RATING SYSTEM USING REVIEWS - SENTIMENT ANALYSIS

Biswas Biswajeet Dash[1], Bishwajit Biswal[2], Ashaya Sinha[3], Ms. Gurpreet Kaur[4]

[1,2,3]Students, [4]Assistant Professor

[s]Department of Information Technology, [3]Department of Computer Science Engineering

BMCEM, Sonipat, India

Abstract: Today, the reviews of customers plays a vital role in enhancing global communications amongst consumers and also influencing consumer buying patterns. Famous E-commerce giants like Amazon, Flipkart, etc. provide a platform to consumers to share their experience or reviews and provide clear picture about the performance of the product for future buyers to get some basic idea about the product from those reviews. In order to extract valuable insights from a large set of reviews, classification of reviews into positive, neutral and negative sentiment is required. Sentiment Analysis is a computational study to extract some information from the text and try to understand customer sentiment about that product. The data used in this study are online product reviews collected from Amazon, Flipkart, Twitter, etc. This research study aims to gauge the public's sentiment in the form of reviews and provide an overall rating (i.e. positive, negative or neutral) according to the reviews that people write. In this study, we design a system that use algorithms to analyze the sentiments and try to provide rating to those reviews.

I. INTRODUCTION

In today's world, Machine Learning is one amongst the most well-liked topics in the market. There is a lot of things in our life which uses machine learning. Basically, Deep Learning is a broader a part of computer science and Deep Learning. It makes human works easier and also save a lot of time. In this research I would like to tell my reviews about a topic called Sentiment Analysis in Machine Learning. Sentiment Analysis is a frequent term in text classification and is essentially uses natural language processing (simply referred as NLP) + machine learning to interpret and classify emotions in text information.

II. DESCRIPTION OF TOOLS:-

Here is a description of tools used to design my system:
- PYTHON

Basically, Python is an interpreted, high level, general purpose programming language. Its design philosophy emphasizes code-readability with the use of important indentation. Python is dynamic and garbage-collected. It supports multiple programming paradigms, including procedural (or structured), object-oriented and functional programming. It made programming easy for coders because python has a number of libraries that enable coders to write program more efficiently and quickly with the use of those libraries.
- DJANGO

Basically, Django is a high-level Python web framework that encourages people for rapid development and also a clean pragmatic design. It was built by experienced developers at the Lawrence Journal-World newspaper, Adrian Holovaty and Simon Willison. This framework takes care of much of the hassle in web development, so that people can focus on writing the app directly without needing to reinvent the wheel. It's also free and open source web framework. It is ridiculously fast, reassuringly secure and exceedingly scalable framework.

- NLTK

NLTK is a popular platform for building programs which can easily work with normal human language data. It provides easily operated interfaces to over 50 corpora and lexical resources such as WordNet, along with a collection of text processing libraries for classification, tokenization, stemming, tagging, parsing, and semantic reasoning, wrappers for industrial-strength NLP (Natural Language Processing) libraries, and also an active discussion forum to discuss problems and help people.

III. LITERATURE REVIEWS[1]

Xing Fang and Justin Zhan [1] presents a study to tackle the problem of sentiment polarity categorization. A process for sentiment polarity categorization is showed with detailed description. Study aims to find polarity using sentiment sentences extraction and POS tagging. Algorithms used Naïve Bayesian classifier, Random forest, Support vector machine.

Wang [2] in his study predicted that the polarities of sentiments by using the SentiDiff that considered the relation between textual information and sentiment diffusion patterns on Twitter messages. A sentiment reversal was explored to analyze the sentiment diffusion of tweets. This method collected tweets and retweets from Beijing Intelligent Starshine Information Technology Corporation, China.

Zulfadzli Drus and Haliyana Khalid [3] presents in their study that lexicon based method is basically called as an unsupervised learning method. This lexicon method does not require any training data since it only depends on the dictionary. In most of the studies, Sentiwordnet and TF-IDR method are used for conducting sentiment analysis. This approach calculates the sentiment score based on the occurrences of the terms in the text data with other positive or negative words in the pre-developed polarity lexicons like Sentiwordnet [5].

Chhinder Kaur and Anand Sharma [4] in their study where they perform the experiment for analysing sentiments on

www.ijtre.com

39

different tweets using Python programming and NLTK library (which is used for pre-processing of tweets). They analyse the tweets dataset by using Textblob Library and after that show their results which is positive, negative, neutral sentiments through different visualizations.

Zeenia Singla, Sukhchandan Randhawa and Sushma Jain [7] shows in their research on over 4,000,00 unstructured reviews of Mobile Phones have been extracted from Amazon.com. Then the data was filtered to remove noisy data and also pre-processing the data to evaluate sentiment of the reviewers using supervised learning. The reviews were classified using machine learning classification models like Naïve Bayes, Support Vector Machine (SVM) and Decision Tree and after that cross validated to find the best classifier for this purpose.
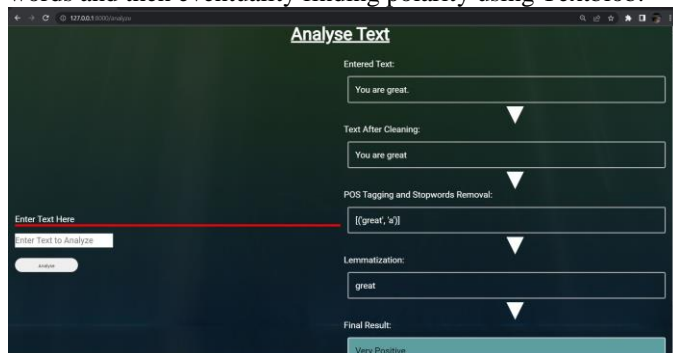
IV. PROPOSED WORK

Rating Prediction System involves multiple phases.
- Analyse Text
- Upload CSV
- Compare

Analyse Text
Analyse text involves working on text, clean the text using cleaning, tokenize the words, remove stop words and then POS tagging. Perform lemmatization on this POS tagged words and then eventuality finding polarity using Textblob.

Upload CSV
System here provides the ability to input the file explicitly. File extension allowed txt, excel or comma separated values (csv) files. Vader Analysis is performed and results are generated on screen.

Compare Models
Here System compare the results provided by different models and then check the accuracy for different models.

1. Textblob
Basically, Textblob is a Python (2 and 3) library which is used for the procession of textual data. This library provides a simple API used for diving into common natural language

processing (NLP) tasks such as part-of-speech tagging, noun phrase extraction, sentiment analysis, classification, translation, and more.

2. Vader Model [6]
VADER (Valence Aware Dictionary for Sentiment Reasoning) is a model which is used for text-sentiment analysis that is sensitive to both polarity (positive/negative) and also intensity (strength) of emotion. It is present in the NLTK package and also can be applied directly to unlabelled text data.

3. SentiWordNet
SentiWordNet is an opinion lexicon which is derived from WordNet database where each term is associated with some numerical scores which indicates positive, objective and negative sentiment information. This has two scores: positive score and negative score. According to these two scores (i.e pos_score and neg_score), the sentence is defined as positive, negative or neutral.
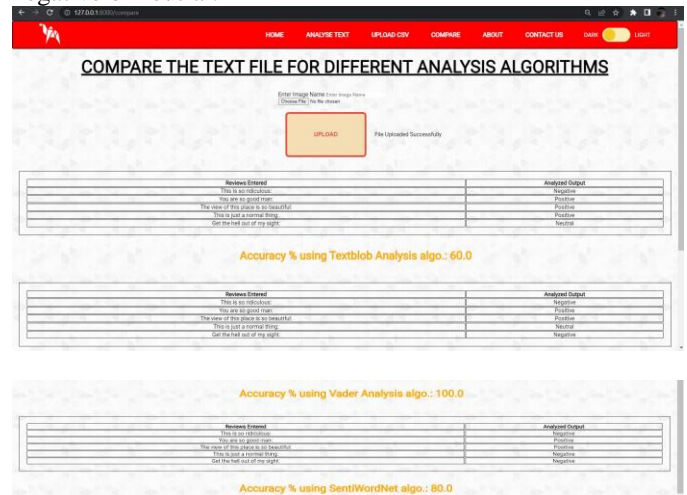
V. METHODOLOGY

Different models used:
1. Textblob library

2. VADER (Valence Aware Dictionary for Sentiment Reasoning) Model

3. SentiWordNet

1. Textblob
Let take a sentence as example and understand the methodology of Textblob library.

Example: You are great b.

Cleaning: it removes all the unnecessary letters, symbols which can cause error in reading text correctly.
You are great

Tokenization: It converts the words present in sentence into tokens.
['You', 'are', 'great']

Stop Words removal: It removes the stop words which are not useful like and, or, the, can, be etc.
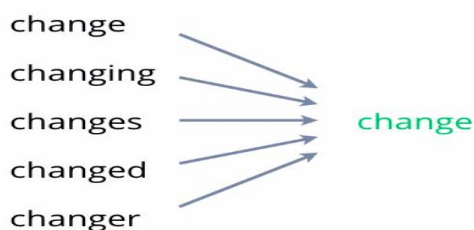
POS Tagging: It finds and club together words by adjectives, nouns, verbs and adverbs. It separates all words based on grammar. It creates dictionary.
[('You', 'PRP'), ('are', 'VBP'), ('great', 'JJ')]

Lemmatize: It reduces words to its smallest unit as possible for faster efficiency.
Here "great" is already in its smallest unit.

Polarity: It involves finding the words and check whether those words are positive words or negative words and the it provide polarity to that sentence which later on used for checking that the polarity is negative, neutral or positive. Polarity value lies in between [-1,1].
[('great', 'a')]

Here keyword 'great' is responsible for polarity determination of the complete sentence thereby generated positive.

2.   VADER Model
VADER is a lexicon and a rule-based sentiment analysis tool for social media text. This lexicon has been built manually, by aggregating ratings coming from 10 human annotators. Since it contains only 7000 words so it is not so extensive. This takes the sentence directly and after that it process and provide polarity.
Since it is specially designed for social media, so it also covers emojis and abbreviations (e.g. "lmao", "lol" etc.) that other dictionaries normally don't add. That's the main advantage of this Model.
Here is a sample of VADER dictionary:

WORD	POLARITY
attractions	1.8
attractive	1.9
attractively	2.2
avoided	-1.4
avoider	-1.8
avoiders	-1.4

The first column is the target word and the second is the polarity according to the word.
In this dictionary, the scores can range from -4 to 4 instead of usual -1 or 0 to 1 range.

3.   SentiWordNet
SentiWordNet assigns scores to WordNet synsets instead of assigning scores to words. Each synsets has both a positive and a negative score, ranges between 0 and 1.
Since this is based on WordNet, so it inherits all the good features, like the separation between different POS tags.
This assigns different scores to the same word depending on its different meanings.
For example, a word "attractive" can have atleast two meanings:
-Pleasing to the eye or mind through its beauty.
-Second id the property of magnet, i.e. the ability to attract or repel.
So, in SentiWordNet, you will find two scores according to the meaning.

POS	SYNSET ID	POSITIVITY	NEGATIVITY	TERMS	DEFINITION
a	00166146	0.875	0	attractive#1	pleasing to the eye or mind...
a	00169955	0	0	attractive#3	having the properties of a magnet...

VI. CONCLUSIONS AND RESULTS

The three algorithms are used and analyzed to eventually generate results as per convenience.
This enables us to get emotions of texts whether from any social media or E-Commerce websites. Irrespective of the nature of content, whether structural or unstructural, proper error removing and text cleaning operations are applied. Thereby generated results with more than 50% accuracy. System also enabled to provide a file containing multiple texts in a single file and can provide a detailed analysis with charts to get the whole outcome of a scenario.

Sentence Input	Textblob	Vader	SentiWordNet
This is so ridiculous:	Negative	Negative	Negative
You are so good man:	Positive	Positive	Positive
The view of this place is so beautiful:	Positive	Positive	Positive
This is just a normal thing:	Positive	Neutral	Negative
Get the hell out of my sight:	Neutral	Negative	Negative

REFERENCESs

[1] Xing Fang and Justin Zhan (2015). Sentiment analysis using Product review data. Fang and Zhan Journal of Big Data (2015)

[2] L. Wang, J. Niu, and S. Yu, "SentiDiff: Combining Textual Information and Sentiment Diffusion Patterns for Twitter Sentiment Analysis", IEEE Transactions on Knowledge and Data Engineering, Vol. 32, No. 10, pp. 2026-39. 2019.

[3] Zulfadzli Drus and Haliyana Khalid (2019). Sentiment Analysis in Social Media and Its Application: Systematic Literature Review. In The Fifth Information Systems International Conference, 2019

[4] Chinder Kaur and Anand Sharma (2020). Twitter Sentiment Analysis on Coronavirus using Textblob. https://www.researchgate.net/publication/339998775

[5] Agarwal, Basant, Namita Mittal, Pooja Bansal, and Sonal Garg. (2015) "Sentiment Analysis Using Common-Sense and Context Information." Journal of Computational Intelligence and Neuroscience 9 (2015).

[6] Hutto, C.J. & Gilbert, E.E. (2014). VADER: A Parsimonious Rule-based Model for Sentiment Analysis of Social Media Text. Eighth International Conference on Weblogs and Social Media (ICWSM-14). Ann Arbor, MI, June 2014.

[7] Zeenia Singla, Sukhchandan Randhawa, Sushma Sentiment Analysis of Customer Product Reviews Using Machine Learning. In International Conference on Intelligent Computing and Control (I2C2), 2017.s