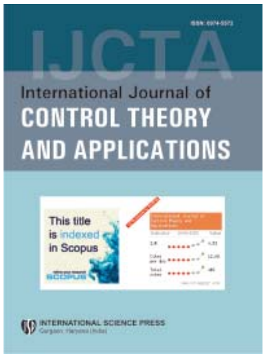


International Journal of Control Theory and Applications

ISSN : 0974–5572

© International Science Press

Volume 9 • Number 42 • 2016

Twitter Mining for Categorized Multiple Event Detection

Devi Vinod[a] and Jisha P. Abraham[a]

[a]*Department of Computer Science and Engineering, M.A College of Engineering, Kothamangalam, Kerala, India*

E-mail: devivinod707@gmail.com, jishaanil@gmail.com

Abstract: Twitter is online news and social networking service where users can post and read messages that are not more than 140 characters called "tweets". Twitter is used to find the latest news and world events faster. Every day around 400 million tweets are sent worldwide and twitter has become one of the rich sources of information's. Currently there are some existing works are done in the area of social data mining or twitter mining for the detection of events but they follows keywords attributed to an event, and these methods are searching for specific event, for example haze detection, or traffic notification etc, so the benefits are limited. Here a more advanced semantic based NLP is proposed for finding categorized multiple events detection. Since the regular algorithms cannot be used for automatic categorization of multiple events simultaneously, our only option is to device an algorithm based on user specified categories that accepts a list of categories and events prior the categorization. for this algorithms meant for clustering or more text oriented algorithms such as TF-IDF can be used in conjunction to our prioritization approach to implement our goal and also proposes a method for checking the genuine tweets to avoid unwanted tweets and thereby improving the accuracy and efficiency. Live twitter data is used for the real time event detection.

Keyword: *Big data; Hadoop; Spark; TF-IDF; Categorization.*

1. INTRODUCTION

Twitter is an online news and social networking service where users post and read short 140-character messages called "tweets". Registered users can post and read tweets, but those who are unregistered can only read them. Every day, around 400 million tweets are sent worldwide, which has become a rich source for detecting, monitoring and analysing news stories and special (disaster) events. In 2016, Twitter had more than 310 million monthly active users. In the big data era, scientific and social data could complement each other for enhanced data analysis and scientific discovery. For example, when thick smog and mist shroud some areas, we could see an increasing social engagement from the Web and online social media, either commenting or communicating the environmental situation. The event detected from the social data could alert and trigger analysis processes on scientific observation data, or confirm results from scientific analysis. The whole process could be accomplished using information technologies and infrastructure services. Existing research within this field follows keywords attributed to an event, monitoring temporal changes in word usage and are detecting only a specific events. The objective of our approach is to introduce a new approach for detecting categorized multiple events using a customized algorithm with advanced semantic search and here only considered the genuine tweets. Existing algorithmic approaches for detecting a specific event based on following key words attributed to an event is less efficient and it works only for single event detection so many of the special events are get avoided. Here multiple categories of events are detected. Since the regular existing algorithms cannot be used for automatic categorization of multiple events simultaneously, our only option is to device an algorithm based on user specified categories that accepts a list of categories and events aprior the categorization, for this algorithms meant for clustering or more text oriented algorithms such as TF-IDF can be used in conjunction to our prioritization approach to implement our goal and also proposes a method for checking the genuine tweets to avoid unwanted tweets and thereby improving the accuracy and efficiency. Live twitter data is used for the real time event detection.

A social network is a social structure. Social networking data are utilized in a variety of ways, such as in business marketing and location-based services etc. The social networking on mobile platforms such as mobile phones, with up to 6 billion mobile-cellular subscriptions, greatly facilitate the communication of real time and customized information, and ignites strong interests in academic research on the mobile geoweb[1]. The network of social media users has been considered to be a low-cost “Geo-sensor” network [2]. In the past several years, the number of social networking websites has increased significantly. Its goal is develop a new approach for detecting multiple events using social data in low cost.

2. RELATED WORK

Some works are done in the field of image processing’s to detect natural events by using satellite images. Main drawback of this approach is the high cost for the performance. Natural events like haze etc can be detected using this method. The aerosol optical thickness in the visible spectrum is a surrogate for fine aerosol concentrations, especially under pollution conditions with low mixing height, and can be measured with high ground sampling density with the help of satellite sensors for the detection of haze. The SIPHA code was developed for such application on high resolution satellite images and allows quantification of the aerosol optical thickness over land, snow and sea. The code compares multi-temporal satellite data sets and evaluates radiometric alterations due to the optical atmospheric effects of aerosols. But the main drawbacks are that they are costly and require high video quality.

To detect events from social data , there are some works are done using the concept of space time scan statistics(STSS)[3] for detecting spatial and temporal clusters, because clusters occurs only if a there is a relevant cluster occurred. Space-time scan statistics (STSS) is used as an alternative method for event detection. This technique looks for clusters within the dataset across both space and time, regardless of tweet content. It is expected that clusters of tweets will emerge during spatio-temporally relevant events, as people will tweet more than expected in order to describe the event and to spread information. The special event used as a case study is the 2013 London helicopter crash. A spatio-temporally significant cluster is found relating to the London helicopter crash. Although the cluster only remains significant for a relatively short time, it is rich in information, such as important key words and photographs. The method also detects other special events such as football matches, as well as train and flight delays from Twitter data. These findings demonstrate that STSS is an effective approach to analyzing Twitter data for event detection.

Framework based approach already exist, SDI approach[4] that use the users as sensor perspective, the tweet content, as special kind of sensor data, could be mined and fused in the Sensor Web environment, allowing social data to be used inside a spatial data infrastructure (SDI). In SDI, Sensor Web technologies can provide real-time or near real-time spatial data to support timely decision-making. The standards-based interoperable architecture adopted in big data analytics provides a distributed and coordinated data analysis environment, where various big data analysis functions are exposed as services and accessed through standard protocols [5]. The main drawback is that it is specific to event detection and based on a keyword based event as attribute, that requires more prior knowledge about that specific event.

3. PROPOSED WORK

Twitter mining[6] for categorical multiple event detection is a new approach for detecting multiple events from social data. The existing works are focusing to detect only a specific single event and the search is based on the keyword attributed to the event and that requires more detailed prior knowledge about the event. Area of social data mining is really promising because of the increasing use and ease of availability of the internet and social Medias among the common peoples. Geo-web can be consider as a low cost network, so event detection from social data or twitter data can be done easily and with low cost.

The objective of our approach is to introduce a new approach for detecting categorized multiple events using an algorithmic approach with advanced semantic search and propose a method for checking the genuinity of tweets because about six percentages of the total tweets are spam tweets. Existing algorithmic approaches for detecting a specific event based on following key words attributed to an event is less efficient and focusing only to a single event.

3.1. Architecture

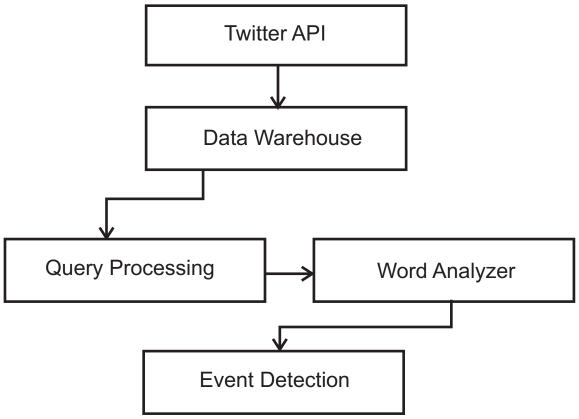


Figure 1: Architecture

This is the architecture of the proposed system. Here live twitter stream data’s are used for the twitter mining purpose. Real time data ensure the detection of real time events. Real time public generated data’s are available from the open twitter API’s if having a valid twitter account. Using a valid twitter account it is possible to access the real time twitter data based on the access permission of that account. Apache spark is used as the framework. Spark is a fast in-memory cluster computing framework for large scale data processing. After configuring the twitter and spark application, live twitter data is streamed and it used for the further processing’s. Data’s are stored in data frames in json (Java Script Object Notation) formats. Json format provide data’s to be in a organized format which in human readable and easily accessible form. In data analysis it goes for the word analyzer phase and checks for the occurrence of the event.

3.2. IMPLEMENTATION DETAILS

Modules of the proposed systems are following:

1. **Data Collection :** To improve the efficiency, real time twitter data is collected for the processing and detection of social events. Using Flume, we can fetch data from various services and transport it to centralized stores (HDFS and HBase). This chapter explains how to fetch data from Twitter service and store it in HDFS using Apache spark streaming. Framework used is apache spark. Apache Spark is an open source cluster computing framework for processing big data. Data collection is the first part of the project. This is divided into two parts, the twitter module and the Storage module.

Twitter: This module is responsible for creating a twitter application so that it provides the credentials for our analysers to access the real time data and store it as and when required. For creating twitter application first we need a twitter account. The method of authentication we use is Oauth. Here twitter and spark applications are configured and linked. Twitter data's are streamed using the category names as query keywords and the data's get categorized.

Data Storage: This module uses Apache Fume, along with HDFS and PIG to collect data and store it in NoSQL / HDFS format. The data is stored so that it can be accessed for further processing. Here data’s are stored in json format by converting the bulk row data to human readable and organized json format.

Data collection module has the following steps:

a) Create a twitter Application

b) Install / Start HDFS

c) Configure Flume

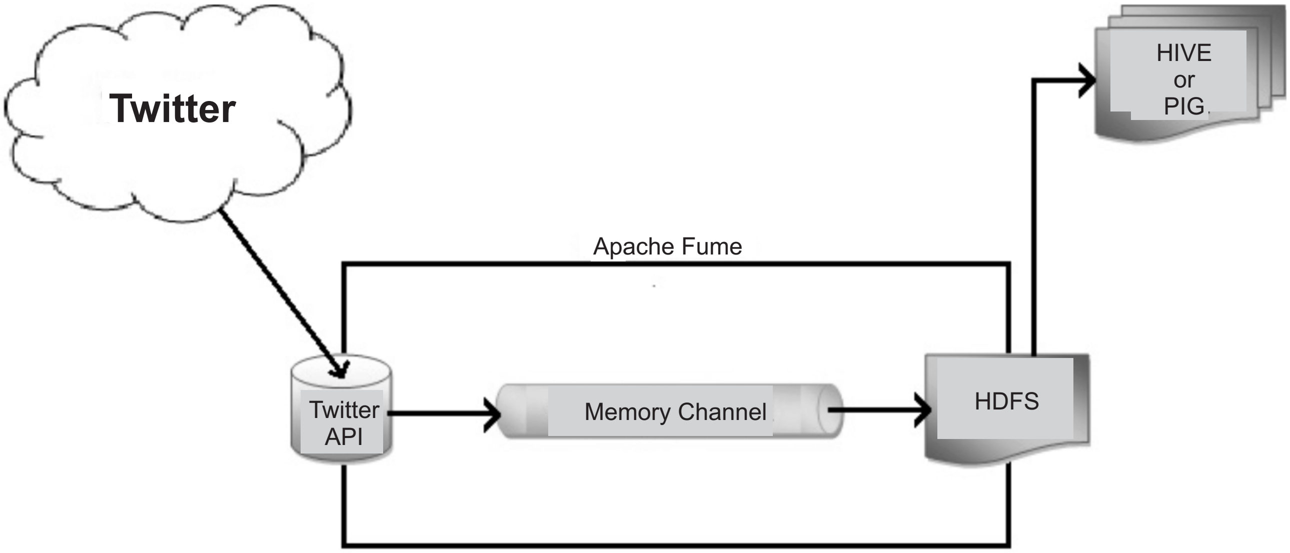


Fig. 2 Data collection

2. **Data Preprocessing:** Data pre-processing is a data mining technique to convert data’s into understandable and suitable format for data processing purposes. Data pre-processing technique used are:

Filtering: Filtering task performs a genuinity check on the tweet; it filters the genuine tweets from the input.

Users can post any tweet as they wish and there is no method to check the genuinity of a tweet. Sometime tweets may be fake; in order to consider the genuinity here we assume that user’s tweets whose accounts which are genuine are taken as genuine. Here a user account is considered as genuine based on the number of followers, if the number of followers is greater than a particular threshold the account is taken as genuine and all its tweets are genuine.

Tokenization: Tokenization is the process of breaking the text into word or individual terms. Here RegexTokenizer allows more advanced tokenization based on the regular expressions.

Stop word Removal: The module is deal with the removal of the unnecessary word or stop words. Stop words are the most commonly used words in a language. A feature transformer called stop word remover that filter out the stop words from the input.

3. **Data Analysis and Event Detection :** The Analysis module is responsible for the overall analysis of the twitter data. It is subdivided as follows:

Query Processing: This module is responsible for retrieving the data from the Data Store and prepares it for processing. Data is collected by giving a query keyword and is stored in the specified document. It then tokenized based on the regular expression. Genuine tweets are filtered from the overall tweets and made this as input for the further processing's.

Analysis: This module dose the necessary analysis to identify the particular event and store it temporarily. This module performs event categorization using a customized algorithm. Algorithms meant for clustering or more text oriented algorithms such as TF-IDF[8] can be used in conjunction to our prioritization approach to implement our goal and also proposes a method for checking the genuine tweets to avoid unwanted tweets and thereby improving the accuracy and efficiency. Live twitter data is used for the real time event detection

3. Algorithm

DsDt	:	Disaster DataSet
DiDt	:	Disease DataSet
Tid	:	Tweet Id

Step 1: Collect_Dataset()

Step 2: Read(DsDt,DiDt)

Step 4: Parallelize(DataSet)

Step 5: Stop_Words_Remover(DataSet)

Step 7: Process(DataSet)

Step 8: Display_Result()

Collect_Dataset

Step 1: Open a Stream to Twitter

Step 2: Authenticate the Stream

Step 3: Filter the Stream based on Keyword

Step 4: Store the Stream as JSON File.

Read(Dataset)

Step 1: Open the JSON File

Step 2: Read the Data

Step 3: Convert the Data to String.

Stop_Words_Remover(DataSet)

Step 1: Read Data

Step 2: Declare Stop Words.

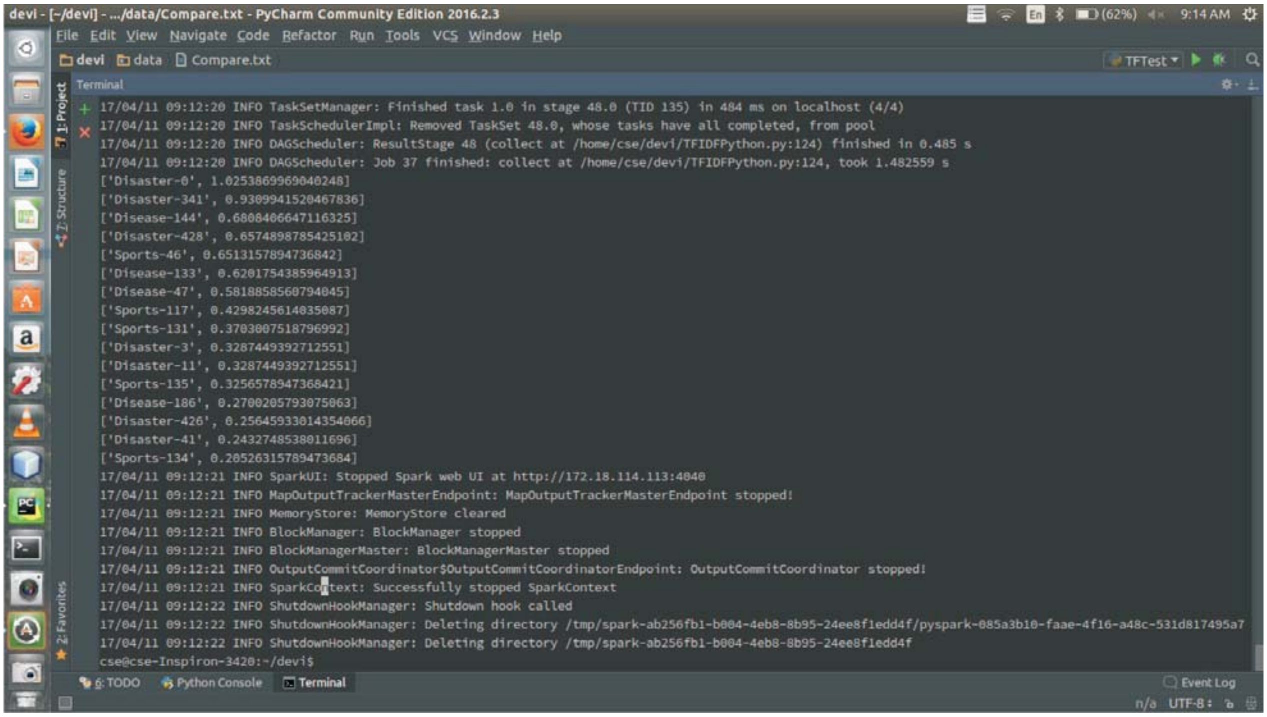
Step 3: Check For Stop Words.

Step 4: Remove Stop Words.

```
Process(DataSet)
Dct : Array of Tweets Weighs, Term Frequencies.
Doc : Array of Dct and IDF Weights.
BoW :  Bag Of Words
Function Term_Weight(BoW)          Tdict  =  null;
For each item  in BoW:        dict[item]  =  get_weight(Tdict[item]) + 1.0
                               Doc[item]  =  get_weight(Doc[item] + 1.0
                                     len  =  length(BoW)
For each item in dict:       Tdict[item]  =  Tdict[item] / len
                                     Dct  =  add(Tdict[item])
end
Function Prediction(BoW)           Qdict  =  null
For each item in BoW:        Qdict[item]  =  get_weight(Qdict[item]) + 1.0
                                     len  =  lenght(BoW)
For each item in dict:       Qdict[item]  =  Qdict[item] / len
                                     sim  =  0
For each doc in Dct:               point  =  0.0
                                   Tdict  =  Dct[1]
for each item in Qdict:
if item in Tdict:                  point  =  point + ( Qdict[item] / Dct[item])
                                             + (Ddict[item] / Dct[item] )
                                     sim  =   add(Dct[0] , point)
end
```

4. RESULT

Real time twitter data's are streamed after the authentication, for this it need a valid twitter account and need to create a twitter application using this account. After the data's are streamed the first step is the filtering of tweets. Only genuine tweets are to be considered since tweets are messages and twitter is a social network users can post anything they wanted so some may be fake posts. But it's not possible for make any rules for tweet posting and no method to check the genuinity of the tweet here checks the genuinity of the account users who posted the tweet by assuming that a user who is genuine will always post genuine tweets. This genuinity is checked based on the number of followers of the account owner. If the number of followers is greater than a particular threshold he can be considered as genuine one and his tweets are genuine. The filtered tweets are passed for the further processing of the event detection. Next step is categorization of this tweets into multiple groups such as diseases, disasters etc. this multiple automatic categorization is not possible with regular algorithms. For this here devised a customized algorithm for multiple event categorizations. For this algorithms meant for clustering or more text oriented algorithms such as TF-IDF can be used in conjunction to our prioritization approach to implement our goal. After the multiple event categorization this categories are prioritized based on the count of occurrence and social significance. Then the event categories are notified based on the priority.


devi - [~/devi] - .../data/Compare.txt - PyCharm Community Edition 2016.2.3
File Edit View Navigate Code Refactor Run Tools VCS Window Help
devi
data
Compare.txt
TFTest
Terminal
1: Project
7: Structure
2: Favorites
6: TODO
Python Console
Terminal
Event Log
n/a UTF-8

```
17/04/11 09:12:20 INFO TaskSetManager: Finished task 1.0 in stage 48.0 (TID 135) in 484 ms on localhost (4/4)
17/04/11 09:12:20 INFO TaskSchedulerImpl: Removed TaskSet 48.0, whose tasks have all completed, from pool
17/04/11 09:12:20 INFO DAGScheduler: ResultStage 48 (collect at /home/cse/devi/TFIDFPython.py:124) finished in 0.485 s
17/04/11 09:12:20 INFO DAGScheduler: Job 37 finished: collect at /home/cse/devi/TFIDFPython.py:124, took 1.482559 s
['Disaster-0', 1.0253869969040248]
['Disaster-341', 0.9309941520467836]
['Disease-144', 0.6808406647116325]
['Disaster-428', 0.6574898785425102]
['Sports-46', 0.6513157894736842]
['Disease-133', 0.6201754385964913]
['Disease-47', 0.5818858560794045]
['Sports-117', 0.4298245614035087]
['Sports-131', 0.3703007518796992]
['Disaster-3', 0.3287449392712551]
['Disaster-11', 0.3287449392712551]
['Sports-135', 0.3256578947368421]
['Disease-186', 0.2700205793075063]
['Disaster-426', 0.25645933014354066]
['Disaster-41', 0.2432748538011696]
['Sports-134', 0.20526315789473684]
17/04/11 09:12:21 INFO SparkUI: Stopped Spark web UI at http://172.18.114.113:4040
17/04/11 09:12:21 INFO MapOutputTrackerMasterEndpoint: MapOutputTrackerMasterEndpoint stopped!
17/04/11 09:12:21 INFO MemoryStore: MemoryStore cleared
17/04/11 09:12:21 INFO BlockManager: BlockManager stopped
17/04/11 09:12:21 INFO BlockManagerMaster: BlockManagerMaster stopped
17/04/11 09:12:21 INFO OutputCommitCoordinator$OutputCommitCoordinatorEndpoint: OutputCommitCoordinator stopped!
17/04/11 09:12:21 INFO SparkContext: Successfully stopped SparkContext
17/04/11 09:12:22 INFO ShutdownHookManager: Shutdown hook called
17/04/11 09:12:22 INFO ShutdownHookManager: Deleting directory /tmp/spark-ab256fb1-b004-4eb8-8b95-24ee8f1edd4f/pyspark-085a3b10-faae-4f16-a48c-531d817495a7
17/04/11 09:12:22 INFO ShutdownHookManager: Deleting directory /tmp/spark-ab256fb1-b004-4eb8-8b95-24ee8f1edd4f
cse@cse-Inspiron-3420:~/devi$
```

Figure 3: Multiple event categorized and prioritized result

5. CONCLUSION

Twitter mining for categorized multiple event detection introduce a new approach for detecting categorized multiple events using a customized algorithm with advanced semantic search using algorithms meant for clustering or more text oriented algorithms such as TF-IDF can be used in conjunction to our prioritization approach to implement our goal and also proposes a method for checking the genuine tweets to avoid unwanted tweets and thereby improving the accuracy and efficiency because about six percentage of tweets are fake tweets. Existing regular algorithmic approaches for detecting a specific event based on following key words attributed to an event is less efficient. Many of the special events are getting avoided. Here multiple events are categorized and prioritized based on this priority events are notified.

REFERENCES

[1] M. F. Goodchild(2007),Citizens as sensors: the world of volunteered geogra- phy, GeoJournal, *vol. 69, no. 4, pp. 211–221*

[2] Y. Georgiadou, J. H. Lungo, and C. Richter(2013) Citizen sensors or extreme publics? Transparency and accountability interventions on the mobile geoweb,*Int. J. Digit. Earth, vol. 7, no.* 7.

[3] T. Cheng and T. Wicks (2014) Event detection using Twitter: A spatio-temporal approach,*PLoS One, vol. 9, no. 6, pp. E97807.*

[4] Peng Yue, Senior Member, Chenxiao Zhang, Mingda Zhang, Xi Zhai, and Liangcun Jiang(2015) An SDI Approach for Big Data Analytics: The Caseon Sensor Web Event Detection and Geoprocessing Workflow.

[5] M. Botts, G. Percivall, C. Reed, and J. Davidson(2007) OGC sensor web enablement: Overview and high level architecture, *Open Geospatial Consortium Inc., USA, OpenGIS White Paper 07-165, 2007, 14pp*

[6] Abhishanga Upadhyay,Luis Mao,Malavika Goda Krishna MINING DATA FROM TWITTER

[7] Juan Ramos ,Using TF-IDF to Determine Word Relevance in Document Queries Department of Computer Science, Rutgers University, 23515 BPO Way, Piscataway, NJ, 08855

[8] Fabrizio Sebastiani. Machine Learning in Automated Text Categorization. ACM Computing Surveys, Vol. 34, No. 1, March 2002: 18-33.