TweetMogaz v2: Identifying News Stories in Social Media

Eslam Elsawy[1], Moamen Mokhtar[1], Walid Magdy[2]

[1]BadrIT inc., Alexandria, Egypt
{eslam.ashraf, moamen.mokhtar}@badrit.com
[2]Qatar Computing Research Institute, Qatar Foundation, Doha, Qatar
wmagdy@qf.org.qa

ABSTRACT

TweetMogaz is a news portal platform that generates news reports from social media content. It uses an adaptive information filtering technique for tracking tweets relevant to news topics, such as politics and sports in some regions. Relevant tweets for each topic are used to generate a comprehensive report about public reaction toward events happening. Showing a news report about an entire topic may be suboptimal for some users, since users prefer story-oriented presentation. In this demonstration, we present a technique for identifying stories within a stream of microblogs on a given topic. Detected tweets on a news story are used to generate a dynamic pseudo-article that gets its content updated in real-time based on trends on Twitter. Pseudo-article consists of a title, front-page image, set of tweets on the story, and links to external news articles. The platform is running live and tracks news on hot topics including Egyptian politics, Syrian conflict, and international sports.

Keywords

TweetMogaz, Twitter, Story detection, Clustering, Arabic

1. INTRODUCTION

Interest in monitoring news from social media has increased in recent years. For example, Twitter was shown to be one of the fastest methods for spreading news [3]. Different methods and applications were developed for extracting useful information about events happening and discussions on news from social media [4, 5, 6, 8, 9]. TweetMogaz[1] [6] is one of these platforms that offers users a news portal entirely generated from social media to follow ongoing news with no bias to a given opinion. The platform uses an adaptive method [8] for tracking tweets relevant to events happening in hot regions such as Syria and Egypt, and popular domains such as sports. The platform generates a comprehensive report on the top shared content on Twitter related to a given topic including most popular tweets, images, videos, and links.

In this demonstration, we present the second version of TweetMogaz [6], which applies a story identification approach for detecting individual stories discussed on Twitter within each topic to be displayed to users as a set of *pseudo-articles* instead of just one comprehensive report. Displaying news in this form allows users to maintain the experience of standard news websites but with the advantage of unbiased and rich social-generated content. A hierarchical clustering algorithm is used to identify hot stories that have enough content on Twitter. Each cluster of tweets is used to generate a real-time pseudo-articles containing most popular tweets as the content, top shared video/image as the article front image, links in tweets as the links to external related articles, and an automatically generated title as news headline. Our story identification algorithm is applied each 15 minutes on TweetMogaz to either detect new stories or update existing ones with emerging new content. We also propose a mechanism to prevent the detection of duplicate stories.

The contributions in this demonstration can be listed as follows: 1) An online real-time story identification service is proposed; 2) A duplicate events detection mechanism is introduced to deliver only distinct events to users; 3) A novel method is presented for displaying tweets relevant to a news story in the form of pseudo-article, which has dynamic content that gets updated automatically over time.

2. RELATED WORK

A related work to our story identification problem is event detection [4, 11]. Two main approaches have been studied for event detection: *document-pivot* and *feature-pivot*. The former cluster documents into clusters, each is related to a specific event and then tries to detect features related to the event from the cluster. The latter extracts high frequency terms in documents stream and groups them to events. We mainly focus on feature-pivot approach, since it fits our task better, where documents are of short length.

Weng et al. proposed a system named EDCoW [11]. The system builds wavelet signals for terms. Then it computes the cross correlation between signals to cluster associated signals together as events. Although this approach showed effectiveness and could be applied for our story identification task, it requires high computational cost, which may be not efficient for a real time application. Li et al. proposed a system named Twevent [4]. The system detects burst phrases based on frequencies. It performs KNN clus-

[1]www.tweetmogaz.com

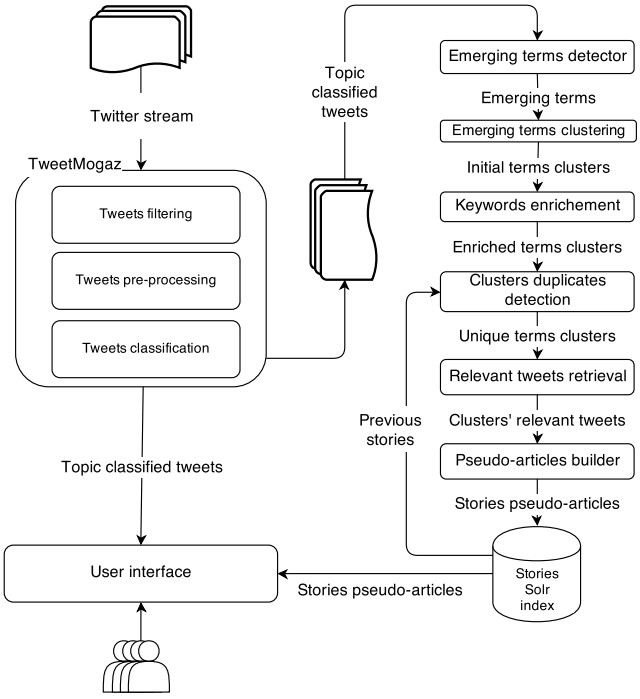

Figure 1: News stories detection architecture

tering to produce disjoint clusters. This approach may not fit our application, where fuzzy clustering is required since terms in different clusters can overlap. Li et.al. [5] proposed an event detection system dedicated for Crime and Disaster related Events (CDE). They crawled Twitter by searching for specific keywords describing crimes and disasters. Then tweets are classified to CDE and non-CDE. Their clustering algorithm depended mainly on spatial and temporal features of tweets.

3. SYSTEM ARCHITECTURE

The full system architecture of TweetMogaz v2 is shown in Figure 1. The left part of the architecture shows the main components of TweetMogaz v1, which is responsible for retrieving relevant tweets to the tracked news topics [6, 8]. The right part represents our main contribution to the old version, which is the story identification system.

3.1 TweetMogaz v1

As introduced earlier, TweetMogaz [6] is a platform for tweets filtering and search [7]. "Mogaz" (موجز) means in Arabic "summary" or "digest", which means that the platform presents a digest of tweets relevant to certain topics. An input of a predefined set of keywords representing a broad topic (e.g. *Syrian conflict*) is prepared, and an adaptive information filtering technique is used to identify relevant tweets from a stream of tweets [8]. Identified tweets are used to generate a comprehensive report (Mogaz) about what is discussed on the topic on Twitter. The platform applies state-of-the-art normalization techniques for Arabic social text [2, 1], and uses Solr [2] for indexing the full stream of tweets to enable effective and efficient search.

[2]http://lucene.apache.org/solr/

3.2 Story Identification System

Several components are integrated to produce an effective and efficient system for identifying emerging stories within a stream of tweets on a given topic. For a set of relevant tweets to topic x in a time-window w, denoted as $T_x(w)$, sets of tweets within $T_x(w)$ representing individual news stories are identified through the following steps:

1. Detecting emerging terms: Most frequent terms, after applying stopwords removal and stemming, appearing in $T_x(w)$ are extracted. This set of terms represents the main discussed stories in w and is referred to as $E_x(w)$.

2. Clustering emerging terms: Emerging terms $E_x(w)$ are clustered to group related terms representing a certain story together. Term co-occurrence probability is used as the distance measure for applying clustering using a fuzzy agglomerative hierarchical clustering technique [10]. For each term t_i in $E_x(w)$, distance to other terms are calculated as shown in equation 1. If mutual $distance(t_i, t_j)$ and $distance(t_j, t_i)$ are less than a given threshold (selected as 0.6), t_j is added to the same cluster with t_i. Remaining terms in $E_x(w)$ are compared to all terms of the cluster and a term gets added if distance between any of the cluster terms and this term is less than the threshold. This leads to the presence of fuzzy clusters, where a term can exist in multiple clusters. Any cluster contains less than three terms is discarded. Also clusters are limited to a number of 6 terms at most, where the closest 6 terms are kept and the remaining are discarded from the cluster. This clustering method does not require setting a predefined number of clusters, where the number of clusters depends on the selected distance threshold.

$$distance(t_i, t_j) = 1 - P(t_j|t_i) = 1 - \frac{count(t_i, t_j)}{count(t_i)} \quad (1)$$

3. Enriching small clusters by additional keywords: To improve the representation of clusters with less than 6 terms, we apply query expansion to enrich the cluster with additional keywords. We create an *AND* Boolean query with all keywords in the cluster to search the tweets set $T_x(w)$. Top terms achieving the highest TFIDF are selected from the result tweets and then added to the cluster making it containing a total of 6 terms. TFIDF is calculated as shown in equation 2.

$$TFIDF(t_i) = tf_{search}(t_i) \cdot log(\frac{N}{df(t_i)}) \quad (2)$$

where, $tf_{search}(t_i)$ is the term frequency of term t_i in the search results of the Boolean query. $df(t_i)$ and N are the number of tweets containing the term t_i and the total number of tweets in $T_x(w)$ respectively.

4. Detecting clusters that can lead to duplicate stories: Performing story identification periodically allows the detection of recent stories. However, sometimes duplicate stories are detected. To overcome this problem we apply the following steps: (1) Keywords of each cluster are used to search $T_x(w)$ with a BM25 retrieval model, and top 100 results are retrieved. (2) Vector representation of terms is constructed for each cluster using the terms appearing in results list more than 10 times. (3) Cosine similarity is computed between vectors of each two clusters, and if similarity is more than 0.5, clusters are merged into one. This threshold was selected based on experimenting different values.

The output from previous steps produces set of clusters

Figure 2: Sample of pseudo-article (translated from Arabic to English)

that potentially map to different news stories. Terms in each cluster are then used to search the Solr index for relevant tweets in the past 48 hours. All tweets achieving a retrieval score (BM25 score) higher than a predefined threshold are considered for the generation of the pseudo-article as described later.

The process of clustering is applied every 15 minutes to cope with emerging stories happening in news. Cluster duplication test is applied between new and old clusters to prevent the generation of duplicate pseudo-articles. Detected duplicate clusters with old ones, are used to update the content of the old article while maintaining the chronological order of the identified stories.

3.3 Pseudo-Articles Generation

The input of this component is a set of tweets relevant to a potential news story, and the objective is to generate an article-like document that can show users the news story and how people think about it. The typical components of an article are: a title, a front image, article body, and possibly links to related content. We use the relevant tweets to each cluster to generate these components as follows:

- **Article body:** We display tweets relevant to the story as the body. Tweets are sorted by normalized retweet count, which is the number of retweets divided by the age of the tweet in minutes. This ensures displaying the most popular and recent tweets on the top. A "see more" button is offered to allow users to read all retrieved tweets on the story.
- **Front image:** We have three options for selecting a front image to the article according to availability: 1) Top tweet with an image is used. 2) Top tweet with link to YouTube video is used by taking the video thumbnail as the front image. 3) Top tweet with link to a news article is used by extracting the front image of the article.
- **Links to related articles:** This is simply achieved by extracting all links to news articles embedded in relevant tweets and sorting them by the number of occurrences in the tweets. This offers the reader external content to read driven by the trend on Twitter.
- **Title of the article:** The title of the pseudo-article is generated using the top tweet text after pruning the name mentions and links from its text.

Figure 2 shows an example of a generated pseudo-article. As shown the article is talking about a story happened in the Egyptian politics topic. Some of the displayed tweets are supporting and others are opposing the news. This feature is unique for TweetMogaz over standard news websites.

4. SYSTEM PERFORMANCE

The performance of our system depends a lot on the parameters and thresholds we use in our approach. These parameters can tune the performance of our system to be recall-based vs. precision-based. Since our platform is a public live service, we decided to select thresholds that would achieve high precision.as this is more acceptable by users. Based on manual evaluation of our platform over few months, the number of identified stories varies according to the events happening related to the topic on a given date. On average 10 stories per topic per day are identified. Around 80% of these stories map clearly to real-life events, while 20% may be seen as general tweets on the topic that do not relate to a specific story. It is unlikely to deliver duplicate stories except for few incidents that can happen around only 5% of the time. This shows the effectiveness of our duplicate stories detection approach.

5. DEMONSTRATION

TweetMogaz is a live news portal of tweets accessible for public at `http://www.tweetmogaz.com`. It collects Arabic tweets and generates comprehensive reports about news happening in different Arabic regions, including Egypt, Syria, and UAE and international sports.A stream of up to 12 million Arabic tweets are collected daily, and processing is performed online for filtering information to different topics, generating comprehensive reports to each topic, and identifying hot stories in each one. The website is entirely in Arabic, since it is directed to the Arabic region. All information on the website gets automatically updated every 15 minutes to cope with the trends on Twitter.

6. REFERENCES

[1] K. Darwish and W. Magdy. Arabic information retrieval. *Foundations and Trends® in Information Retrieval*, 7(4), 2014.

[2] K. Darwish, W. Magdy, and A. Mourad. Language processing for arabic microblog retrieval. In *CIKM*. ACM, 2012.

[3] H. Kwak, C. Lee, H. Park, and S. Moon. What is twitter, a social network or a news media? In *WWW*. ACM, 2010.

[4] C. Li, A. Sun, and A. Datta. Twevent: Segment-based event detection from tweets. In *CIKM*. ACM, 2012.

[5] R. Li, K. H. Lei, R. Khadiwala, and K.-C. Chang. Tedas: A twitter-based event detection and analysis system. In *(ICDE)*. IEEE, 2012.

[6] W. Magdy. Tweetmogaz: a news portal of tweets. In *SIGIR*. ACM, 2013.

[7] W. Magdy, A. Ali, and K. Darwish. A summarization tool for time-sensitive social media. In *CIKM*. ACM, 2012.

[8] W. Magdy and T. Elsayed. Adaptive method for following dynamic topics on twitter. In *ICWSM*, 2014.

[9] T. Sakaki, M. Okazaki, and Y. Matsuo. Earthquake shakes twitter users: real-time event detection by social sensors. In *WWW*. ACM, 2010.

[10] R. Sibson. Slink: an optimally efficient algorithm for the single-link cluster method. *The Computer Journal*, 16(1), 1973.

[11] J. Weng and B.-S. Lee. Event detection in twitter. In *ICWSM*, 2011.