TwIST: A Mobile Approach for Searching and Exploring within Twitter

Radhika Gopi
Department of Computer Science
University of Regina
Regina, SK, Canada S4S 0A2
gopi200r@uregina.ca

Orland Hoeber
Department of Computer Science
University of Regina
Regina, SK, Canada S4S 0A2
orland.hoeber@uregina.ca

ABSTRACT

Popular user-generated content services like Twitter have made it easy for people to post and browse short messages while on the go. However, the ability to search is limited by the query box and search results list paradigm. In this paper, we propose TwIST (Twitter Information Search Tool), which has been designed to support interactive search and exploration within Twitter. TwIST is a mobile application that works in two modes: simple search and exploratory search. The searcher can easily change between these modes by rotating the device, making it easy to access the mode that will support the desired search activity. While the simple search mode is dominated by the common search results list format, the exploratory search mode uses topic modelling, visualization, and interactive filtering to support the searcher in finding the information they are seeking.

CCS Concepts

•Information systems → Search interfaces; Information extraction; Retrieval on mobile devices;

Keywords

Search interfaces, topic modelling, mobile computing

1. INTRODUCTION

Social networking services such as Twitter serve as an important source for user-generated content. Their public, unfiltered, and open aspects make them a valuable source for public opinion on a wide range of topics. However, finding useful information on Twitter is limited by a search interface that follows the traditional list based representation, requiring the user to evaluate the search results tweet by tweet. While list based displays can be effective when presenting topically focused search results, they are less effective when there is ambiguity within the information displayed [7]. Due to the short and cryptic nature of tweets and the breadth of information posted within Twitter, search results that include a mix of relevant and irrelevant posts are commonplace. As such, we suggest that searching within such data may be more effective if visual and interactive tools are provided that allow the searcher to focus their efforts on the intended topic, filtering out irrelevant tweets.

While mobile devices are constrained in terms of memory, processing power, and screen size, they have become extremely popular as information seeking tools due to their portability, connectivity, and intuitive operation. When designing a mobile search app, it is important to provide an interface that makes it easy to identify and navigate among the information presented, and to remain focused on why one might want to search for information in a mobile context. In terms of searching within Twitter, we have observed that one may be interested to know three things: (1) What are the most recent posts that use the search term? (2) What are the major topics mentioned in these posts? and (3) How are the posts related in terms of the topics?

In this paper, we propose TwIST, a Twitter Information Search Tool that provides both simple search and exploratory search modes, enabling the searcher to address these questions as they seek relevant tweets that satisfy their information seeking goals. The simple search mode was designed to support the lookup-based IR model [2], which is useful when the search strategy focuses on the lookup of specific information [12] or the verification of facts [20]. It allows the user to issue query and inspect the tweets in a list based representation. The exploratory search mode was designed to support the exploratory search model [12], which extends the search activities beyond information lookup towards learning and investigation. This mode helps the user to recognize relevant information by displaying a set of topics extracted from the tweets, along with their relative importance within the search results. Interactive selection of relevant topics allows for the exploration of the search results and a comparative analysis of the topics. These topics can subsequently be used to interactively refine the query.

2. RELATED WORK

2.1 Search within User Generated Content

In a systematic comparison of search behaviour in Twitter and the web, Teevan et al. found that searching within user generated content generally focuses on temporally relevant information (e.g., related to current news and events) and social information (e.g., related to opinions and the posts of people of interest) [17]. Traditional search interfaces may not be the best option in these cases; there is a need to

CHIIR '16, March 13-17, 2016, Carrboro, NC, USA

DOI: http://dx.doi.org/10.1145/2854946.2854991

develop new interfaces that focus on supporting complex search activities using machine learning to extract relevant information and visualization to convey this information to the searcher[11]. Approaches that use faceted browsing to leverage existing or dynamically extracted data attributes can be an effective method for supporting information seeking tasks [1]. Hearst et al. has demonstrated that such facets can help structure the search results and aid in the reformulation of queries [8].

Topic modelling has been shown to be an effective technique for summarizing user generated content. Methods for extracting topics by aggregating hashtags has helped to identify topically relevant streams of posts [13]. However, the over-reliance on hashtags require that posters use them effectively. Other topic-based approaches, such as Eddi [4] and TweetMotif [5], follow a clustering approach to extract the topics, and display them to the user for further exploration. A common theme among these approaches is the use of topics to summarize the results, and as a tool to help the searchers to refine their queries.

2.2 Mobile Based Search Applications

Although conducting searches using mobile devices is more difficult than on a computer, people have shown an interest in using mobile devices to perform complex information behaviours such as browsing, downloading, and sharing content [16]. Amidst the advantage of the immediacy of information, there are a limited number of mobile applications supporting search activities. For those that do exist, displaying the results in categories provides an effective overview of the information, and has resulted in an increase in retrieval performance [6, 9]. Such categorical views help the searcher to find relevant information faster than list based views. Visual representations of search results [10] may also help the searcher in their information seeking processes, even with limited prior knowledge about the information being shown.

3. TWIST

TwIST has been designed to support the searcher's knowledge discovery and decision making activities [10] through interactive visualization of the search results, leveraging topic modelling, dynamic categorization, faceted browsing, and interactive query refinement. Two modes of operation are available, depending on the orientation of the device. The details of these two modes are explained in the remainder of this section.

3.1 Simple Search Mode

When the device is held in upright position (portrait mode) TwIST defaults to the simple search mode, with the primary interface elements following the query box and search results list paradigm (see Figure 1). This orientation provides more vertical than horizontal space, and is dominated by the list of tweets that match the searcher's query. The overall design is meant to replicate the search interface currently used by the Twitter app, but with some simple modifications and additions. Upon submitting a query, the app fetches the most recent 100 posts using the Twitter API [18], with the results sorted by their submission date and time (most recent first). These are displayed in a scrollable list, and include relevant metadata such as author information and date/time. In order to help the searcher to decide on the relevance to the query, any use of the query terms within the tweet contents

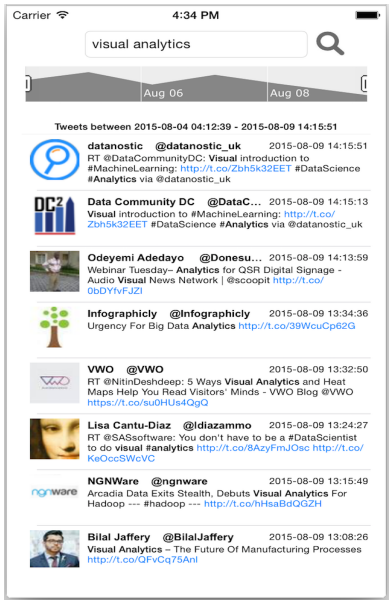

Figure 1: The simple search mode provides a query box and search results list, as well as a timeline visualization.

are highlighted in bold.

Given the importance of the temporal aspect of the tweets, a timeline is provided to allow the searcher to observe such patterns visually. This timeline represents time on the x-axis and the aggregated frequency of the tweets on the y-axis. The aggregation scale is chosen automatically depending on the temporal range of the tweets within the search results. The timeline supports interactive zoom and filter operations, allowing the searcher to dynamically adjust the timeframe of the tweets shown in the search results list. The simplicity and familiarity of the overall interface was designed to enable learnability.

3.2 Exploratory Search Mode

When the device is rotated to landscape mode, the search interface is transformed into the exploratory search mode (see Figure 2). This orientation provides more horizontal than vertical space, making room to provide interactive visualization features to support exploration and analysis activities. The exploratory search mode was designed following Shneiderman's information seeking mantra: "overview first, zoom and filter, then details-on-demand" [15]. Topics automatically extracted from the search results provide a framework for the overview of the search results. Both temporal zooming and topic-based filtering are supported. At any time, the searcher can view the specific details of any of the tweets. The primary goal is to support information filtering tasks [3], which can help the searcher to focus the search results on the specific topics that are of interest at the specific point in time. In addition, the exploratory search activities of learning the topics hidden in the search results, investigating the relationship between the topics, and looking up of results with respect to the topics [12] are all supported.

Due to the limited processing power of the mobile device, the computing necessary to perform topic modelling on the

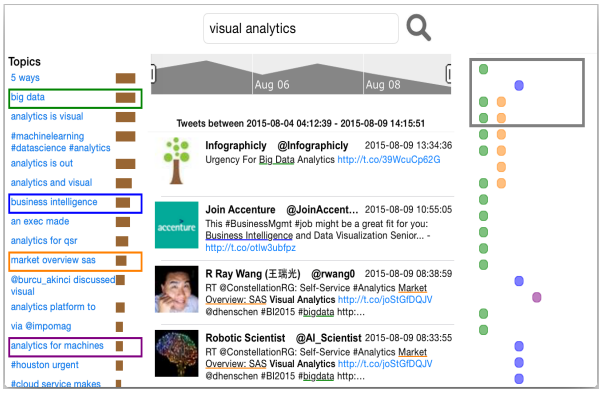

Figure 2: The exploratory search mode provides an overview of the topics (left), the list of tweets (middle), and a comparison view (right). A timeline filter is included at the top.

tweets is implemented on a server. The TwIST app communicates with this server using REST, loading the search results first, and the additional information regarding the topics once the modelling is complete. The topic modelling algorithm itself operates in three steps, which are described in the following subsections.

3.2.1 *Preprocessing of Tweets*

Given that the topic modelling is meant to extract meaningful textual topics from the set of search results, it is important to first clean the data to remove any content that may result in meaningless topics. All symbols, emoticons, incomplete URLs, and single characters other than numbers are removed from the data. Since retweets may result in significant repetition within the data, distinct tweets are selected for the topic modelling process.

Many tweets include links to external content, so we use this information to supplement the topic modelling process. All URLs included within the tweets are followed, and the titles of the corresponding web pages are added to the tweet. In the cases where this information was already written in the body of the tweet itself, the information is not duplicated.

3.2.2 *Clustering*

The next step in the process is to group similar tweets into topic clusters, such that each cluster represents a specific and independent topic. Given the variability within the search results sets, it is necessary to perform such clustering in an unsupervised manner, and without a predetermined number of clusters. As such, nearest neighbour clustering [14] is used. The clustering begins by computing the similarities between tweets, where each tweet is represented as a feature vector of terms, weighted using TF-IDF. The distance is computed using cosine similarity according to the following formula:

$$sim(FV_i, FV_j) = \frac{FV_i \cdot FV_j}{||FV_i||\ ||FV_j||}$$

where FV_i and FV_j are feature vectors of tweets i and j in the collection.

Calculating the pair-wise similarity between all tweets results in a similarity matrix n $\times$ n where n is the number of tweets in the collection (for this work, $n = 100$). This similarity matrix is then sorted in ascending order over each column, with any entries greater than a pre-defined similarity constant removed. At this point, each column of the matrix will represent a cluster of tweets that are most similar to each tweet. Since this representation will include multiple repetitive clusters, the final step is to merge similar clusters. Each column of the matrix is compared to all other columns to find the overlapping clusters that are sufficiently close in terms of overall per-tweet similarity. Duplicates found in this way are merged into the original.

3.2.3 *Topic Extraction*

Using the tweet contents that produced the clusters in the previous step, uni-grams, bi-grams, and tri-grams are extracted from the text. The bi-grams and tri-grams are computed in consideration of any content that was removed from the original tweet (e.g., punctuation, emoticons), as well as the boundary between the tweet itself and the titles of any URLs added to the tweet in the pre-processing step. For each such n-gram, its frequency within the cluster is calculated.

Part-of-speech (POS) tagging is then used to identify key n-grams that can be used as topic-describing phrases. With each term in each n-gram tagged in this way, the n-grams are then passed to a syntactic filter that only retains those n-grams that contain at least a single noun phrase. This noun phrase is considered a subject descriptor, and is therefore a good representation of some topic within the search results. All n-grams remaining after this filtering are sorted by their frequency within the cluster and then their length. As a final step, the top n-gram from each cluster is chosen as topic, and the frequency of occurrence of each topic is calculated within the entire search results set.

3.2.4 *Displaying Topics*

The topics extracted in the previous step are displayed along with an horizontal histogram that visually depicts the relative importance of each topic in relation to the search results set. Tapping on any topic will filter the search results to only show those that make use of the selected phrase. Doing so for multiple topics will produce a combined list, sorted chronologically. If the searcher recognizes any topic that is particularly relevant to what is being sought, a new search using this topic as the query can be initiated with a long-tap on the topic.

3.2.5 *Visual Comparison of Topics*

When multiple topics are activated within the exploratory search mode, it may be useful to help the searcher to understand the relationship between these topics and the tweets from which they were extracted. Visual encoding within the search results list, along with a visual comparison view are provided in the exploratory search mode for this purpose. Each topic selected is encoded using a distinct colour [19] and the use of these topics within the tweets are underlined using this same colour. This allows the searcher to visually identify where these topics are being used in the tweets. In the case where the topic came from the URL, the URL within the tweet is underlined.

Within the comparison view, the tweets of the selected topics are displayed in a table format, with each column representing a topic and each row representing the tweets. Each

tweet belonging to a topic is represented as a coloured circle that corresponds to the topic. This visual representation makes it easy to visually compare the topics to one another (e.g., viewing the column similarities and differences) as well as compare the tweets (e.g, viewing the row similarities and differences). A scrollable box is placed over the comparison view to show which tweets are being displayed in the search results list, providing context to what will be found if the searcher scrolls within the filtered search results.

4. CONCLUSIONS

In this paper, we have presented our work on the development of a mobile Twitter search interface. TwIST is a dual-mode search app that can be transformed from a simple search interface to an exploratory search interface simply by rotating the orientation of the device. Within the exploratory search mode, the topics are automatically extracted from the search results, with the importance of each in the search results visually encoded. Interactive selection of a topic filters the search results, and when multiple topics are selected, a comparison view is provided to support the searcher in understanding the relationships between topics and tweets.

The key limitation in this work is the efficiency of the unsupervised topic modelling. Given the limited computing capacity of mobile devices, we have moved this computation to a server. Even so, the system must still download all of the tweets, follow any embedded links, and then iterate over the data multiple times to discover the topics. With a search results set of 100 tweets, the system currently takes approximately 12 seconds to display the topics that are necessary for the exploratory search mode. In order to hide this delay from the searcher, the topic modelling is initiated even while searching within the simple search mode.

There are three elements of future work that are ongoing. We are working to improve the efficiency and accuracy of the topic modelling aspect of this work. A comprehensive user evaluation of the TwIST interface is currently in the planning stages. Finally, we plan to make it possible for a Twitter user to provide their login credentials in order to support searching and exploration within their personalized Twitter feed.

5. REFERENCES

[1] F. Abel, I. Celik, G.-J. Houben, and P. Siehndel. Leveraging the semantics of tweets for adaptive faceted search on Twitter. In *Proceedings of the International Conference on The Semantic Web*, pages 1–17, 2011.

[2] M. J. Bates. The design of browsing and berrypicking techniques for the online search interface. *Online Review*, 13(5):407–424, 1989.

[3] N. J. Belkin and W. B. Croft. Information filtering and information retrieval: Two sides of the same coin? *Communications of the ACM*, 35(12):29–38, 1992.

[4] 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 *Proceedings of the Annual ACM Symposium on User Interface Software and Technology*, pages 303–312, 2010.

[5] B. O. Connor, M. Krieger, and D. Ahn. TweetMotif: Exploratory search and topic summarization for Twitter. In *Proceedings of the International AAAI Conference on Weblogs and Social Media*, pages 2–3, 2010.

[6] E. W. De Luca and A. Nürnberger. Supporting information retrieval on mobile devices. In *Proceedings of the International Conference on Human Computer Interaction with Mobile Devices & Services*, pages 347–348, 2005.

[7] S. Dumais, E. Cutrell, and H. Chen. Optimizing search by showing results in context. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 277–284, 2001.

[8] M. Hearst, A. Elliott, J. English, R. Sinha, K. Swearingen, and K.-P. Yee. Finding the flow in web site search. *Communications of the ACM*, 45(9):42–49, 2002.

[9] T. Heimonen. Mobile Findex: Facilitating information access in mobile web search with automatic result clustering. In *Proceedings of the International Conference on Human Computer Interaction with Mobile Devices and Services*, pages 397–404, 2007.

[10] O. Hoeber. Human-centred web search. In C. Jouis, I. Biskri, J.-G. Ganascia, and M. Roux, editors, *Next Generation Search Engines: Advanced Models for Information Retrieval*, pages 217–238, 2012.

[11] O. Hoeber. Visual search analytics: Combining machine learning and interactive visualization to support human-centred search. In *Proceedings of Beyond Single-Shot Text Queries: Bridging the Gap(s) Between Research Communities Workshop*, pages 37–43, 2014.

[12] G. Marchionini. Exploratory search: From finding to understanding. *Communications of the ACM*, 49(4):41–46, 2006.

[13] K. D. Rosa, R. Shah, B. Lin, A. Gershman, and R. Frederking. Topical clustering of tweets. In *Proceedings of the SIGIR Workshop on Social Web Search and Mining*, pages 1–8, 2011.

[14] N. Roussopoulos, S. Kelley, and F. Vincent. Nearest neighbor queries. In *Proceedings of the International Conference on Management of Data*, pages 71–79, 1995.

[15] B. Shneiderman. The eyes have it: A task by data type taxonomy for information visualizations. In *Proceedings of the IEEE Symposium on Visual Languages*, pages 336–343, 1996.

[16] A. Smith. The best (and worst) of mobile connectivity, 2012. http://www.pewinternet.org/files/old-media/Files/Reports/2012/PIP_Best_Worst_Mobile_113012.pdf Accessed: 2015-08-22.

[17] J. Teevan, D. Ramage, and M. R. Morris. #TwitterSearch: A comparison of microblog search and web search. In *Proceedings of the International Conference on Web Search and Data Mining*, pages 35–44, 2011.

[18] Twitter. Twitter search API. https://api.twitter.com/1.1/. Accessed: 2015-08-18.

[19] C. Ware. *Information visualization: Perception for design*. Elsevier, 2nd edition, 2012.

[20] R. W. White and R. A. Roth. Exploratory search: Beyond the query-response paradigm. *Synthesis Lectures on Information Concepts, Retrieval, and Services Series*, 3(1):1–98, 2009.