MAQSA: A System for Social Analytics on News

Sihem Amer-Yahia, Samreen Anjum,
Amira Ghenai, Aysha Siddique,
Sofiane Abbar
QCRI, Doha, Qatar
{syahia, sanjum, aghenai, asiddique,
sabbar}@qf.org.qa

Sam Madden, Adam Marcus
MIT, Cambridge, USA
{madden, marcua}@csail.mit.edu

Mohammed El-Haddad
AlJazeera Network, Doha, Qatar
mohammed.haddad@aljazeera.net

ABSTRACT

We present MAQSA, a system for social analytics on news. MAQSA provides an interactive topic-centric dashboard that summarizes news articles and social activity (e.g., comments and tweets) around them. MAQSA helps editors and publishers in newsrooms understand user engagement and audience sentiment evolution on various topics of interest. It also helps news consumers explore public reaction on articles relevant to a topic and refine their exploration via related entities, topics, articles and tweets. Given a topic, e.g., "Gulf Oil Spill," or "The Arab Spring", MAQSA combines three key dimensions: *time, geographic location,* and *topic* to generate a detailed activity dashboard around relevant articles. The dashboard contains an annotated comment timeline and a social graph of comments. It utilizes commenters' locations to build maps of comment sentiment and topics by region of the world. Finally, to facilitate exploration, MAQSA provides listings of related entities, articles, and tweets. It algorithmically processes large collections of articles and tweets, and enables the dynamic specification of topics and dates for exploration. In this demo, participants will be invited to explore the social dynamics around articles on oil spills, the Libyan revolution, and the Arab Spring. In addition, participants will be able to define and explore their own topics dynamically.

Categories and Subject Descriptors

H.5 [**Information Interfaces and Presentation**]: Miscellaneous

General Terms

Design

Keywords

Social Analytics, Sentiment Analysis, Topic Extraction, Data Visualization

SIGMOD'12, May 20–24, 2012, Scottsdale, Arizona, USA.
Copyright 2012 ACM 978-1-4503-1247-9/12/05 ...$10.00.

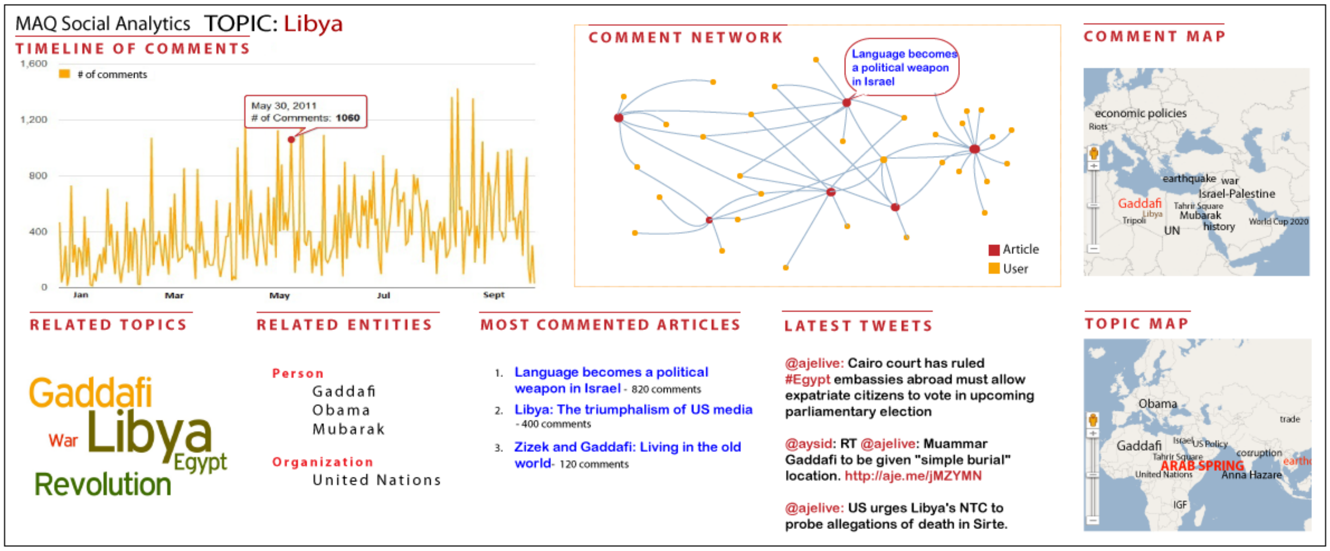
1. INTRODUCTION

The proliferation of social media is dramatically changing the way people produce and consume the news. As the barriers to content production fall and users are given the ability to express their opinions, editors, publishers and news consumers are finding it harder to *gather* and *follow* what is going on in the world. Users are included at every stage of newsmaking and consumption, both as citizen journalists that disseminate news via traditional and social media, and as commenters that participate in ephemeral networked audiences around particular articles. For users interested in understanding news trends, the notion of *topic* is of key importance, as there will be many different articles or posts about a particular event or idea (e.g., the Arab Spring, the Gulf Oil Spill, or the Occupy Wall Street movement.) A topic may be represented as a collection of articles in a single newspaper, as a set of articles in many different papers, as hashtags on social media like Twitter, or as a page or set of pages on Facebook. As journalists cover different angles of a story, they may wish to explore topics from different points of view, perhaps by different populations in different geographic regions. In this demo, we describe MAQSA, a news aggregator for both newsrooms and news consumers that presents several different views of a collection of articles, summarizing the frequency of comments and tweets about a particular topic or topics, showing their location on a map, grouping them by sentiment, and linking them to related articles, entities, tweets, and topics.

Major news websites recognize the growing importance of social media in disseminating and consuming news. Many websites now provide analytics in the form of the aggregate number of tweets, Facebook likes, and Google +1's for each article. These aggregates provide a *non-exploratory, article-centric* understanding of social activity created around a story. An essential departure is to offer annotated topic-centric analytics in order to guide users in their social exploration of news.

Our input is a database of news articles $\mathcal{A}$ and a collection of user comments $\mathcal{C}_a$ for each article $a \in \mathcal{A}$. Given a set of articles relevant to a topic (e.g., opinion pieces relevant to "Oil Spill," overview articles on "Libya" and "The Arab Spring"), we produce an interactive topic-centric activity dashboard that summarizes the actions of a large number of users in a way that appeals both to editors and publishers in newsrooms, and news consumers.

There are three key dimensions we exploit in building an aggregation-based summary of news articles and their com-



1. Language becomes a political weapon in Israel - 820 comments
2. Libya: The triumphalism of US media - 400 comments
3. Zizek and Gaddafi: Living in the old world- 120 comments

@ajelive: Cairo court has ruled #Egypt embassies abroad must allow expatriate citizens to vote in upcoming parliamentary election

@aysid: RT @ajelive: Muammar Gaddafi to be given "simple burial" location. http://aje.me/jMZYMN

@ajelive: US urges Libya's NTC to probe allegations of death in Sirte.

Figure 1: Topic Activity Dashboard

ments: *time, geographic location,* and *topic*. Time is essential for understanding changing topic trends and sentiment evolution. Geographic location identifies trends across news consumers and stories. Finally, related topics are the glue that binds related articles and comments. `MAQSA` uses time to power an annotated comment timeline. It combines time and geographic location to display aggregate sentiment in user comments. Finally, article topics are used to allow navigation to related articles, entities, tweets, and topics.

There are several challenges we addressed in building `MAQSA`. Fundamentally `MAQSA` must *join* a collection of articles by topic, location, and sentiment. Unfortunately, articles are largely unstructured, so our primary focus is on extracting structure from text to allow us to perform this join operation. Once articles are joined and aggregated, we can use those aggregates to construct our visualization. In `MAQSA` we had to address three structure extraction challenges:

1. *Topic discovery*, where we group articles by topic. We explore and adapt several well-known methods such as tf*idf [7], Latent Dirichlet Allocation (LDA) [3], and OpenCalais.[1]
2. *Sentiment extraction* from article comments, to group articles and comments by user opinion. Existing methods to automatically identify the polarity of text rely on pre-defined dictionaries of positive and negative words. In our experience, users express sentiment for different article categories (e.g., disasters or political events) using different words. In `MAQSA`, we learn topic-specific terms via an algorithm that refines an initial term dictionary.
3. *Article grouping*, where related articles, users, and topics are used to provide users with recommendations for additional reading or people to follow. Since the number of news articles relevant to a date range or topic is large, we must quickly find and aggregate articles and related user actions.

Our demonstration will allow participants to interact with a dashboard for several predefined topics and time ranges. Participants will be able to refine time granularity and topics, and explore related topics, entities, articles, and tweets. Participants can also define and select new topics to explore.

2. USER INTERFACE AND DEMO

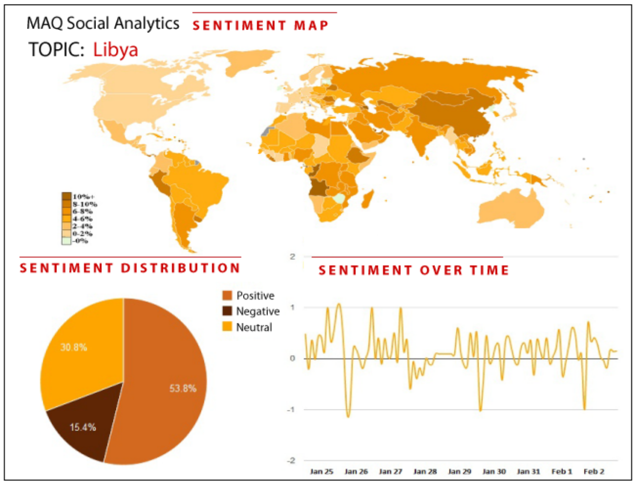


Figure 2: Aggregate Comment Sentiment Views

The primary interface of our demo is shown in Figures 1 and 2. Attendees will have the opportunity to select topics, e.g. *"Libya."* and interact with activity and sentiment dashboards. The example depicted in the figures is built from opinion articles from Al Jazeera English.[2]

The top left part of Figure 1 shows a timeline of comment activity for articles relevant to *"Libya."* Users can zoom in and out to explore different time granularities, and can click on different date in the graph to see the number of comments and articles. Details on a specific date, `May 30,2011`

[1] *http://www.opencalais.com*

[2] *http://english.aljazeera.net/indepth/opinion/*

in this case, are displayed in the remaining parts of the interface. Immediately to the right of the timeline is a social graph that enables the understanding of comments activity on that day, at the granularity of individual users and articles. To its right is a `Comment Map` showing the locations of the commenters, and the most common words they use in their comments that day (Al Jazeera English provided us with article comments and IP addresses of commenters.) Below that map is a `Topic Map` shows a geographic distribution of the most common topics covered by articles that received comments on the selected date. Observe that comment maps are different from topic maps extracted from articles being commented as they provide more insights on the concerns of news commenters beyond the topics covered by articles they read. To the left of the topic map is a collection of related objects. In the center, `Most Commented Articles` are the ones that received the highest number of comments on the selected date. The `Latest Tweets` are the ones mentioning those articles or covering their topics. All the way to the left, `Related Topics` extracted from articles that received comments on the selected date, are displayed as a word cloud. Viewers can click on a topic and immediately see a dashboard for it. Finally, `Related Entities` extracted from the articles are shown.

The top part of Figure 2 shows the sentiment distribution for articles relevant to *"Libya."* Sentiment is extracted from comments posted on those articles and plotted in three different ways: on a map based on location of commenters, as a pie chart, and as a timeline. Given this high-level overview of our interface, we now turn to the design of `MAQSA`.

3. MAQSA DESIGN

In this section, we introduce the data model, the architecture, and the main challenges in building `MAQSA`.

3.1 Data Model

`MAQSA` manages a set of articles $\mathcal{A}$, a set of topics $\mathcal{T}$, a set of users $\mathcal{U}$, and a set of comments $\mathcal{C}$. An article is a tuple with several fields, including a unique identifier *aid*, a *title*, a *post_date*, an *author*, and *content*. Topics are also tuples, containing the topic id *tid*, and a human-readable *description*. A comment is a tuple with a comment id *cid*, a user id *uid*, article foreign key *aid*, text fields for *subject* and *message*, a *post_location*, a *timestamp*. Additionally, we build tables that track the association between articles and topics, and between articles and the locations they mention.

3.2 Architecture

`MAQSA` relies on three components: *Topic Extraction*, *Activity Frequency*, and *Sentiment Extraction.*

The *Topic Extraction* module uses a combination of techniques to extract topics from existing and incoming articles. We implemented three methods: *LDA*, *tf*idf* and *OpenCalais*. Currently, topic extraction is done offline. Section 3.3 discusses the three methods we implemented and the need for an online incremental topic extraction.

The *Activity Frequency* module implements a scalable and efficient interactive topic-centric exploration of news. When a user selects a date, articles that received at least one comment that day are extracted and used to aggregate topics and user comments. We use Gephi[3] to plot a graph where an edge represents a user posting a comment on an article on a particular day. The *Activity Frequency* module implements topic aggregation from both comments and articles in order to extract the topics of the day, the topics covered by comments that day, related entities, related articles, and related tweets. This module relies on an adaptive ranking algorithms for articles that only re-ranks articles if the number of comments they received on a day is higher than the least commented articles on a previous day.

The *Sentiment Extraction* module relies on fine-tuned dictionaries containing words and their sentiment polarity. Sentiment is first extracted from each comment and aggregated at three levels: article, day, and geographic region. Our baseline approach to extract sentiment from comments uses term dictionaries from Sentistrength[4] and UPitt.[5] Those dictionaries contain words that express generic positive and negative sentiment. However, based on our experience in the news domain, users express sentiment for different topics (e.g., disasters or political events) using different words. For example, the word *power* is more commonly used to express positive sentiment in Sports than in Politics. Our next step is to adapt and extend the approach used in [5] to refine our baseline dictionary and improve the accuracy of extracted sentiment. We are developing an algorithm for learning topic-specific terms in a scalable way.

3.3 Topic and Entity Extraction

To identify sets of related articles, we extract topics and entities from article titles and contents. We have implemented *LDA*, *tf*idf*, and *OpenCalais* for topic extraction, and are using *OpenCalais* for entity extraction. We plan to conduct a series of user studies to determine which methods users find most appropriate.

LDA is a probabilistic generative method that uses a Bayesian network to discover a set $\mathcal{T}$ of latent topics from articles in $\mathcal{A}$, each of which viewed as a document formed by the words it contains. LDA outputs the probability of a topic generating each word in an article, as well as the probability of an article being about a topic. We associate each article $a \in \mathcal{A}$ with a topic signature $T_{sign}(a) = \{(t, score(a,t)) | \forall t \in \mathcal{T}\}$ where $score(a,t)$ is the relevance of a to t. Similarly, we use *tf*idf* with word stemming and stopword removal. Like with LDA, each article $a \in \mathcal{A}$ is characterized by a set of pairs $(t, score(a,t))$ where $score(a,t) = tf(t,a) * idf(t,\mathcal{A})$. We plan to use similar approaches to extract topics from comments and build topic maps as in Figure 1. We also plan to run a user study to compare topics extracted using all three methods and compare the results to decide which method works best in the news context.

We also plan to develop an algorithm in the spirit of [1] to assign topics incrementally to articles as they are added.

For entity extraction, we use OpenCalais, a web API that, when given a text document, extracts entities such as places, persons and organizations in text. OpenCalais can also be for topic extraction, as it identifies a set of topics in addition to named entities. Finally, OpenCalais identifies the *locations* mentioned in each article.

3.4 Generating the Visualization

The visualization is keyed by topic, which is supplied by the user. Given a topic t, we extract a set of relevant arti-

[3] *http://www.gephi.org*

[4] *http://sentistrength.wlv.ac.uk/*

[5] *http://www.cs.pitt.edu/mpqa/subj_lexicon.html*

cles A_t, their comments, and the entities and locations they reference. Out of those articles, a set of articles $S \subseteq A_t$ correspond to a user-selected time range in the comment timeline. These articles represent the current set being displayed in our visualization and changes every time the user selects a different date. We use Google Chart Tools[6] to generate the interactive charts in our figures.

Articles that are related will have similar topics. We use this idea to identify clusters of similar articles. To measure similarity, we compute the topic signature of a set of articles S, denoted $T_{set}(S) = \{(t, score(S,t)) | \forall t \in \mathcal{T}\}$ where $score(S,t) = avg_{a \in S} score(a,t)$. Sets of related articles will have more concentrated topic score distributions. Similarly, we define the topic signature of a comment $T_{sign}(c)$ and of a collection of comments $T_{set}(C)$. As future work, we plan on building a time- and topic-based article clustering algorithm that uses these signature methods.

The size of each topic in the word cloud shown in Figure 1 is determined using its relative score in $T_{set}(S)$. We use Wordle[7] to generate word clouds. The topic map is generated using a combination of the article location (returned by OpenCalais) and topics in $T_{set}(S)$. We are currently experimenting the generation of topic maps using TagMaps.[8]

Each article receives comments from users located in different geographic areas. We group comments by geographic region according to the location of users who authored the comments. We then compute the topic signature for each set of comments obtained that way, and display each location and topic signature on the comment map. To extract the tweets about a set of articles S, we use the top scoring topics in $T_{set}(S)$ to query Twitter. To generate the sentiment dashboard in Figure 2, we group comments by location and time window, and compute the average sentiment in each location or time period. The sentiment map is generated using Google Chart Tools.

4. RELATED WORK

Others have worked on the problem of extracting entities and sentiments, and generating visualizations from Twitter and other social streams, but have not taken a *topic-centric* approach to viewing a collection of news articles with a focus on their user comments in the way we propose.

Prior work by a subset of the authors [6] focused on an analytics dashboard for a collection of Twitter feeds; in this demo we focus on news articles and their comments, and extend our ideas with more sophisticated topic and entity extraction techniques. We also combine topics extracted from articles and comments with user and article location, to show geographic distributions. Other similar works includes Statler [8] for timeline display of tweets during large-scale events, and Eddi [2] for extracting topics from tweets.

Several tools on extracting entities and using them to explore news are available online. TextMap[9] is an entity-based search engine that extracts entity references in news sources and produces visualizations that analyze the relationships between entities. NewsExplorer[10] clusters news articles into groups of related articles and identifies entities such as people, places and organizations for each group. In our work, we are using open source tools such as OpenCalais for entity extraction.

Several works explore sentiment extraction from articles and microblogs. Vox Civitas [4] by Diakopoulos et al., provides a timeline-based sentiment over time visualization of events discussed on microblogs. TextMap carries out sentiment analysis for entities in a timeline fashion across domains such as business, health, crime, and politics. Similar work is done by Opinion Crawl,[11] which crawls the web for a given topic, and focuses on generating sentiment analysis and related headlines. While Opinion Crawl focuses on sentiment analysis for news articles and blogs, our system focuses on sentiment analysis of user comments.

Timelines, charts, and maps are commonly used to visualize social media topics. TextMap represents sentiment distribution of news entities via heatmaps and charts. Similarly, Opinion Crawl and TwitInfo [6] use charts and maps for sentiment distribution. Tag clouds are also used by systems like Opinion Crawl to represent most relevant or discussed topics. Another interesting approach to represent related topics in a geo context is shown by Trendsmap,[12] which visualizes twitter trends on the world map.

5. CONCLUSION

MAQSA provides an *on-the-fly trend visualization* of user activity in a collection of news articles, focusing on a topic *topic-centric*, *interactive* view. It combines a number of entity, sentiment, and topic extraction technologies to join together related articles. Our demo will allow users to input topics of interest and interact with the visualization to explore trends and relationships in topics and users.

6. REFERENCES

[1] A. Ahmed, Q. Ho, J. Eisenstein, E. P. Xing, A. J. Smola, and C. H. Teo. Unified analysis of streaming news. In *WWW*, pages 267–276, 2011.

[2] M. S. Bernstein, B. Suh, L. Hong, J. Chen, S. Kairam, and E. H. Chi. Eddi: Interactive Topic-based Browsing of Social Status Streams. In *UIST '10*, 2010.

[3] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

[4] N. Diakopoulos, M. Naaman, and F. Kivran-Swaine. Diamonds in the rough: Social media visual analytics for journalistic inquiry. In *VAST 2010.*

[5] N. Godbole, M. Srinivasaiah, and S. Skiena. Large-scale sentiment analysis for news and blogs. In *Proceedings of the International Conference on Weblogs and Social Media (ICWSM)*, 2007.

[6] 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 *CHI*, pages 227–236, 2011.

[7] G. Salton and C. Buckley. Term-weighting approaches in automatic text retrieval. *Information Processing and Management*, 24(5):513–523, 1988.

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

[6] *http://code.google.com/apis/chart/interactive/docs/gallery.html*

[7] *http://www.wordle.net/*

[8] http://tagmaps.research.yahoo.com/

[9] *http://www.textmap.com*

[10] *http://emm.newsexplorer.eu*

[11] *http://www.opinioncrawl.com*

[12] *http://www.trendsmap.com*