75$ to the most of the experiments. However, we also conducted an experiment to investigate the effect of this factor in the results.

[2] `http://www.ark.cs.cmu.edu/TweetNLP`

4.2 Evaluation metrics and baselines

We applied the proposed method (denoted as **MGraph**) to the dataset tweets to generate a representative summary for each of the contained events. In particular, we ranked the images according to their DivRank score and kept the top N as the summary. We evaluated the average performance of our method in a similar manner as [14] by calculating the following metrics:

- **Precision (P@N)**: The percentage of images among the top N that are relevant to the corresponding event, averaged among all events. We calculate precision for N equal to 1, 5, and 10.
- **Success (S@N)**: The percentage of events, where there exist at least one relevant image amongst the top N returned, for N=10.
- **Mean Reciprocal Rank (MRR)**: Computed as $1/r$, where r is the rank of the first relevant image returned, averaged over all events.
- **α-normalized Discounted Cumulative Gain**: α-nDCG@N measures the usefulness, or gain, of the returned images based on their position in the summary (N=10).
- **Average Visual Similarity**: AVS@N measures the average visual similarity among all pairs of images in the top N selected images, averaged over all events. Lower AVS values are preferable since they imply higher diversity in terms of visual content.

We compare the proposed **MGraph** scheme with several methods for image ranking. Note that we applied the same filtering and deduplication steps to all methods. More specifically, we evaluated the following summarization methods:

- **Random:** randomly selects N images from the (filtered) set of images as the summary set.
- **MostPopular:** picks up the most popular images in terms of reposts. This corresponds with ranking based on the score of Equation 4.
- **LexRank:** uses the graph $G = \{\mathcal{V}, \mathcal{E}\}$ of Section 3.6 and ranks the nodes using the LexRank algorithm [8]. Then, it selects the top N nodes that contain images.
- **TopicBased:** selects the most relevant messages from the most significant topics according to the score of Equation 6.
- **P-TWR:** ranks images in descending order using the weighting scheme described in [14].
- **S-TWR:** groups the tweets of each event into subclusters and select the highest ranked tweet of each cluster using the weighting scheme of [14].

4.3 Results

Table 1 contains several precision-oriented metrics for both `MGraph` and the competing methods. Not surprisingly, the worst results for all the metrics are those of `Random` selection. Regarding P@N the best results were achieved from `MGraph`. For P@1, popularity-based methods, such as `MostPopular` and `P-TWR`, achieved very good results as would

Table 1: Comparison of summarization methods in terms of precision. Bold values indicate the highest performing method for the given metric.

Method	P@1	P@5	P@10	S@10	MRR
Random	0.391	0.400	0.405	0.800	0.562
MostPop	0.522	0.469	0.446	0.848	0.669
LexRank	0.456	0.452	0.420	0.847	0.611
TopicBased	0.457	0.473	0.469	0.847	0.620
P-TWR	0.521	0.486	0.437	0.826	0.673
S-TWR	0.478	0.452	0.435	0.869	0.661
MGraph	**0.587**	**0.518**	**0.544**	**0.913**	**0.728**

be expected. This means that the image having the highest value of popularity, has a higher possibility of being relevant to the event. However, the performance of these two methods drops significantly for P@5 and P@10. This is explained by the fact that although some image might be considered as irrelevant or marginally relevant, it could still attract the attention of OSN users for a number of other reasons (e.g., it could be funny), and would therefore be highly ranked by popularity-based methods. Success for the top 10 retrieved images is high for all methods, even for the `Random` one. However, even in this case our method achieves a better value of S@10. The average mean reciprocal rank (MRR) is also higher for our method, with the popularity-based method achieving the next best results. Note that the average performance for this metric for the popularity based methods is benefiting from the cases that the most popular image is relevant. This mainly occurs when the number of reposts of an image gets extremely high values, e.g., hundreds or thousands of reposts. However, in events that there are no such images the performance drops significantly. In contrast our method handles successfully such cases, as it does not solely rely on the popularity of images, but also considers their association with the underlying topics.

Table 2 presents a comparison among methods in terms of their diversity performance. According to it, `MGraph` achieves the best value of α-nDCG@10, with `S-TWR` having the second best performance. This indicates that the use of the DivRank algorithm resulted in a more diverse set of relevant images compared to the other methods. Compared to the `S-TWR` method that aims to achieve diversity by using clustering of images, our method achieves an α-nDCG score that is improved by a factor of 7%. It is noteworthy that this improvement is achieved without sacrificing precision, as P@10 compared to `S-TWR` is also improved by 25%. In case of average visual similarity between images the best result is obtained by `S-TWR`. Our method has somewhat worse performance in terms of AVS@5, where it is ranked second, while for AVS@10, it is ranked third. The worst results in terms of AVS are obtained using `LexRank`. This is reasonable as `LexRank` is based on the PageRank algorithm, and hence it favours images that are highly connected, i.e. images that are highly similar in terms of visual content. One should be cautious regarding the interpretation of AVS: although it is a reasonable measure of diversity, it is solely based on the use of visual features, hence it might not be able to capture the users' perception. In addition, it is expected that the inclusion of irrelevant images in the set of selected, would result in lower values for AVS, but this is obviously not desirable.

Table 2: Comparison of summarization methods in terms of diversity. Bold values indicate the highest performing method for the given metric.

Method	α-nDCG@10	AVS@5	AVS@10
Random	0.657	0.024	0.019
MostPop	0.717	0.022	0.018
LexRank	0.685	0.081	0.056
TopicBased	0.689	0.035	0.027
P-TWR	0.717	0.020	0.016
S-TWR	0.722	**0.011**	**0.010**
MGraph	**0.774**	0.018	0.021

Table 3: Performance of MGraph across different event categories.

Category	P@10	α-nDCG	AVS@10
Law & Politics	0.536	0.729	0.047
Arts & Entertainment	0.700	0.721	0.048
Science & Technology	0.800	0.896	0.059
Disasters & Accidents	0.450	0.492	0.013
Sports	0.500	0.624	0.025
Miscellaneous	0.368	0.606	0.053

The events in the test dataset belong to six categories, as shown in Table 3. Each of these categories has different characteristics, thus the performance of our method differs among them. For example, the Arts & Entertainments category is more prone to duplicate messages and images, e.g., tweets with images of celebrities shared by users. The best P@10 measure is obtained for events about Science & Technology, but this should be taken with caution, as this category contains very few events. The second best P@10 is obtained for events about Arts & Entertainment. This can be explained by the fact that these events refer mostly to celebrities and the corresponding images usually depict them in a manner that is relevant to the event. Regarding average visual similarity, the best value is achieved for events about disasters & accidents. This is easily explained, taking into account that images of this type, e.g., earthquakes, can be very different in terms of their visual information even in cases they refer to the same event.

Finally, we study how parameter d of DivRank affects the precision and diversity of MGraph, using different values of d, from 0 to 1, and calculating P@10, S@5, MRR and α-nDCG@10 for each of them. The results are depicted in Figure 2. The worst results for all metrics are obtained for $d = 0$. Essentially, in this marginal case, the re-ranking procedure of DivRank is not performed as the first part of Equations 9 and 10 is equal to zero. The best results are achieved for $0.7 \leq d \leq 0.8$, but even for $d > 0.8$ the performance remains almost steady for most of the metrics. The slight decrease for $d > 0.8$ can be explained by the fact that for such extreme values of d, DivRank attempts to create a more diverse set of images, thus it is more likely to introduce some irrelevant images in the top ranks of the result set.

5. CONCLUSIONS AND FUTURE WORK

We presented MGraph, a framework for the generation of visual summaries for real world events using messages from social media. To achieve this goal we proposed a method that assigns a significance score on each image of the event-

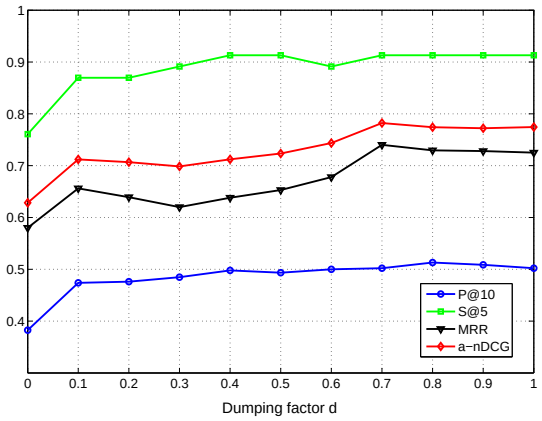

Figure 2: Effect of the dumping factor d on P@10, S@5, MRR and α-nDCG@10.

related set of messages, that maximizes the coverage of the underlying topics and the diversity at the same time.

In future work, we plan to extend the proposed method by using more advanced topic modelling techniques that identify not only topics but also hierarchies and relations between them. Regarding ranking we plan to investigate the use of co-ranking algorithms to rank mutually text and image nodes in a more principled way. Finally, we intend to integrate additional features such as users' popularity, influence and trustworthiness, as recent research indicates that these could improve the results, and especially the quality of the selected images.

6. ACKNOWLEDGMENTS

This work was supported by the SocialSensor and REVEAL projects, partially funded by the European Commission under contract numbers 287975 and 610928.

7. REFERENCES

[1] Celebrating #SB48 on Twitter. `https://blog.twitter.com/2014/celebrating-sb48-on-twitter`, 2014. [Online; accessed 27-Feb-2014].

[2] O. Alonso and K. Shiells. Timelines as summaries of popular scheduled events. In *Proceedings of the 22nd International Conference on World Wide Web (WWW) companion*, pages 1037–1044. International World Wide Web Conferences Steering Committee, 2013.

[3] J. Bian, Y. Yang, and T.-S. Chua. Multimedia summarization for trending topics in microblogs. In *Proceedings of the 22nd ACM International Conference on Information and Knowledge Management*, CIKM '13, pages 1807–1812, New York, NY, USA, 2013. ACM.

[4] D. Chakrabarti and K. Punera. Event summarization using tweets. In *Proceedings of 6th AAAI International Conference on Weblogs and Social Media (ICWSM)*, 2011.

[5] M. S. Charikar. Similarity estimation techniques from rounding algorithms. In *Proceedings of the 34th Annual ACM Symposium on Theory of Computing*, STOC '02, pages 380–388, New York, NY, USA, 2002. ACM.

[6] F. C. T. Chua and S. Asur. Automatic summarization of events from social media. In *Proceedings of 8th AAAI International Conference on Weblogs and Social Media (ICWSM)*, 2013.

[7] M. Dork, D. Gruen, C. Williamson, and S. Carpendale. A visual backchannel for large-scale events. *IEEE Transactions on Visualization and Computer Graphics*, 16(6):1129–1138, 2010.

[8] G. Erkan and D. R. Radev. Lexrank: Graph-based lexical centrality as salience in text summarization. *Journal of Artificial Intelligence Research*, 22(1):457–479, Dec. 2004.

[9] H. Jegou, M. Douze, and C. Schmid. Product quantization for nearest neighbor search. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(1):117–128, 2011.

[10] C. Lin, C. Lin, J. Li, D. Wang, Y. Chen, and T. Li. Generating event storylines from microblogs. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*, CIKM '12, pages 175–184, New York, NY, USA, 2012. ACM.

[11] E. Mantziou, S. Papadopoulos, and Y. Kompatsiaris. Large-scale semi-supervised learning by Approximate Laplacian Eigenmaps, VLAD and pyramids. In *14th Int. Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS)*, pages 1–4, 2013.

[12] 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 on Human Factors in Computing Systems*, CHI '11, pages 227–236, New York, NY, USA, 2011. ACM.

[13] A. J. McMinn, Y. Moshfeghi, and J. M. Jose. Building a large-scale corpus for evaluating event detection on twitter. In *Proceedings of the 22nd ACM International Conference on Information and Knowledge management*, pages 409–418. ACM, 2013.

[14] P. J. McParlane, A. J. McMinn, and J. M. Jose. Picture the scene...: Visually summarising social media events. In *Proceedings of the 23rd ACM International Conference on Information and Knowledge Management*, pages 1459–1468. ACM, 2014.

[15] Q. Mei, J. Guo, and D. Radev. Divrank: The interplay of prestige and diversity in information networks. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '10, pages 1009–1018, New York, NY, USA, 2010. ACM.

[16] J. Nichols, J. Mahmud, and C. Drews. Summarizing sporting events using twitter. In *Proceedings of the 2012 ACM International Conference on Intelligent User Interfaces*, IUI '12, pages 189–198, New York, NY, USA, 2012. ACM.

[17] G. Palla, I. Derényi, I. Farkas, and T. Vicsek. Uncovering the overlapping community structure of complex networks in nature and society. *Nature*, 435(7043):814–818, 2005.

[18] D. R. Radev, H. Jing, M. Styś, and D. Tam. Centroid-based summarization of multiple documents. *Information Processing and Management*, 40(6):919–938, Nov. 2004.

[19] M. Schinas, S. Papadopoulos, Y. Kompatsiaris, and P. Mitkas. StreamGrid: Summarization of large scale events using topic modelling and temporal analysis. In *Proceedings of the 1st International ICMR Workshop on Social Multimedia and Storytelling, Glasgow, UK*, 2014.

[20] C. Shen, F. Liu, F. Weng, and T. Li. A participant-based approach for event summarization using twitter streams. In *Proceedings of NAACL-HLT*, pages 1152–1162, 2013.

[21] E. Spyromitros-Xioufis, S. Papadopoulos, K. I., G. Tsoumakas, and I. Vlahavas. A comprehensive study over vlad and product quantization in large-scale image retrieval. *IEEE Transactions on Multimedia*, 16(6):1713–1728, 2014.

[22] X. Xu, N. Yuruk, Z. Feng, and T. A. J. Schweiger. Scan: A structural clustering algorithm for networks. In *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '07, pages 824–833, New York, NY, USA, 2007. ACM.