International Journal on Applicationsin Information and Communication Engineering
Volume 2:Issue 6: June 2016, pp 6-9. www.aetsjournal.com ISSN (Online) : 2394 - 6237

Design and Implementation of Sentiment Analysis Technique Over Twitter Data

Aiswarya.S , Dr A.Prabhu

Abstract — Sentiment analysis in a data streams is aimed to detect an author's attitudes, emotions and opinions from text in real time. To reduce the labeling efforts needed in the data collection phase, active learning is often in applied streaming in scenarios, where a learning algorithms is allowed to select a new examples to be a manually labeled in order to improves the learner performance. Even though there is many online platforms which performed a sentiment analysis, there is no publicly available interactive online platforms for dynamic adaptive sentiment analysis, which would be able to handle changes in data streams and adapts its behavior in over time. This paper described a cloud Flows, a cloud based scientific workflows can platform, and its extensions are enabling the analysis of data streams and active learning. Moreover, by utilizing the data and workflows sharing in Cloud Flows, the labeling of examples can be distributed through crowd sourcing. The advanced features of cloud flows is demonstrated on a sentiment analysis use case, using actives learning with a linear Support Vector Machine for a learning sentiment classification models to be applied in micro blogging data streams.

Keywords: Twitter, Sentiment Analysis, Hadoop, Map reduce, HDFS.

I. INTRODUCTION

Live in societies, where the textual data in the internet is grows at rapid places and many companies are tried to using these deluges of data to extracts people views towards their product. Online social network platforms, with their large scale repository of user generated content, can provide unique opportunity to gain insights into an emotional "Pulse of Nation" indeed the main global community.

A great source of unstructured text information's is included in social networks, where it is unfeasible to manually analyze such amounts of data. There is large number of social networks websites that enable user to contributes, modified and grade the contents, as well as to express their personal opinions about specific topics. Some examples included blogs, forum, products review site, and social networks, like Twitters. Twitter (San Francisco, CA, USA) is a micro blogging sites that offer in the opportunity for these analyses of expressed mood, and previous study have shown in that geographical, weekly, and seasonal patterns of positive and negative effects can be observed.

This paper introduced a cloud based Scientifics workflow platforms, which is able to perform online dynamic adaptive sentiments analysis of micro blogging posts. Even though there are many online platforms which applied to sentiment analysis on micro blogging texts, there is still no such platform that could be used for online dynamic adaptive sentiment analysis and would thus been able to handles change in data streams and adapted its component over times. In order to provide continuous updating of the sentiment classifiers with timing we used active learning approaches. In this paper, we address these issues by presenting an approach to interactive stream based sentiment analysis of micro blogging messages in a cloud based scientific workflow platform Cloud Flows. With the aim of to minimize the efforts required to apply labels to tweets; this browser based platforms provide an easy ways to share the results and a Web interface for labeling tweets.

II. PROBLEM DEFINITION

This project focused on using Twitter, and the most popular micro blogging platforms, for the tasks of sentiment analysis. The tweet are important for analysis because data arrived at a high frequency and algorithm that process them must do a under very strict constraint of storage and time. It will be shown in how to automatically collect a corpus for a sentiment analysis and opinion mining purposes and then perform linguistics analysis of the collected corpus. All public tweets posted in twitter are freely available through a set of APIs provided by Twitter. Using the corpus, sentiment classifiers, is constructed that able to determine positive and negative and neutral sentiments.

III. DEVELOPMENT METHODOLOGY

In the Proposed systems we proposed two Latent Dirichlet Allocation (LDA) models, for ground and Background LDA (FB-LDA) and Reason Candidates and Background LDA (RCB-LDA). The FB LDA models can filter out background topics and then extracts foreground topics to reveals possible reasons. To give more intuitive representations, the RCB-LDA modeled can ranks a set of reason candidates expressed in natural languages to provide sentence level reasons. Our proposed models were evaluated on real Twitter data. Experimental results show that our models can mine possible reasons behind sentiment variation.

Aiswarya.S , PG Scholar, Department of CSE, Annai Mathammal Sheela Engineering College, Namakkal, Tamilnadu.
(Email: aiswariayas@gmail.com.)
Dr A.Prabhu,Head of the Department, Department of CSE, Annai Mathammal Sheela Engineering College, Namakkal, Tamilnadu.

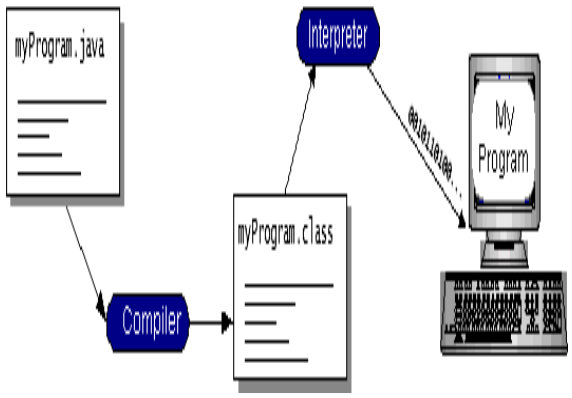

Fig.1 Common Language Infrastructure

Every Java interpreter, whether it's a development tool or a Web browser that can run applets, is an implementation of the Java VM. Java byte codes help make "write once, run anywhere" possible. You can compile your program into byte codes on any platform that has a Java compiler. The byte codes can then be run on any implementation of the Java VM. That means that as long as a computer has a Java VM, the same program written in the Java programming language can run on Windows 2000, a Solaris workstation, or on an iMac.

1) The Java Platform

A *platform* is the hardware or software environment in which a program runs. We've already mentioned some of the most popular platforms like Windows 2000, Linux, Solaris, and Macros. Most platforms can be described as a combination of the operating system and hardware. The Java platform differs from most other platforms in that it's a software-only platform that runs on top of other hardware-based platforms.

The Java platform has two components:

- *The Java Virtual Machine (Java VM)*
- *The Java Application Programming Interface (Java API)*

You've already been introduced to the Java VM. It's the base for the Java platform and is ported onto various hardware-based platforms.

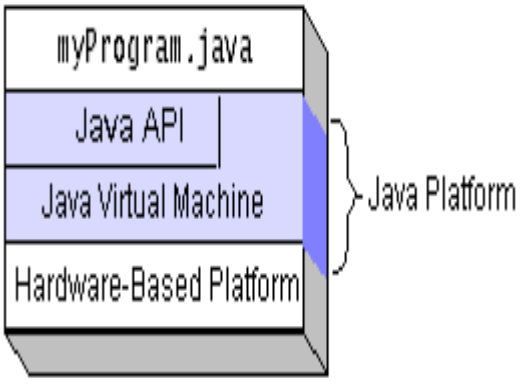

Fig.2 Compiler Design

Native code is code that after you compile it, the compiled code runs on a specific hardware platform. As a platform-independent environment, the Java platform can be a bit slower than native code. However, smart compilers, well-tuned interpreters, and just-in-time byte code compilers can bring performance close to that of native code without threatening portability.

The Java API is a large collection of ready-made software components that provide many useful capabilities, such as graphical user interface (GUI) widgets. The Java API is grouped into libraries of related classes and interfaces; these libraries are known as *packages*.

IV. KEY EXPOSURE IN BIG DATA SHARING SYSTEM

The issue of key exposure is more severe in a ring signature scheme: if a ring member's secret key is exposed, the adversary can produce valid ring signatures of any documents on behalf of that group. Even worse, the "group" can be defined by the adversary at will due to the spontaneity property of ring signature: The adversary only needs to include the compromised user in the "group" of his choice.

ACTIVE LEARNING

In active learning, the learning algorithm periodically asks an oracle (e.g., a human annotator) to manually label the examples which he finds most suitable for labeling. Using this approach and an appropriate query strategy, the number of examples that need to be manually labeled is largely decreased. Typically, the active learning algorithm first learns from an initially labeled collection of examples. Based on the initial model and the characteristics of the newly observed unlabeled examples, the algorithm selects new examples for manual labeling. After the labeling is finished, the model is updated and the process is repeated for the new incoming examples. This procedure is repeated until some threshold (for example, time limit, labeling quota or target performance) is reached or, in the case of data streams, it continues as long as the application is active and new examples are arriving.

In our software, the active learning algorithm first learns from the Stanford smiley labeled data set as an initial labeled data set. According to this initial model, the algorithm classifies new incoming tweets from the data stream as positive or negative. Tweets, which come from the data stream, are split into batches. The algorithm selects most suitable tweets from a first batch for hand-labeling and puts them in a pool of query tweets. The process is repeated for every following batch and every time the pool of query tweets is updated and the tweets in the pool are reordered according to how suitable they are for hand-labeling. When the user decides to conduct manual labeling, she is given a selected number of top tweets from the pool of query tweets for hand-labeling. The user can label a tweet as positive, negative or neutral. After the labeling, labeled tweets are placed in the pool of labeled tweets and removed from the pool of query tweets. Periodically, using the initial and manually positively and negatively labeled tweets from the pool of labeled tweets, the model is retrained. This process is repeated until it is terminated by the user.

As a result, the exposure of one user's secret key renders all previously obtained ring signatures invalid (if that user is one of the ring members), since one cannot distinguish whether a ring signature is generated prior to the key exposure or by which user. Therefore, forward security is a necessary requirement that a big data sharing system must meet. Otherwise, it will lead to a huge waste of time and resource.

V. ACTIVE LEARNING ANALYSIS WORKFLOW USING SQL SERVER

A database management, or DBMS, gives the user access to their data and helps them transform the data into information. Such database management systems include dBase, paradox, IMS, SQL Server and SQL Server. These systems allow users to create, update and extract information from their database.

A database is a structured collection of data. Data refers to the characteristics of people, things and events. SQL Server stores each data item in its own fields. In SQL Server, the fields relating to a particular person, thing or event are bundled together to form a single complete unit of data, called a record. Each record is made up of a number of fields. No two fields in a record can have the same field name.

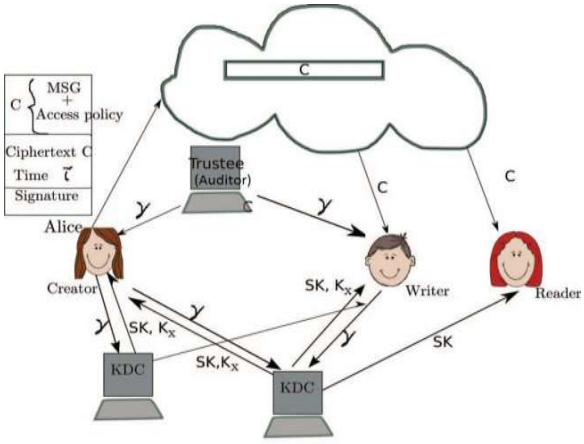

Fig 3: Our secure and authenticated cloud storage model

VI. RESULTS AND DISCUSSIONS

1) Output Design

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any system results of processing are communicated to the users and to other system through outputs. In output design it is determined how the information is to be displaced for immediate need and also the hard copy output. It is the most important and direct source information to the user. Efficient and intelligent output design improves the system's relationship to help user decision-making.

Designing computer output should proceed in an organized, well thought out manner; the right output must be developed while ensuring that each output element is designed so that people will find the system can use easily and effectively.

1. When analysis design computer output, they should Identify the specific output that is needed to meet the requirements.
2. Select methods for presenting information.
3. Create document, report, or other formats that contain information produced by the system.

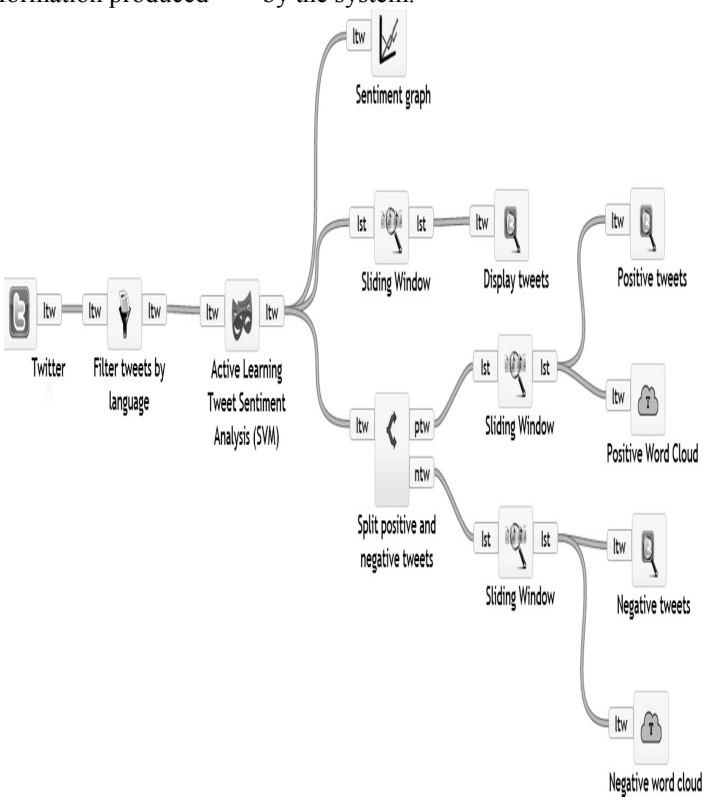

Fig.4 The Twitter sentiment analysis workflow

2) Input Design

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data in to a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides security and ease of use with retaining the privacy.

After getting a siganature ,they can upload our files. The file can be stored in the data center.

VII. CONCLUSION

It is proposed to stream real time live tweets from twitter using Twitter API, and the large volume of data makes the application suitable for Big Data Analytics. The FB-LDA model can filter out background topics and then extract foreground topics to reveal possible reasons. To give a more intuitive representation, the RCB-LDA model can rank a set of reason candidates expressed in natural language to provide sentence-level reasons. Our proposed models were evaluated on real Twitter data. Experimental results showed that our

models can mine possible reasons behind sentiment variations. A method to predict or deduct the location of a tweet based on the tweet's information and the user's information should be found in the future.

REFERENCES

[1] T. Wilson, J. Wiebe and P. Hoffmann, "Recognizing contextual polarity in phrase-level sentiment analysis," in Proceedings of HLT and EMNLP. ACL, **(2005)**, pp. 347–354

[2] C. C. Tao, S. K. Kim, Y. A. Lin, Y. Y. Yu, G. Bradski, A. Y. Ng and Kunle Olukotun, "Map-reduce for machine learning on multicore", In NIPS, vol. 6, pp. 281-288,2006.

[3] L. Jimmy, and A. Kolcz, "Large-scale machine learning at twitter", In Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data, ACM, pp. 793-804,2012.

[4] B. Jiang, U. Topaloglu and F. Yu, "Towards large-scale twitter mining for drug-related adverse events", In Proceedings of the 2012 international workshop on Smart health and wellbeing, ACM, pp. 25-32, 2012.

[5] L. Bingwei, E. Blasch, Y. Chen, D. Shen and G. Chen, "Scalable Sentiment Classification for Big Data Analysis Using Naive Bayes Classifier", In Big Data, 2013 IEEE International Conference on, IEEE, pp. 99-104,2013.

[6] Á. Cuesta, David F. and María D. R-Moreno, "A Framework for Massive Twitter Data Extraction and Analysis", In Malaysian Journal of Computer Science, pp. 50-67,2014.

[7] S. Michal and A. Romanowski, "Sentiment analysis of Twitter data within big data distributed environment for stock prediction", In Computer Science and Information Systems (FedCSIS), 2015 Federated Conference on, IEEE, pp. 1349-1354,2015.

[8] T. Mohit, I. Gohokar, J. Sable, D. Paratwar and R. Wajgi, "Multi-Class Tweet Categorization Using Map Reduce Paradigm", In International Journal of Computer Trends and Technology, pp. 78-81, 2014.

[9] D. Jeffrey and S. Ghemawat, "MapReduce: simplified data processing on large clusters", Communications of the ACM 51.1,pp. 107-113,2014.

[10] B. Yingyi, "HaLoop: Efficient iterative data processing on large clusters", Proceedings of the VLDB Endowment 3.1-2, pp. 285-296,2011.

[11] T. Maite, "Lexicon-based methods for sentiment analysis", Computational linguistics 37.2, pp. 267-307,2011.

[12] R. Tushar and S. Srivastava, "Analyzing stock market movements using twitter sentiment analysis", Proceedings of the 2012 International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2012). IEEE Computer Society, **2012**.

[13] D. Pessemier and Martens "MovieTweetings: A Movie Rating Dataset Collected From Twitter", Ghent University, Ghent, Belgium, **2013.**

[14] Twitter. Twitter Search API, available at https://dev. twitter.com /rest/public/search.

[15] V. D. Katkar, S. V. Kulkarni, "A Novel Parallel implementation of Naive Bayesian classifier for Big Data", International Conference on Green Computing, Communication and Conservation of Energy, 978-1-4673-6126-2 - IEEE, pp. 847-852,2013.