Twitter Trend Detection

Mentor: Nazrul Haque Athar

- **Suhas Reddy Inturi (201301171)**
- **Chilupuri Anil Reddy (201301238)**
- **Kaja Varun Kumar (201301214)**

Problem Statement:

Detecting trends/breaking news on social media stream, Twitter for example. Among the flooded stream of posts on social media, the task is to identify the trending and significant posts using hashtags and popular keywords. This can layout as identifying ones based on user input or breaking on the charts using corpus of tweets.

Solution:

1) Database Collection:

Database that we have taken consists of tweets which have 26 different aspects. They are:

1) contributors
2) truncated
3) text
4) is_quote_status
5) in_reply_to_status_id
6) id
7) favorite_count
8) source
9) Retweets
10) coordinates
11) timestamp_ms
12) entities
13) in_reply_to_screen_name
14) id_str
15) retweet_count
16) in_reply_to_user_id
17) favorited
18) retweeted_status
19) user
20) geo
21) in_reply_to_user_id_str
22) lang
23) created_at
24) filter_level
25) in_reply_to_status_id_str
26) place

Out of these 26, we have selected 6 important aspects. They are:
1)Text 2) Retweets 3)Timestamp_ms 4)Entities 5)retweet_count 6)created_at

Data is further processed so that we only have the above mentioned ‘tags’ available in the Dataset.

2)Database Pre-processing:

In the first phase, we found the frequency of all the words(hereby will be called as “tokens”). Frequency of occurrence of all the tokens is calculated.The top most frequent tokens are considered and they are further processed. Parts of Speech of the most frequent tokens is now taken into consideration. Tokens which come under the categories like Conjunction, Preposition, Interjection, Articles, Pronoun have been given less priority. And the remaining tokens are highly prioritized.

2.1) Categorising the classified words:

All the tokens that have been given less priority are considered as “Stop words”. The task before us is to identify these “Stop words” from the DataSet.

In the training phase, a large dataset is taken. For that Dataset, frequency of occurrence of all the tokens and also their parts of speech is identified. Then words with higher frequency and those which come under the afore mentioned less priority case are considered to be “Stop Words”.

3)Analysis of Data :

We are now left with tokens from which the trend has to be detected based upon the word occurrence frequency and HashTags. This is a 3-step process. We first identify “Bursty” keywords, i.e. keywords that appear in tweets at an unusually high rate. Subsequently, it groups bursty keywords into trends based on their co-occurrences. In other words, a trend is identified as a set of bursty keywords that occur frequently together in tweets. After a trend is identified,
we extract additional information from the tweets that belong to the trend, aiming to discover interesting aspects of it. 3 steps are described in detail in the following paragraphs.

3.1)Detecting Bursty Keywords:

A keyword is identified as bursty when it is encountered at an unusually high rate in the stream. For example, the keyword 'NBA' may usually appear in 5 tweets per minute, yet suddenly exhibit a rate of 100 tweets/min. Such 'bursts' in keyword frequency are typically associated with sudden popular interest in a particular topic and are often driven by emerging news or events. For example, a sudden rise in the frequency of keyword 'NBA' may be linked to an important NBA match taking place. We treat bursty keywords as 'entry points' for trend detection. In other words, whenever a keyword exhibits bursty behavior, we consider this an indication that a new topic has emerged and seeks to explore it further Effective and efficient detection of bursty keywords is thus crucial to our solution's performance. To detect bursty keywords, we developed an algorithm, Queue Burst, with the following characteristics: (i) One-pass. Stream data need only be read once to declare when a keyword is bursty. (ii) Real-time. Identification of bursty keywords is performed as new data arrives. No optimization over older data is involved. (iii) Adjustable against 'spurious' bursts. In some cases, a keyword may appear in many tweets over a short period of time simply by coincidence. The algorithm is tuned to avoid reporting such instances as real bursts. (iv) Adjustable against spam. Spam user groups repetitively generate large numbers of similar tweets. The algorithm is tuned to ignore such behavior. (v) Theoretically sound. QueueBurst is based on queuing theory results.

3.2) From Bursty keywords to Trends :

Using QueueBurst, we compute a set of bursty keywords K_t at every moment t. Some of these keywords correspond to the same trend. For example, keywords 'NBA', 'Lakers', 'Orlando' and 'game' may be bursty at the same time, all of them occurring in tweets commenting on a Lakers vs Orlando match that is taking place. Twitter-Monitor periodically groups keywords $k \in K_t$ into disjoint subsets K^i_t of K_t, so that all keywords in the same subset appear in the same topic of discussion. Given subsets $\{K^i_t\}$, a trend is identified by a single subset K^i_t.

Bursty keywords are grouped together by algorithm Group-Burst. To group bursty keywords, GroupBurst assesses their co-occurrences in recent tweets. For this purpose, a few minutes' history of tweets is retrieved for each bursty keyword and keywords that are found to co-occur in a relatively large number of recent tweets are placed in the same group. Since enumerating all possible groupings proves to be very expensive for such a real-time task, GroupBurst pursues a greedy strategy that produces groups in a small number of steps.

4)DataSet Details :

Data Size : 4GB

Number of Tweets available : More than 4 crore 32 lakh.

5)Implementation :

In analysing the trends from the vast amount of data available, we have to consider the Data(tokens) and also the timestamp(Time at which tweet has been tweeted). So this is Data-Time series analysis.

We have split the trend detection process into two parts. Data has been partitioned into HashTags and Tweets without HashTags(Hereby referred as TweetData). Trend analysis has been done on these two parts simultaneously. Initially, a mapping has been created from HashTag or Data without HashTag to TimeStamp.TimeStamp is in the format of “dd-mm-yyyy”. This notation has been converted into milliseconds using python.

Following is the series of tasks that have been performed on the data.

5.1) Detection of Bursty Keywords:

First task afore us in this stage is to remove the Stop words. Most common words in a language which occur frequently but do not convey any valuable information are called as Stop Words. In order to remove stop words, following process has been followed.

5.1.1)Stop Words Removal :

A part of the data that we have has been considered as Training data which is used to detect Stop Words. Each and every tweet has been partitioned into tokens and a word count is maintained for all these tokens. Simultaneously we also obtained the “Parts of Speech” of all the tokens. Tokens which come under the category of Pronouns,Articles,Prepositions, Conjunctions and those with higher frequency have been detected and are appended to a “Stop Word” list.

5.1.2)Analysing Data without Stopwords :

Based on the TimeStamp, data for every hour is retrieved from the dataset(with no stopwords) and top “n” most occurring tokens (might also include hashtags) will be displayed as the trends for that hour. While doing this process, HashTags are given considerably more preference than TweetData.

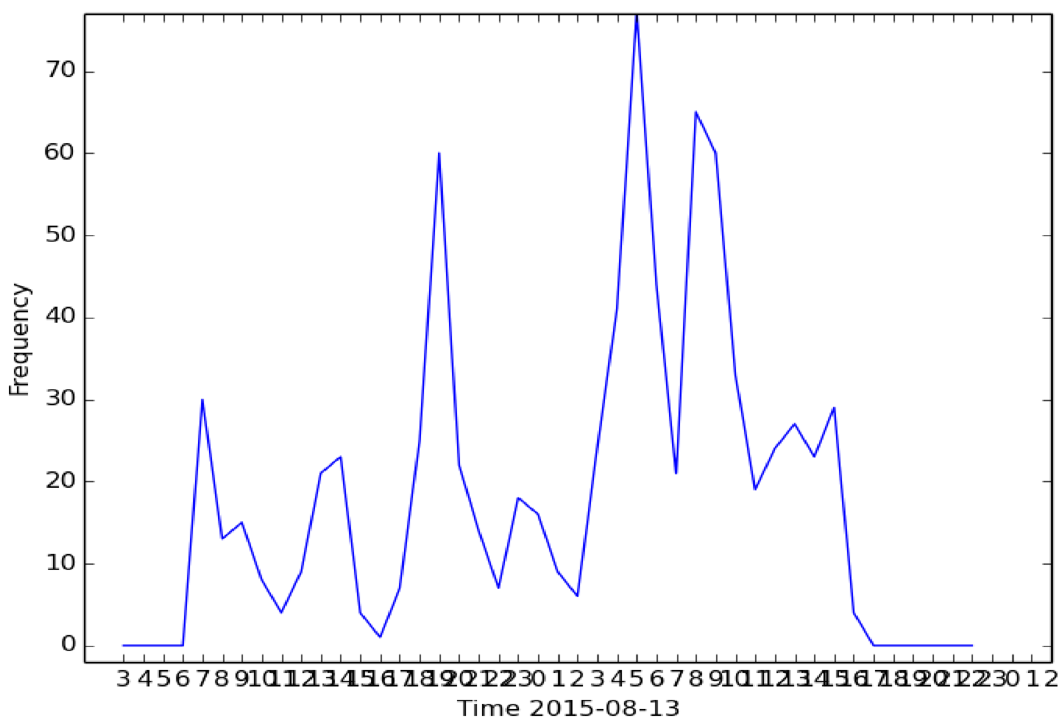
As an addition to identifying keywords, we have also plotted graphs that indicate the peak time frequency of the Bursty keywords. These graphs would describe the time at which a particular keyword has started occurring more frequently along with its count in the tweets.

5.2)Bursty keywords to Trends :

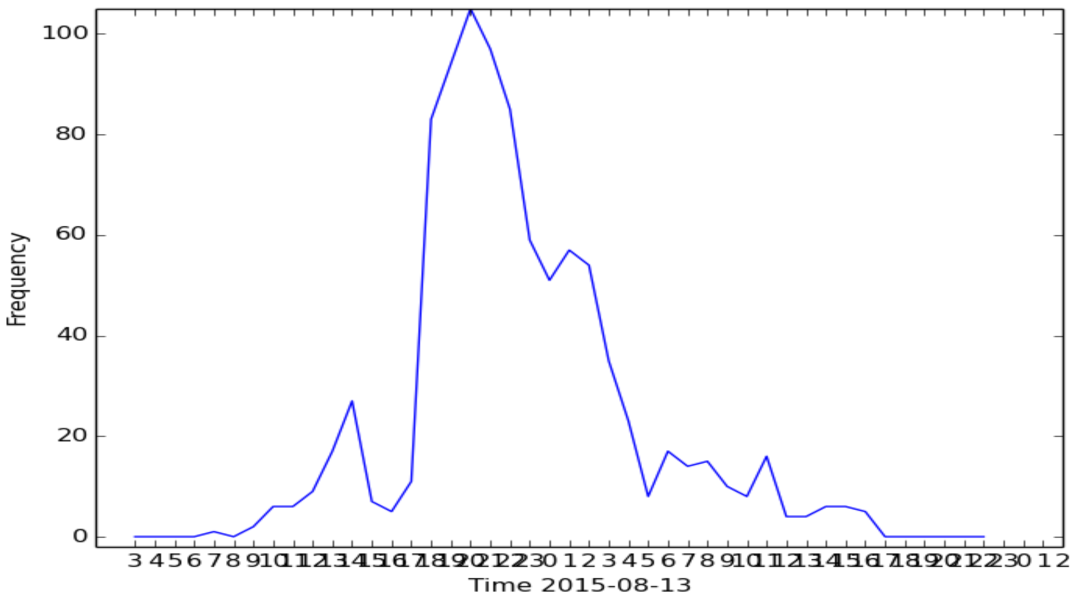
Top 5 Bursty keywords that we have acquired in the previous phase are considered to be the trends in that particular hour. In this way, for every hour, trends get updated. Accordingly, the graphs will be plotted for these statistics. Some example graphs have been added to this report in the end.

Example Graphs :

Trend : Soundsgoodfeelsgood



Trend : LeftHandersDay



Trend : SharetheLove