International Journal of Scientific Engineering and Research (IJSER)
www.ijser.in
ISSN (Online): 2347-3878, Impact Factor (2014): 3.05

Real Time Analysis of Social Media Text Using Stream Computing

Neha Borole

Student of ME, Shah and Anchor Kutchhi Engineering College, Mumbai, India

Abstract: ***Social media measurement or 'social media monitoring' is an active monitoring of social media channels for information about a company or organization, usually tracking of various social media content such as blogs, wikis, news sites, micro-blogs such as Twitter, social networking sites, video/photo sharing websites, forums, message boards, blogs and user-generated content in general as a way to determine the volume and sentiment of online conversation about a brand or topic. Real time analysis of social media application requires methodology which can speedup processing and reduce latency. By using stream computing concept, this goal can be achieved.***

Keywords: Data stream computing, Event stream Computing, processing node, processing elements, events

1. Introduction

Many social media sites are available these days. Among various micro blogging sites, twitter and Facebook are very much popular. Many users post millions of messages on such sites. Using these millions of messages, we can understand what is going on at various locations and areas. Social media monitoring allows users to find insights into a brand's overall visibility on social media, measure the impact of campaigns, identify opportunities for engagement, assess competitor activity and share of voice, and be alerted to impending crises. It can also provide valuable information about emerging trends and what consumers and clients think about specific topics, brands or products.

This is the work of a cross-section of groups that include market researchers, PR staff, marketing teams, social engagement and community staff, agencies and sales teams. Several different providers have created tools to facilitate the monitoring of a variety of social media channels from blogging to internet video to internet forums. This allows companies to track what consumers are saying about their brands and actions. Companies can then react to these conversations and interact with consumers through social media platforms.

There is a large class of existing and newly emerging applications that require sophisticated, real-time processing of high-volume data streams. Although these applications have traditionally been served by "point" solutions through custom coding, infrastructure software that specifically target them have also recently started to emerge in the research labs and marketplace. Stream computing is a new paradigm necessitated by new data-generating scenarios, such as the ubiquity of mobile devices, location services, and sensor pervasiveness. A crucial need has emerged for scalable computing platforms and parallel architectures that can process vast amounts of generated streaming data.

2. Stream Computing

Stream computing is a new paradigm necessitated by new data-generating scenarios, such as the ubiquity of mobile devices, location services, and sensor pervasiveness. A crucial need has emerged for scalable computing platforms and parallel architectures that can process vast amounts of generated streaming data.

The stream processing computational paradigm consists of assimilating data readings from collections of software or hardware sensors in stream form (i.e., as an infinite series of tuples), analysing the data, and producing actionable results, possibly in stream format as well. In a stream processing system, applications typically act as continuous queries, ingesting data continuously, analysing and correlating the data, and generating a stream of results.

There are two types of stream computing mechanism that is Data stream computing and event stream computing.

A. Data Stream Computing

Data stream computing is able to analyze and process data in real time to gain an immediate insight, and it is typically applied to the analysis of vast amount of data in real time and to process them at a high speed. Many application scenarios require big data stream computing.

Figure 1 describes the architecture of stream processing. Here storage is optional which keep state of processed data.

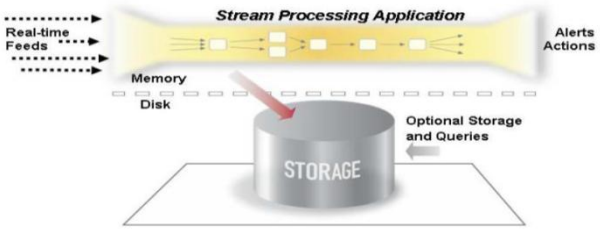

Figure 1: Straight through processing paradigm [1]

System S is data stream computing platform developed by IBM. In June 2007, IBM announced its stream computing system, called System S [2][8]. This system runs on 800 microprocessors and the System S software enables software applications to split up tasks and then reassemble the data into an answer.

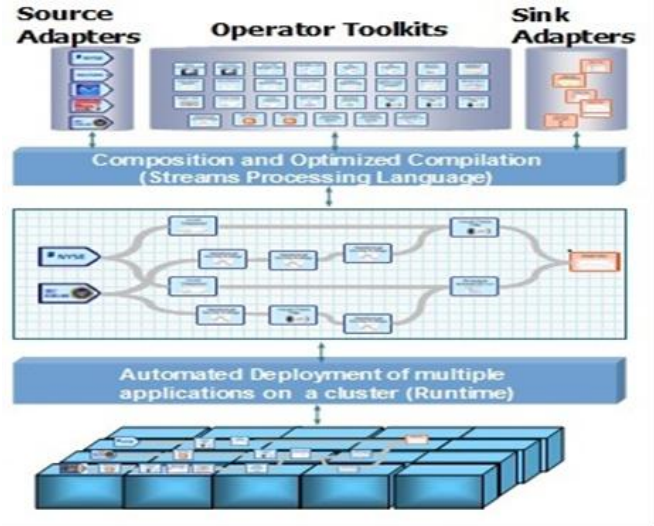

Figure 2: Architecture of System S [2]

System S supports applications written using the Streams Processing Language (SPL)[2][8].System S runtime provides an execution substrate for streaming applications, which includes services such as high performance data transport, resource allocation and scheduling, advanced job management, high availability, and security. System S provides an eclipse-based IDE for developing streaming applications using the SPL language. The development environment also includes support for interacting with the System S runtime via application launch capabilities and visualization of running jobs.

System S[2][8] includes web-based interfaces as well as command line tooling for configuring and administering System S instances in multi-user environments.

B. Event Stream Computing

Complex event processing (CEP) delivers high-speed processing of many events across all the layers of an organization, identifying the most meaningful events within the event cloud, analyzing their impact, and taking subsequent action in real time. Esper[5] is an Event Stream Processing (ESP) and event correlation engine (CEP, Complex Event Processing).Complex event processing (CEP) is an emerging network technology that creates actionable, situational knowledge from distributed message-based systems, databases and applications in real time or near real time. CEP can provide an organization with the capability to define, manage and predict events, situations, exceptional conditions, opportunities and threats in complex, heterogeneous networks.

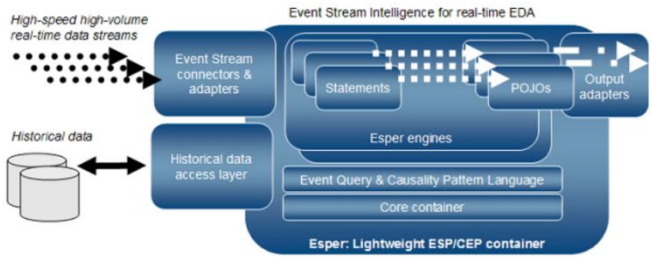

Figure 3: Architecture of Esper

Esper is written in java language. One can register the query statement and their corresponding library. Esper works like an inverted database. Event stream come into the engine and is run through the live queries. Queries can be windowed on time or length.Esper accepts different event representations, POJO2 events, java.util.Map events, object array events and XML events. Esper used EPL3 to write SQL like statements to be run in the engine. Esper engine parameters can be tuned by the configuration xml file. Many parameters can be altered in the runtime as well using Configuration Operations object.

3. Methodology

This methodology contains steps that are data collection, real time data analysis, data sentiment calculation, visualization. Now, consider, example of twitter sentiment analysis.

- In real time twitter sentiment system, system fetch data from external web services i.e. Twitter. To stream messages from twitter, System requires connection with twitter stream API. There are two services offered by twitter to collect data, one is using twitter search services and second one is twitter streaming API.
- In real time twitter analysis, creating connection with twitter API, authentication processing, streaming data, capturing appropriate data, these steps take place.
- In twitter sentiment calculation, tweets which are collected are used to calculate sentiment and polarity of tweet. Using sentiment algorithm we can extract sentiment polarity and language of tweet.
- Visualization is done by plotting the result in form of graphs. After sentiment, the CSV file of sentiment contains polarity and confidence. Using these two parameters, we can plot graph which will describe the popularity of particular politician. Various charting tools we can use here to plot the graph.

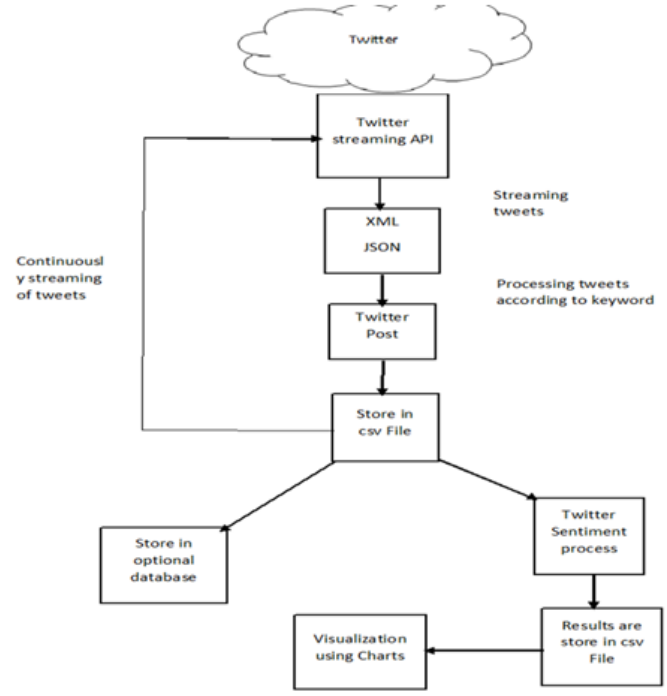

Figure 4: System flow diagram

In above data flow diagram of system, system creates connections with twitter streaming API. After that, tweets are fetched in system. Tweets then processed by CEP engine by searching specific keyword in it. Tweets with specific keyword are stored in CSV file with id, text and date.

In sentiment process, data from previously saved CSV file is fetched by system. Then sentiment calculation is done on each text. Results of each text calculation are stored in CSV file with user id.

In visualization layer, plotting of twitter sentiment results are take place.

4. Report and Analysis

Following chart diagrams explain twitter sentiment analysis of president Obama. This algorithm is implemented by using event stream computing. This application is implemented using java and esper. Esper is event stream processing engine. It increases speed of fetching data and processing data.

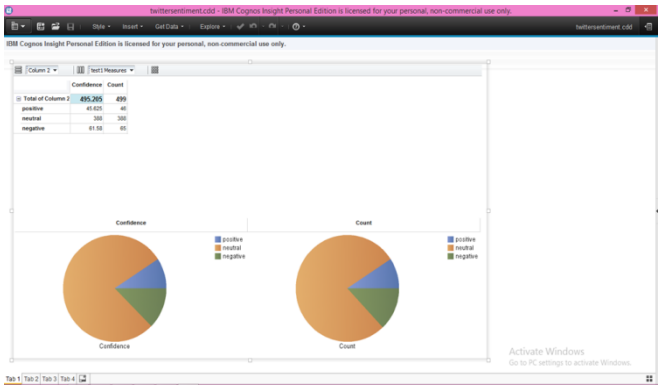

Figure 5: Pie diagram of twitter sentiment analysis

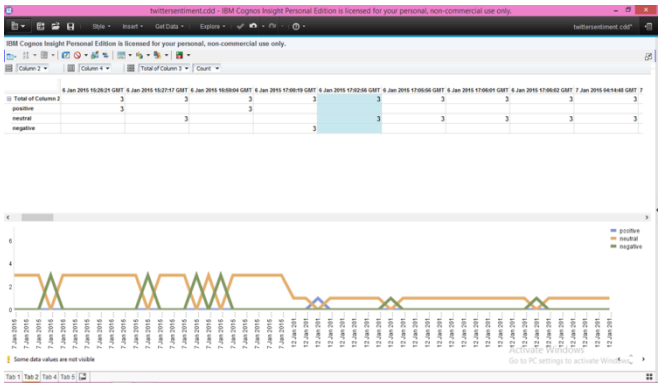

Figure 6: Chart diagram of date vs polarity

5. Conclusion

This paper focuses on real time stream computing which can be good option for traditional relational database system or even to batch processing. Real time twitter analysis not only stream twitter messages in real time but also calculate sentiments of tweets in real time.

This paper even describes about different type of stream computing processors. We can use different stream computing processor for different applications. Selection of processor depends on requirements of application and how complex that application is.

To implement real time application stream computing can be good option as it reduce latency and save storage space.

In traditional twitter sentiment analysis, data is first collected in database. Then data is fetched according to query followed by pre-processing, classifying and further steps. But in this aspect collecting and processing had done at the same time which saves lot of time.

References

[1] Stonebraker, U. C¸ etintemel, and S. Zdonik, “The 8 Requirements of Real-Time Stream Processing,” SIGMOD Rec., vol. 34, no. 4, pp. 42–47, Dec 2005.
[2] Barıs¸Güç,B. ; Ranganathan, A. “Realtime, scalable route planning using a stream-processing infrastructure ” in 2010, 13th International IEEE Intelligent Transportation Systems (ITSC), Conference on Publication Year: 2010 , Page(s): 986 - 991L.
[3] Neumeyer, B. Robbins, A. Nair, and A. Kesari, “S4: Distributed Stream Computing Platform,” in 2010 IEEE Data Mining Workshops (ICDMW), Sydney, Australia, Dec. 2010, pp. 170 –177
[4] Chauhan, J. ; Chowdhury, S.A. ; Makaroff, D. “Performance Evaluation of Yahoo! S4: A First Look” in 2012 IEEE P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC), Seventh International Conference Publication Year: 2012 , Page(s): 58 – 6
[5] Arun Mathew, 123059007,Dept of Computer Science and Engineering,IIT Bombayarunmathew@cse.iitb.ac.in”,Benchmarking of Complex Event Processing Engine -Esper”.
[6] Jayashri Khairnar, Mayura Kinikar, “ Machine Learning Algorithms for Opinion Mining and Sentiment Classification”, International Journal of Scientific and Research Publications, Volume 3, Issue 6, June 2013 1 ISSN 2250-3153
[7] StreamComputingPlatforms,Applications,andAnalytics.[Online].Avaliable:http://researcher.watson.ibm.com/researcher/view_project_subpage.php?id=2534.
[8] Toyotaro Suzumura and Tomoaki Oiki,” StreamWeb: Real-Time Web Monitoringwith Stream Computing” in 2011 IEEE International Conference on Web Services, Publication Year: 2011