Movie Sentiment Analysis

**Kush Bavishi,Udhav Chandel,Nagashree Devayani Urs,Pallavi Hemmige**

Computer Science and Engineering Department, M.S Ramaiah Institute of Technology, Bangalore, India

Abstract**:** This paper presents a study of supervised learning and machine learning techniques in classifying movie reviews by semantic meaning. We use movie reviews from Rotten Tomatoes as our data set and classify text by subjectivity/objectivity and negative/positive attitude. We propose different approaches in extracting text features such as bag-of-words model, using large movie reviews corpus, restricting to adjectives and adverbs, handling negations, bounding word frequencies by a threshold. We evaluate their effect on accuracy by using a machine learning method- Naive Bayes. We reduce our test dataset using Map-Reduce technique on the Hadoop framework.

We conclude our study with, the sentiments found within comments, feedback or critiques to provide useful indicators for many different purposes. These sentiments can be categorized either into five categories: negative, somewhat negative, neutral, somewhat positive, and positive. In this respect, a sentiment analysis task can be interpreted as a classification task where each category represents a sentiment

Keywords: Sentiment, Review, Visualization, Analysis

1. **Introduction**

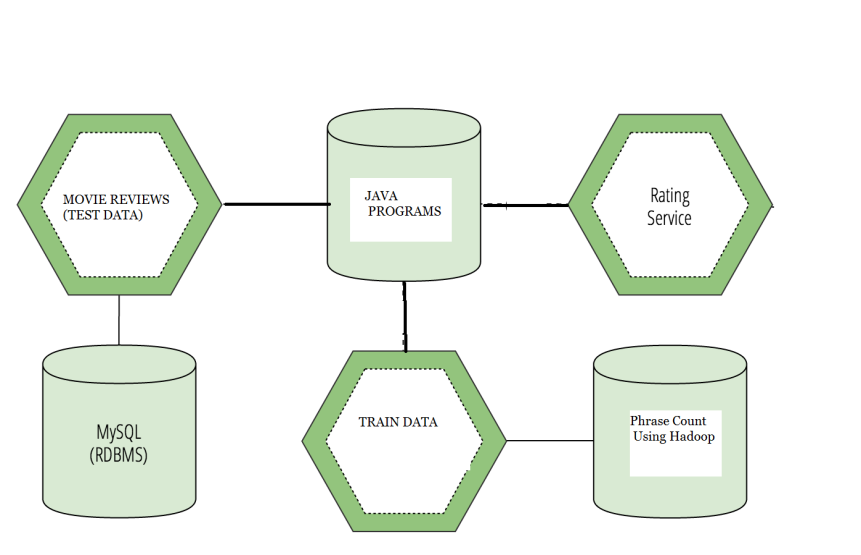
The ways people express their opinions and sentiments have changed in the past few years thanks to the arrival of social networks, web communities, and other online media. The extraction of knowledge from the large amount of information on the Web can be a key factor to those who want to engrave an idea into the people of the world and thus reaching out to many. The automatic analysis of online opinions requires a understanding of language text. Sentiment analysis aims to benchmark your sentiment analysis ideas on the Rotten Tomatoes dataset. You are asked to label phrases on a scale of five values: negative, somewhat negative, neutral, somewhat positive, and positive. Obstacles like sentence negation, sarcasm, terseness, language ambiguity, are taken into account in the project.

Here the goal is to analyze textual content from the perspective of the opinions and viewpoints they hold. The dataset is comprised of tab-separated files with phrases from the Rotten Tomatoes dataset. In the train and test data set each phrase has a PhraseId. Each sentence has a SentenceId. Phrases that are repeated are only included once in the data. (a) train.tsv contains the phrases and their associated sentiment labels. We have additionally provided a SentenceId so that you can track which phrases belong to a single sentence.

(b) test.tsv contains just phrases. You must assign a sentiment label to each phrase. The sentiment labels are that are used in this project are as follows:0 – negative,1 - somewhat negative,2 – neutral,3 - somewhat positive,4 – positive.

The aims are: To construct a database of reviews (words/phrases) on the keywords, Machine learning techniques which are based on a training set and will determine the sentimental value of the sentence/word/phrases, to provide a sentimental value after thorough analysis of the movie review data set which can help an individual choose a movie.

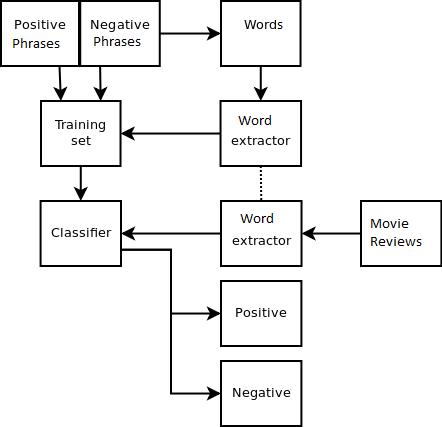
1. **Design**



**Figure-4: System Architecture of Sentiment Analysis**

The Sentiment Analysis Movie Architecture primarily consists of the following key components which are hadoop,mySQL,Java using eclipse. The hadoop framework involves implementing the phrase count on the train data set and comparing the values using java with the test data set. Hadoop implements the Map-Reduce algorithm on the data sets and helps in simplifying the data to be processed. The values in the data set are then extracted and put into the mySQL database. Our java program then compares the values and provides the final sentimental analysis values.

The purpose of the implementation is to be able to automatically classify a review as a positive or negative review sentiment wise. The classifier needs to be trained thus,we need a list of manually classified reviews. The list of word features need to be extracted from the reviews. It is a list with every distinct words ordered by frequency of appearance. The Naive Bayes classifier uses the prior probability of each label which is the frequency of each label in the training set, and the contribution from each feature.

Now that we have our classifier done, we can try to classify a review and see what the sentiment type output is. Our classifier is able to detect that this review has a certain sentiment value accordingly. Hence we obtain our sentiment value for the movie review. 

**Figure-3: Flowchart of Sentiment Analysis of Data**

The above Flowchart shows us the steps in the sentiment analysis.

1. Extract the phrases from the test data: Our test data has several extra columns from which we are extracting the required data and writing it onto a file.

2. Map-reduce operations performed on the test data: Our test data will be reduced by implementing the map-reduce operation on phrases using Hadoop to give us the count of non-unique phrases.

3. Putting Train data into the database: The data that is in the data set is transferred or put in to the mySQL database with the help of a java code or java program.

4. Calculating the sentimental value: Finally, the sentiment value is calculated for the particular movie review and hence the sentiment analysis is available to the user.

III.**Implementation**

**A. Tools Introduction**

Hadoop framework:

Apache Hadoop is an open-source software framework written in Java for distributed storage and distributed processing of very large data sets on computer clusters built from commodity hardware. All the modules in Hadoop are designed with a fundamental assumption that hardware failures (of individual machines or racks of machines) are commonplace and thus should be automatically handled in software by the framework.

SAP Lumira:

SAP Lumira (formerly called SAP Visual Intelligence) is a self-service, [data visualization](http://searchbusinessanalytics.techtarget.com/definition/data-visualization) application for business users. SAP Lumira grew out of [SAP Business Explorer](http://searchsap.techtarget.com/definition/BEx), the company's previous, but more basic, self-service data visualization tool. According to SAP, the two are meant to go hand-in-hand.

MySQL: Is an open-source relational database management system.

Xampp server:

The Xampp is a free and open source cross-platform web server solution stack package developed by Apache Friends, consisting mainly of the Apache HTTP Server, MySQL database, and interpreters for scripts written in the PHP and Perl programming languages.

Ubuntu Operating System:

Ubuntu Linux operating system and distribution,with unity as its default desktop environment for personal computers including smart phones in later versions.

**B. Technology Introduction**

Java: (Sun Java 7):

Java is a set of computer software and specifications developed by Sun Microsystems, later acquired by Oracle Corporation that provides a system for developing application software and deploying it in a cross-platform computing environment.

**C. Overall view of the project in terms of implementation**

The sentimental value is obtained according to the sentimental analysis scale that we have utilized i.e somewhat negative, negative, positive, and somewhat positive and so on. Hadoop is the framework that helps us to reduce any redundancy and a java program was written with the help of algorithms such as Naïve Bayes which acts a classifier. Thus the sentimental value of the movie review is obtained.

**D. Explanation of Algorithm and how it is been implemented**

Naive Bayes Algorithm : It is a popular method for text categorization, the problem of judging documents as belonging to one category or the other (such as spam or legitimate, sports or politics, etc.) with word frequencies as the feature.

Bag Of Words: : The bag-of-words model is commonly used in methods of document classification, where the (frequency of) occurrence of each word is used as a feature for training a classifier.

Ngram.:In the fields of computational linguistics and probability, an *n*-gram is a contiguous sequence of *n* items from a given sequence of text or speech. The items can be phonemes, letters, words or base pairs according to the application. The *n*-grams typically are collected from a text or speech corpus.

An *n*-gram of size 1 is referred to as a "unigram"; size 2 is a "bigram" (or, less commonly, a "digram"); size 3 is a "trigram". Larger sizes are sometimes referred to by the value of*n*, e.g., "four-gram", "five-gram", and so on.

**E. Information about the implementation of Modules**

Number of Modules:In the movie sentiment analysis we use 4 modules. They are:

1. Extract the phrases from the test data

2. Map-reduce operations performed on the test data

3. Putting Train data into the database

4. Calculating the sentimental value

Modules description

* Extract the phrases from the test data: Our test data has several extra columns from which we are extracting the required data and writing it onto a file
* Map-Reduce operations performed on the test data: Our test data will be reduced by implementing the map-reduce operation on phrases using Hadoop to give us the count of non-unique phrases.
* Putting Train data into database: The train data will be extracted and entered into the database using JDBC concepts.
* Calculating the sentimental value: Each of the phrases from the output of the reduced test data will be checked against the train data and the final sentimental value will be calculated.

IV**.Testing and Comparisons**

Software testing is an investigation conducted to provide stakeholders with information about the quality of the product or service under test. Software testing can also provide an objective, independent view of the software to allow the business to appreciate and understand the risks of software implementation. Test techniques include the process of executing a program or application with the intent of finding software bugs (errors or other defects).

Software testing involves the execution of a software component or system component to evaluate one or more properties of interest. In general, these properties indicate the extent to which the component or system under test:

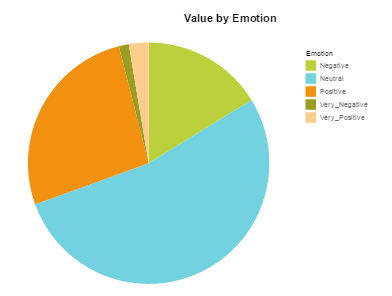
* meets the requirements that guided its design and development,
* responds correctly to all kinds of inputs,
* performs its functions within an acceptable time,
* is sufficiently usable, can be installed and run in its intended environments.

When validating a sentiment analysis system, the testing methodology is crucial. The data source, cleanliness of language, how it is scored, subject matter and volume of data tested are all significant variables that can dramatically affect results.

For an optimal test, the data source should closely match the intended uses. For example, if your intended application is analysis of online dialog, the data used to test system accuracy should also be sourced from online dialog. Volume of data tested is also important, and a general rule of thumb here is “the more data the better the test”.

In our project, testing is primarily performed by simplifying the data sets and running our program. On running the program we find the trend in our data set of the movi review. We notice the time that is taken during the implementation.

After testing and comparisons we finally get these results using SAP Lumira.



**Figure-4: Categorization using SAP Lumira**

.

V**. Results**

The results of the analysis is primarily a categorization of the different categories of sentiments:

* + Negative
  + Somewhat negative
  + Neutral
  + Somewhat positive
  + Positive

The count of the phrases associated with each sentiment, and finally a value signifying the overall sentiment value, with graphical representation by using lumira tools.

VI.**SCOPE AND FUTURE WORK**

Much of the current research is focusing on the area of sentiment analysis. People are intended to develop a system that can identify and classify opinion or sentiment as represented in an electronic text. An accurate method for predicting sentiments could enable us, to extract opinions from the internet and predict online customer’s preferences, which could prove valuable for economic or marketing research.

Future researches can be carried out to generate better and fast models for higher order n-grams. Hence movie sentiment analysis can be improved in the future.

VII. **CONCLUSION**

In conclusion, the sentiment analysis project gives us the ability to guage the sentiment value of a particular review on the Rotten Tomatoes website. Hence a person will be able to determine the sentiment value of a particular movie. It not only helps assess the viewer the movie he would want to watch based on its sentiment, but also provides an overall categorization of the critics review based on his sentiment of the view. It helps us get a clear understanding of the application of hadoop, the usage of machine learning algorithms and finally visualization of data by SAP Lumira. The sentiment analysis is the future of analytic research, and the analysis on movie reviews is just about to get bigger.

**ACKNOWLEDGMENT**

We would like to thank our HOD Dr. K.G.Srinivasa(Professor). Without their push and motivation, this project would have not been completed. Their continuous support and help made this project feasible. Without his support we would not have been able to excel in our project. Lastly, we would like to thanks all our friends and teammates for believing in each other and our capabilities.

**References**

[1]Amazon online retailer web site. http://www.amazon.com.

[2] Digg social networking site. <http://www.digg.com>.

[3] Natural language toolkit. <http://www.nltk.org>.

[4]Rottentomatoes movie review site. <http://www.rottentomatoes.com>.

[5] Twitter social networking site. <http://www.twitter.com>.

[6]http://www.cisco.com/web/offer/email/43468/5/Cloud\_Computing\_in\_Higher\_Education.pdf : Cloud Computing in Higher Education: A Guide to Evaluation and Adoption

[7] A. Bagga and B. Baldwin. Algorithms for scoring coreference chains. In In The First International Conference on Language Resources and Evaluation Workshop on Linguistics Coreference, pages 563– 566, 1998.