

Sentiment Analysis of Application Reviews On Google Playstore

Sangeeta Panigrahi, Saarim Momin, Pooja Patil, Prof Prachi Kshirsagar

Department of Information Technology, Mumbai University, Mumbai, Maharashtra, India

ABSTRACT

Our day-to-day life has always been influenced by what people think. Ideas and opinions of others have always affected our own opinions. As the Web plays an increasingly significant role in people's social lives, it contains more and more information concerning their opinions and sentiments. The distillation of knowledge from this huge amount of unstructured information, is also known as opinion mining and sentiment analysis. Nowadays, with the rapid evolution of smart phones, mobile applications (Mobile Apps) have become essential parts of our lives. However, it is difficult for consumers to keep track and understand the app sphere because new apps are entering market every day. So sentiment analysis of application reviews on google playstore will help the developers of the applications to keep their particular applications up to date in order to keep their particular application in the top lists and also help the customers to select the most popular application.

Keywords: Sentiment analysis, Learning approaches and Polarity

I. INTRODUCTION

Posting reviews online has become an increasingly popular way for people to express opinions and sentiments towards the products bought or services received. Analyzing the large value of online reviews would produce useful actionable knowledge that could be of economic values to vendors and other interested parties.

Generally speaking, sentiment analysis aims to determine the attitude of a speaker or a writer with respect to some topic or the overall contextual polarity of a document. The attitude may be his or her judgment or evaluation, affective state that is to say, the emotional state of the author when writing, or the intended emotional communication that is to say, the emotional effect the author wishes to have on the reader.

Everything the project will be expected to do the earlier stage will comprise of the extraction of the user reviews for the purpose of analysis on it. The user reviews will consists of the text feedback. The

sentiment analysis will be carried on which will create score for each text review generated.

The score calculated from these reviews will help the developer to for future evaluation and any changes to their application. For example the developer of the application may been keen to know

1. What do people think about the particular application?
2. How is the overall impact of the application whether its positive or negative?
3. What are people suggestions for improving the application or what does people prefer the future update for the application should be like?

The main need to collect opinions for particular applications and draw conclusions is to be aware that what are the people like/dislike, is been the most important aspect in today's scenario. The objective of this paper is to discuss concept of sentiment analysis of application reviews on google playstore and perform comparative study of its various techniques.

II. METHODS AND MATERIAL

1. Literature Review

The various methods or approaches that are used over a period of time to help the developers find the sentiments for the opinion and sentimental scores vary on various factors. The various types of Methods are

1. Lexicon based techniques.
2. Machine learning based techniques.

Lexicon based techniques use a dictionary to perform entity-level sentiment analysis. This technique uses dictionaries of words annotated with their semantic orientation (polarity and strength) and calculates a score for the polarity of the document. Usually this method gives high precision but low recall.

Learning based techniques require creating a model by training the classifier with labeled examples. This means that you must first gather a dataset with examples for positive, negative and neutral classes, extract the features/words from the examples and then train the algorithm based on the examples.

Choosing which method to use heavily depends on the application, domain and language. Using lexicon based techniques with large dictionaries enables us to achieve very good results. Nevertheless they require using a lexicon, something which is not always available in all languages. On the other hand Learning based techniques deliver good results nevertheless they require obtaining datasets and require training.

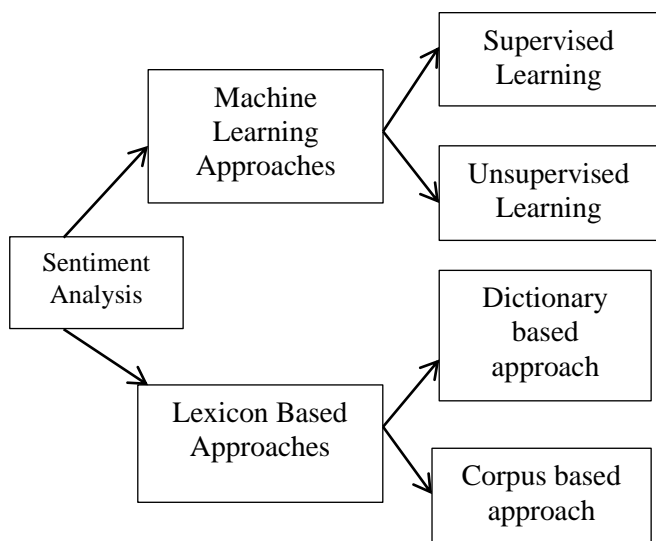


Figure 1. Approaches of sentiment analysis

Machine learning tasks are typically classified into three broad categories, depending on the nature of the learning "signal" or "feedback" available to a learning system. These are:

Supervised learning: The computer is presented with example inputs and their desired outputs, given by a "teacher", and the goal is to learn a general rule that maps inputs to outputs.

Unsupervised learning: No labels are given to the learning algorithm, leaving it on its own to find structure in its input. Unsupervised learning can be a goal in itself (discovering hidden patterns in data) or a means towards an end.

Sentiment lexicon - Sentiment words or phrases (also called polar words, opinion bearing words, etc.) E.g., Positive: beautiful, wonderful, good, amazing. Negative: bad, poor, terrible, cost an arm and a leg. Many of them are context dependent, not just application domain dependent. Two main ways to compile such lists:

Corpus-based approaches : Often used as a double propagation between opinion words and the items they modify require a large corpus to get good coverage.

Dictionary-based methods: Typically use wordnet's synsets and hierarchies to acquire opinion words and usually do not give domain or context dependent meanings.

III. RESULTS AND DISCUSSION

1. Proposed System

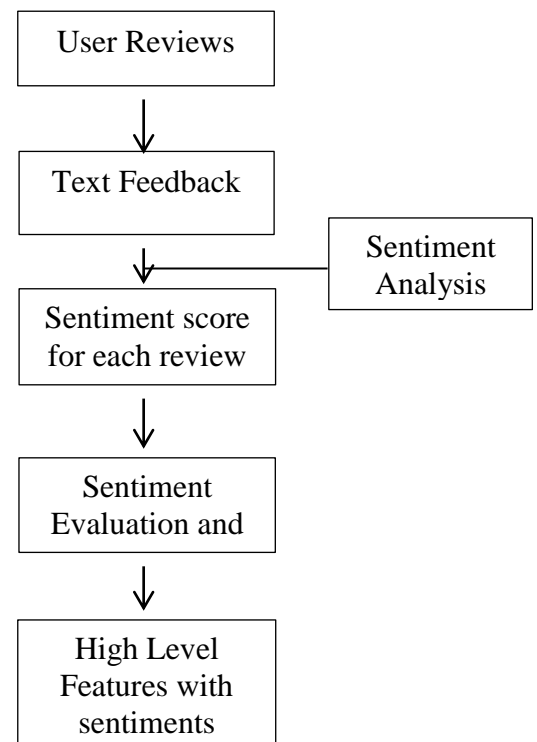


Figure 2. Overview of this approach

As above diagram explains, everything the project will be expected to do the earlier stage will comprise of the extraction of the user reviews for the purpose of analysis on it. The user reviews will consists of the text feedback. The sentiment analysis will be carried on which will create score for each text review generated.

The evaluation and the estimation process will then help the developer to study the various possible areas of improvement and expertise, which will help the developer to create a better application in future for the users. The high level features and information can be extracted on the basis of the information generated and sentiment scores are evaluated.

2. Processing Steps

There are standard methods involved in above techniques. Those are as follows:

2.2.1 Data Collection

The data collection (data RAW) from google playstore is collected for analysis.

2.2.2 Preprocessing

Data in the form of raw comments is acquired by using the python library which provides a package for simple processing through application interface.

A comment acquired by this method has a lot of raw information in it which we may or may not find useful for our particular application. It comes in the form of the python “dictionary” data type with various key-value pairs. Since a lot of information we only filter out the information that we need and discard the rest.

2.2.3 Classification

For the purpose of classification we will classify the comments into three types

- **Positive** : If the entire review has a

positive/happy/excited/joyful attitude or if something is mentioned with positive connotations. Also if more than one sentiment is expressed in the comment but the positive sentiment is more dominant. Example: “It is a

great application and works perfectly fine on my phone!!”.

- **Negative** : If the entire comment has a negative/sad/displeased attitude or if something mentioned with negative connotations. Also if more than one sentiment is expressed in the comment but the negative sentiment is more dominant. Example: “I did not like the application as it provides too many tabs and lags too much on my phone”.

- **Neutral/Objective** : If the creator of review expresses no personal sentiment/opinion in the comment and merely transmits information. Example: “I recently downloaded this application on my new phone”.

2.2.4 Feature Extraction

- **Tokenization**: It is the process of breaking a stream of text up into words, symbols and other meaningful elements called “tokens”. Tokens can be separated by whitespace characters and/or punctuation characters. It is done so that we can look at tokens as individual components that make up a tweet.

- **Sentence Splitting** : Splits a sequence of tokens into sentences.

- **Parts-of-Speech Tagging** : POS-Tagging is the process of assigning a tag to each word in the sentence as to which grammatical part of speech that word belongs to, i.e. noun, verb, adjective, adverb, coordinating conjunction etc.

- **Named Entity Recognition(ner)** : Recognizes named (Person, Location, Organization, Misc) and numerical (Money, Number, Date, Time, Duration, Set) entities. With the default annotators, named entities are recognized using a combination of CRF sequence taggers trained on various corpora. Also Implements a simple, rule-based NER over token sequences building on Java regular expressions. The goal of this Annotator is to provide a simple framework to allow a user to incorporate NE labels that are not annotated in traditional NL corpora. For example, a default list of regular expressions that we distribute in the models file recognizes ideologies (Ideology), nationalities (Nationality), religions (Religion), and titles (Title).

• **Stop-words removal** : Stop words are class of some extremely common words which hold no additional information when used in a text and are thus claimed to be useless . Examples include “a”, “an”, “the”, “he”, “she”, “by”, “on”, etc. It is sometimes convenient to remove these words because they hold no additional information since they are used almost equally in all classes of text, for example when computing prior-sentiment-polarity of words in a comment according to their frequency of occurrence in different classes and using this polarity to calculate the average sentiment of the tweet over the set of words used in that comment.

IV. CONCLUSION

The approach for extracting app features mentioned in user reviews and their associated sentiments. These can help app analysts and developers to analyze and quantify users’ opinions about the single app features and to use this information e.g., for identifying new requirements or planning future releases.

A number of works has been done for the informal reviews and blogs. But we look forward to work on the application world and help to provide users with more refined and critic free applications on Google playstore. As the user reviews of apps vary from category to category, the proposed procedure is efficient from that point of view. The result is sufficient up to the mark for judging the Android App and the developers are also able to predict the problem and the improvement needed in the app for its popularity within less time. There is major advantage of sentiment analysis in mobile environment of analyzing the reviews of users using Google apps.

V. REFERENCES

- [1] International Journal of Innovative Research in Advanced Engineering (IJIRAE) ISSN: 2349-2163 Issue 1, Volume 2 (January 2015) <http://www.ijirae.com/volumes/Vol2/iss1/28.JACS10092.pdf>
- [2] International Journal Of Computer Science And Applications www.researchpublications.org
- [3] International Journal of Emerging Engineering Research and Technology Volume 3, Issue 1, January 2015, PP 51-55 Sentiment Analysis Based on Dictionary Approach
- [4] Neha S. Joshi et al, / (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 5 (4) , 2014, 5422-5425 A Survey on Feature Level Sentiment Analysis
- [5] International Journal of Computer Applications (0975 – 8887) Volume 121 – No.20, July 2015 Sentiment Analysis on Social Media and Online Review