

Computer Science & Engineering Department

COE 632- Advanced Database Systems

Instructor: Dr. Ghassan Z. Qadah

TwiNN: Automated Real-time Journalism using Topic Detection and Tracking (TDT) in Twitter

Student name: Suha Ahmed

ID: g00041093

Fall 2010

Automated Real-Time Journalism, using Topic Detection and Tracking (TDT) in Twitter

# ABSTRACT

Twitter as a microblogging platform has huge potential to become a collective source of intelligence that can be used to obtain opinions, ideas, facts, sentiments, and news. In this paper, we studied the role of Twitter as a news portal bearing in mind its global users network, its extraordinary short response time, and its short concise 140 character message. We applied 2 methods to detect news topic: undirected graph using lucene indexer and prefix tree using estMax method [1]. We provided a description of every method and its qualitative results

***Index Terms—microblogs, trend detection, TDT, Twitter, Data mining.***

# INTRODUCTION

The need for up-to-date brief news information is increasing. In the past CNN, BBC, and other TV news channel were competing to “*break*” the news. But after the internet revolution, people started to turn to their computers to get the latest news. This create new source of news internet website and RSS feeds becomes our new news channel and with the emerging of social network like Twitter and facebook a new meaning is given to the phrase “breaking news”. With Twitter we do not have one news reporter but thousands or even millions of news reporters who report news as they are happing in front of their own eyes and for free!

Twitter is a user-generated content system that allows its users to share short text messages, called ***tweets***, for a variety of purposes, including daily conversations, URLs sharing and information news. Below is a list of Twitter facts and figures which makes it more of news medium than a social network [2]:

* Twitter now has 105,779,710 registered users.
* New users are signing up at the rate of 300,000 per day.
* 180 million unique visitors come to the site every month.
* 75% of Twitter traffic comes from outside Twitter.com (i.e. via third party applications.)
* Twitter users are, in total, tweeting an average of 55 million tweets a day.
* Twitter's search engine receives around 600 million search queries per day.
* Of Twitter's active users, 37 percent use their phone to tweet.

Considering these facts, big companies are competing to get hold of Twitter data. Last year, Google reportedly paid $ 15 million for access to Twitter real-time stream of all its users’ tweets, Microsoft paid $ 10 million, and Yahoo joined later with a cash and revenue-share deal [3]. Moreover, in Nov. 19, 2010, Social media data company, Gnip has announced a partnership with Twitter that will allow them to sell up to 50% of Twitter’s data stream for $360,000 per year, or 10% of all messages for $60,000 per year to anyone who can afford the price [4].Twitter firehouse represents a valuable resource for data mining. Using the minded Twitter data business can get insights into market trends ,user demographic and many valuable information in fact Dell claimed that $9 million of their sales for 2009 came directly through Twitter and Facebook combined[5].

# RELATED WORK

Despites Twitter young age, four and half years, it becomes the interest of numerous research studies. In [6] they classify Twitter messages which deal with trending topics. They also analyzed the demographics of the authors of such Twitter messages and attempted to map a Twitter trend into what's going on in the real world.

In [7] they investigated real-time interaction of events such as earthquakes in Twitter and proposed an algorithm to monitor tweets and to detect a target event. They devised a classifier of tweets based on features such as the keywords in a tweet, the number of words, and their context and produced a probabilistic model for the target event that can find the center and the trajectory of the event location.

In addition, [8] describes a novel approach to news recommendation that uses real-time micro-blogging activity, from Twitter, as the basis for promoting news stories from a user’s favorite RSS feeds.

Although [9] has the same motivation as ours, they used different methodology. Their novel topic detection technique, permit retrieving in real-time the most emergent topics expressed by Twitter community by means of extracting a set of terms from the tweets and model the terms life cycle according to an aging theory intended to mine the emergent terms. They classify a term as emerging if it frequently occurs in the specified time interval and it was relatively rare in the past. Furthermore, they analyzed the social relationships in tweeter network with the well-known Page Rank algorithm in order to determine the authority of the users.

Finally, [10] gives an impressive quantitative study on the entire Twitter sphere and information diffusion on it. In [10] they crawled the entire Twitter site and obtained 41.7 million user profiles, 1.47 billion social relations, 4262 trending topics, and 106 million tweets. In their analysis to Twitter follower-following topology, they found a non-power-law follower distribution, a short effective diameter, and low reciprocity, which all mark a deviation from known characteristics of human social networks [11]. They ranked users by the number of followers and by PageRank and found the two rankings to be similar. On the other hand, ranking by retweets differs from the previous two rankings, indicating a gap in influence inferred from the number of followers and that from the popularity of one’s tweets. They also analyzed the tweets of top trending topics and classified the trending topics based on the active period and the tweets and they found out that the majority (over 85%) of topics are headline news or persistent news in nature. One of the interesting results they found about retweets reveals that any retweeted tweet is to reach an average of 1000 users no matter what the number of followers is of the original tweet. Once retweeted, a tweet gets retweeted almost instantly on next hops, signifying fast diffusion of information after the 1st retweet.

All the above research were either providing quantitative data about Twitter [6][10] or trying to detect news topics [9] or rank them [10] but none of the papers have taken in to consideration Twitter streaming data nature. For example [9] use a technique similar to our undirected graph method even though estMax method can be more appropriate for Twitter streaming data if it was implemented properly.

# PROBLEM DEFINITION

The main goal of this project is to detect and identify trending topics from streaming Twitter data. Then, use the identified topic to acquire more details using Twitter search API [12][13] and allow the tracking of any topic of the identified trending topic. Determining trending topics can be considered a type of First Story Detection (FSD), a subset of the larger problem known as Topic Detection and Tracking (TDT) [14]. The problem of TDT consists of three tasks: segmentation task, detection task and tracking task. In this project the methodologies we used consist of the three tasks [14].

# METHODOLOGIES

## Segmentation Task

In TDT, the segmentation task mainly addresses the problem of automatically dividing a text stream into topically homogeneous blocks [14]. Given Twitter 140 character messages, we do not need to do segmentation. But looking at motivation behind segmentation, which is facilitating topic detection and tracking task [14], we found that we needed to do preprocessing in order to facilitate topic detection and tracking task.

### Filtering

The data was collected from the Twitter streaming API using Twitter4j an unofficial Java library for the Twitter API [15]. We filtered Twitter messages as they arrive on real time. First, the tweets were preprocessed to remove URL’s, unicode characters, usernames, and punctuation. Next, words which are trigram or less were removed from the tweet text as well as words which contain repeating characters such as “hhhhhhhhiiii”. Then, a hash table with frequently occurring stop words in twitter was used to remove stop words form the tweet. When we used estMax method we only processed and filtered English words any tweet written in language other than English was not considered in our estMax topic detection and tracking engine and the reason is described below.

Initially, we filtered the tweets without stemming and translated them using Google translate API but because Google translate is REST API, translating every tweet took 3 sec given that we get on average 30 tweet per second this is a huge lost we cannot afford. Then we collected 10000 tweets and sent them to be translated in different thread although this will not improve the speed drastically but it will reserve the tweets for processing whenever possible. Unfortunately, Google translate API is a rate limited API and cannot process stream data. Thus, we decided to translate topics after detection which implies grouping tweets that belongs to the same language to gather. We used Langdetect API (open source) to detect English text from non English one and process them separately based on the method used for topic detection as we will see in the next section.

Also, we did not apply stemming for undirected graph method using lucene indexer but we stemmed estMax tweets to improve the detection task as much as we can. Although stemming will alter text nature a bit.

## Detection Task

In TDT, event detection is the problem of identifying stories in several continuous news streams that pertain to new or previously unidentified events [14]. Our to topic detection problem is part of TDT event detection category called “*on-line detection*” which is applied on stream data. We used two methods to detect new topic, undirected graph using Apache lucene indexer and prefix tree using estMax method. The two methods are described below.

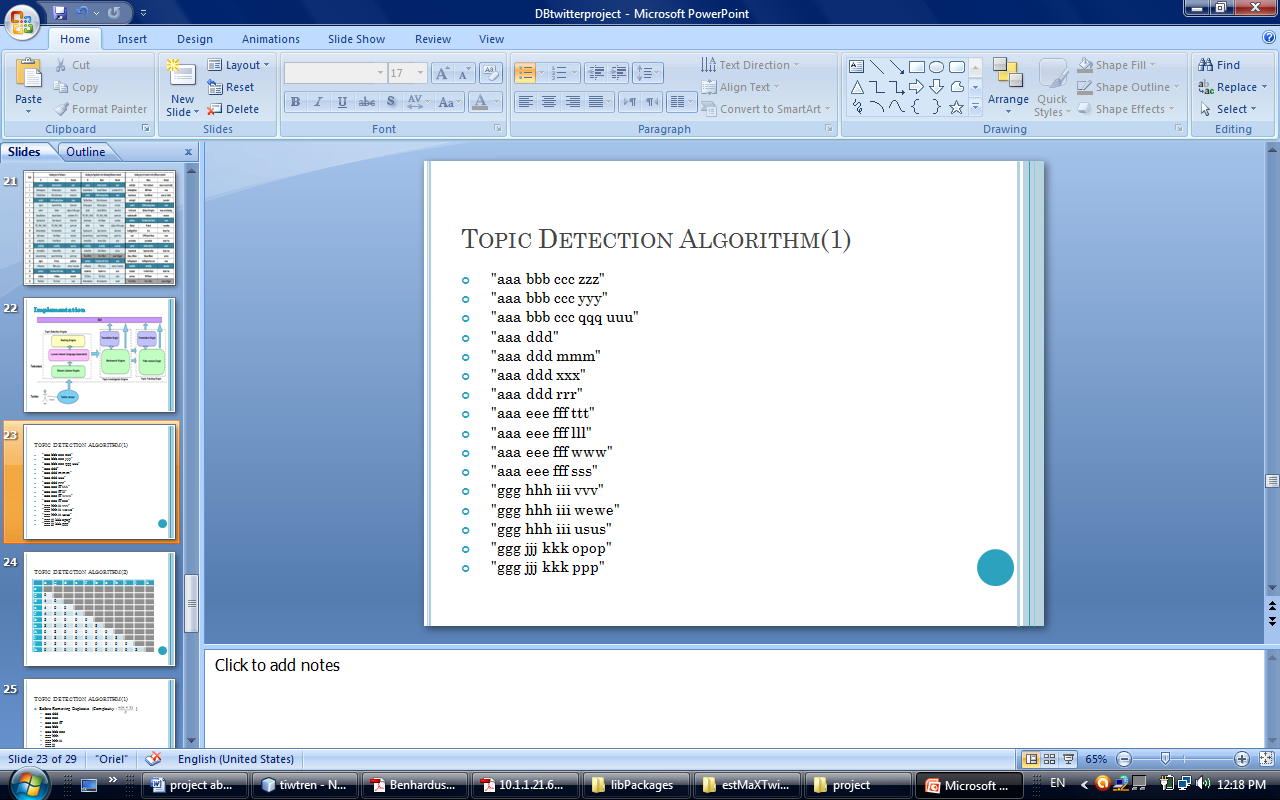


Figure 2(a) List of tweets

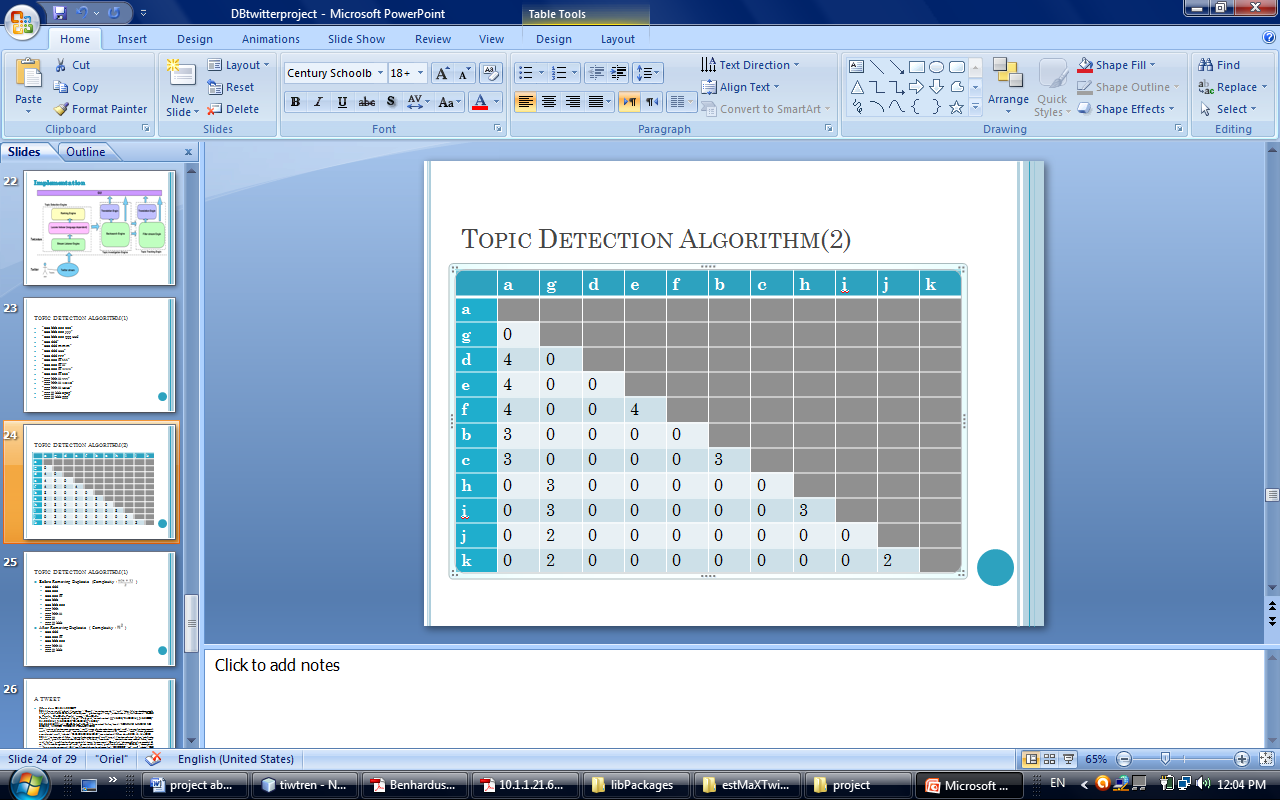


Figure 2 (b) Frequent 2 Topic sets

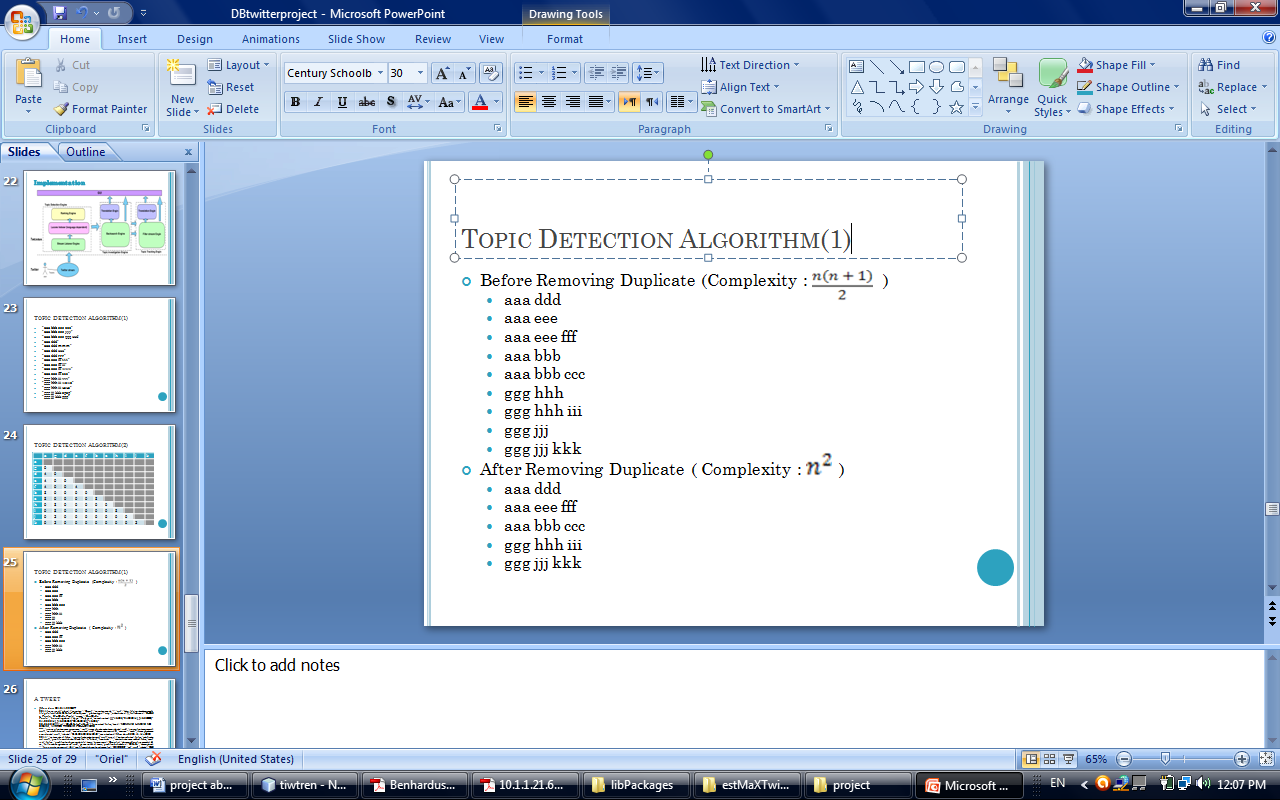


Figure 2 (c) Frequent topics before and after removing duplicates.

### Undirected Graph using Lucene Indexer method

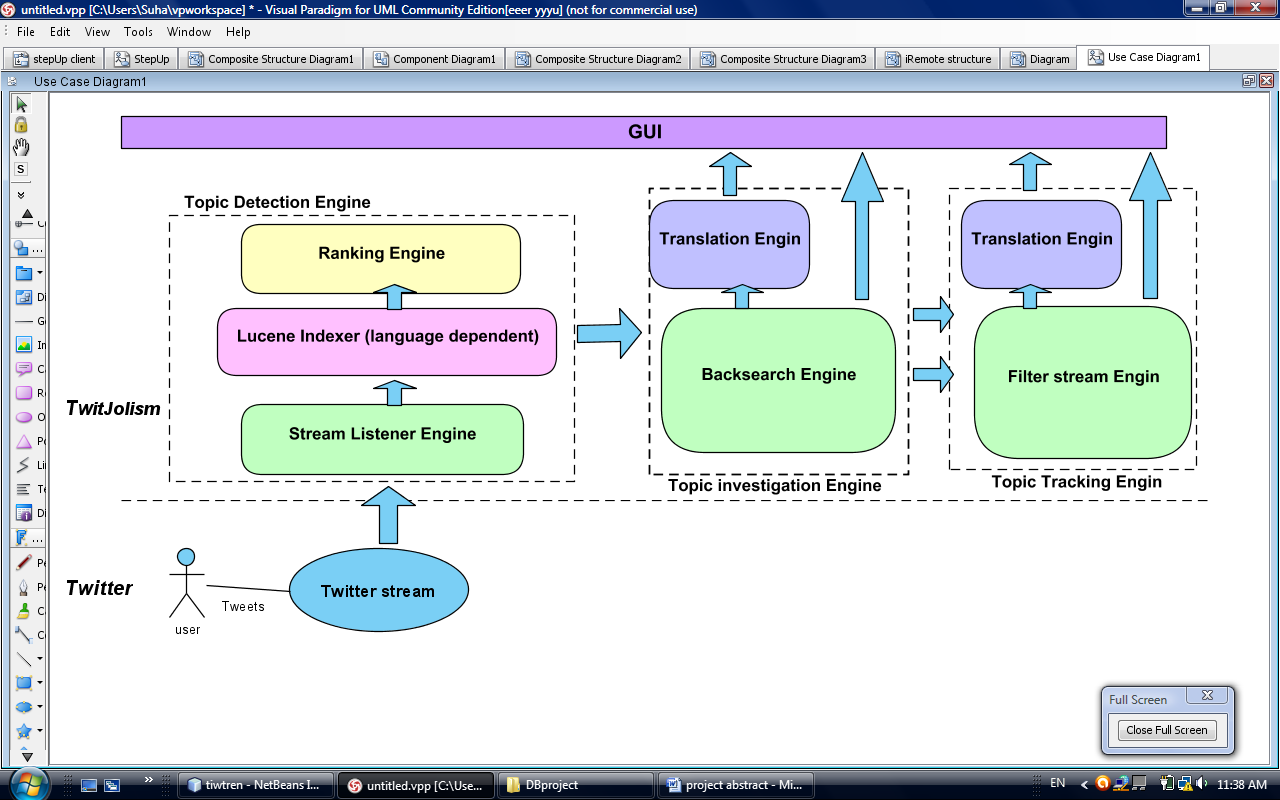


Figure 1. Lucene Topic detection Engine

The topic detection task using this method is performed as following (Fig. 1): The tweet arrive to the stream listener engine, it get filtered and stored. Once we collected 10000 tweets we sent them to the lucene indexer module and start collecting new set of tweets. In lucene indexing model, we performed a language dependent indexing. Using lucene languages analyzers, we managed to remove stop words from the following languages: Arabic, Brazilian Portuguese, Czech, Danish, German, Greek, English, Spanish, Persian, Finnish, French, Hungarian , Italian, Japanese, Korean, Dutch, Norwegian, Portuguese, Romanian, Russian, Swedi ,Turkish, Chinese. Then, we created index of words for the 10000 tweets and computed the document frequency for every word in the index. Document frequency here represents in how many tweets did the word of interest appeared. To reduce processing time and computation complexity, we collected words whose frequency is greater than σ. Then, we computed frequent two items ( sets that consists of 2 words only) set for those words (see Fig. 2 (b)) and again the 2 items sets whose frequency above or equal to a cut off frequency ϕ were used to create the undirected graph . Using these topic sets super sets were produced and labeled as topic (see Fig. 2 (c)). The algorithm we used to produce the super item sets was of complexity and the algorithm we used to remove duplicate topics was of complexity n2. After that, topics were ranked based on their words document frequency.

There are four problems related to this method:

First, it introduces false positive error which is detecting a topic that does not exist because of the use of the undirected graph. Example, if sets {aaa,bbb} and {aaa,ccc} frequencies is greater than the cut off frequency σ which mean there are at least σ tweets mentioned {aaa,bbb} and there are also at lease σ tweet mentioned {aaa,ccc} the method create the super set {aaa,bbb,ccc} and lable it as topic even though there is no direct edge between {bbb} and {ccc}.

The second problem is that we have to wait until we collect n tweets before we starts processing which really undermine Twitter real time messages advantages. Reference [9] also uses the same concept.

Third, the method consumes lots of memory the maximum n tweets that we were able to process was 40000 tweets which took 28 minutes to collect and 43 minutes to process.

Fourth, all the identified frequent item topics will be discarded unless the user chooses to track one of them and in this case the topic tracking engine will track the specified topic/s. One problem with this approach is the need to construct the frequent topic set all over again every time we run the engine and we may identify topics that we have identified in the previous runs.

To address the above problems we decided to use different method for topic detection, one that considers real time streaming data. Thus, we used estMax.

### Prefix Tree using estMax Method

estMax method track the set of maximal frequent item set MFIs instantaneously over an online data stream. The method maintains the set of frequent item sets by a prefix tree and extracts all MFIs without any additional superset/subset checking mechanism. Upon processing a new transaction, those frequent item sets that are matched maximally by the transaction are newly marked in their corresponding nodes of the prefix tree as candidates for MFIs. At the same time, if any subset of a newly marked item set has been already marked as a candidate MFI by a previous transaction, it is cleared as well. By employing this additional step, it is possible to extract the set of MFIs at any moment. A detailed description of the method is available in [14].

We used the method to detect a list of frequent topics in Twitter and we reached the following results:

First, the method is computationally expensive. It relay heavily on breadth depth traversing the tree and power set computation which can degrade performance. Not only that but also there are many variables that need to be updated every time a new tweet is received.

Second, as the prefix tree size gets bigger the method becomes slower. Also, there is no obvious way to use the method in concurrent environment because the method is iterative in nature.

Third, the method needs very clean data. In other words, extremely noisy stream such as twitter may not harness the usefulness of this method unless it is coupled with an exceptionally good filter which is not handy. This problem was identified by the fact that the entire topics detected consisted of one word only. In other words spelling mistake or other type of sentence structure mistakes reduced the possibility of identifying relation between words. This type of false negative error clearly indicates the need for a better filter.

## Tracking Task

In TDT, tracking task is defined as the task of identifying all of the events in a corpus of stories; it is fundamentally similar to the standard routing and filtering tasks of Information Retrieval (IR). But in TDT tracking is considered difficult task due to the fact that events rather than queries are tracked, and in that events have a temporal locality that more general queries lack)[14].

In our case, topic tracking is more of a IR filtering task. Thus, we handle it using Twitter filter stream API. Once a list of topics detected, we can track them using Twitter filter stream API to get more details about a specific topic. Moreover, once we have enough information about a specific topic we can query Twitter search database to get information regarding the topic of interest.

The problem for us with this task was identifying new useful information automatically and constructing a good filer. For now, we leaved the problem of identifying new information in filtered data for the user.

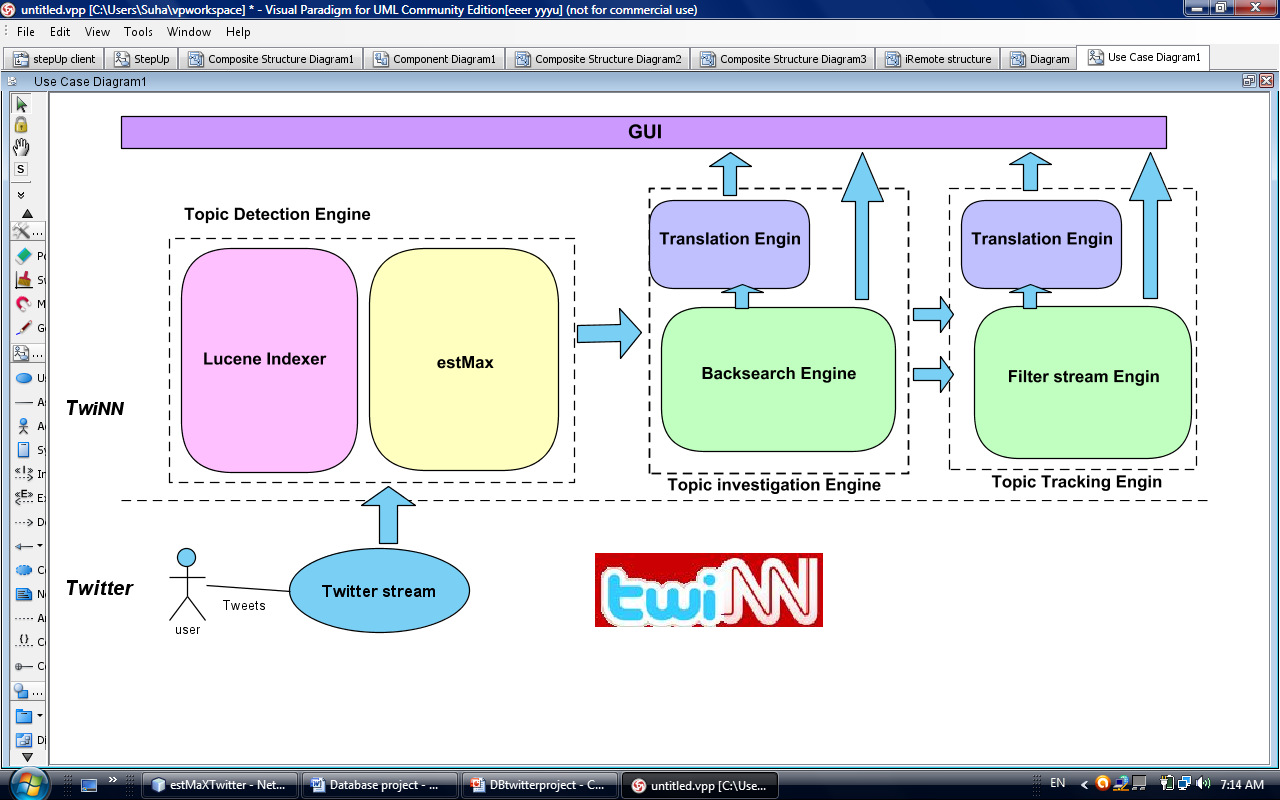


Figure 3. TwiNN Architecture

# RESULTS

A preliminary evaluation is carried out on an implementation of the two topic detection techniques used that shows promising results. In this section, we give some qualitative results for both topic detection methods:

### First, Undirected Graph using Lucene Indexer method:

In the below table we can see some of the identified topics by the undirected Graph using Lucene Indexer method and Google trend results for the same topic. We can clearly see climb in the same period the topic was detected.

|  |  |  |
| --- | --- | --- |
| Time Topic detected | Method Topic | Google Trend results for the same period |
| 2011-01-03 10:42:42 | justin selena |  |
| 2011-01-11 23:28:42 | palin blood sarah libel |  |
| 2011-01-12  14:33:41 | australia qld floods |  |
| 2011-01-13 00:16:24 | ronaldinho flamengo |  |

### Prefix Tree using estMax Method

In the below table we can view some of the detected frequent unigrams by the Prefix Tree using estMax Method and Google trends results for the same topic. We can clearly see climb in the same period the topic was detected.

|  |  |  |
| --- | --- | --- |
| Time Topic detected | Method Topic | Google Trend results for the same period |
| 2011-01-16 00:00:01 | nelson |  |
| 2011-01-16 09:39:30 | zodiac |  |
| 2011-01-16 09:39:30 | aaliyah | Figure Google Hot topic list in 2011-01-16 at 20:42 |

# IMPROVEMENTS AND SUGGESTION

The door for improvement is wide open. First of all, a good filter is really the essence of such a project. For example, the topic picked above to qualify performance were not picked automatically rather the author went through the list of detected topics to see which one could convey a meaningful news topic.

Second, language translation and processing (NLP) could benefit this project vastly. And the application of introducing NL is countless. For instance, connecting language to a geographical location, we can know emergent topics in different locations around the world. Also, translating all tweet will give us global topical news.

Third, a project such as this needs expensive computation with fast response time, the only way to satisfy this is by parallel processing.

# CONCLUSION

Two of our era main characteristics are the exponential data generation and the need for automated real time data processing. Detecting topical news automatically can solve one aspect of this dilemma. In this paper, we proposed two topic detection techniques which can be used to automatically detect topical news. We also presented a primal qualitative performance results for each method. We also identified areas of improvement for each method and for the topic detecting task in general.

# REFERENCE

1. Ho Jin Woo, Won Suk Lee, "estMax: Tracing Maximal Frequent Item Sets Instantly over Online Transactional Data Streams," IEEE Transactions on Knowledge and Data Engineering, vol. 21, no. 10, pp. 1418-1431, Oct. 2009, doi:10.1109/TKDE.2008.233
2. “Twitter User Statistics REVEALED” [http://www.huffingtonpost.com/2010/04/14/Twitter-user-statistics-r\_n\_537992.html](http://www.huffingtonpost.com/2010/04/14/twitter-user-statistics-r_n_537992.html)
3. http://gadgetidol.com/mobile-application/Twitter-firehose-too-intense-take-a-sip-from-the-garden-hose-or-sample-the-spritzer.html
4. http://www.marketingpilgrim.com/2010/11/gnip-gets-its-hands-on-Twitters-firehose.html
5. http:// www.pcpro.co.uk/news/enterprise/.../Twitter-earned-dell-9-million
6. M. Cheong, V. Lee, “Integrating Web-based Intelligence Retrieval and Decision-making from the Twitter Trends Knowledge Base,” In Proceedings of CIKM 2009 Co-Located Workshops: SWSM 2009, pp. 1-8,2009.
7. T. Sakaki, M. Okazaki, Y. Matsuo,“Earthquake shakes Twitter users: real-time event detection by social sensors,” In Proceedings WWW 2010, April 26–30, 2010, Raleigh, North Carolina, USA.
8. Owen Phelan, Kevin McCarthy, Barry Smyth, “Using Twitter to Recommend Real-Time Topical News,” RecSys’09, October 23–25, 2009, New York, New York, USA.
9. Mario Cataldi , Luigi Di Caro , Claudio Schifanella “Emerging Topic Detection on Twitter based on Temporal and Social,” MDMKDD '10 ,In proceedings of the Tenth International Workshop on Multimedia Data Mining ACM ,New York, NY, USA, 2010
10. H. Kwak, C. Lee, H. Park, and S. Moon, “What is Twitter, a Social Network or a News Media?” WWW 2010, April 26–30, 2010, Raleigh, North Carolina, USA.
11. M. E. J. Newman and J. Park. “Why social networks are different from other types of networks.” Phys. Rev. E, 68(3):036122, Sep 2003.
12. Twitter API Documentation,http://apiwiki.Twitter.com/w/page/22554679/Twitter-API-Documentation
13. [The Twitter Streaming API — How It Works and Why It’s A Big Deal (Update 2010-11-21)http://borasky-research.net/2010/01/06/the-Twitter-streaming-api-how-it-works-and-why-its-a-big-deal/](http://borasky-research.net/2010/01/06/the-twitter-streaming-api-how-it-works-and-why-its-a-big-deal/)
14. J. Allan, R. Papka, and V. Lavrenko, “On-line New Event Detection and Tracking,” In Proceedings of ACM SIGR, pp. 37-45, 1998.
15. www.twitter4j.org

## Appendix 1

# How does Twitter works?

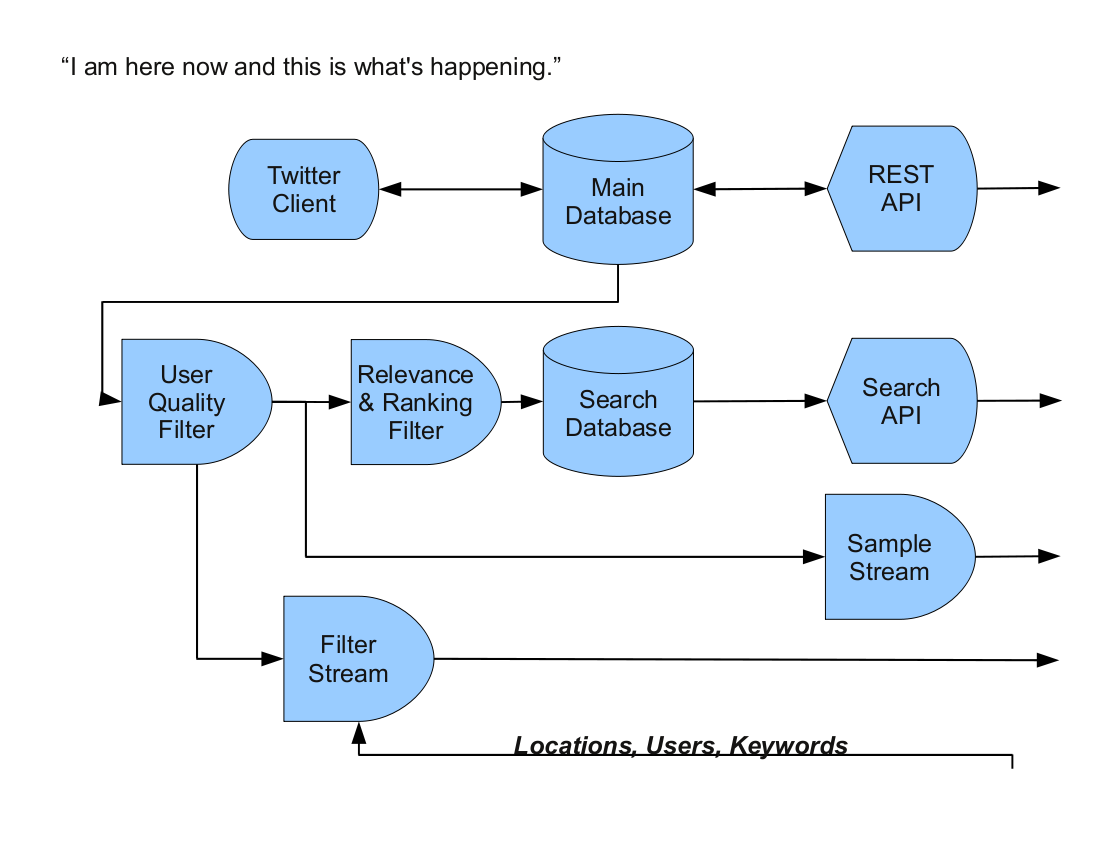


Figure 4

The basic flow of tweets is as follows [12][13]:

A user publishes a tweet tagged with a time, the name of the user, the user it was sent to if it was a reply, and, if the user has enabled geotagging, the user's location. First, the tweet goes into the main database (firehouse). Once it is in the main database, it is also sent into a user quality filter which removes tweets which Twitter deems to be of low quality. Then the tweet is forwarded into relevance and ranking filter. If this filter also passes, the tweet is sent to the search database, to be indexed for Twitter search. The top arrow on the right in Figure 1 represents the REST API that enable you to create, read, update or delete tweets from the main database. users are allowed to make 150 requests per hour by default using this API while white listed accountant and IP addresses allowed 20,000 requests per hour (there is a process you can go thorough to get your account or IP to be white listed). The second arrow on the right represents the search API which permits looking up tweets from the past in the search database which is read-only database. The Search Rate Limit isn't made public to discourage unnecessary search usage and abuse. The bottom two arrows represent the streaming API. The streaming API is also a read-only. Unlike the search API, the streaming API is not buffered through the search database. The raw tweets pass through the same user quality filter as used to qualify tweets for the search database, but they do not go through the relevance and ranking filter. Thus, there are more tweets available to users of the streaming API than there are to users of the search API. Then it split into two streams, the sample stream and the filter stream which both are subset of the Firehose. A subscriber simply reads the sample stream and the filter stream directly from Twitter. The subscriber can filter the tweets coming out of the stream by keywords, lists of users that created the tweets, or locations of geotagged tweets.The filter stream default access level allows up to 400 track keywords, 5,000 follow userids and 25 0.1-360 degree location boxes. Increased access levels allow 100,000 follow userids (“shadow” role), 400,000 follow userids (“birddog” role), 10,000 track keywords (“restricted track” role), 200,000 track keywords (“partner track” role), and 200 0.1-360 degree location boxes (“locRestricted” role).On the other hand, the sample stream default access level ‘Spritzer’  allow 1% of the firehose and the higher access level ‘Gardenhose’ access level allow 5% of the firehose which is good enough for data mining purpose. To grantee a ‘Gardenhose’ access level you have to contact Twitter and describe your use case and sign EULA which I have done and now I have ‘Gardenhose’ access level for my project.