Journal of Theoretical and Applied Information Technology
15th February 2019. Vol.97. No 3
© 2005 – ongoing JATIT & LLS

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

SAMA: A TWITTER BASED WEB SEARCH ENGINE

FALAH AL-AKASHI

University of Kufa, Faculty of Engineering, Department of Electronics, Iraq

E-mail: falahh.alakaishi@uokufa.edu.iq

ABSTRACT

How can a model efficiently identify relevant references in the hundreds of millions of Twitter messages that are posted every day? In this paper, we intend to address this fundamental research question, as well as introduce SAMA, a scalable search model that uses Twitter streams. Real-time topic detection is an important function for all search engines, and extracting topics from Twitter raises new challenges. As a huge temporal data flow, Twitter has many various types of topics, as well as a lot of noise. Current sophisticated search engines with high computational complexity are not designed to handle such large data flows efficiently. Twitter provides many opportunities for people to engage with real-time world events through communication and information sharing, as well as tools for dealing with its data. However, little is understood about the external links available in Twitter content, and this affects topic engagement. As of today, Twitter posts and its external links is very limited using upon traditional search engine despite the fact that content of micro-blogging presented by Twitter is very curious and useful for some queries rather than content of traditional Webs. In this paper, we propose a platform for modeling URL and inverse message frequencies and Twitter external references, which allows us to use a novel self-content detection algorithm for link authorities. Our model can make use of a new source of Web references, and experiments verify the effectiveness of the model in real time topic detection of Twitter social content. In our evaluations, we investigate the impact of different features on retrieval performance, and highlight tweet features that have high precision for both adhoc and diversity tasks: 77% and 78% respectively.

Keywords: *Twitter, Topic Detection, Social Search Model, Web References*

1. INTRODUCTION

The degree of interaction between users on the Web has increased exponentially over the past decade, and there are many challenges when dealing with such vast amounts of data. While much of the data is limited and due to private interaction such as emails and blogs, there has been a recent surge in social communications data driven by Twitter and Facebook. To date, only a few search engine applications have attempted to improve user experiences with social media and related data.

Facebook and Twitter are the current leaders in social networking, with over 500 million users each. In social applications, users post an abundance of information on these networks, which can be used to define their online personality. A social application can learn user preferences through static information like book and movie interests, and/or dynamic information such as user locations. Another significant feature is user social circles, including the posts of friends to friends. From a search engine perspective, social interaction data can be very helpful for personalizing results for users. However, in addition to being social communication platforms, Facebook and Twitter enable third party applications to track interactions and search for users who share the same interests by applying REST authentication. Facebook has taken two major steps that have impacted the search field. In September 2009 they made their data available to third party services [1], and in September 2010 they began adding web links to external web resources, that were based on user recommendations [5]. Despite these developments, however, little has been done with this data since.

In late 2009, researcher [19] launched an algorithm called ‘Facebook Results’ that indexed authenticated user and friends’ wall posts, comments and interests. The data was somewhat disorganized, and using the related data enriched the user experience. While social data on the web is still relatively new, there have been some attempts to apply it to enhance web searching and topic sources. A San Francisco based start-up [18] employed ‘User-Rank’ to measure the influence of users on their circle of friends, and to determine their score and the topics that were most influential in User-Topic-Ranking. Due to a recent development, web search engines can now use such information to determine the authority of social data. In October 2010, Bing and Facebook announced the "Bing Social Layer', which provides the ability to search for people on Facebook, and computes the related links that a user's friends had liked in Bing's search results [20].

Micro blogging in Twitter have emerged as large information sources for exploring and analysing news-related topics [13, 15]. Socially, Twitter is used as a major platform for publishing and disseminating information related to various topics such as politics or sport events. For trending topics, thousands of Twitter messages (tweets) are posted per second. That means, the number of posts published per day typically exceeds several hundred million. Thus, searching for tweets that are relevant to a given topic is a non-trivial research challenge. Furthermore, Twitter search engines have faceted other two challenges: The first one is related to how they display their search results in categories. The second one is related to the behaviour of Twitter search compared to Web search. Researcher [15] revealed that users exhibited different search behaviour on Twitter than when they search the Web. For example, queries on Twitter are significantly shorter than those for Web searching. Users typically use 1.64 words to search on Twitter and 3.08 words on the Web; this is due to the 140 character limitation per Twitter message. As long keyword queries can become restrictive, people tend to use broader and fewer keywords for searching.

Given the drawbacks of query terms provided by Twitter, researchers are now investigating alternative search interfaces, and a new interface to categorize the tweets in the personal timeline of a user into topics has been introduced. This area is developing rapidly, and improved interfaces will enhance search experiences for both information creators and users. Therefore, social searching and search interfaces that involve geographical contents are sensitive areas we will consider in advance.

This paper is organized as follows: Section 2 indicates the requirements of real-time searching, and Section 3 describes links analysis on Twitter. Section 4 details the approach to our proposal. Section 5 describes query processing and discusses the experimental results, and Section 6 presents concluding remarks.

2. REQUIRENTS OF THE REAL-TIME APPROACH

We begin by describing the objective requirements of real-time searching, which to a large extent dictate the design of the SAMA1 search engine. With the exception of the first, these requirements are different than those of a web search engine.

- **Low-latency and high-throughput query processing:** When it comes to searching users demand quick results, and to succeed a search engine must be able to work with large query volumes. Web search engines meet the requirements of fast query response time and high query throughput, and sometimes also improve real-time results as well.

- **High volume rate and instance data retrieved**: In a real-time search documents can be retrieved very quickly, and there are often sudden spikes similar to the 'flash crowd' effect. Regardless of the response rate, users expect the index to be searchable within a few seconds, which means the indexer must achieve both low latency and high throughput. This requirement is different than common assumptions in typical search environments where indexing can be considered a batch operation. Although modern web crawlers achieve high throughput, crawled content is not available for searching immediately; Web search engines face this issue when crawling real-time contents in social media such as Twitter. Depending on the type of content, with the exception of social content, an indexing delay of minutes, hours or even days might be acceptable. This allows engineers to trade off latency for throughput when running indexing jobs on batch systems such as Map Reduce [13]. Alternatively, researchers found that more resources can be applied to reduce indexing latency using special [12], but these alternatives still did not improve the speed enough for real-time searching.

- **Synchronous reads and writes:** Real-time search engines must simulate large volumes of synchronous reads and writes. This means index structures must be continuously updated

[1] http://site.uottawa.ca/~falak081

as contents are delivered, and index structures accessed quickly to serve several queries simultaneously. This is in contrast to a highly concurrent web search deployed in a static index structure. As a simple solution, indexes can be deployed in atomic swaps from older to newer index content, which makes it fairly easy to design architectures where the indexes are never read from and written to concurrently.

- **Dominance of the temporal signal**: The very nature of a real-time search engine means that the timestamp of the content is a signal to order results. By default, such an engine would display hits in reverse chronological order, and even when other relevance signals are incorporated to improve the ordering (as with Twitter), the temporal signal remains dominant. This is unlike web searching, where the timestamp of a web page has a relatively minor role in determining the relevance ranking (news searches being the obvious exception). The practical implication of this is that the query evaluation algorithm traverses inverted index postings in reverse chronological order. Researchers [11] proposed that these factors are the standard requirements of real-time web search engines.

3. LINK ANALYSIS ON TWITTER

The artificial intelligence behind search engines considers repetition as confirmation that something is true. Thus, using the same user name and handle gives them greater weight and authority in search results. When a company or brand name is part of the user name and handle, its legitimacy with search engines also increases.

Witter references acting as handles for users can be defined as direct or indirect links from a tweet to a peer reviewed suitable resource online. They can be first and second-order references, based on if there is an intermediate web page between the tweet and the target reference. Researchers showed that in a sample of tweets collected from 28 academic researchers, including URLs, 6% match their definition of twitter citations; that is, they linked directly or via an intermediate webpage to a peer-reviewed article. Moreover, they suggested that linking to a peer-reviewed publication is only one dimension of citing with Twitter, and they are interested to discuss other alternatives. All URLs included in tweets are a form of reference, and analyses may focus on the types of resources that are referenced in URLs. URLs in tweets act as external citations (where the tweet includes a reference and the external source receives a citation). Retweets can be interpreted as a form of inter-Twitter citations (internal citations).

A user who retweets to another user publishes a reference, and the first user gets a citation. In general, users retweet for different reasons such as information diffusion, or they use retweets as a “means of participating in a diffuse conversation”. Yet, retweet analysis is not easy to perform, due to the lack of format standardization. User names within tweets symbolized by @ also resemble references; for example, in tweets like “Just read an interesting paper by @sampleuser”. To date, they cannot be automatically distinguished from other messages, and therefore must be excluded from current analyses. In the following section, we analyze test sets of tweets with respect to the first two types of Twitter citations: URLs in tweets (external citations) and retweets (internal citations).

4. THE PROPOSAL APPROACH

The advent of services such as Twitter and Facebook has made it very easy for researchers to create real-time data in the form of micro-posts. As mentioned previously, Twitter users write whatever they wish in 140 characters, and whether they publicize a link to an external article they like or just share a passing thought, the information is extremely valuable from a search engine real-time perspective. However, as different projects over the past few years have shown, this information is much richer if it is mined and presented to the users in real-time [3].

Twitter has updated its content to allow shared internal and external links to content, and similarly, the target links can use backlinks to social media profiles or companies to boost their credibility. However, we will describe our algorithm for both external links related to Web references, and internal links denoted by user identifiers by @. As a consequence, our algorithm can compute the impact of each posted link.

Building a real-time indexing approach requires access to raw micro-post data. Fortunately, as a real-time data service, Twitter provides such data for periods of seven days to researchers who request it. Since this data must be processed and presented in real-time, our system avoids dealing with it all at once. There are ~90M tweets generated daily; 1,040 per second [13]. As of 2013, at about 140 bytes apiece 12.6 GB of tweet-data is created every day. Having petabytes of data, and dealing with a corpus of this size is trivial for any modern search engine [10]. Thus, our system connects to the Twitter index to retrieve data for a particular topic regarding a user query in periods of one day. However, are all tweets equally important? From our perspective, the important tweets on a particular topic must be rated many times between users, and cycled within a period. More specifically, the tweets that point to external links have higher impact than others, as users are more likely to request diverse topics through web pages since they are more natural than short tweet texts. Similarly the tweets that point internally to other posts have the least diversity and the topics refer to immediate events. However, by combining co-occurrence and information gain with a time decay factor, it is possible to analyze user tweets and derive the related topics in real-time.

We assume that we do not own the Twitter Data and we access Twitter data only through a Twitter REST API call. Thus, the tweets in the result set are the most recent tweets that contain the one or more of the query terms. In order to compute the trustworthiness of the source of a tweet, we model the entire Twitter interlinks as a graph of three layers as shown in Figure 1. Each layer in this model corresponds to one of the features of a tweet: the content, the user, and the links that are part of that tweet. The user layer consists of the set of all users. The tweet layer consists of the content of the tweets that involved some features of a tweet that are found to do well in determining the trustworthiness of that tweet. The links layer consists of the external links and URLs for a specific topic and user [23].

As with traditional indexing, to achieve this process the algorithm must involve crawling, parsing, extracting any external links if present and indexing them as internal tweets and followers if otherwise, and building the index. The system processes every user contribution, including the text, tags, and all the other structured data in the collection. This stage is similar to the anchor texts in web page indexing, but instead of anchor texts we use tweet texts, and instead of PageRank we use User Rank. Though traditional search engines, such as Google and Microsoft, are aware of the importance of tweet data, they have not worked with it strongly. Real-time data and related topics help users discover information about current topics better than the classical web data available in Web resources.

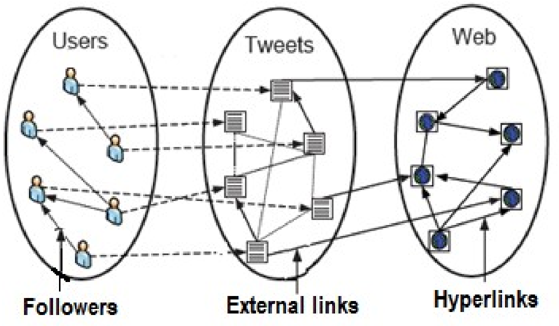
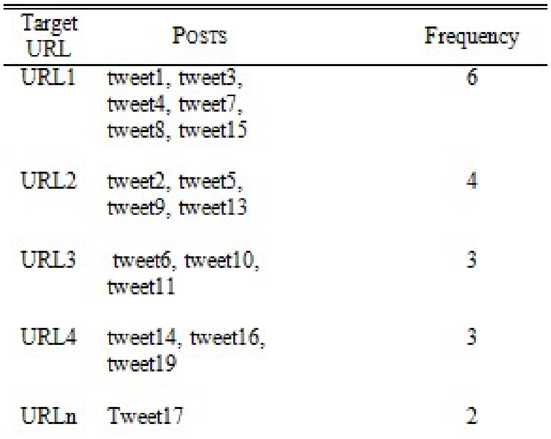


Figure 1: Graph of Interlinks in Twitter

Since the data must be retrieved and indexed in real-time, and ranked and presented to the users from the most to least recent, our system crawls, parses and builds the index on-the-fly. Though the process of building a real-time index is slower than creating a classical index, the data is fresher and the algorithm more realistic. We used two algorithms: one for indexing the external links, which has more impact than the second in which the topic are normalized to be more diverse. The second index involves tweets that represent more and fresher data. In terms of the first index, as shown in table (1), we build a pool of vectors for all tweets that involve external references, with each vector in the pool represented by three parts. The first part is a unique URL link to the external pages represented by some tweets that are aggregated in part two, and the third part represents the frequency of the URL with respect to the tweets. In information retrieval, vector space is an efficient model for indexing and ranking such data [7].

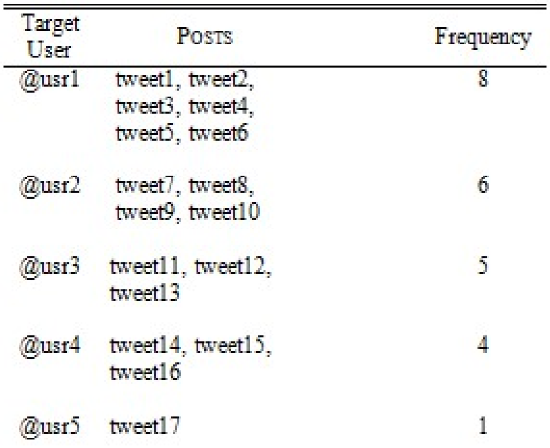
The second index, as shown in table (2), represents the internal tweets that transform users as followers. Users often post a message for an activity several times within a short period, and some messages hold links for temporal references presented in other social media such as Facebook, Instagram, or Pinterest. Such tweets were ranked in the second category in the resulting list and following the category of external URLs, as shown in Section 10.

Table: Statistics of External Tweets Index Table



Target URL	Posts	Frequency
URL1	tweet1, tweet3, tweet4, tweet7, tweet8, tweet15	6
URL2	tweet2, tweet5, tweet9, tweet13	4
URL3	tweet6, tweet10, tweet11	3
URL4	tweet14, tweet16, tweet19	3
URLn	Tweet17	2

Table 2: Statistics of Internal Tweets Index Table



Target User	Posts	Frequency
@usr1	tweet1, tweet2, tweet3, tweet4, tweet5, tweet6	8
@usr2	tweet7, tweet8, tweet9, tweet10	6
@usr3	tweet11, tweet12, tweet13	5
@usr4	tweet14, tweet15, tweet16	4
@usr5	tweet17	1

5. TWEET FEATURES SELECTION

In this section, we present an overview of the different features that are selected by the SAMA search engine approach. We found two types of features: topic-insensitive features that were evaluated before a query is issued and characterized as syntactical, semantic or contextual properties of a Twitter message, and topic-sensitive features that were computed at query time and provide the relevance of a Twitter message with respect to the query.

5.1. Topics-Insensitive Features

- **Total number of tweets:** The total number of tweets a person has posted, reposted, or replied to in a period.
- **Directed communications:** The number of tweets symbolized by @, which indicate interpersonal activities between the user and others in a period.
- **Hashtag communications**: This is defined as the ratio of tweets that contain at least one hashtag (#), to the total number of tweets from a person in a period.
- **Reformer tweets:** This is the ratio of users who share tweets about themselves to the total number of tweets by a person in a period. If a tweet contains any of the 24 self-referencing pronouns (e.g. I, me, we, us) then it is classified as a Reformer tweet and discarded
- **Informer tweets:** We identified informer tweets as those containing any of the person pronouns (e.g. he, she, it, them). This is computed as the ratio of informer tweets to the total number of tweets by a person in a specific period.
- **Tweet length**: The number of characters in a tweet may also be an indicator of the relevance. We hypothesize that the length of a Twitter message correlates with the amount of information it conveys.
- **Special Characters:** whether the tweet contains a question mark, exclamation mark, smile or frown.

5.2. Topics-Sensitive Features

Query processing is an essential phase to find the topic of a query [5]. It includes detecting the type of query, query searching, query normalization and query expansion. In our view, Twitter does not always perform effective ranking, so to calculate the retrieval score for a pairing of topic and tweet and filtering non-relevant tweets, we employ the coefficient similarity model.

For each URL, given a query sequence X=$x_1 \ldots x_n$ and a tweet sequence Y=$y_1 \ldots y_m$, we can say the tweet Y is highly relevant if there is a strictly increasing sequence $i_1 \ldots i_m$ of indices of Y, such that for all j=1… m, we have $x_{ij}=x_j$.

Similarly, given two sequences of tweet X and tweet Y, we can say that a query sequence Q is a common subsequence of X and Y if Z is a subsequence of both X and Y. Thus, the vector will be ranked high.

Different contexts were used for validating the query terms in each vector [8], with each context assigned a different weight. Since we have a merged list for each URL, we process all tweets as one document. To compute the weight of each URL represented by its tweets, we used the coefficient similarity between query Q and a tweet vector represented by a URL in Table (1) and Table (2). To impose the mentioned features, we used TF-IDF coefficient similarity which is weighted by proximity of the query keywords in the tweet. The intuition is that a tweet that contains most of the query terms is more relevant to the query than a tweet that contains fewer of the query terms. Proximity of the query keywords in the tweet is a very important feature when judging the relevance due to the low likelihood of repeating of query terms in the tweet, as follows:

$$Similarity(Q,D) = \frac{n(Q \cap D)}{\sqrt{n(D).n(L)}} \qquad (1)$$

where Q denotes a query string, D represents a tweet vector, $n(D)$ is the number of tweets belonging to the same URL and $n(L)$ is the number of tweets in the entire vector. We try to normalize TF-IDF similarity score by exponentially decaying the TF-IDF similarity based on the proximity of the query terms in the tweet for verbosity of the final weight, as follows:

$$Rank_{Q,D} = 2(1 - Similarity(Q,D)) \qquad (2)$$

The following algorithm was used to rank our tweets collection:

Inputs: Individual tweets from primary vertical URL

Output: Aggregated result (ranked list)

```
// federated Tweets T;    // federated Score fs
// federated Vector (V);  // verticals selected based on fs($V_i$)
forEach (V) read (T) ; n ← n + 1
sortVerticals.add($T_n$); end
for n=1 to len(T) do
if Vector (n)> fs then aggregate.List[j] = Vector (n)
j ← j + 1; end
for i=1 to j
rankList = aggregate.list($V_i$)
if (rankList) then sorted(rankList) → list.add(i)
else Remove ($V_i$). end
return RankList
```

In terms of query expansion, researchers claim that optimizing query terms will help to find relevant results [9]. As we explained previously[2], Wikipedia articles are connected together to form high diversification and complete topics, and Wikipedia writers join similar or related articles using inter-links. Similarly, Wikipedia articles can extend current articles to others using shared-links, and a target article often points back to a source article.

If we assume article (A) in Wikipedia has a link that points to article (B), and article (B) has a link that points back to article (A), then articles A and B are topically related. We used this to collect article forward names and backward names in one query expansion list. Similarly, Wikipedia writers often use different strategies to title articles, so highly similar articles can have different titles and similar content. We used these variations to expand queries; if the initial ranked list contains results from Wikipedia, the title of article will be used to collect other corresponding results. For example, Wikipedia writers entitle the article 'Lipoma' as 'Fatty Tumor, 'Fatty Lipoma', Lypoma', 'Lipomatous Neoplasm', 'Lipomas' and 'Lipomatosis'. Thus, if a user queries the system and the system returns a Wikipedia document relevant to the query, then the title of the document will be used to expand the query by calling other titles.

However, in our model, query expansion and reformulation was not applied to all queries; only when the initial ranking list was short and the initial query returns at least one Wikipedia document as relevant. For example, the initial result list for the query 'angular chelitis' is too short; using the expansion query 'angular stomatitis' will increase the number of results.

6. EXPERIMENTAL EVALUATION

Topics in information retrieval specified in Twitter posts are evaluated for two tasks: adhoc,

[2] https://trec.nist.gov/pubs/trec21/papers/uottawa.web.final.pdf

and diversity. The diversity retrieval task is similar to the adhoc task, except in its judging process and evaluation metrics. The goal of the diversity task is to return a ranked list of pages that, together, provide complete coverage for a query while avoiding excessive redundancy in the result list. The relevance judgment and primary effectiveness measure for both tasks are determined by calculating the graded precision on the top ten results or the top k (P@k). Documents can be judged Nav (navigational), Key (top relevant), HRel (highly relevant), Rel (minimally relevant) or Non (not relevant). Researchers [2, 4, 6] showed that the relevancy of Twitter results can be determined by calculating the graded precision in subset result. The Normalized Documents Cumulative Gain (NDCG) metric maximizes the relevancy by evaluating all ranks together; so the graded relevance levels of the results in the top ten must be considered. The normalized CG and DCG metrics clearly show the share of ideal performance given by the IR technique, and make statistical comparisons possible [21]. Similarly, the relevance of results for a given query is determined by the best performing Twitter searcher, which means the relevance of a vertical is represented by the maximum graded precision of its results.

Our runs were submitted to the adhoc and diversity tasks were judged according to the judging criteria of both tasks. This additional judging allows us to make direct comparisons between runs optimized for the two tasks, supporting efforts to determine if the different judging criteria and evaluation measures identify genuine differences. For example, figure 2, 3, and 4 provide a scatter comparing the performance of the runs under nDCG@1, nDCG@10, and nDCG@20 the primary effectiveness measures for the adhoc and diversity tasks respectively. While the values are correlated, there are clear differences in the relative performance of runs under the three measures.

However, for the final evaluation the binary relevance of a topic is determined by a threshold; we assumed the minimum graded precision of 0.5 was relevant. The threshold was determined because some queries, particularly navigational, target small sets of relevant topics. If no topics have exceeded the threshold, the top topic with the maximum relevance is selected as the relevant topic. For training and testing our model, 5% of the gold standard query topics were randomly picked from Million Query Track3 for training the model and another 5% to test the trained model (the remaining data is reserved for the experiments4). Since we do not like to penalize tweets that do not contain external URLs, or user information that we were unable to collect, we impute the missing feature values using population average. We normalize the Feature Score to lie between 0 and 1.

As shown in Tables (3) and (4), the Mean Average Precision is high and identical, though for a few queries (not listed) the precision is low as the type of queries was unpopular and they were selected from specific individual resources., In information retrieval systems, the relevancy of documents generally depends on the user's perspective. It is difficult to determine all relevant documents that satisfy all users' needs in one relevancy judgment, since users might have different points of view at different times. Based on TREC evaluation our previous approach included two tasks, and the graded precision of 0.41 was the best of all approaches. We listed the precision at (1), (5), (10) and (20). The tested and training sets of queries involved both tasks and different relevance complexities. Figures 2, 3, and 4 show the Normalized Cumulative Gain metric per topic at "1", "10", and "20", respectively, using 200 topics for training. As we can see, we have substantial and significant results for both tasks over all previous runs; for instance. A Twinder search engine [24] got a precision for all selected features ranged between "0.1956 " and "0.3827". The author(s) did not mention in which score the precision was calculated; but any ways, our precisions at all scores are better and greater. However, little is understood about Twitter search engine itself and its ranking algorithm; but searching a query in Twitter will results lots of trash and irrelevant results that obviously presented based on different issues.

[3] https://trec.nist.gov/data/million.query09.html
[4] https://trec.nist.gov/data/web/12/queries.151-200.txt

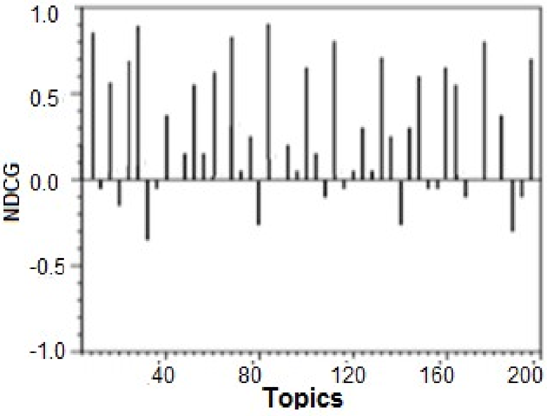


Figure 2: Normalized Discounted Gain @1 per topic

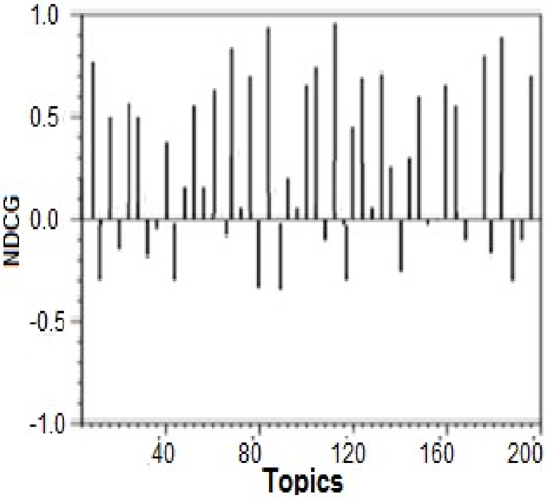


Figure 3: Normalized Discounted Gain @10 per topic

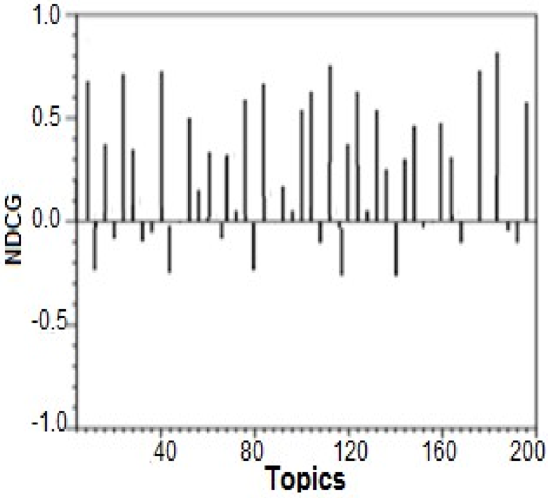


Figure 4: Normalized Discounted Gain @20 per topic

Table 3: The precisions in our run using Diversity task

Run ID	P@ 1	P@ 5	P@ 10	P@ 20
Sama	0.61	0.57	0.51	0.78

Table 4: The precisions in our run using adhoc task

Run ID	P@ 1	P@ 5	P@ 10	P@ 20
Sama	0.77	0.63	0.64	0.58

7. ANALYSIS OF TOPICS IN TWITTER

We identified all tweets that included an URL as a link to a website6 as an external reference. In Twitter, URLs are often abbreviated with a shortened URL, such as Bit.ly. Shortened URLs were resolved to create a list of all URLs in the datasets, and a basic categorization scheme was developed to classify the types of websites the URLs in tweets are pointing to. Each URL in Twitter was classified according to either *Blog or Post* schemes. This category is used for all types of blogs and blog posts, as well as private commentaries on personal websites. The *Facebook* category is used to link to content in Facebook, and if a URL could not be accessed it was marked as an *Error* with this category. The *Media* category was applied for all types of multimedia data, including photos, videos, other types of visualizations and graphics. *Spam* refers to non-official content and the link is forwarded to pornographic content. The *Advertisement* category is used to display advertisements for tools, platforms and houses, while the *Scholar* category is used to publish scientific journals and information about a conference. *Home-Page* category is used for official websites; that is, the home pages of external domains. The *Slides* category is used for links to presentation slides, such as presentation sharing platforms like Slideshare. *Location* points to addresses or locations on maps. The *Twitter* category is comprised of links to subpages of Twitter, such as Twitter profiles and Twitter-related websites. The *Other* category is not specific, and includes everything that does not belong to the other categories. Figures 5, 6, and 7 show the categories of tweets in different years.

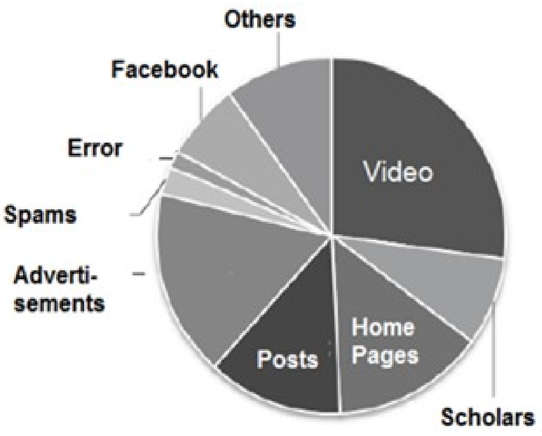


Figure 5: Categories of URLS 2015 using ONE MILLION QUERY TRACK

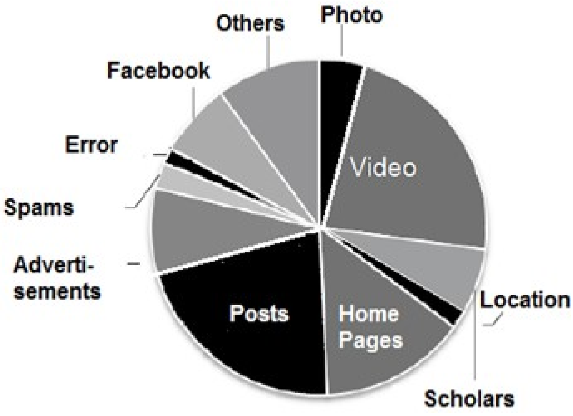


Figure 6: Categories of URLS 2016 using ONE MILLION QUERY TRACK

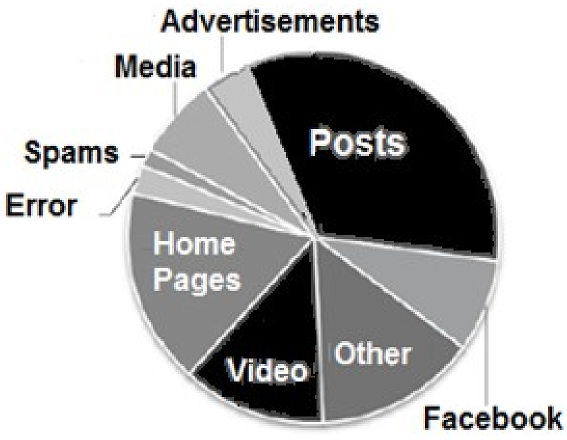


Figure 7: Categories of URLS 2017 using ONE MILLION QUERY TRACK

8. EXPERIMENTAL SETUP

SAMA is a real-time, retrieval-based engine used by Wikipedia and Twitter. It is built on the open-source Apache Lucene search engine[5], and adapted to meet the requirements discussed in Section II. SAMA is specifically designed to handle all real-time content, including content from social media such as Twitter and Facebook. We implemented a few enhancements that could be applicable to the general case. SAMA exists in the architecture shown in Figure (7) that depicts Twitter real-time search. In 2015, Google made many of the top Tweets searchable, and its search results are most likely to come up in the top few lines of instant query answers.

Tweets enter the ingestion pipeline, where they are tokenized and annotated with additional metadata (e.g. language). To manage large volumes the tweets are hash partitioned across SAMA servers, which indexes the tweets immediately after they have been processed. The search service performs relevance filtering and personalization instantly, using three types of signals:

- ○ *Static signals* are directly added at indexing time.
- ○ *Resonance signals* are dynamically updated over time.
- ○ *Informational signals are* dynamically added and provided at search time.

A component in SAMA pushes dynamic resonance signals to servers to make the contents seem current. At query time, our front-end server query processor parses a user's query and passes it to multiple SAMA servers with the user's local social graph. The servers use a ranking function that combines relevance signals and a user's local social graph to compute a personalized relevance score for each tweet. The highest-ranking, most-recent tweets are returned to the front-end processor, which merges and re-ranks the results before returning them to the front-end user. In production, SAMA servers receive load from the front ends while simultaneously indexing new tweets and other contents from the ingestion pipeline. Overall, we typically observe a ten second

[5] https://apache.lucene.org

indexing latency (i.e. from tweet creation time to a searchable tweet), and about a 50 ms query latency.

The total distributed search architecture for all our big datasets is beyond the scope of this paper; for an informal description, we refer the reader to a Twitter real-time engineering blog post. SAMA simply returns a list of tweets that do not satisfy a Boolean query. SAMA is written in Java, Javascript and C-sharp, primarily for three reasons: to take advantage of the existing Lucene Java codebase, to fit into Twitter's JVM-centric development environment, and to take advantage of the easy-to-understand memory model for concurrency offered by Java and the JVM. Although this decision has inherent performance challenges, with careful engineering and memory management we believe it is possible to build systems that are comparable to real-time performance. SAMA, like most modern retrieval engines, maintains its inverted index concurrently. Postings remain in the front-end before being forwarded to the user in forward chronological order (most recent last), then are traversed backwards (most recent first); this is accomplished by maintaining a pointer to the current end of each posting list. We believe this index organization is an interesting and noteworthy aspect of SAMA. Our model supports different query lengths: one term, two term and phrase, and results are returned in reverse chronological order (most recent first). Query evaluation is relatively straightforward, and we were able to use our Lucene query model algorithm for conjunctive queries corresponding to intersections of posting lists, disjunctive queries corresponding to unions and phrase queries corresponding to intersections with positional constraints. Lucene provides abstractions for posting lists and traversing postings, and we provide implementation for our custom indexes; otherwise, we can reuse existing Lucene query evaluation code. The actual query evaluation algorithm isn't particularly interesting, but how we handle concurrency (i.e. concurrent index reads and writes) in a multi-threaded framework is certainly worth discussing. The following section addresses this.

9. SYNCHRONIZATION MANAGEMENT

An important component of real-time searching is the ability to handle index writes (ingest new tweets) and index reads (query processing) synchronously in multi-threaded environment. However, this only applies to the active index segment ingesting tweets, as he other index segments are read-only and do not have concurrency induced issues: multiple query processing threads can traverse postings concurrently. The complex problem of concurrent index reads and writes can be simplified by limiting writes to a single thread; that is, one writer thread ingests tweets and updates the inverted index; but queries are usually concurrently processed on separate index reader threads. In this context, it is important that index reader threads are presented with an up-to-date and consistent view of the index structures. Though this can achieved through synchronization mechanisms, there is typically a trade-off between the degree of synchronization and performance.

Too much synchronization hinders performance, and too little can lead to inconsistent or incorrect results. Thus, determining the correct balance is perhaps the most difficult aspect of concurrent programming. Fortunately, Java provides a memory model for concurrency, which we made use of. The indexing of a new tweet by a single index writer thread proceeds as follows: The model first looks up the corresponding dictionary entry for each term in the tweet. The terms are then mapped to term ids in the dictionary, where they serve as indices into the parallel arrays holding the term data. If the pointer is at the tail of the current posting list a new posting is added, and if there isn't enough space additional slices are allocated. If a term has never been encountered it is added to the dictionary and assigned the next available term id, and a slice is allocated in the first pool for the new posting. In both cases (i.e. existing or new term), the term occurrence count and the tail pointer of the postings list are then updated. Once all terms in a tweet have been processed we increment the posting variable that holds the largest current document id encountered, indicating that the tweet has been successfully ingested. Concurrent queries are handled by separate threads (one per thread). Each query evaluation thread begins by reading the posting variable, then looking up the posting list tail pointer corresponding to each term. These are used to initialize the appropriate posting list abstractions in Lucene, after which query processing begins.

There are two important aspects to the SAMA consistency model. First, individual posting lists must always be internally consistent. For example, the tail pointers of the posting lists should always be valid. Secondly, consistency must be maintained across postings lists; an index reader should see a consistent view of the index structures up to a certain, well-defined point. Thus, SAMA guarantees search correctness over all tweets with document ids to be less than or equal to the posting at the point when the index reader begins query processing.

Maintaining consistency in a multi-threaded environment is challenging [11], as thread execution can occur in arbitrarily interleaved orders, and there is no guarantee when memory writes from one thread are visible to another thread. A potential solution is to make tweet indexing an atomic operation, so the index reader threads are guaranteed to see consistent, up-to-date index structures. However, this level of synchronization overly decreases performance, and it is not practical for the volume of tweets we need to handle.

10. EXPERIMENTAL RESULT

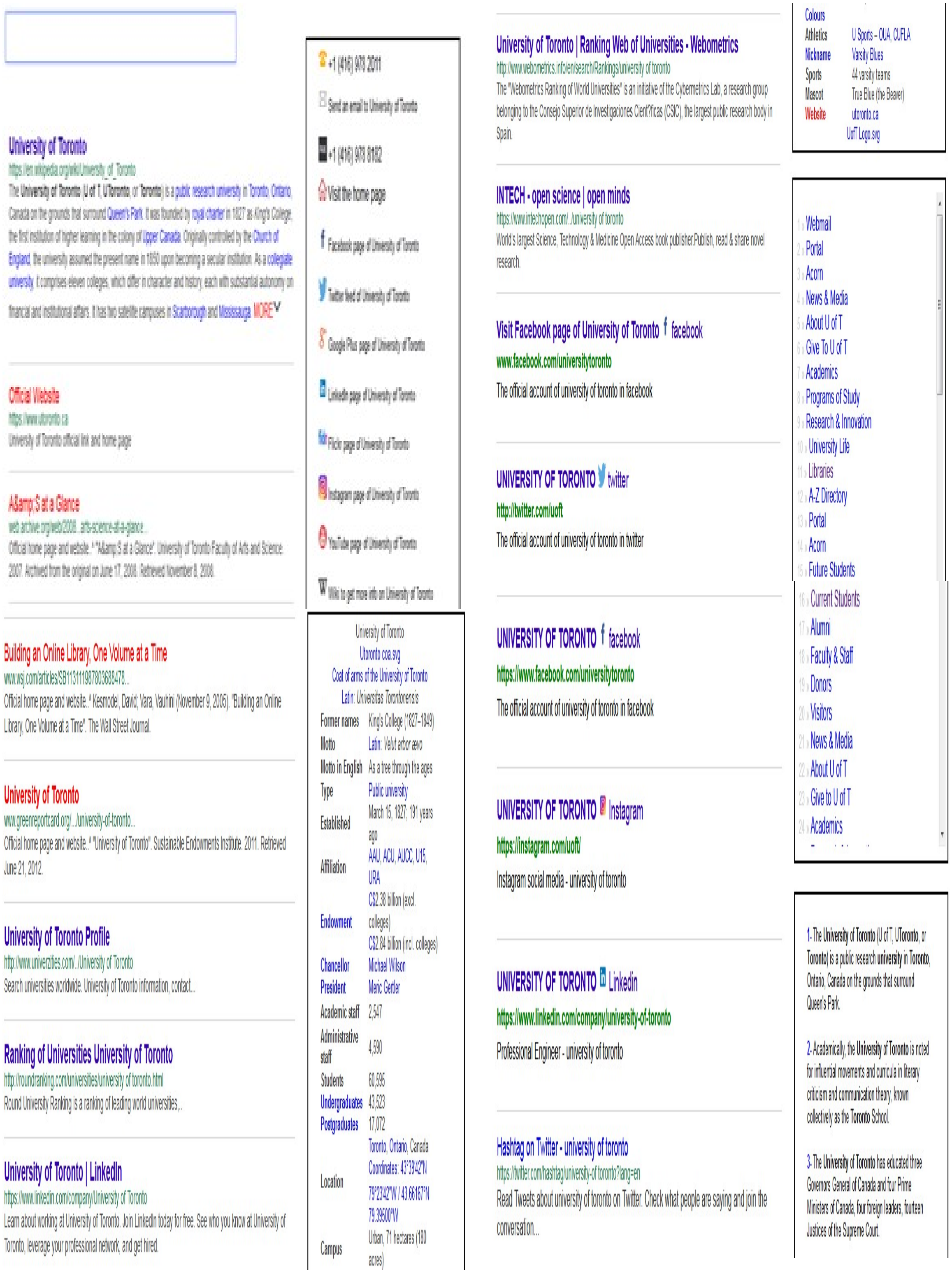
Typically, users of Web search engines must sift through long ordered lists of document snippets returned by their searches. The search engine community has explored document grouping as an alternative method to organize retrieval results, but this has yet to be deployed on most major search engines. Some search engines, such as Yahoo, organize their output into custom blocks based on categorized document levels. However, this does not indicate how the categories are created, or how well they correspond to users' interests [22]. In this section, we finalize our approach by introducing a new interface to our approach that dynamically groups retrieved results into blocks according to user preferences.

Information seeking is inherently imprecise, because when users launch search systems they often have only minimal understanding of how they can best achieve their goals. This issue will only increase when the model deals with Twitter. Since the average length of search queries submitted to Twitter search engines is lower than in a traditional Web search, we need to understand the information behind the query.

Our model infrastructure is designed to be scalable by processing intensive tasks, so a user interface would help them recognize what kind of results they can retrieve to address their information needs. Previously, very little was known about what makes an effective search result interface, but now there is information about which method works best from a usability perspective. Result presentation has been thoroughly investigated as a post retrieval document visualization technique [22].

Currently, traditional search engines such as Google show only the posts sent by members, not the external links involved in the posts. Our model shows the posts and external references in different groups, with each group ranked differently. However, our search result interface, as shown in figure 8, groups the final list of documents according to their hosts in the index of references and based on document topics. In this section of the paper, we show the section that belongs to the twitter results, embedded with other results which are beyond the scope of this paper.

As we can see in figure (8), some results are presented as references because some posts are involved and boosted by external links; likewise, some results are only involved textual posts without any external links that target some external Web pages. Often, referenced results are very important for some distinctive queries; for instance, a query "trombone for sale", the results that cited by Twitter users are more relevant than the classical results in traditional search engines. User name are presented to certify the poster and to help and influence the user to realize and recognize postings and tweets because not all references are officially trusted. For example, a poster by CTV News which indicated the citation is very high relevant since the poster shows some curious knows about the University of Toronto. However, not all results in our interface are presented directly from Twitter; but alternatively, they derived from some results as home page, if available. In terms of user preference, we organized the interface based on the type of a query; for example, results for queries events, shopping, news... etc. are preferred to be on the top rather than on the bottom.



University of Toronto
https://en.wikipedia.org/wiki/University_of_Toronto
The University of Toronto (U of T, UToronto, or Toronto) is a public research university in Toronto, Ontario, Canada on the grounds that surround Queen's Park. It was founded by royal charter in 1827 as King's College, the first institution of higher learning in the colony of Upper Canada. Originally controlled by the Church of England, the university assumed the present name in 1850 upon becoming a secular institution. As a collegiate university, it comprises eleven colleges, which differ in character and history, each with substantial autonomy on financial and institutional affairs. It has two satellite campuses in Scarborough and Mississauga. MORE

Official Website
https://www.utoronto.ca
University of Toronto official link and home page

A&amp;S at a Glance
web.archive.org/web/2008...arts-science-at-a-glance...
Official home page and website.. ^ "A&amp;S at a Glance". University of Toronto Faculty of Arts and Science. 2007. Archived from the original on June 17, 2008. Retrieved November 8, 2008.

Building an Online Library, One Volume at a Time
www.wsj.com/articles/SB113111987803688478...
Official home page and website.. ^ Kesmodel, David; Vara, Vauhini (November 9, 2005). "Building an Online Library, One Volume at a Time". The Wall Street Journal.

University of Toronto
www.greenreportcard.org/.../university-of-toronto...
Official home page and website.. ^ "University of Toronto". Sustainable Endowments Institute. 2011. Retrieved June 21, 2012.

University of Toronto Profile
http://www.univerzities.com/../University of Toronto
Search universities worldwide. University of Toronto information, contact...

Ranking of Universities University of Toronto
http://roundranking.com/universities/university of toronto.html
Round University Ranking is a ranking of leading world universities,..

University of Toronto | LinkedIn
https://www.linkedin.com/company/University of Toronto
Learn about working at University of Toronto. Join LinkedIn today for free. See who you know at University of Toronto, leverage your professional network, and get hired.

☎ +1 (416) 978 2011

✉ Send an email to University of Toronto

+1 (416) 978 8182

Visit the home page

Facebook page of University of Toronto

Twitter feed of University of Toronto

Google Plus page of University of Toronto

LinkedIn page of University of Toronto

Flickr page of University of Toronto

Instagram page of University of Toronto

YouTube page of University of Toronto

Wiki to get more info on University of Toronto

University of Toronto
Utoronto coa.svg
Coat of arms of the University of Toronto
Latin: *Universitas Torontonensis*

Former names	King's College (1827–1849)
Motto	Latin: *Velut arbor ævo*
Motto in English	As a tree through the ages
Type	Public university
Established	March 15, 1827; 191 years ago
Affiliation	AAU, ACU, AUCC, U15, URA
Endowment	C$2.38 billion (excl. colleges) C$2.84 billion (incl. colleges)
Chancellor	Michael Wilson
President	Meric Gertler
Academic staff	2,547
Administrative staff	4,590
Students	60,595
Undergraduates	43,523
Postgraduates	17,072
Location	Toronto, Ontario, Canada Coordinates: 43°39′42″N 79°23′42″W / 43.66167°N 79.39500°W
Campus	Urban, 71 hectares (180 acres)
Colours	
Athletics	U Sports – OUA, CUFLA
Nickname	Varsity Blues
Sports	44 varsity teams
Mascot	True Blue (the Beaver)
Website	utoronto.ca

UofT Logo.svg

University of Toronto | Ranking Web of Universities - Webometrics
http://www.webometrics.info/en/search/Rankings/university of toronto
The "Webometrics Ranking of World Universities" is an initiative of the Cybermetrics Lab, a research group belonging to the Consejo Superior de Investigaciones Cient?ficas (CSIC), the largest public research body in Spain.

INTECH - open science | open minds
https://www.intechopen.com/../university of toronto
World's largest Science, Technology & Medicine Open Access book publisher.Publish, read & share novel research.

Visit Facebook page of University of Toronto facebook
www.facebook.com/universitytoronto
The official account of university of toronto in facebook

UNIVERSITY OF TORONTO twitter
http://twitter.com/uoft
The official account of university of toronto in twitter

UNIVERSITY OF TORONTO facebook
https://www.facebook.com/universitytoronto
The official account of university of toronto in facebook

UNIVERSITY OF TORONTO Instagram
https://instagram.com/uoft/
Instagram social media - university of toronto

UNIVERSITY OF TORONTO Linkedin
https://www.linkedin.com/company/university-of-toronto
Professional Engineer - university of toronto

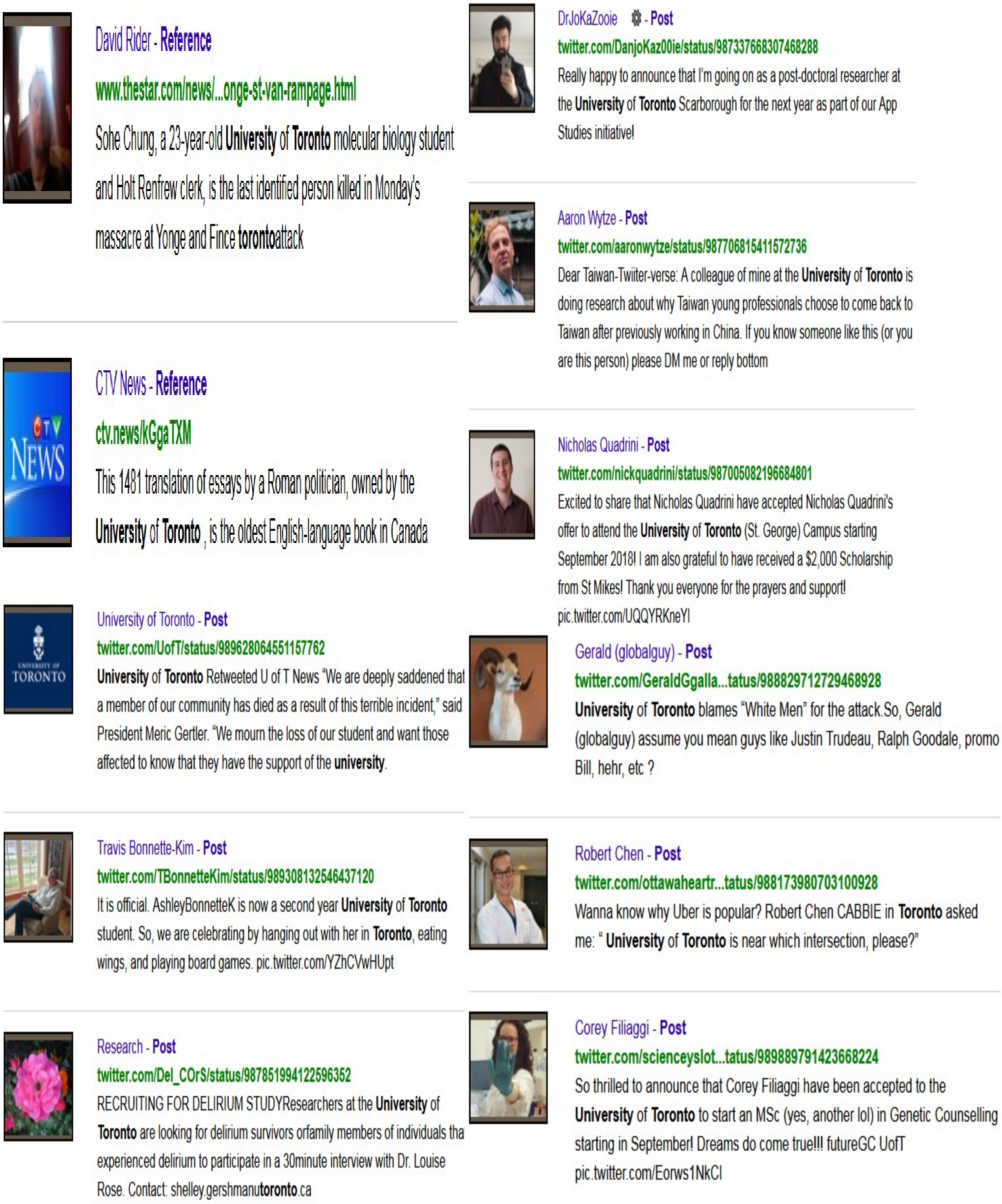
Hashtag on Twitter - university of toronto
https://twitter.com/hashtag/university-of toronto?lang=en
Read Tweets about university of toronto on Twitter. Check what people are saying and join the conversation...

1 » Webmail
2 » Portal
3 » Acorn
4 » News & Media
5 » About U of T
6 » Give To U of T
7 » Academics
8 » Programs of Study
9 » Research & Innovation
10 » University Life
11 » Libraries
12 » A-Z Directory
13 » Portal
14 » Acorn
15 » Future Students
16 » Current Students
17 » Alumni
18 » Faculty & Staff
19 » Donors
20 » Visitors
21 » News & Media
22 » About U of T
23 » Give to U of T
24 » Academics

1- The **University** of **Toronto** (U of T, U**Toronto**, or **Toronto**) is a public research **university** in **Toronto**, Ontario, Canada on the grounds that surround Queen's Park.

2- Academically, the **University** of **Toronto** is noted for influential movements and curricula in literary criticism and communication theory, known collectively as the **Toronto** School.

3- The **University** of **Toronto** has educated three Governors General of Canada and four Prime Ministers of Canada, four foreign leaders, fourteen Justices of the Supreme Court.



David Rider - **Reference**

www.thestar.com/news/...onge-st-van-rampage.html

Sohe Chung, a 23-year-old **University** of **Toronto** molecular biology student and Holt Renfrew clerk, is the last identified person killed in Monday's massacre at Yonge and Fince **toronto**attack

CTV News - **Reference**

ctv.news/kGgaTXM

This 1481 translation of essays by a Roman politician, owned by the **University** of **Toronto** , is the oldest English-language book in Canada

University of Toronto - **Post**

twitter.com/UofT/status/989628064551157762

University of **Toronto** Retweeted U of T News "We are deeply saddened that a member of our community has died as a result of this terrible incident," said President Meric Gertler. "We mourn the loss of our student and want those affected to know that they have the support of the **university**.

Travis Bonnette-Kim - **Post**

twitter.com/TBonnetteKim/status/989308132546437120

It is official. AshleyBonnetteK is now a second year **University** of **Toronto** student. So, we are celebrating by hanging out with her in **Toronto**, eating wings, and playing board games. pic.twitter.com/YZhCVwHUpt

Research - **Post**

twitter.com/Del_COrS/status/987851994122596352

RECRUITING FOR DELIRIUM STUDYResearchers at the **University** of **Toronto** are looking for delirium survivors orfamily members of individuals tha experienced delirium to participate in a 30minute interview with Dr. Louise Rose. Contact: shelley.gershmanu**toronto**.ca

DrJoKaZooie ⚙ - **Post**

twitter.com/DanjoKaz00ie/status/987337668307468288

Really happy to announce that I'm going on as a post-doctoral researcher at the **University** of **Toronto** Scarborough for the next year as part of our App Studies initiative!

Aaron Wytze - **Post**

twitter.com/aaronwytze/status/987706815411572736

Dear Taiwan-Twiiter-verse: A colleague of mine at the **University** of **Toronto** is doing research about why Taiwan young professionals choose to come back to Taiwan after previously working in China. If you know someone like this (or you are this person) please DM me or reply bottom

Nicholas Quadrini - **Post**

twitter.com/nickquadrini/status/987005082196684801

Excited to share that Nicholas Quadrini have accepted Nicholas Quadrini's offer to attend the **University** of **Toronto** (St. George) Campus starting September 2018! I am also grateful to have received a $2,000 Scholarship from St Mikes! Thank you everyone for the prayers and support! pic.twitter.com/UQQYRKneYI

Gerald (globalguy) - **Post**

twitter.com/GeraldGgalla...tatus/988829712729468928

University of **Toronto** blames "White Men" for the attack.So, Gerald (globalguy) assume you mean guys like Justin Trudeau, Ralph Goodale, promo Bill, hehr, etc ?

Robert Chen - **Post**

twitter.com/ottawaheartr...tatus/988173980703100928

Wanna know why Uber is popular? Robert Chen CABBIE in **Toronto** asked me: " **University** of **Toronto** is near which intersection, please?"

Corey Filiaggi - **Post**

twitter.com/scienceyslot...tatus/989889791423668224

So thrilled to announce that Corey Filiaggi have been accepted to the **University** of **Toronto** to start an MSc (yes, another lol) in Genetic Counselling starting in September! Dreams do come true!!! futureGC UofT pic.twitter.com/Eorws1NkCI

Figure 8: Our model interface

11. RELATED WORKS

Twitter has been launched for more than one decade and involved lots of researches among the research community. Researchers initially studying micro-blogging streams to predict the type of posts discussed on Twitter [26]. This issue is also demonstrated by our proposed approach; in which, it is involved two types of posts (internal blogs and external references). Others researchers evolved trends [27]; or detecting influential users on Twitter [28]. Twinder [24] is another example of search engine for Twitter stream used number of features that were proposed by previous research [29]. However, the selected investigative features showed that improving recall and precision were possible to make high impact on retrieval performance. The researchers also developed a number of novel semantic measures to further boost their retrieval effectiveness. Researchers imply some applications to utilize the blogs and to enrich traditional news media with information from Twitter [11], or detecting unpredictable events such as earthquakes [12]. However, posts in twitter are used rarely to enhance or to improve search relevancy. Tevaan et al. [30] compared the search behaviour on Twitter with traditional Web search behaviour as discussed in the introduction. Bernstein et al. [3] proposed an interface that allows for exploring tweets by means of tag clouds. However, their interface is targeted towards browsing the tweets that have been published by the people whom a user is following and not for searching the entire Twitter corpus. Jadhav et al. [31] developed an engine that enriches the semantics of Twitter messages and allows for issuing SPARQL queries on Twitter streams. RAProp, algorithm proposed by Srijith et a. [23] combined two orthogonal features of trustworthiness: trustworthiness of source and trustworthiness of content, in order to filter out irrelevant results and spam. RAProp works by computing a Feature Score for each tweet and propagating that over a graph that represents content-based agreement between tweets, thus leveraging the collective intelligence embedded in tweets. RAProp improved the precision of the returned results significantly over the baselines in both mediator and non-mediator models. Semantic search enriched a strategy to provide faceted search capabilities on Twitter [32]. Others [33] investigate features such as Okapi BM25 relevance scores or Twitter specific features (length of a tweet, presence of a hashtag, etc.) in combination with RankSVM to learn a ranking model for tweets. In an empirical study, they found that the length of a tweet and information about the presence of a URL in a tweet are important features to rank relevant tweets. Tagging in Twitter is also important, researchers showed that they specifically used for filtering and directing content so that it appears in certain streams [34].

12. CONCLUSION

The goal of this paper is to create a real-time indexing approach that uses Twitter posts to facilitate finding another source of relevant documents. As of right now, posts in twitter are used rarely by the traditional search engines to enhance or to improve their search relevancy. We introduced a SAMA search engine, which analyzes various features to determine the relevance and value of Twitter messages for a given topic, and demonstrated the scalability of our model. We extensively investigated two features: topic-sensitive and topic-insensitive, and gained insights into the importance of these features for retrieval effectiveness. Adding external links to tweets often improves the SEO of user tweets and the target webpage. However, one of the advantages that obtained by SAMA model is embedding real-time tweets and posts as well as its external-links with our regular Web results that showed through our experiments are very relevant.

We used different indexing algorithms that exploit queries and Twitter posts, and we believe our approach will enhance and improve the design and functionality of future Web search engines. In this work, we build a model that can identify relevant documents from Twitter collections, using various representation techniques based on document content, query structures (i.e. the variation of query terms occurring in the tweet content based on query structures), and applicable knowledge available in Twitter. We demonstrated that Twitter post structure and content provide valuable information that helps determine relevant data in Web collections. We find relevant pages based on link impact, a novel idea that improves term and inverse document frequency, as well as other similar weighting schemes used by traditional systems. One drawback we faced in our approach that not all external references are relevant but some rare pages are categorized spams, advertisements, errors or so-called black-hat.

ACKNOWLEDGMENTS

This work was collaborated between the University of Ottawa and the University of Kufa who developed "SAMA[6]" search engine. An earlier version of this algorithm appeared in the Proceedings of the Text Retrieval Conference (TREC 2011 and TREC 2012) in Gaithersburg, Maryland, USA. Many thanks to Twitter for its contribution, help, and providing its data collection. We also wish to thank the Text Retrieval Conference for providing us with training and testing data, as well as their support and positive criticism. We gratefully acknowledge the assistance of Professor Diana Inkpen and the IT support staff at the University of Ottawa for extending our research project. Finally, many thanks to our anonymous reviewers for their useful ideas and valuable support.

REFERENCES

[1] M. Milajevs, and G. Bouma,_Real Time Discussion Retrieval from Twitter‘. In Proceedings of the International World Wide Web Conference Committee (IW3C2). 2013.

[2] M. V. Vieira, B. M. Fonseca, R. Damazio, P. B. Golgher, D. de Castro Reis, and B. Ribeiro-Neto, "Efficient search ranking in social networks," in Proceedings of the Sixteenth International Conference on Information and Knowledge Management (CIKM 2007), Lisbon, Portugal, 2007, pp. 563–572.

[3] D. Horowitz and S. D. Kamvar, "The anatomy of a large-scale social search engine," in Proceedings of the 19th International World Wide Web Conference (WWW 2010), Raleigh, North Carolina, 2010, pp. 431–440.

[4] T.-Y. Liu, "Learning to rank for information retrieval," Foundations and Trends in Information Retrieval, vol. 3, no. 3, 2009, pp. 225–331.

[5] J. M. Kleinberg, "Authoritative sources in a hyperlinked environment," Journal of the ACM, vol. 46, no. 5, 1999, pp. 604–632.

[6] L. Wang, J. Lin, and D. Metzler, "A cascade ranking model for efficient ranked retrieval," in Proceedings of the 34th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2011), Beijing, China, 2011, pp. 105–114.

[7] V. N. Anh, O. de Kretser, and A. Moffat, "Vector-space ranking with effective early termination," in Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2001), New Orleans, Louisiana, 2001, pp. 35–42.

[8] V. N. Anh and A. Moffat, "Simplified similarity scoring using term ranks," in Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2005), Salvador, Brazil, 2005, pp. 226–233.

[9] H. Turtle and J. Flood, "Query evaluation: Strategies and optimizations," Information Processing and Management, vol. 31, no. 6, 1995, pp. 831–850.

[10] A. Ritter, S. Clark, Mausam, and O. Etzioni, "Named entity recognition in Tweets: An experimental study," in Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing (EMNLP 2010), Edinburgh, Scotland, 2011, pp. 1524–1534.

[11] M. Busch, K. Gade, B. Larson, P. Lok, S. Luckenbill, J. J. Lin, Earlybird: Real-Time Search at Twitter, 2012 IEEE 28th International Conference on Data Engineering.

[12] M. Shokouhi and L. Si, Federated Search, Foundations and Trends in Information Retrieval, pp. 1–102, 2011.

[13] WANG, Xun; ZHU, Feida; JIANG, Jing; and LI, Sujian. Real Time Event Detection in Twitter. (2013). In Proceedings of the 14th International Web-age information management Conference, pp.502-513, 2013 Proceedings.

[14] J. Arguello, F. Diaz, M. Shokouhi, Integrating and Ranking Aggregated Content on the WebUNC Chapel Hill,Yahoo! Labs, Microsoft Research, 2012.

[15] D. Sentiment, Mining within Social Media for Topic Identification, IEEE Fourth International Conference on Semantic Computing, 2010, pp. 394 – 401.

[16] Lei-Lei Shi; Yan Wu; Lu Liu; Xiang Sun; Liang Jiang, Event Detection and Key Posts Discovering in Social Media Data Streams, International Conference on Internet of

[6] http://eecs.uottawa.ca/~falak081

Things (iThings) and IEEE , 2017, pp. 1046 – 1052.

[17] F. Al-akashi, Using Wikipedia Knowledge and Query Types in a New Indexing Approach for Web Search Engines, PhD Thesis, University of Ottawa, 2014.

[17] S. Somani; S. Jain, Resolving identities on Facebook and Twitter, Tenth International Conference on Contemporary Computing (IC3), 2017 , pp. 1 – 3.

[18] A. Rao, N. Spasojevic, Z. Li, and T. Dsouza, "Klout score: Measuring influence across multiple social networks", Big Data (Big Data), 2015 IEEE International Conference on, IEEE, 2015 , pp. 2282-2289.

[19] G. Zhou, “Topics and Influential User Indentification in Twitter using Twitter Lists”, Master thesis, Waseda University, 2014.

[20] Ch. Jouis, I. Biskri, J. Ganascia, M. Roux. “Next Generation Search Engines: Advanced Models for Information Retrieval”, Book, 2012.

[21] K. Jaana and J. Kalervo, “Cumulated Gain-Based Indicators of IR Performance”, Research notes, University of Tampere.

[22] Zamir, O. (1999), _Clustering Web Documents: A Phrase Based Method for Grouping Search Engine Results‘. Doctoral thesis, University of Washington.

[23] Srijith Ravikumar, Kartik Talamadupula, Raju Balakrishnan, Subbarao Kambhampati. RAProp: Ranking Tweets by Exploiting theTweet/User/Web Ecosystem and Inter-Tweet Agreement. In Proceedings of the 13th CIKM Conference. ACM transaction 978-1-4503-2263-8/13/10., 2013, http://dx.doi.org/10.1145/2505515.2505667.

[24] Ke Tao, Fabian Abel, Claudia Hauff, Geert-Jan Houben. Twinder: A Search Engine for Twitter Streams. Web Information Systems, 2012.

[25] M. Bekkali and A. Lachkar. “Web Search Engine-Based Representation for Arabic Tweets Categorization”. From Social Data Mining and Analysis to Prediction and Community Detection, 2017, pp. 79-101

[26] H. Kwak, C. Lee, H. Park, S. Moon. “What is twitter, a social network or a news media?” In Proceedings of the WWW conference, ACM, 2010, pp. 591-600

[27] M. Mathioudakis and N. Koudas. “Twitter monitor: trend detection over the twitter stream”. In Proceedings of the SIGMOD, ACM, 2010, pp. 1155-1158.

[28] J. Weng., P. Lim, J. Jiang, Q He. “Twitter rank: finding topic-sensitive influential twitters”. In Proceedings of the WSDM, ACM 2010, pp.261-270.

[29] Y. Duan, L. Jiang, T. Qin., M. Zhou., Y. Shum. “An empirical study on learning to rank of tweets”. In Proceedings of the COLING, Association for Computational Linguistics, 2010. pp. pp. 295-303.

[30] J. Teevan, D. Ramage, R. Morris. “TwitterSearch: a comparison of microblog search and web search”. In Proceedings of the WSDM, ACM, 2011, pp. 35-44.

[31] A. Jadhav, H. Purohit, P. Kapanipathi, P. Ananthram, A. Ranabahu, V. Nguyen, N. Mendes, G. Smith., M. Cooney, A. Sheth. “Twitris 2.0 : Semantically Empowered System for Understanding Perceptions From Social Data”. In Proceedings of the Semantic Web Challenge. 2010.

[32] F. Abel, I. Celik, P. Siehndel, “Leveraging the Semantics of Tweets for Adaptive Faceted Search on Twitter”. In ISWC, Bonn, Germany, Springer 2011.

[33] Y. Duan, L. Jiang, T. Qin, M. Zhou, Y. Shum. “An empirical study on learning to rank of tweets”. In Proceedings of the Association for Computational Linguistics, 2010, 295-303.

[34] J. Huang, K.. Thornton, E. Efthimiadis. “Conversational Tagging in Twitter“. In Proceedings of the 21st ACM Conference on Hypertext and Hypermedia (HT), Toronto, Ontario, Canada, 2010, June 13-16.