The scope of the system is set to social media sites, in particular a site called Twitter. Twitter is a micro blogging social media site that allows users to post frequent short messages also known as “micro blogs”. The content of such a site is an extraordinarily large number of small textual messages, posted by millions of users, at random or in response to perceived events or situations. These trends can be discovered using statistical analysis of mass of posts. The system has been developed to summarize the trending events or situations on Twitter in order to provide a quick overview of the generalized opinion shared by millions of users on that particular trend as it can be very tedious and humanly impossible to go through all the tweets on that trend. Also, users are likely to encounter spam, posts in other languages, rants, and other sources of misinformation. So the system provides the users with the generalized opinion of the people who have been talking on that topic through an automated generated summary by mapping the collected information on already available resources and thereby preserving context.

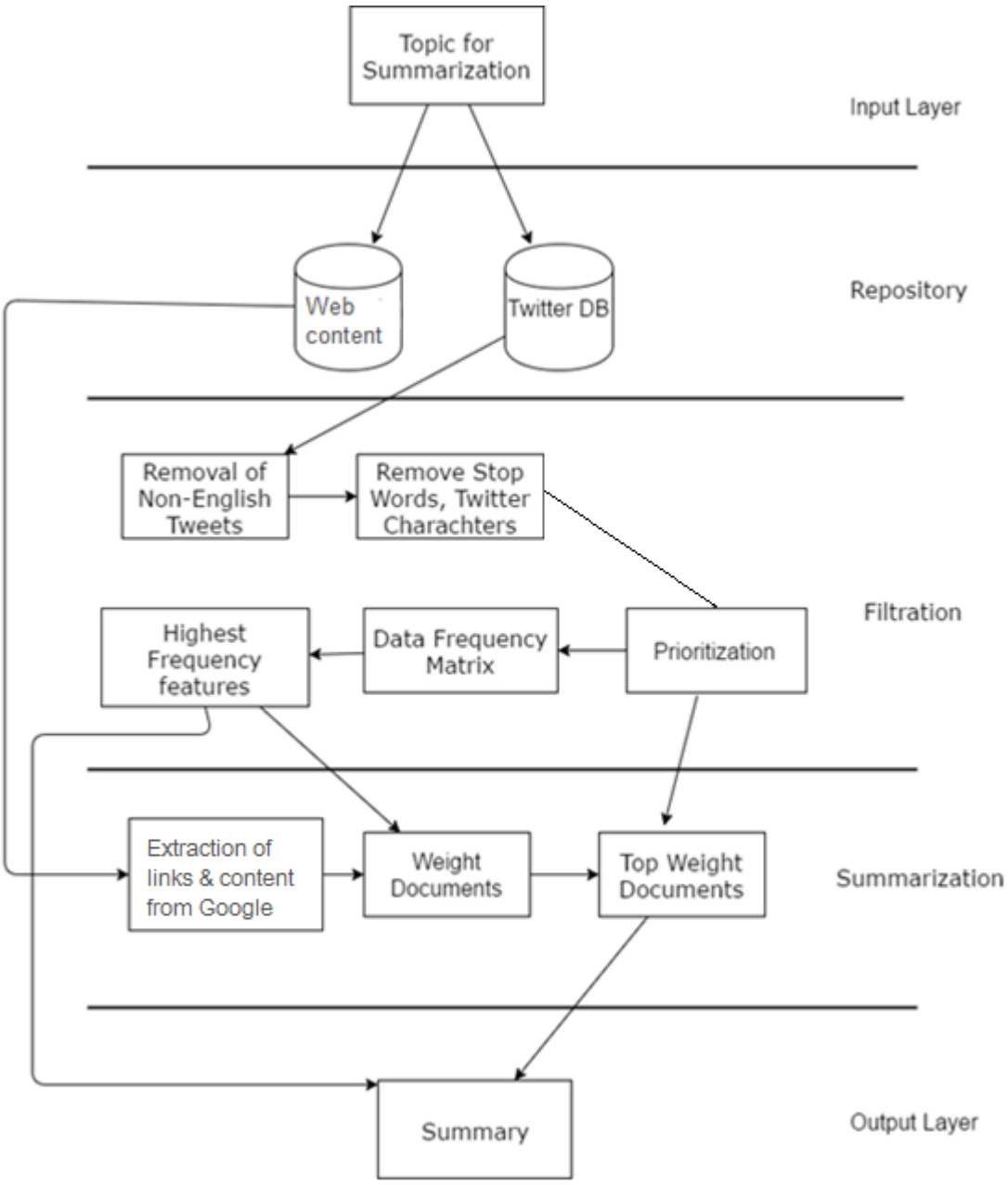
The scope of this system involves:

- **Extraction:** The tweets related to the trending topic from Twitter are extracted using existing Twitter API.

- **Filtration:** The tweets extracted are filtered using algorithm(s) to remove spams, posts in other languages, rants, and other sources of misinformation.

- **Web-scraping**: This component of the system deals with extraction of all the content related to the entered search term from the web.

- **Summarization:** The tweets collected after filtration are mapped on pre-existing web-based via web scraping content to provide a generalized view which is then followed by summarization of top trending tweets to provide an overview of the opinions of twitter users.



Software Architecture Diagram

The software architecture diagram illustrates the flow of the software modules and how the project will be developed in various stages. It represents how the modules of the system interact with each other. The diagram also shows the flow of usage of software along with working of subsystems.

The various layers are as follows:

- **The Input Layer**: This is the first step of the system. The user enters the topic

for summary on the interface which is then carried forward for extraction from twitter and pre-existing web content.

- **The Filtration Layer:** The tweets collected after extraction are then processed further to remove non-English tweets, spams, rants, irrelevant data and other sources of misinformation. The system removes all the non-English tweets, removes stopping words, stems those tweets and generates a frequency count of all the words after filtration; thereby generating a set of highest repeating phrases which have the maximum weight from the collected tweets.
- **The Summarization Layer:** The tweets collected after filtration are mapped on pre-existing web-based content to provide a generalized view of the users. This is the most distinguishing feature of our system. This is then followed by summarization of top trending tweets to provide an overview of the opinions of twitter users.
- The **data repository** is responsible for providing the necessary content to the system i-e tweets for filtration and web content for the final summarization layer.

HARDWARE AND SOFTWARE REQUIREMENTS

- **Hardware Requirements:**
  - ✓ Server machine (PC, Laptop, Tablet)
  - ✓ 10 GB hard disk
  - ✓ 1GB RAM
  - ✓ Net connectivity
- **Software Requirements:**
  - ✓ Operating System (Windows 7/XP and above)
  - ✓ R
  - ✓ User Interface : Shiny app package
  - ✓ Microsoft Office: MS Word, MS Excel
  - ✓ Quantitative analysis : Quanteda package
  - ✓ ROAuth : Tweets extraction
  - ✓ RCurl, XML : Web Scraping
  - ✓ Microsoft Project Professional
  - ✓ Star UML

User Interface Description

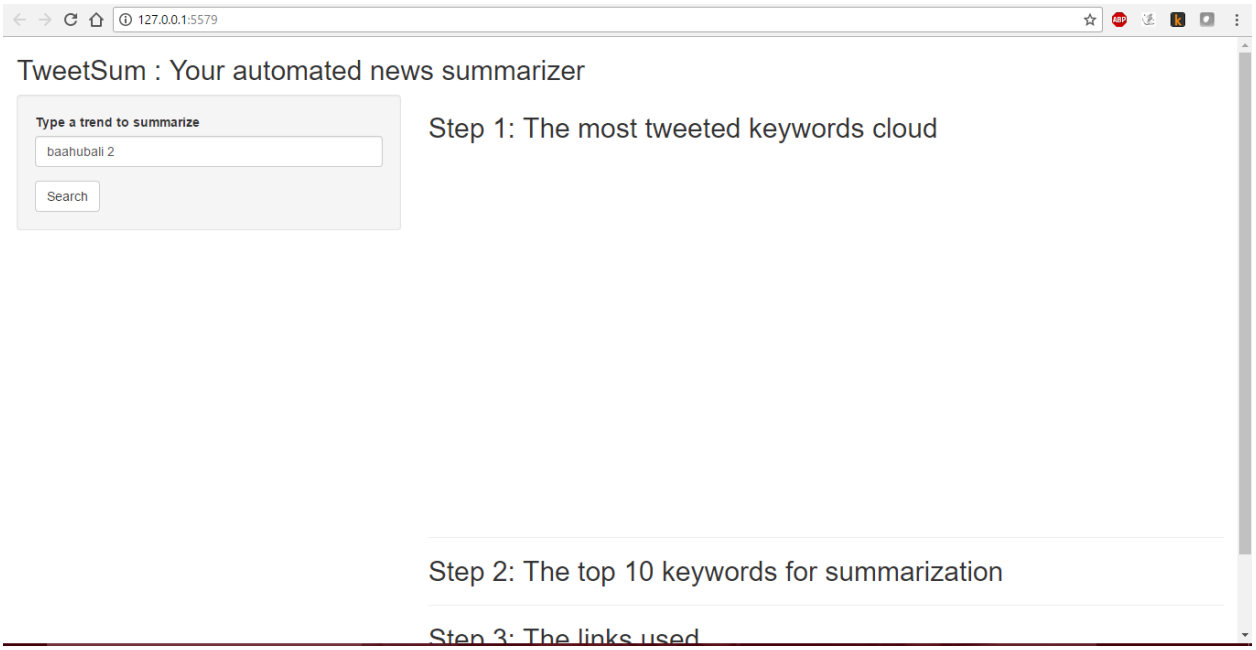


Fig (a) User Interface Description

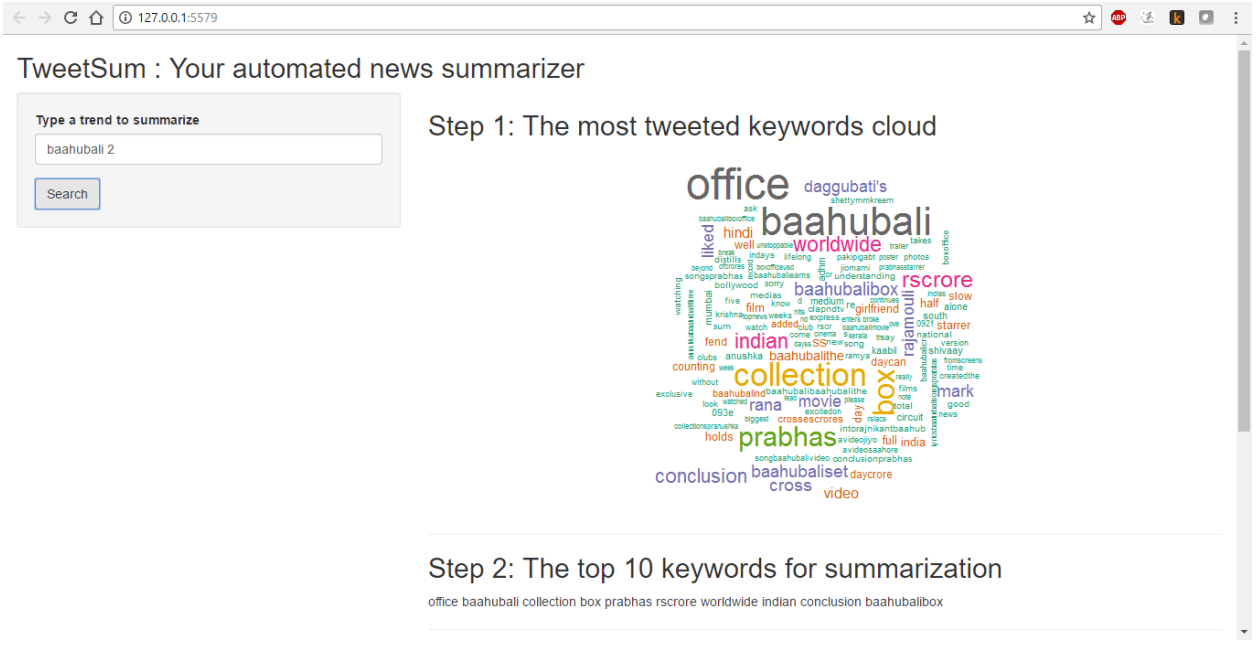


Fig (b) User Interface Description

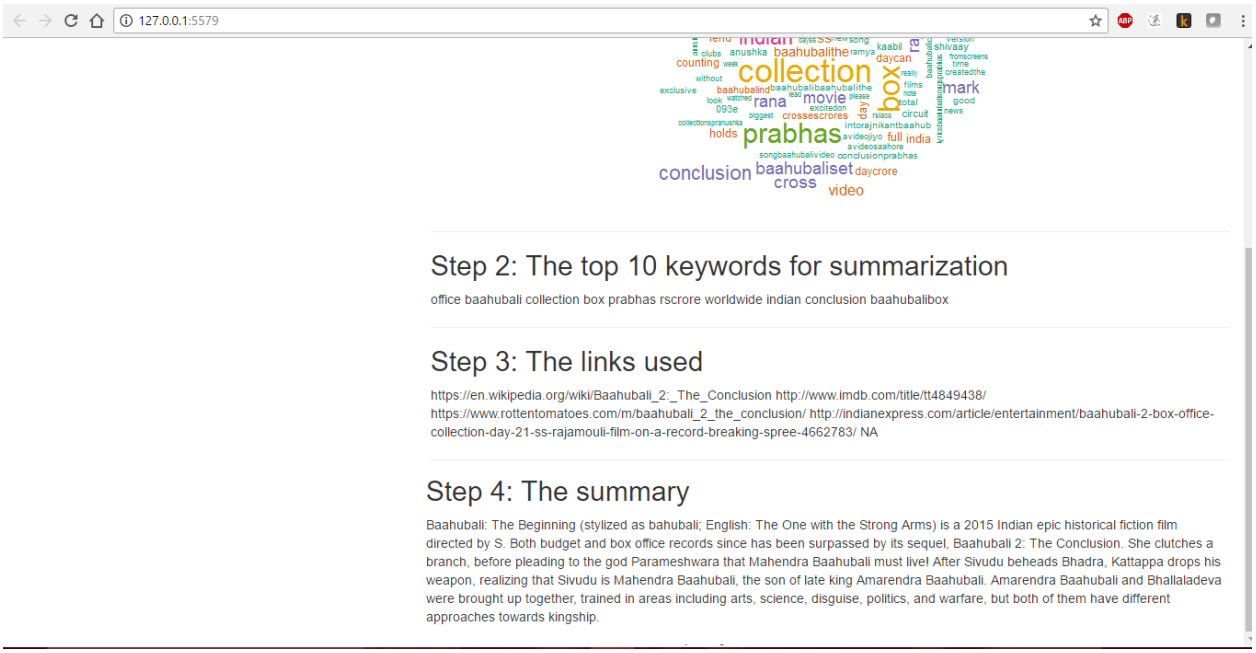


Fig (c) User Interface Description

The output generated by the system is displayed in the following levels:

- Word-cloud displaying all the top features which are spoken about most frequently on Twitter.
- Top 10 features from the Word-cloud.
- Top 10 referred links which the system chooses as the basis of our summarization.
- The actual crisp summary.

Approach

A massive amount of content, related to any subject, is present in form of various blogs, websites, social media posts, news articles etc. on the World Wide Web. The information content multiplies with every passing second, due to which it becomes increasingly inconvenient to find pertinent information. Summary of this information content will assist the user to get an overview of the content in a short span of time. This summarization entails to retrieval of information and generating a condensed interpretation of the retrieved information. Much research is present on the various techniques of summarization, but the system we propose summarizes on the basis of highest frequency key words used in tweets posted relevant to the topic searched. Our system, in brief, involves extraction of topic relevant tweets and content present on available websites, followed by generating highest frequency keywords from the extracted tweets and using them to summarize the extracted content. Our system enables the user to enter a topic for summarization and displays a summary of that topic.

1.1. System Architecture

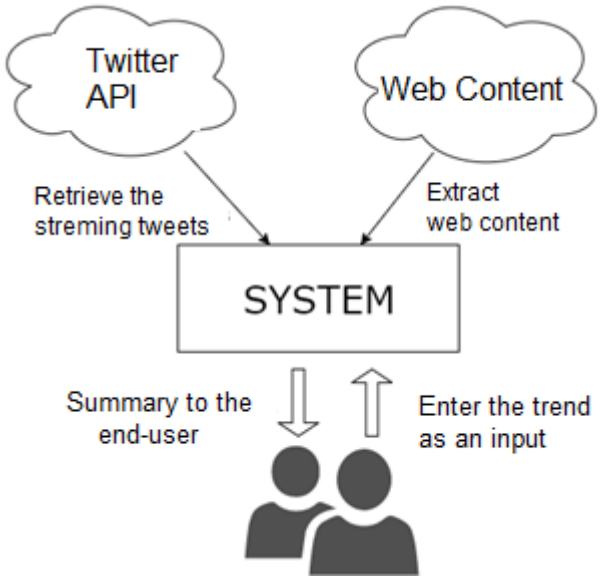


Fig 5.1 System Architecture Overview

Our system consists of two main components that deal with Twitter repository and web scraped content cloud to extract the required contents from their respective databases by the means of HTTP based APIs.

Twitter API

Access to the real time Twitter feed is required for the summarization of the trending tweets. By the means of third party application such as REST API, users can access the web interface and can extract information or perform their required task. REST API assists any application by providing access to read and write Twitter content. The application via REST API can create a new tweet, extract posted tweets, read public user profiles, obtain follower data etc. Our proposed system has used OAuth to access the REST API. One of the main reason for using OAuth is that it is application-only authentication; the application makes API request on the behalf of the users therefore the users are not required to share their login credentials with a third party application thereby maintaining user security. Our system restricts the API to retrieve tweets on a trending topic. The number of tweets retrieved with the help of this API is around 450 posts in a time span of 10 seconds. These retrieved twitter posts are stored in text format.

Web Content Scraping

This component of the system deals with extraction of all the content related to the entered search term from the web. This can broadly include Wikipedia pages, news articles, blog posts, etc. To capture a sense of all the content available related to the entered search term, the process is carried out in two levels.

At the first level is a typical 'Google Search' through the programming interface. For e.g. R programming makes use of the 'getGoogleURL' and 'getGoogleLinks' functions to search for the given topic. It mimics a normal web search but directly posts the search results on the R interface. These results typically include a Wikipedia page followed by popular and recent write-ups on the topic. These results form the basis of the summary. At the second level of processing top 10 links are selected from the Google search results and the content on these web pages is scraped for summary. The RCurl package in R programming is one of the already available packages that allows content extraction on supplying it with a URL. Hence, at this level of processing, we get all the web content available on the World Wide Web from trusted sources and pertaining to the topic.

User

The primary stakeholder of the architecture is the user. The user enters the phrase for obtaining summary and hence is the primary source for the input. Further, phrase specific summary, top tweets, data frequency cloud etc. are returned to the user.

Main Algorithm and Methodology

Our proposed system captures the context of the topic by summarizing pre-available Web content on the basis of features generated from user tweets. It hence nullifies the need for natural language processing. The entire process entails to three major steps: Extraction, Filtration and Summarization as detailed herewith.

Extraction of Tweets from Twitter

This is the first step of the system. The user enters the topic for summary on the interface which is then carried forward for extraction from twitter. Publicly available Twitter third party applications are used for this extraction of tweets. OAuth is the official API used for extraction. The API is supplied with the string variable that stores the phrase and over 450 posts are retrieved in a span of 10 seconds. The API performs a sequential search for tweets based on the phrase and hence 450 tweets containing the phrase are returned. The tweets are initially stored in a .json file. These tweets are then used by loading into a text file for further processing.

Extraction of Web content

The distinguishing feature of our proposed system is the usage of already available content to form the summary. The available content that the system uses is information available on the Internet related to that topic. To capture a sense of all the content available related to the entered search term, the system performs a Google search of the keyword entered by the user and captures web content from the top 10 links of this Google search. The pages are stored in the form of a text file and used later during the summarization step.

Filtration

The tweets collected after extraction are then processed further to remove non-English tweets, spams, rants, irrelevant data and other sources of misinformation. The system removes all the non-English tweets, removes stopping words, associates a level of significance on the tweets

by influential users, and generates a frequency count of all the words after filtration; thereby generating a set of highest repeating phrases which have the maximum weight from the collected tweets.

Stop word elimination: The process of removing certain words unnecessary to text processing is the process of stop word removal. Examples of stop words are “want”, “who”, “the” etc.

Removal of non-English tweets: The collected tweets are restricted to English language only for further processing. All the tweets in any language but English are eliminated during this process.

Removal of Twitter characters: This process rejects all the special characters (@, #) present in those collected tweets which are irrelevant for the summarization process, but occur in almost all the tweets. Hence such characters are removed so that they do not make it to the top features for summarization process.

After processing, these tweets are converted into a corpus consisting of only relevant information from the tweets so as to optimize the summarization process. Later, the corpus consisting of the collected data is converted into a data-frequency matrix which consists of the top features along with the frequency with which they are being repeated in the tweets. These features are stored in descending order with respect to their associated frequency so as to generate the top “n” features that can be carried forward for the summarization process. The topmost repeated words having the highest frequency form the base for the proposed summarization process. The output after this stage is the processed data frequency matrix.

The significance of this step is to remove the clutter from the tweets available to gain clarity on the focus area corresponding to the subject.

Summarization

As mentioned in 5.1.3.2 the content extracted from the web is stored as a document. After segmentation of that document, each sentence present in the document is represented as an individual document. The top frequency words obtained in 5.3 are then mapped to each document in order to assign weights to the documents. After the weights are assigned to each document, the top ‘n’ weighing documents are selected for the summary. The sentences present in the selected documents are printed as a summary for the keyword entered.

Example:

Top 10 links:

https://en.wikipedia.org/wiki/Donald_Trump

http://www.independent.co.uk/topic/DonaldTrump

http://www.politico.com/news/donald-trump

https://www.theguardian.com/us-news/donaldtrump

https://www.nytimes.com/2017/05/19/opinion/sunday/donald-trump-middle-east-frank-bruni.html?_r=0

https://lawfareblog.com/what-james-comey-told-me-about-donald-trump

http://edition.cnn.com/2017/05/18/opinions/trump-has-finally-got-what-he-wanted-opinion-dantionio/index.html

http://www.express.co.uk/latest/donald-trump

http://www.cnbc.com/donald-trump/

http://www.biography.com/people/donald-trump-9511238

Wikipedia Document *(The* ***example*** *uses Wikipedia as the base for summary):*

https://en.wikipedia.org/wiki/Donald_Trump

Donald John Trump is an American businessman, television personality, politician, and the 45th President of USA. Trump won the general election on November 8, 2016, in a surprise victory against Democratic opponent Hillary Clinton. Trump announced his campaign slogan, "Make America Great Again". Trump first publicly expressed interest in running for political office in 1987. In June 2015, Donald Trump launched his campaign for the 2016 presidential election, and quickly emerged as the front-runner among 17 candidates in the Republican primaries.

After Segmentation:

Document 1: Donald John Trump is an American businessman, television personality, politician, and the 45th President of United States of America.

Document 2: Trump won the general election on November 8, 2016, in a surprise victory against Democratic opponent Hillary Clinton.

Document 3: Trump announced his campaign slogan, "Make America Great Again".

Document 4: Trump first publicly expressed interest in running for political office in 1987.

Document 5: In June 2015, Donald Trump launched his campaign for the 2016 presidential election, and quickly emerged as the front-runner among 17 candidates in the Republican primaries.

Keywords	Weight of each keyword
"Trump"	5
"Election"	2
"President"	2
"America"	3

Top frequency keywords

Document	Weight
Document 1	13
Document 2	7
Document 3	8
Document 4	5
Document 5	9

Weight mapping and assignment

We need a summary of 3 lines (say), we therefore print the top three weighing documents.

Summary:

Donald John Trump is an American businessman, television personality, politician, and the 45th President of United States of America.

In June 2015, Donald Trump launched his campaign for the 2016 presidential election, and quickly emerged as the front-runner among 17 candidates in the Republican primaries.

Trump announced his campaign slogan, “Make America Great Again”.

Programming Languages Used for Implementation

TweetSum uses R programming for its implementation. The R language is an open source programming language and software environment for statistical computing and graphics that is supported by the R Foundation for Statistical Computing. The R language is widely used among statisticians and data miners for developing statistical software and data analysis. Polls, surveys of data miners, and studies of scholarly literature databases show that R's popularity has increased substantially in recent years.

The statistical capabilities in built in R coupled with packages like Quanteda allowed us to do a thorough quantitative analysis of our collected data. R also supported extraction of data through Twitter and other websites through its RCurl and XML packages. The Shiny package allowed us to build an interface for the entire project. Due to these benefits we used the R language.

Tools Used

Various Tools used for designing, programming, building and documenting the MES are:

1. R
2. Languages: HTML, CSS, Java
3. User Interface : Shiny app package
4. Microsoft Excel: for storing the tweets
5. Microsoft Word: Documentation
6. Quantitative analysis : Quanteda package
7. ROAuth : Tweets extraction
8. RCurl, XML : Web Scraping

CONCLUSION & FUTURE SCOPE

Conclusion

In today's world of explosive news and trends sweeping the entire world and the generation of plethora of views and content for every topic, summarization of trends has become the need of the hour. This summarization can be deemed useful only if it provides a contextual summary of the topic after taking into consideration the user point of view. If summarization is done giving weightage to what the users consider important then the summary can be deemed useful. This decision of giving importance to specific content from within the available content is done using user tweets pertaining to that content subject. These tweets enable the weighing of content important to the users and hence make the summary concise and to the point. By using already available contextual and concise content the system eliminates the need for Natural Language Processing (NLP) hence removing much processing needed to be done on the content. The system leverages the existing editorials and summarizes it based on user's importance. Going forward Twitter tweets could be replaced by a combination of all social media platforms capable of capturing the user reactions and can provide a more holistic view to the summary generated. This algorithm aims to fulfil the need of a quick summary of pre-existing web content based on user views and user interactions via social media websites.

Future Work

- Feedback can be taken from the users in a field after the summary is presented. This will help in improving the efficiency and performance of the system.
- Also, our proposed system is a web application. Thus in future iOS as well as Android application can be created for the same.

References

[1] Open domain event extraction from twitter (2012)
http://dl.acm.org/citation.cfm?id=2339704

[2] Twitter API: twitteR Documentation

https://cran.r-project.org/web/packages/twitteR/twitteR.pdf

[3] Web data extraction, applications and techniques: A survey (2010)
http://www2.ic.uff.br/~bazilio/cursos/progweb/material/web-data-extraction-survey.pdf

[4] XML: Tools for Parsing and Generating XML Within R and S-Plus

https://cran.r-project.org/web/packages/XML/index.html

[5] Developing an Approach to Harvesting, Cleaning, and Analyzing Data from Twitter Using R (2017)

http://isedj.org/2017-15/n3/ISEDJv15n3.pdf#page=42

[6] Automatic Twitter Topic Summarization (2014)
http://ieeexplore.ieee.org/document/7023580/

APPENDIX

I) Minimum System Requirements

- Server machine (PC, Laptop, Tablet)
- 10 GB hard disk
- 1GB RAM
- Working Internet connection
- User device
- Operating System: Windows 7/XP and above
- R
  - Quantitative analysis : Quanteda package
  - ROAuth : Tweets extraction
  - RCurl, XML : Web Scraping
- User interface: Shiny app package

II) User's Manual

The manual guides the user on how to use the application:

Step 1:

Open the desktop application.

Step 2:

Enter the trending keyword based on which you want your summary generated in the search field provided on the top left corner of the system.

Step 3:

Once you are done entering the keyword, click "Search".

Step 4:

The output generated by the system is displayed in the following levels:

- Top 10 tweets on Twitter based on the trending keyword.
- Word-cloud displaying all the top features which are spoken about most frequently on Twitter.
- Top 10 features from the Word-cloud.
- Top 10 referred links which the system chooses as the basis of our summarization.
- The actual crisp summary.