Youtube Video Ranking: A NLP based System

Selvakumar K, Rajesh M, Eshwar S, Shraveen BS

Abstract**:** ***YouTube is an acclaimed video information source on the web among various social media sites, where users are sharing, commenting and liking/dis-liking the video along with the continuous uploading of videos in real-time. Generally, the quality, popularity and relevance of results obtained from searching a query are obtained based on a rating system. Now and then few irrelevant and substandard videos are ranked higher because of higher views and likes. To address this issue, we put forth a sentiment analysis approach on the user comments based on Natural Language Processing. The suggested analysis will be helpful in providing a desirable result to the search query. The effectuality of the system has been proved in this paper using a data driven approach in terms of accuracy.***

Keywords**:** ***Natural Language Processing, YouTube, Sentiment Analysis, Vader.***

I. INTRODUCTION

Sentiment analysis has become a common buzz word in the recent days. It has become so common that a layman can make sense out of it. There are a lot of existing systems for sentiment analysis till date. It has been surveyed that around 45 different parameters can be tuned to analyse Facebook data to precisely predict its sentiment. But sentiment analysis is not that simple. Consider simple systems like twitter, Facebook and other social media or movie datasets, here sentiment analysis is constrained to a single context where the sentiment can be for or against a political view, likes or dislikes of a common topic, friendship analysis, or a mere prediction of a movie being positive or negative. In all these fields the context is very simple and precisely defined. But when it comes to YouTube, things get tricky. A video on YouTube has several contexts, it can be a film, it can be a fashion video, it can be a lifestyle video or it can be videos which actually need negative comments. For example, lets assume someone has posted an “harassment incident” on YouTube, since the comments can be negative doesn’t mean the video is negative. The dislikes avalanche can be due to content there. But the video is positive and it does spread awareness. We watch youtube videos in our life from day to day and so it is important to analyse its sentiment.

But who can this be done? Based on what parameters can youtube videos be analysed for? That is where various parameters come to picture. Though this article is meant to focus on the ensemble of like dislike ratio and the Natural language processing outcomes of the comments from the YouTube data which is provided by google cloud, there are various parameters that can be used to analyse YouTube videos:

- Transcript of the video (translation or speech-to-text)
- Likes/Dislikes
- Comments
- Share captions
- Share comments
- Plus post comments
- Social media related analysis

II. LITERATURE SURVEY

The buzz words today have been AI, ML and all the other technologies. The beauty of Natural Language processing is that it is a subset of techniques from all of these. One of the fastest growing streams in computer science is said to be sentimental analysis. This specific feature helps to identify the type of response users give to a product launched in the market. A product now a days reaches millions of users around the world and each user has his/her own opinions about it on whether it is good, bad or neutral. This helps to find the overall success rate of the product based on the user’s responses. Getting every user’s input and finding the average success rate by conventional methods is tedious and inaccurate but Sentiment Analysis helps nullify that issue.[1] This can not only be used for products but also for any purpose that requires feedback. This plays a major role in election campaigns. It helps the political party analyze their stand when compared to their competitors. Social media plays a major role in providing millions and billions of bytes of information for sentiment analysis. Sentiment Analysis has its origin from the public opinion analysis of the 20th century and text subjectivity analysis of the 90’s. Sentiment analysis was initially used for analyzing product reviews and now is used in social media texts in Twitter, FaceBook etc., Stock markets, elections, disasters, medicine, software engineering and cyberbullying are also some areas where sentiment analysis plays a major role.

Sentiment analysis is a combination of methods, techniques and tools for finding and getting information such as opinions and attitudes from the language spoken by the subject. Sentiment analysis classifies results based on opinion polarity such as positive, negative or neutral.

Manuscript published on November 30, 2019.
***** Correspondence Author

Selvakumar K*, Computer Science and Engineering, Vellore Institute of Technology, Vellore, India.

Rajesh M, Computer Science and Engineering, Vellore Institute of Technology, Vellore, India.

Eshwar S, Computer Science and Engineering, Vellore Institute of Technology, Vellore, India.

Shraveen BS, Computer Science and Engineering, Vellore Institute of Technology, Vellore, India.

Sentiment analysis typically works on a product or a service whose review has been made public. [1]

The sentiment analysis has been done on a large scale using the information from newspapers and blogs. The project presented here assigns the scores to each unique entity in the text corpus and indicates whether the given context has a positive or a negative opinion.

The system consists of three phases namely:

1) Sentiment Identification Phase which associates expressed opinions with each relevant entity

2) Sentiment Aggregation Phase and

3) Scoring Phase - scores each entity relative to others in the same class.

Content we get through news is seldom neutral but most probably good or bad. Computers now a days can't still completely comprehend the natural language text but statistical analysis of relatively simple sentiment cues can provide a meaningful sense. This system determines the public sentiment on each of the hundreds of thousands of entities and how this sentiment varies with time.

[2]Algorithmic Construction of Sentiment Dictionaries - Calculating the reference frequencies of adjectives with positive and negative connotations is the core method used for the sentiment index. Sentiment-alteration hop eliminates the ambiguous terms by counting the polarity strength of the candidate terms. Path based analysis of synonyms and antonym sets from WordNet is used to expand small candidate seed lists into positive and negative words into full sentiment lexicons.

Evaluation of significance - this project provides validity of it's sentimental evaluation by providing evidence which is done by correlating their index with several real-world events like results of professional baseball and basketball games, stock market indices and seasonal effects. Positive correlation proves that the sentiment analyzer can accurately measure public sentiment.

Sentiment Index Formulation - It uses a technique called justaposition of sentiment terms and entities and frequency weighted interpolation with world happiness levels to score entity sentiment. [2]

Sentiment analysis done using twitter data. Microblogging is an emerging new trend. It's popular among internet users and is a very popular communication tool. These users share opinions on different aspects of life everyday and hence these websites are a rich source of data for opinion mining and sentiment analysis. Twitter is one such microblogging platform. This project automatically creates a corpus for sentiment analysis and opinion mining purposes. Linguistic analysis is done one the collected corpus and a sentiment classifier is built based on it. The sentiment classifier determines positive, negative and neutral sentiments for the document. These techniques have been proven to be efficient and perform better than the previously proposed methods. The analyzer basically works with the english language by default but can be used with any other language.

Users of these social media sites basically write about their life, share opinions on variety of topics and discuss current issues Social Media or Microblogging sites like FaceBook, Instagram, Twitter etc., are popularly used by users to write about their lives, post photos, share opinions, discuss on current issues and spread awareness. These sites provide sophistication to communicate with other microblogger members around the world and hence it has replaced the conventional communication methods. Hence social media sites now a days are rich gold mines for gathering information about any topic. This data can be used for marketing and social studies. A dataset is formed from the messages collected in twitter. Twitter contains a very large number of short messages.

Under the [3]manufacturing sector questions like, people's opinion about a company's product, what more are people expecting their product to be etc., pop up. Similarly in political parties, the members would be interested to know how the people are reacting to their schemes, opinions on current debates and what is their image in society and how strong is their people support etc., The answer to all these questions can be easily and accurately derived from social media based on the likes and dislikes and user comments on the specific topic, company or product.

A corpus of 300000 text posts from Twitter was created where the texts where evenly split automatically between three sets of texts such as the one's with positive, negative and neutral emotions where the positive emotions contain happiness, amusement etc., whereas the negative ones contain sadness, anger, disappointment etc., and neutral texts are those that only state facts and do not express any emotions[3]

This paper brings in a new approach to sentimental analysis by making use of

SVM - Support Vector Machines to combine diverse sources of potentially pertinent information which includes several favorability measures for phrases and adjectives.

Models using these features are combined with unigram models which have been shown to be effective in the past and lemmatized versions of unigram models. When this was experimented on the movie review data from epinions.com, it showed that hybrid SVMs with real-valued favorability measures showed greater performance, producing the best results published using the data. When another experiment on a smaller dataset such as music reviews was conducted, the results notified that taking into account the topic information into such models may also lead to improvement.

Large amounts of research activities have been devoted to recognize favourable and unfavourable sentiments towards particular subjects within natural language texts.

There are [4]numerous and varied areas of application for such analysis, ranging from newsgroup flame filtering and informative augmentation search engine responses to analysis of public opinion trends and customer feedback. For all these tasks, classifying the tone as positive, negative or neutral is an important step. Opinions in natural language processing are very often expressed in subtle and complex ways, presenting challenges which may not be easily addressed by simple text categorization approaches such as n-gram or keyword identification approaches.

Recognizing the semantic impact of words or phrases is a challenging task but, in many cases, the overarching sentiment of a text is not the same as that of decontextualized snippets.

Negative reviews may contain positive phrases while the meaning of the phrase is still negative. For Eg: Sarcasm. SVM plays a major role in identifying the positive and negative sentences and phrases accurately.

The current approach stresses on the use of a diverse variety of information sources and SVMs provide the required tools to bring these sources together. The results of a variety of experiments are presented, using both data which is not topic annotated and data which has been hand annotated for topic. Now comparing with the former the present approach may allow further improvements to be gained given knowledge of the topic of the text.[4]

Sentiment Analysis using [5]Deep Convolution Neural Networks. Sentiment analysis done on short texts such as single sentences like twitter messages is a challenging task, reason being the limited contextual information that they normally contain. This paper proposes the use of a new idea known as deep convolution neural networks which exploits information from character to sentence-level information to perform sentiment analysis of short texts. This approach has been applied to two corporas namely the Stanford Sentiment Tree Bank(SSTB) which contains movie review sentences and the Stanford Twitter Sentiments(STS) which contains titter messages. While testing with SSTB the results had an accuracy of 85.7% and fine grained classification, with 48.3% accuracy. The accuracy achieved using the STS approach turned out to be 86.4%.

The proposed network named Character to Sentence Convolutional Neural Network(CharSCNN), uses two layers to extract required features from words and sentences of any size. The proposed network can easily explore the word embeddings produced by unsupervised pre-training. The effectiveness of CharSCNN for sentiment analysis of texts was calculated by using two domains: movie review sentences and twitter tweets. It was able to achieve state of the art results in both domains. This experiment uses unsupervised pre-training, contribution of character-level features and effectiveness of sentence level features to detect negation.[5]

SENTIWORDNET 3.0 is a lexical resource explicitly used for supporting sentiment classification and opinion mining applications. Previous version is the [6]SENTIWORDNET 1.0 which is also publicly available for research purposes.

Both versions are the result of automatically annotating all WORDNET synsets according to their degree of positivity, negativity and neutrality.

SENTIWORDNET 1.0 and 3.0 uses different versions of WORDNET which they annotate automatically which now includes a random-walk step for refining the scores. SENTIWORDNET 3.0 when used with WORDNET 3.0 showed an improvement in accuracy of 20% with respect to SENTIWORDNET 1.0.

The different versions of SENTIWORDNET are:

1. SENTIWORDNET 1.0 publicly made are available for research purposes.

2. SE NTI WORD NET 1.1 and 2.0 was never released publicly

3. SENTIWORDNET 3.0

Version 1.0 uses WORDNET 2.0 and automatic annotations where carried out with weak supervisions and semi-supervised learning algorithm.

Version 3.0 uses WORDNET 3.0 and the results of the semi-supervised learning algorithm are only intermediate step of the annotation process, since they are fed into an iterative random walk process that is run to convergence. SENTIWORDNET 2.0 and 3.0 are the output reached after the convergence.[6]

III. TOOLS AND METHODOLOGY

A. Dataset

We do not use any predefined dataset rather we use the Vader lexicon dictionary which uses Natural Language processing to predict the outcome of the word. The words are assigned a sentiment and we normalize it between -1 and 1, with 15 as the normalization parameter. We arrived at this value using trial and error method.

B. Google Cloud Platform

The system inherently combines youtube comments along with its sentiment and likes as heuristics for a perfect ranking. Scraping of data can be done but this can become less efficient as the comment keeps on loading and we would have to make our javascript run in a loop until the show more button fades. This method is fine for normal video with just in the range of 1000s of comments and comment like count under it. But these techniques become time consuming when it goes for millions of comments and corresponding like count. Here google, provides the solution. We have used google cloud platform to retrieve comments and their statistics. We also use view count as a parameter but assign less weight to it.

C. VADER Sentiment Analysis Library

Sentiment analysis of text is a really big field, however the tools used really boils down to just two approaches the lexical approach and machine learning approach. And Vader is lexical approach-based library.

[7]Valence Aware Dictionary and sEntiment Reasoner (VADER) is a rule-based and lexicon-based sentiment analysis tool. This library in Python has been specially created to be able to apply to the text expressed in social media. This library makes use of sentiment lexicon (list of lexical features like words etc.,) which are usually labelled positive or negative according to their semantic operation. Vader uses human-centric method for sentiment analysis, combining empirical validation and combining qualitative analysis by human raters and wisdom of the crowd.

In the lexical approach we have a dictionary of sentiment, with a target of mapping each word to sentiment. These sentiments can be a range of scores or intensities (numerical) or positive, negative, neutral (categorical). In this approach we try to assess the sentiments of texts without the need of anything else. In lexical approach we decide what the whole sentence sentiment score is based on the sentiment score of each word. The main advantage is the use a dictionary of emotions instead of a trained model using labelled data.

Sentiment scores or emotional intensity of words (including acronyms and slang) are measured on the scale ranging from -4 to +4, where most negative is -4 and most positive is +4 and the midpoint 0 represents neutral sentiments. Since the sentiment scores can be arbitrary the creators of VADER sentiment analysis have made use a number of human raters and the wisdom of crowd averaging their ratings for each word.

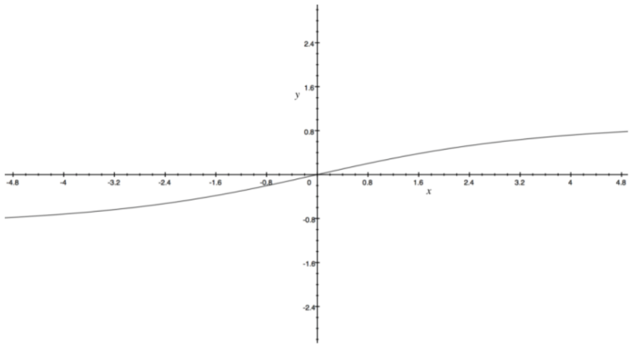
Sentiment scores of a sentence ranges from -1 to +1, where most negative is -1 and most positive is +1. The sentiment score of each sentence is calculated by adding the sentiment score of each sentiment-bearing word and then applying a normalization. Once such normalization that can be used is Hutto's normalization,

Figure 1: Normalization formula

$$\frac{x}{\sqrt{x^2+\alpha}}$$

Where a is normalization constant and x is sum of sentiment scores of each sentiment-bearing word in the sentiment.

Figure 2: Plot of normalization with change in X values



From the above graph we can conclude that it gets closer to +1 or -1 as x grows larger. Thus, we can say that VADER works best for short documents, tweets and social media comments, not on large documents.

VADER sentiment analysis takes into account of other contextual elements like capitalization, modifiers, and punctuations which can also impart some emotion. The effects of these are quantified using human raters. VADER sentiment analysis takes into account the sentiment score of each sentence proportional to each question marks and exclamation points ending the sentence. VADER also takes into account the capitalization of words in sentences and decrements or increments their sentiment score accordingly. VADER uses degree modifiers, they do so by using a booster dictionary which contains a set of human rated boosters and dampeners. The effects of these boosters and dampeners also depends on its distance to the word. In VADER there is also a shift in polarity due to the use of "but". So, VADER uses a but checker, which reduces the valences of the words before but to 50% and the words after to 150% of their values.

D. Algorithm and Implementation

Before starting with the natural language processing part, i.e Sentiment analysis of the comments. We have to fetch the data from the google API, but we have to extract what we need. The data retrieval algorithm for getting view count of a video, dislike, like is given below.

Algorithm 1: Youtube Video View count, dislike, like count retrieval

Input: api endpoint, api key

Output: counts of the parameters

```
procedure get statistics
        response = getRequest(apiEndPoint, apiKey)
        view_count = response[video][item][stats][‘viewCount’]
        like_count = response[video][item][stats][‘likeCount’]
        dislike_count = response[video][item][stats][‘viewCount’]
    return view_count, like_count, dislike_count
```

The above same procedure can be used for comment retrieval too. The parameter given above to retrieve is changed to comment threads. The data retrieved till now contributes a lot already to the youtube ranking system. But this how youtube already ranks their videos, combined with tagging. Youtube uses a special system to rank their videos where users give tags for their videos. These tags are then associated with the context and these provide a matrix representation. It uses Eigen vector-based ranking.

Algorithm 2: Polarity Score Computation

Input: Youtube comments as text, videoid

Output: csv file of {comment, score, likeCount}

```
procedure get Polarity Scores
        video_c_threads = getComments(GoogleCloud, videoid)
        for videoComment in video_c_threads:
        data cleaning of videoComment
        array.append(tokenization(videoComment))
        //initialize hashmap with comment with score -1, likeCount = 0
        hashMap = hashMap(arrayComments, -1,0) //initial values
        for VideoComment in comments:
        score = 0
        for token in VideoComment.split(“ ”):
        score = score + vader.get(token)
        finalScores.append(normalize(score))
        csvFile.add(comments, finalScores, likeCount)
    return csvFile
```

Every sentence consists of two polarities. Positive and negative. Polarity score is something that assigns the skewness that assigns positive or negative score. 1 is positive, -1 is negative and 0 is neutral. VADER does very good with emojis and slang of the people. The normalization we have used in the above algorithm is $\frac{x}{\sqrt{x^2+\alpha}}$

x → sum function of the polarity scores of the words of sentence

α → normalization param after trial and error (value = 15)

Since vader gives values between -4 and 4 we have normalized between -1 and 1.

Example: “The video is great!”

The → 0, Video → 0, Is → 0, Great → 2.87

X → 2.87

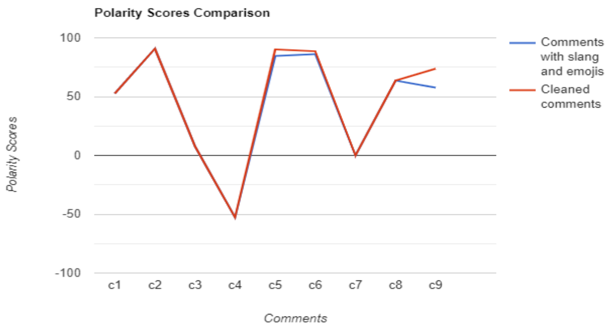
α → 15

Normalization = 15 / (sqrt(2.87 * 2.87 + 15))
Value = 0.83939

IV. RESULTS AND DISCUSSION

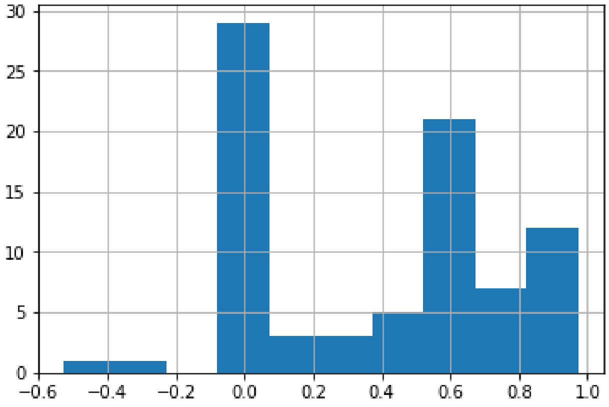
As discussed earlier, our analysis consists of two parts, one with emojis and slangs and other part is without the emojis and slangs. In this article, we refer the embedded emoji part as text and cleaned version as clean text. As text and clean text has more or less the same polarity score we can infer that VADER performs well with emojis and slang. We can also infer that exclamation plays a major part,

Figure 3: Line graph of polarity scores before and after cleaning



The above graph interprets extreme evidence for our intuition that emojis and exclamations of youtube comments can be very well recognized by VADER. The deviation occurs only with exclamations. This again provides us with an intuition that number of exclamations is directly proportional to the emotion of the user, thus contributing to more of the polarity scores and thus the deviation.

Figure 4: Histogram distribution of video sentiment



The histogram distribution of the comments shows that this video is more skewed towards neutral and positive comments. This histogram as a representation of data can be coherently used with already existing ranking algorithm to rank the best videos higher.

V. CONCLUSION

We again insist on the fact that YouTube ranking is solely based on view count and some other parameters. Our article clearly changes this perspective and solely concentrates on how emotion of an user towards a video is important in ranking the video. Our system can be integrated with the existing system as we converge towards user experience and leave out the facts of increasing views through software. Since our comment is also weighted based on the likes it received, we take in the fact of how many people support the comment.

VI. FUTURE SCOPE

The system recommended in this paper has the potential to scale to multiple platforms where user sentiments can be used as a metric for ranking. The system can be very well integrated with Google making our system as an API as a service or it can be used for youtube video ranking as well. The future scope of the system can further extend to recommendations based on what kind of sentiment an user is more interested in and whether he watches videos based on videos or likes or the inherent cumulative approach that we proposed.

REFERENCES

1. Mika V. Mantyla, Daniel Graziotin, Miikka Kuutila, “The Evolution of sentiment analysis - A review of Research Topics, Venues and top cited papers”, February 2018.
2. Namrata Godbole, Manjunath Srinivasaiah, Steven Skiena,”Large-Scale Sentiment Analysis for news and blogs”, January 2007.
3. Alexander Pak, Patrick Paroubek, “Twitter as a Corpus for Sentiment Analysis and Opinion Mining”, May 2010.
4. Tony Mullen, Nigel Collier, “Sentiment Analysis using support vector machines with diverse information sources”, July 2004.
5. Cicero Nougueira dos Santos, Maira Gatti, “Deep Convolution Neural Network for Sentiment Analysis of short texts”, August 2014.
6. Stefano Baccianella, Andrea Esuli and Fabrizio Sebastiani, “SentiWordNet 3.0: An enhanced Lexical Resource for Sentiment Analysis and opinion mining”, January 2010.
7. VADER Sentiment Analysis Explained, http://datameetsmedia.com/vader-sentiment-analysis-explained/
8. VADER Documentation, https://www.nltk.org/_modules/nltk/sentiment/vader.html

AUTHORS PROFILE

Selvakumar K, currently working as an Associate Professor in Vellore Institute of Technology (VIT), Vellore. Completed Ph.D., Degree from Anna University, Chennai-25. Also, completed Master of Engineering in Computer Science from Anna University, Chennai; Bachelor of engineering from University of Madras. His field of specialization is Mobile Ad-hoc networking with Soft Computing approaches. Moreover, interested in Internetworking Technologies, Network Protocols and Design, Group Communication, Distributed Computing and Web Mining.

Rajesh M, is currently pursuing Bachelor of Technology in Computer Science and Engineering (4th year) at Vellore Institute of Technology, Vellore. Email Id: rajesh.marudhachalam@gmail.com .He is currently working on projects using concepts such as Natural Language Processing, Visual Cryptography and Statistics. His current domain of interests includes *Data Science, Artificial Intelligence and Machine Learning.*

Eshwar S, is currently pursuing Btech. CSE (4th Year) in VIT University, Vellore. Email id: northeasteditsofficial@gmail.com . He is currently working on projects based on Natural language processing and artificial intelligence. His current research interests include Data science, Machine learning, Web development, Android development and Virtualization.

B.S.Shraveen, is currently pursuing Bachelor of Technology in Computer Science and Engineering(4th year) at Vellore Institute of Technology, Vellore.
E-Mail: shraveen1998@gmail.com . He is currently working on projects using concepts such as Natural Language Processing, Visual Cryptography and Statistics. His current domain of interests include, Game Development, App development, Ethical Hacking and Cyber Security.