ISSN: 2278 – 7798
International Journal of Science, Engineering and Technology Research (IJSETR)
Volume 5, Issue 4, April 2016

Text Mining using Side Information from Twitter Tweets

Udhaya Sandhya [1], **Dr. S. Revathi** [2]

***Abstract*— Twitter is a popular social media application which allows users to interact with each other using short messages. These messages are called tweets, they are large repository of social data that is a great starting point for the process of text mining because of its public availability and openness. Side information in tweets such as username and time are useful for text mining process. Trending topics, user comments, updates, reviews all over the world from users having different background are collected from the Twitter Server using Twitter Application programming Interface. These tweets are processed using Java Script Object Notation format based serialization techniques and a live dataset is built. Frequent association technique is used to extract the most frequently associated tweets that contains words that are searched. Then sentiment analysis is performed on all tweets in the live dataset to determine the positivity of each tweet. Content based clustering is performed to classify the tweets based on side information such as username and positivity. The results of this text mining application can reveal interesting patterns, which are strongly related tweets according to the search words given as input to the mining program. Clustered tweets based on positivity can reveal unique relationships among desired tweets. This Twitter Tweets based mining application has wide usage in key decision making that can be implemented in various policy making fields where people feedbacks are primary factors.**

***Index Terms*— Clustering, Java Script Object Notation; Sentiment Analysis, Twitter API**

I. INTRODUCTION

Twitter is a social web and mobile platform where people can communicate using short 140 character messages that usually contain user's ideas, reviews on current products, and general public comments on recent happenings in that particular region or all over the world. It is a reliable source of social data that is a good starting point for social web based mining because of its availability for public in the web world. Well processed Twitter data is interesting because tweets are posted at rapid speed of happening, real time scenarios and situations are commented rigorously by users all over the world.

Manuscript Received April ,2016

***Udhayasandhya.P** Deptartment of Computer Science and Engineering, B S Abdur Rahman University, Chennai, India.*

***Dr. S. Revathi.**, Deptartment of Computer Science and Engineering, B S Abdur Rahman University, Chennai, India.*

Since the social network is the latest medium for Information as well as knowledge sharing and people collaboration, it has become a mandatory requirement in daily life. Using side information to extract interesting patterns of relationship or other forms of intelligent information from twitter, can provide a great scope in the knowledge and intelligence gaining process which is useful for taking decisions in several areas such as government strategic and tactical departments, operational sectors of corporate organizations for various business needs. Side information may be of different kinds, such as document information, the links in tweets, user access patterns from web logs, or other non-textual attributes which are embedded into document. Text documents typically occur in the context of a variety of applications in which there may be a large amount of other kinds of attributes or meta- information which can be vital factors in the clustering process. Several text documents contain links among them, which can also be considered as attributes. Such links contain useful information for mining purposes. Side Information that are widely used in this project are username of the twitter account where the tweets are read from the user time line, date and time of the tweets. The required knowledge from the tweet data set using the side information is extracted and consensus is gained. This process of gaining an intelligence is the actual mining process that uses frequent association rule learning technique to determine the tweets that contain the respective words that are searched, in the whole tweets data set. There is a 2nd mining process that involves performing sentiment analysis on all the tweets data set and calculating the positive percentage of tweets. This percentage determines the positivity of the tweets that are later stored in the twitter database. From the positivity value of each tweets, the whole tweet data set is clustered.

Sentiment analysis is the process of classifying the polarity of a given text at the document, sentence, whether the given opinion in the tweet is positive, negative, or even neutral. Beyond polarity, sentiment analysis looks at emotional states such as "confused," "dull," and "happy." A scaling system is used in which the words commonly associated are having a negative, neutral or positive sentiment value on a -10 to +10 scale and when a piece of unstructured text is analyzed using natural language processing with the sentiment analysis algorithm it finds out

the negativity and positivity of the given statement by analyzing the words that are present in the input source.

After the sentiment analysis phase is completed and once the positivity of every tweet is determined then based on the positivity percentile, all the tweets in the data set is clustered by implementing a content based clustering methodology. In this context, clustering is the process of organizing tweets into classes whose members are similar in positivity and username of the twitter account. A cluster is therefore a collection of items which are "similar" between them and are "dissimilar" to the items belonging to other clusters. In certain clustering technique the relating criterion is distance, two or more objects belong to the same cluster if these objects are related according to a given distance, and such a clustering based on geometrical distance is called distance based clustering. Conceptual clustering is where two or more items belong to the same cluster if there is a concept common to all that objects are grouped according to their descriptive concepts, not according to simple similarity measures. The primary goal of clustering is to performing grouping of the unlabeled data.

II. RELATED WORK

Users sharing short messages on social media sites such as Twitter and Facebook have emerged as strong, real-time medium of information and knowledge sharing on the Web. These short messages are bound to reflect a variety of events in real time. Twitter particularly is well suited as a source of practical event content. Here the twitter messages are analyzed to provide the difference between real world event and non-events [1]. This technique relies primarily on a rich family of aggregate statistics of message clusters that are similar. Large experiments were done on the millions of Twitter tweets to depict the effectiveness of the method for surfacing real time event content on Twitter.

Twitter based event detection and event analysis systems helps to find new and unique events that can be analyzed with an intention to find spatial and temporal patterns, and to identify events that are recurring [3]. Detailed information is given for the implementation of functions that can crawl like automatic software bot programs, classify and arrange tweets based on rank and extract geographical locations from tweets, and display some interesting results of our system. Twitter has several unique advantages that differentiate it from other social networking applications, or other information channels.

The tweets are created in real-time on the fly. Tweets have the limit of 140 characters and with advent of popular twitter mobile application, users can tweet and retweet instantly. Next advantage is that the tweets have a wide coverage over events. Twitter has millions of general users and trusted accounts of authorized news agents, organizations and celebrities, they constantly publish new tweets. Thus, tweets cover nearly every aspect of social web life, from international breaking news, local issues, to product reviews. Third advantage is the tweets are not isolated. They are associated with rich side information [1].

Normally Social networks contain a large amount of text content over time due the continuous and rigorous interaction between users. Mining of such social web based data set is more challenging than conventional text streams, because of the presence of both text content and other visually rich media content in the tweet, this issue of event detection is also strongly related to clustering [4]. Two similar problems of clustering and event detection in social streams were discussed in reference [5]. The supervised and unsupervised states for the event detection problem must be studied to understand the technical complexities. The Experimental results that are produced [6], illustrates the fact that the effectiveness of event based network structure in event discovery is purely based on arbitrary user tweets.

Live event detection problem is related to the subject of tweets event detection and tracking. The associated clustering of tweets problem is also related with stream clustering and tries to find out new trends in the text mining for mining of tweets. The key idea is to capture all the important happenings in real life as events in the form of temporal bursts of closely related tweets in a social stream.

III. PROCESSING AND MINING TWEETS

The Twitter APIs provide programmatic access to read and write Twitter data, write a new Tweet, read author profile and read follower data, and has more versatile functions.. It identifies Twitter applications and users using age old access token concepts. A HTTP web Request is used to send the automatically generated keys to the twitter service and the token is received using HTTP web response class provided by the Microsoft .NET framework 4.0. Multi-threading is used to retrieve the tweets as dynamic Java script object notation (JSON) packets which are converted back into routine html. Data sets using the twitter application programming interface (API) since the messages are communicated through an asynchronous medium and the tweet reading program iterates through timeline results in order to build a more complete list of tweets with respective side information [8].

A. Preprocessing Tweets

All the tweets that are read, are stored into the twitter database by creating a live data set that have the primary attributes such as twitter username, tweet, date and time. The tweet attribute is parameterized using SQL Parameters that help in effective text processing in database since the tweets

contain special characters such as "#,%&,',"",\". So during compilation the database doesn't thrown an error.

A windows application form is used to present the tweets and side information from the database. This functionality is provided by the data grid view component which takes a data adapter as its primary input data source. The SQL data adapter serves as a bridge between the Data set that contains the twitter data source and SQL Server database for retrieving and saving data. When the SQL data adapter fills a Data Set with twitter data source it creates the necessary tables and columns for the returned data if they do not already exist. For every column that is presented in the data grid view from the SQL data adapter using the multi select query the column width is adjusted programmatically and the result is displayed using the grid view.

B. Mining Tweets

After the preprocessing phase is completed, the tweets are loaded into the mining application using data grid view. Every tweet is loaded with twitter data and side information into the twitter dataset by adding every row that is presented in the data grid view from the SQL data adapter using the multi select query with joining multiple data tables and the result is displayed using the grid view. The cells of each row within the data set is loaded into the mining application. These cells contains the tweet data. Mining application performs frequent association rule learning mechanism on the complete data set that is pre-loaded into data grid view. The number of words for searching process in determining the frequent association mechanism is defined earlier that can be changed during runtime. After the frequent association process the strong tweets are presented in a new data set called "Twitter-FARLM" and this is displayed in the data grid view. Also the tweet counts for the new data set is calculated. Support and confidence are determined by implementing apriori algorithm concept.

For extracting useful information from twitter dataset the concept of apriori algorithm is implemented. It is a classic algorithm for learning association rules. It is designed to operate on databases involving transactions for e.g. market basket analysis. The whole idea of the algorithm and data mining in general is to extract useful unknown information from large amounts of data. For e.g, there is a tweet in the live dataset built, which contains "Chennai flood" and also contain "Milk required".

From the above given words it can be inferred that the tweets contains about "Chennai flood" also tends to have "Milk required at the same time which is acquired from the association rule below. Support is the percentage of task-relevant tweets for which the pattern that is inferred is true. Support formula is given in (1).

$$Support(w^1, w^2 \ldots \rightarrow w^n) = \frac{\sum_1^n (w^1 \bigcap w^n)}{\sum_1^n Tweets} \quad (1)$$

In this formula w is the words that is to be searched in any given tweet. Minimum 1 word is required to search in the given tweet. In word w, the nth value's practical limitation is 4. In this case, for example w1 is "Chennai Floods" which is the major issue concerned for mining in this instance and w2 is "#Tambaram", the string that points out a location related to the word w1. Confidence is the measure of certainty or trustworthiness associated with each discovered pattern as shown in (2).

$$Confidence(w^1, w^2 \ldots \rightarrow w^n) = \frac{\sum_1^n (w^1 \bigcap w^n)}{\sum_1^n w^1} \quad (2)$$

The frequent association rule learning techniques for twitter tweets from tweet database aims to find the rules which satisfy both a minimum support threshold and a minimum confidence threshold.

C. Uclassify Algorithm

Sentiment Analysis is implemented using uclassify algorithm that is done with the support of uclassify HTTP server [9]. A uniform resource locator (URL) request is sent to the uclassify server with the API key that is automatically generated for each of uclassify program user and the tweet, encoded using standard HTML-encoding mechanism. Then the positivity result is computed for the clustering application. The tweets are categorically classified without any labels after detection phase and this result is useful to compare 2 major parameters, in this context known as side information such as username and positivity value of every tweet present in the database. After this, the tweets are clustered based on their content, here the content is the positivity value. This content based clustering is implemented from the concept of COATES algorithm [1].

D. Dynamic Reporting

Reporting is done by the reporting module which presents the inferred knowledge and Intelligence from the mining process that can play a major role in decision making process depending on context it is being used. Report is generated based on the number of clusters available, with options to view the tweets and its side information. Report presents the statistics yielded by the two mining processes, frequent association rule learning and sentiment analysis based clustering.

IV. IMPLEMENTATION

A. Reading Tweets

The Twitter API in the Twitter server has several methods, such as GET statuses of user timeline, GET statuses from home timeline and GET search tweets, which return a timeline of Tweets data. Such timelines can grow very large, so there are limits to how much of a timeline a client application may fetch in a single request. The tweet reading program iterates through timeline results in order to build a more complete list of tweets and side information. The Twitter GET status "user_timeline" is called by the twitter class file that returns the tweets in prescribed format for the mining process. This method returns a collection of the most recent Tweets posted by the user indicated by the "`user_id`" parameters. User timelines belonging to protected users may only be requested when the authenticated user either "owns" the timeline or is an approved follower of the owner. The timeline returned is the equivalent of the one seen when viewing a user's profile on twitter.com. The key parameters that are used by the tweets reading program is shown in Table I.

Table I. GET user time line parameters for side information with sample values

Variables	Description
user_id	The ID of the user for whom to return results for.
Created_at	Time and Date
Profile_image_url_https	Picture in Tweet
Followers	Number of followers for the user
Time_zone	Time zone like India, USA, Canada
Description	Content of the tweet
Html	Html code for the tweet

The reading module sends the tweets and side information into the collection in the main program using call by value of the method. The mining front end application stores the tweet data in a SQL Server database using the .NET Framework Data Provider for SQL Server as shown in fig 1.

B. Frequent Association Rule Learning

The tweets are loaded into a data set in the data grid view. The data set has the following attributes for every tweet or record by default.

- Username
- Tweet data
- Tweet Date and Time

In the twitter database, tweets table and the tweet date and time table are related using primary key relationship and it is queried using multi select query. The records in the data set are converted into string and then the sentence is chunked into words and the all the words are inserted into a temporary list box.

Then the match words are searched in the list box iteratively if it satisfies the condition then a flag variable is set to '2' in the data table to indicate the presence of the word. Then this process continues iteratively until every tweet is processed. At last the tweets that are having the flag set to 2 is said to have strong frequent association than the rest of the data set and the filtered tweets are displayed as shown in fig 2.

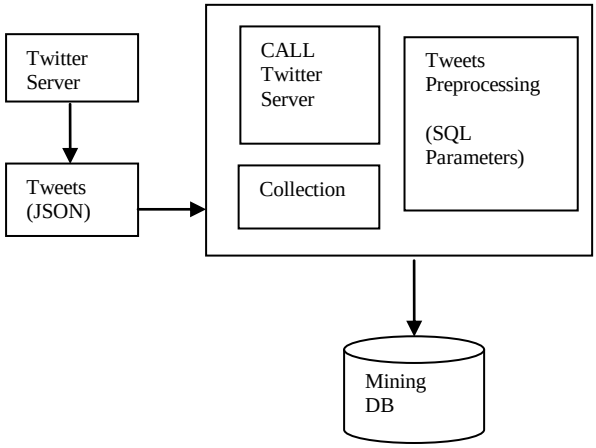

Fig. 1. Asynchronously reading tweets from twitter API using JSON and storing into database after pre-processing tweets.

Support and confidence are calculated using the match words against every tweet and their percentile is determined for the whole live twitter data set. After this step only the filtered tweets are made into a new data set and it is presented into the data grid view.

C. Sentiment Analysis

Sentiment analysis phase implements the uclassify analysis algorithm that works efficiently on large text data set with simple URL requests. Initially the Clustering application loads all the tweets from the database to the windows form application then each row is iterated and the tweets are sent to the uclassify server, with the API key. Creating an uclassify account and a sentiment analysis classifier generates an API key and it is sent across to the server using URI request and the web service performs the analysis and yields the result in XML format. Then using Xlement class provided by .NET framework,each node and

its child node is searched to find the specific attributes and corresponding values that fetch the sentiment analysis. The resulting string containing the positivity value is type casted into double for computation purpose.

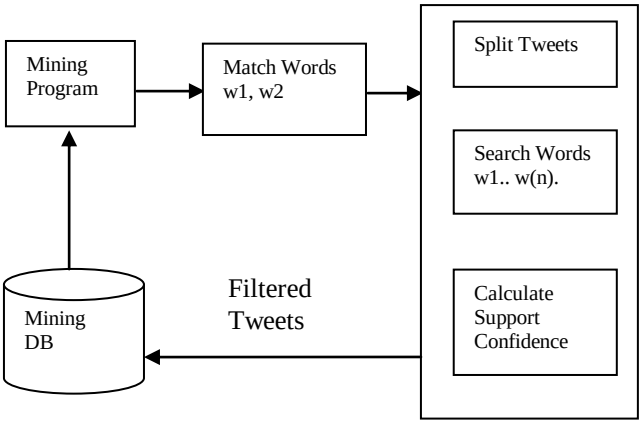

Fig. 2. Frequent Association learning using match words.

D. Content Based Clustering

After the positivity of every tweet is determined using uclassify algorithm all these values are stored in new data table in the twitter database, that is linked to the main data table using primary key relationship. All the tweets are classified using content based clustering technique derived from COATES algorithm, hence the primary objective is to group the data using side information such as username and positivity value as shown in fig 3.

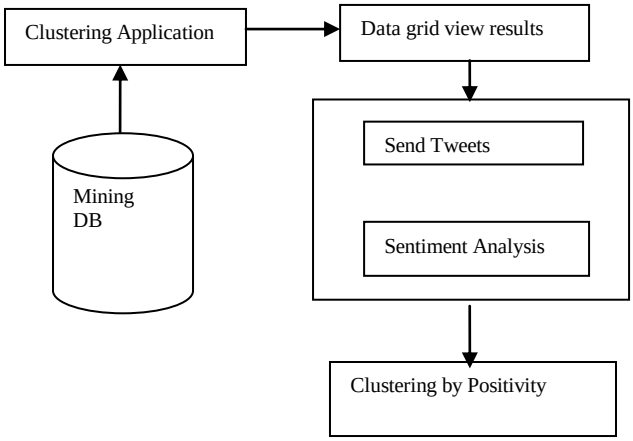

Fig. 3. Clustering using sentiment analysis result.

V. EXPERIMENTAL RESULTS

Both the GET tweets method and the get time method in the reading module returns raw format of the twitter API's result which is stored in dynamic variable format for both the values of tweets and side information. Serialization allows the HTTP twitter result from the server to save the state of an object and recreate it as needed while The Language Integrated Query (LINQ) is an expression that retrieves relevant twitter tweets and side information from the JSON data source. LINQ simplifies the process of selecting specific columns from the vast twitter result that is deserialized in earlier steps of the reading module. LINQ offers a consistent model for working with data across various kinds of data sources and formats. The LINQ query operations has three distinct actions before passing to the storing module.

- Obtain the data source (Twitter JSON)
- Create the query (Side information)
- Execute the query (Sent to the main program's IEnumerable collection)

Finally the real time Data set containing processed twitter tweets 577 of 2 twitter accounts, "actorvijay" and "nytimes" and their respective side information from the twitter server which is stored into 2 tables in the back end SQL server database is retrieved using the retrieving module for building the real time data set that can be used for live mining process as shown in fig 5.

ISSN: 2278 – 7798
International Journal of Science, Engineering and Technology Research (IJSETR)
Volume 5, Issue 4, April 2016

Fig. 5. Sample Tweets and Side Information displayed in console.

This real time data set is a specifically built repository of information that is generated by live comments and views by people all over the world, is a vital input for intelligent and automated decision making processes. Once the miner program is started with the twitter data record having 577 sample tweets after the sample match words w1 and w2 are entered, in this sample, w1 = "Less" and w2 = "than", there were 12 matches in the tweet data set. With having support value as 2.08 % and Confidence value as 100%. In the clustering phase the tweets are all loaded in to the data grid view with NULL values in the sentiment analysis and positivity columns in the clustering data table. The uclassify's URL is encoded and the request is made to the sentiment analysis web service. Then the result for each tweet is sent back to the client. The sample XML response of a tweet record in the data set is shown in fig 6.

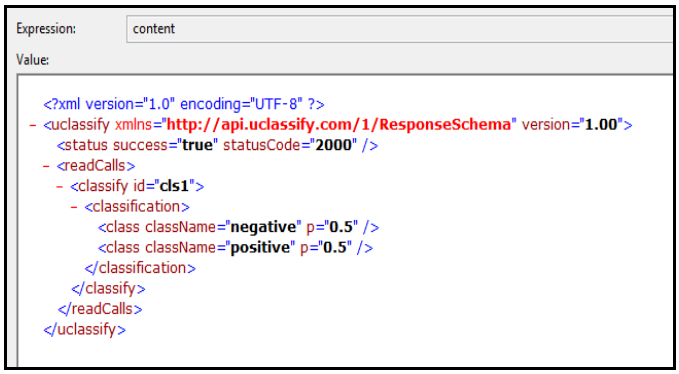

Fig. 6. Sample XML Response with negative and positive values for tweets

Every tweet is mapped with the positive value, a floating point value and converted into percentage. All the tweets with complete side information such as username, date, and time and positivity value are passed on to the clustering application that groups the tweets based on the positivity values as shown in fig 7. All the clustered groups are assigned unique colors in graphical representation. Tweets are grouped into different label less clusters based on 5 scales as shown in Table II. All these data are forwarded to the reporting wizard then the report is generated as a pdf document.

Table II. GET user time line parameters for side information with sample values

Values	Clusters
90.0 +	Group A
70.0 – 89.9	Group B
50.0 – 69.9	Group C
30.0 – 49.9	Group D (Low)
0 – 29.9	Group E (Very Low)

username	twitterdata	tweettime	sentimentanalysis	value
actorvijay	#DubTheriStep https://t.co/2oghNGjWLb	Tue Mar 22 17:2...	positive=50%	50
actorvijay	https://t.co/nlxnCB8dUP	Mon Mar 21 16:...	positive=99.426...	99.4265
actorvijay	#TheriTrailer https://t.co/0sypTk4GYy	Sun Mar 20 14:4...	positive=50%	50
actorvijay	RT @Samanthaprabhu2: https://t.co/7j3...	Sun Mar 20 06:3...	positive=99.616...	99.6168
actorvijay	#Raangu track - https://t.co/B4NbeKW...	Sun Mar 20 06:0...	positive=50%	50
actorvijay	#Chellaakutty lyric video : https://t.co/r...	Sun Mar 20 05:3...	positive=50%	50
actorvijay	#EenaMeenaTeeka track https://t.co/4O...	Sun Mar 20 05:0...	positive=50%	50
actorvijay	#EnJeevan track - https://t.co/3upDCSr...	Sun Mar 20 04:3...	positive=50%	50
actorvijay	#jithujilladi track - https://t.co/tlH1ta7w...	Sat Mar 19 18:3...	positive=50%	50
actorvijay	#jithujilladi single track from 12am .. 15 ...	Sat Mar 19 18:1...	positive=50%	50
actorvijay	#TheriTrailer from tomorrow evening !! ...	Sat Mar 19 13:0...	positive=50%	50
actorvijay	#jithujilladi song poster https://t.co/ILQ...	Fri Mar 18 09:42...	positive=50%	50
actorvijay	#ChellaakuttyPoster2 #Theri https://t.c...	Thu Mar 17 09:...	positive=50%	50
actorvijay	#Chellaakutty Song poster https://t.co/l...	Wed Mar 16 10:...	positive=50%	50
actorvijay	#EenaMeenaTeeka https://t.co/dlwC8tg...	Tue Mar 15 09:3...	positive=50%	50
actorvijay	https://t.co/N6TGfZOKJe	Mon Mar 14 08:...	positive=99.426...	99.4265
actorvijay	#TheriNewPoster https://t.co/31U8UFK...	Sun Mar 13 09:5...	positive=50%	50
actorvijay	#TheriAudio from March 20 !! https://t....	Fri Mar 11 12:25...	positive=50%	50
actorvijay	2 Million + views in just 24 hours. Twink...	Fri Feb 05 18:24:...	positive=48.199...	48.1999
actorvijay	Here it is !! #TheriTeaser - https://t.co/Jf...	Thu Feb 04 18:3...	positive=81.162...	81.1628
actorvijay	Less than 10 mins for #TheriTeaser	Thu Feb 04 18:1...	positive=7.7417...	7.74171
actorvijay	#TheriTeaser from Feb 5th https://t.co/...	Mon Feb 01 16:...	positive=50%	50
actorvijay	#Theri https://t.co/9gQNF25dAI	Sat Jan 16 11:34...	positive=50%	50
actorvijay	https://t.co/0vWTKx99HU	Thu Jan 14 08:0...	positive=99.426...	99.4265
actorvijay	#தெறி #Theri Poster 2 https://t.co/Rr...	Wed Nov 25 11:...	positive=50%	50
actorvijay	#தெறி #Theri https://t.co/ldGzKKfLOd	Wed Nov 25 11:...	positive=50%	50
actorvijay	#Theri !! First look at 4:30pm.	Wed Nov 25 10:...	positive=50%	50
actorvijay	RT @gvprakash: Ilayathalapathy sings hi...	Thu Oct 29 05:1...	positive=99.714...	99.7149
actorvijay	Here it is !! #Jingiliya Song Promo - http...	Thu Sep 24 09:0...	positive=81.162...	81.1628
actorvijay	Here is the new exclusive promo of #Pul...	Wed Sep 23 14:...	positive=95.526...	95.5269
actorvijay	#Puli releasing Worldwide on Oct 1st ht...	Tue Sep 22 16:3...	positive=50%	50
actorvijay	RT @SKTStudios: Entire team of #Puli w...	Wed Sep 02 10:...	positive=71.110...	71.1109
actorvijay	http://t.co/AjhbkCpztf	Wed Sep 02 10:...	positive=99.980...	99.9802
actorvijay	http://t.co/UgCSpxondX	Fri Aug 21 11:13...	positive=99.980...	99.9802
actorvijay	http://t.co/AkGxChlFjn	Fri Aug 21 06:22...	positive=99.980...	99.9802
actorvijay	Here it is !! The Trailer of #Puli https://t....	Wed Aug 19 18:...	positive=80.591...	80.5917
actorvijay	Less than 90 mins for #PuliTrailer !!	Wed Aug 19 17:...	positive=0.9417...	0.941772
actorvijay	Wishing everyone a Happy Independen...	Fri Aug 14 18:31...	positive=94.57%	94.57
actorvijay	#PuliTrailer from Aug 20th!! http://t.co/...	Thu Aug 13 10:...	positive=50%	50
actorvijay	#PuliAudioLaunch http://t.co/BSjb6FkRfn	Sun Aug 02 19:...	positive=50%	50
actorvijay	Now its our turn to work relentlessly an...	Thu Jul 30 08:02...	positive=84.028...	84.0287

Fig. 7. Positivity results based on sentiment analysis with tweet data records.

VI. CONCLUSION

The primary objective of this thesis is to design a system that extracts required knowledge or new interesting patterns from the tweet data set that built on the fly using side information such as username, date, time and gaining a consensus, that can be a vital factor in decision making, also such patterns can be useful input to decision support systems. Frequent association rule learning is implemented to find the tweets that have strong associated match words given in the tweets by live user comments and reviews. Support and confidence values are calculated for whole twitter data set. Content based clustering is done to group the tweets based on the positivity of every tweets after performing sentiment analysis on the whole data set. All the grouping is based on the username and the positivity. The Frequent association technique must be improved with much better grammar and language accuracy while extracting the patterns from the tweets. Efficient methods need to be developed to extract the patterns by considering special characters in the tweets, where generally social media messages are short and will contain #tags, @username. For the clustering purpose, the COATES algorithm can be well implemented in both the content based clustering and anomaly attribute based clustering by extracting side information since COATES algorithm is specifically made for clustering data using side information.

REFERENCES

[1] Charu C. Aggarwal, Yuchen Zhao and Philip S. Yu, "On the Use of Side Information for Mining Text Data", *IEEE Transactions on Knowledge and Data Engineering*, Vol. 26, No. 6, pp. 1415-1429, June. 2014.

[2] Masaki Kohana, Shusuke Okamoto and Masaya Kaneko, "A Clustering Algorithm Using Twitter User Biography", 16th *International Conference on Network-Based Information Systems*, pp. 432-435, September. 2013.

[3] Hsin-Min Lu and Chien-Hua Lee, "A Twitter Hashtag Recommendation Model that Accommodates for Temporal Clustering Effects", *IEEE Intelligent Systems*, Vol. 30, No. 3, pp. 18-25, February. 2015.

[4] Xiaowei Xu, Martin Ester, Hans-Peter Kriegel and Jorg Sander, "A Distribution Based Clustering Algorithm for Mining in Large Spatial Databases", *Proceedings of the 14th International Conference on Data Engineering*, pp. 324-331, February. 2013.

[5] Lian Duan, Deyi Xiong, Jun Lee and Feng Guo, "A Local Density Based Spatial Clustering Algorithm with Noise", *IEEE International Conference on Systems, Man, and Cybernetics*, pp. 978–986, November. 2007.

[6] Rafeeque and P.C. Sendhilkumar S., "A Survey on Short Text Analysis in Web", *Third International Conference on Advanced Computing (ICoAC)*, pp.365-371, December. 2011.

[7] Hongyun Cai, Zi Huang, Divesh Srivastava and Qing Zhang, "Indexing Evolving Events from Tweet Streams", *IEEE Transactions on Knowledge and Data Engineering*, Vol. 27, No. 11, pp. 3001-3015, June. 2015.

[8] Twitter documentation available in link, http://www.dev.twitter.com

[9] Uclassify documentation available in link, http://www.uclassify.com/docs.