Sentiment Categorization through Natural Language Processing

[1]Jyovita Christi, [2]Prof. Gayatri Jain

[1]Information Technology Department,
[1]Gujarat Technological University, L.J. Institute of Engineering & Technology, Ahmedabad, India.
[2]Head of Department (Post Graduation), L.J. Institute of Engineering & Technology.

Abstract : Sentiment is an attitude, thought, or judgement prompted by feeling. Sentiment Categorization, studies people's sentiments towards certain units. Sentiment Analysis isn't an unfamiliar term anymore. Today, smart phones, high speed and affordable Internet and various forums and social networks, have made it very common for people to give voice to their opinions. Therefore, a lot of textual data is available in various forms where people express their opinions. Analysing this data to know the underlying sentiment behind it has also become quite popular these days. Sentiment Categorization involves classifying text as positive or negative towards a certain target. The model proposed here, makes use of a hybrid approach of Natural Language Processing and Machine Learning to achieve an accuracy of 90% for an IMDB movie review dataset. To achieve this accuracy, the system uses methods like: Bag of words model, TF-IDF to calculate the relevance of each term in each sentence, regular expressions to remove punctuations and retain emojis by shifting them to the end of each sentence, tokenization and stemming to break the sentences into tokens and restore all words to their roots, stop word removal to remove words that do not bear any sentiment, and finally the logistic regression algorithm to perform the sentiment categorization into positive and negative.

IndexTerms **- Sentiment Analysis, Opinion Mining, Natural Language Processing, TF-IDF, feature extraction, feature based, stop word removal, tokenization, logistic regression.**

I. INTRODUCTION

Internet, social media and their applications generate data with high-volume, high-velocity, high-variety, high-value, and high variability [1]. Several researchers have shown a keen interest in the exploitation of big social data in order to describe, determine and predict human behaviors in several domains [6, 7]. Almost 80% of internet data is text [8], therefore, text analysis has become a key element for public sentiment and opinion elicitation. Sentiment Analysis can be done by the following three approaches:

1. Lexicon-based approach: Relies on a sentiment lexicon and a collection of known sentiment terms. Most of these approaches use adjectives and verbs as indicators of the semantic orientation of text [9 10 11].

2. Learning-based approach: Uses Machine Learning techniques. In the supervised methods, models are trained using a large labeled dataset and opinion mining is done using Naïve Bayesian classification, maximum entropy principle and Support Vector Machine. In case of an absence of a labeled dataset, unsupervised techniques used. However, both methods fall short when it comes to analysing text with a different context.

3. Hybrid approach: Combines both Lexicon-based and Learning-based approaches. Lexicon-based approaches can be used to create a labeled dataset which in turn can be used to train a Learning-based model. This gives a higher accuracy than using the approaches separately.

In order to perform Sentiment Categorization, Natural Language Processing is required. Natural Language Processing for Sentiment Categorization requires the following tasks to be done:

Tokenization: Creating tokens out of each sentence.

POS-tagging: Tagging each word in the sentences with its part-of-speech.

Stemming and Lemmatization: Removing all embellishments from words and converting them to their base forms.

II. RESEARCH METHODOLOGY

2.1 Literature Survey

Paper 1: A comprehensive analysis of adverb types for mining user sentiments on amazon product reviews.
Authors: Ummara Ahmed Chauhan, Muhammad Tanvir Afzal, Abdul Shahid, Moloud Abdar, Mohammad Ehsan Basiri, Xujuan Zhou

Publication details: World Wide Web Journal (2020)
Summary of paper:
In a sentence, there are many parts of speech like adjective, adverb, noun, etc. This paper deals with which types of adverbs are best for classification of sentiments. This paper uses a dataset of 51,005 reviews about two products, office supplies and musical DVDs, from Amazon. In the conclusion, it shows that using superlative adverbs, an accuracy of 0.86% can be achieved. [1]

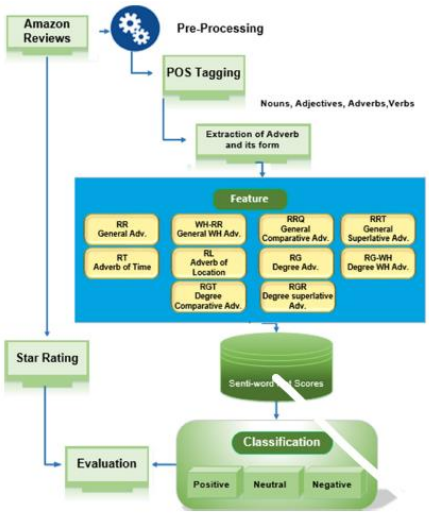

Figure: Proposed System using adverb types [1]

Paper 2: Sentiment analysis using product review data
Author: Xing Fang, Justin Zhan
Publication details: Journal of Big Data, 2015
Summary of paper: General process for sentiment polarity categorization is proposed with detailed process descriptions. Data used in this study are online product reviews collected from Amazon.com. Experiments for both sentence-level categorization and review-level categorization are performed with promising outcomes. [2]

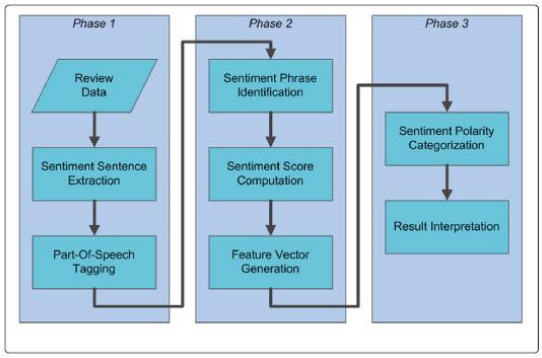

Figure: Sentiment Polarity Categorization Process [2]

Paper 3: A Hybrid Framework for Sentiment Analysis Using Genetic Algorithm Based Feature Reduction
Author: Farkhund Iqbal, Jahanzeb Maqbool Hashmi, Benjamin C. M. Fung, Rabia Batool, Asad Masood Khattak, Saiqa Aleem, Patrick C. K. Hung
Publication details: IEEE 2018
Summary of paper:
This paper proposes an integrated framework which bridges the gap between lexicon-based and machine learning approaches to achieve better accuracy and scalability. To solve the scalability issue that arises as the feature-set grows, a novel genetic algorithm (GA)-based feature reduction technique is proposed. By using this hybrid approach, the feature-set size is reduced by up to 42% without compromising the accuracy. The comparison of this feature reduction technique with more widely used principal component analysis (PCA) and latent semantic analysis (LSA) based feature reduction techniques has shown up to 15.4% increased accuracy over PCA and up to 40.2% increased accuracy over LSA. Furthermore, the sentiment analysis framework is also evaluated on other metrics including precision, recall, F-measure, and feature size. In order to demonstrate the efficacy of GA-based designs, a novel cross-disciplinary area of geopolitics as a case study application for our sentiment analysis framework is proposed. The experiment results have shown to accurately measure public sentiments and views regarding various topics such as terrorism, global conflicts, and social issues. The applicability of our proposed work in various areas including security and surveillance, law-and-order, and public administration is envisaged. [3]

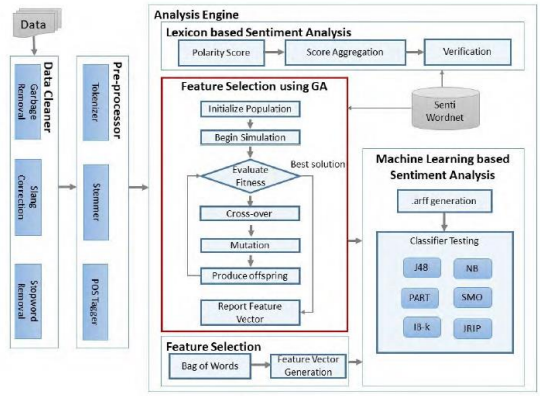

Figure: Proposed System using genetic algorithm[3]

Paper 4: Emotion and sentiment analysis from Twitter text
Author: Kashfia Sailunaz, Reda Alhajj
Publication details: Journal of Computational Science 2019
Summary of paper:
The target of the work described in this paper is to detect and analyze sentiment and emotion expressed by people from text in their twitter posts and use them for generating recommendations. Tweets and replies on few specific topics were collected a dataset with text, user, emotion, sentiment information, etc was created. The dataset to detect sentiment and emotion from tweets and their replies is used and the influence scores of users based on various user-based and tweet-based parameters is measured. Finally, this information to generate generalized and personalized recommendations for users based on their twitter activity. The method used in this paper includes some interesting novelties such as, (i)including replies to tweets in the dataset and measurements, (ii) introducing agreement score, sentiment score and emotion score of replies in influence score calculation, (iii) generating general and personalized recommendation containing list of users who agreed on the same topic and expressed similar emotions and sentiments towards that particular topic. [4]

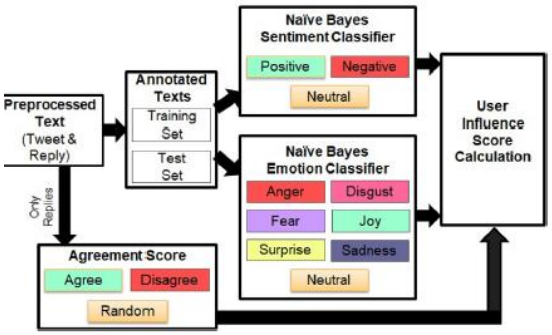

Figure: Proposed method for twitter data [4]

Paper 5: Aspect based Sentiment Oriented Summarization of Hotel Reviews
Author: Nadeem Akhtar, Nashez Zubair, Abhishek Kumar, Tameem Ahmad
Publication details: Procedia Computer Science (2017)
Summary of paper:
This study analyzes the hotel reviews and gives information that ratings might overlook. The reviews and metadata are crawled from website and classified into predefined classes as per some of the common aspects. Then Topic modelling technique (LDA) is applied to identify hidden information and aspects, followed by sentiment analysis on classified sentences and summarization. [3]

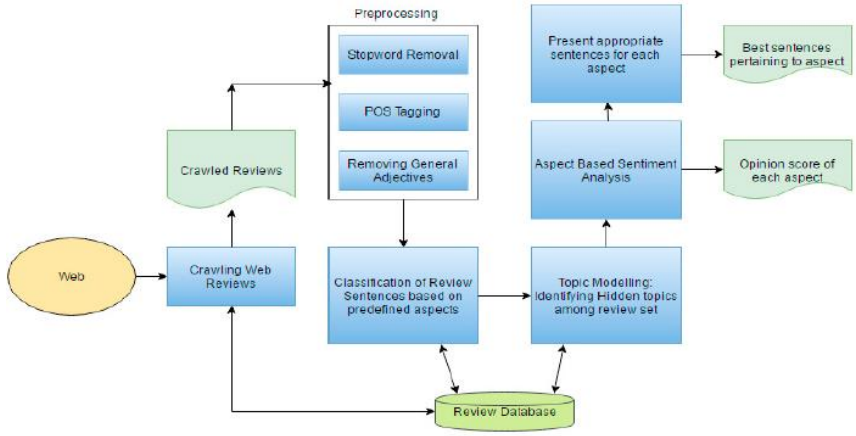

Figure: Basic Underlying Architecture of the Review Processing System[5]

2.2 Comparative Study:

PUBLICATION	TITLE	METHOD	RESULT ACCURACY %
World Wide Web Journal (2020)	A comprehensive analysis of adverb types for mining user sentiments on amazon product reviews	Superlative Adverbs	86
Journal of Big Data (2015)	Sentiment analysis using product review data	Support Vector Machine	83
IEEE (2018)	A Hybrid Framework for Sentiment Analysis Using Genetic Algorithm Based Feature Reduction	Genetic Algorithm	77
Journal of Computational Science (2019)	Emotion and sentiment analysis from Twitter text	Naïve Bayes Classifier	66
Procedia Computer Science (2017)	Aspect based Sentiment Oriented Summarization of Hotel Reviews	Topic Modelling (LDA), Lexicon-based	85

Table 2.1: Comparative Study

III. RESULTS AND DISCUSSION

3.1 Proposed Model

The proposed model is divided into following stages:

1. Loading the dataset
2. Transforming documents into feature vectors
3. Word relevancy using term frequency-inverse document frequency (TF-IDF)
4. Calculating sample TF-IDF
5. Data Preparation
6. Tokenization of documents
7. Document classification via a logistic regression model
8. Load saved model from disk
9. Model accuracy

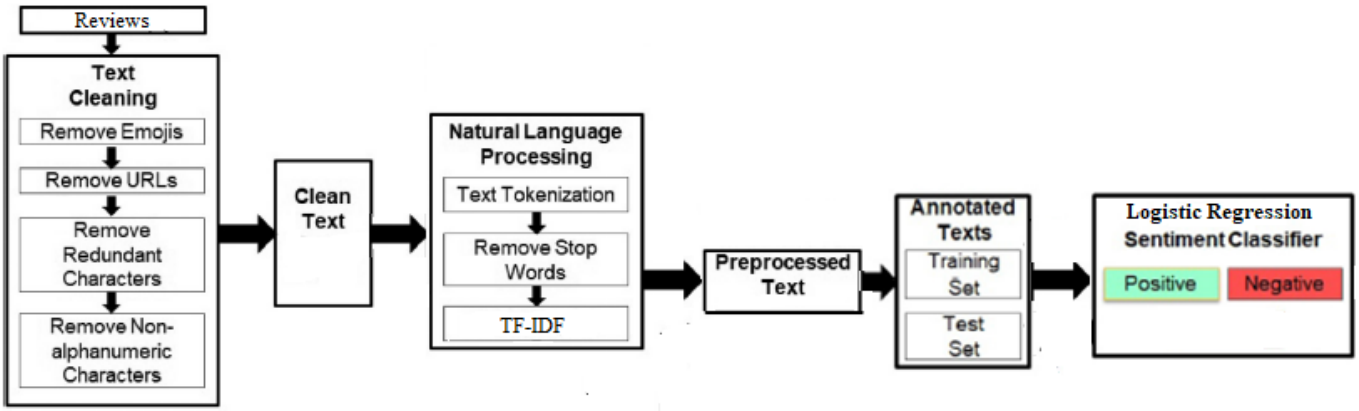

Figure 3.1 Proposed System

3.2 Loading the Dataset

The Dataset used in the implementation is an IMDB movie reviews dataset available at: http://ai.stanford.edu/~amaas/data/sentiment. It contains 1 lakh reviews with 50% positive and 50% negative reviews. The reviews also contain ratings. A review with at least 7 stars out of 10 is labelled positive (1) and a review with at most 4 stars out of 10 is labelled negative (0). A sample of an IMDB movie review is shown below:

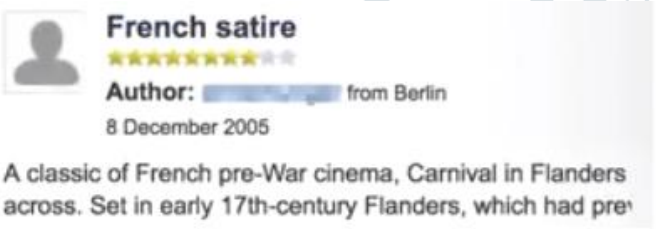

Figure 3.2: IMDB sample review

The dataset is loaded using pandas library. The code snippet below shows how the dataset is loaded and then shows the first 5 reviews as an output:

```
In [1]: import pandas as pd

        df = pd.read_csv('./movie_data.csv')
        df.head(5)
```

Out[1]:

	review	sentiment
0	In 1974, the teenager Martha Moxley (Maggie Gr...	1
1	OK... so... I really like Kris Kristofferson a...	0
2	***SPOILER*** Do not read this, if you think a...	0
3	hi for all the people who have seen this wonde...	1
4	I recently bought the DVD, forgetting just how...	0

Figure 3.3: Loading the dataset

3.3 Transforming documents into feature vectors

This step involves constructing the vocabulary of the bag-of-words model. Since the dataset is very large, it is impossible to fit its output on the screen or print it. Hence, to provide a better understanding of the various steps, code snippets are shown using a sample data.

1. The sun is shining.
2. The weather is sweet.
3. The sun is shining, the weather is sweet, and one and one is two.

The following code uses CountVectorizer to assign a unique number to each word in the above three sentences:

```
In [2]: import numpy as np
        from sklearn.feature_extraction.text import CountVectorizer

        count = CountVectorizer()
        docs = np.array([
                'The sun is shining',
                'The weather is sweet',
                'The sun is shining, the weather is sweet, and one and one is two'])
        bag = count.fit_transform(docs)

In [3]: print(count.vocabulary_)

        {'the': 6, 'sun': 4, 'is': 1, 'shining': 3, 'weather': 8, 'sweet': 5, 'and': 0, 'one': 2, 'two': 7}
```

The following code shows the array containing the frequency of each of the above words in each sentence:

```
In [4]: print(bag.toarray())

        [[0 1 0 1 1 0 1 0 0]
         [0 1 0 0 0 1 1 0 1]
         [2 3 2 1 1 1 2 1 1]]
```

3.4 Word relevancy using term frequency-inverse document frequency

A word that appears too often and in many sentences, is very likely to be irrelevant to the sentiment categorization process. E.g., a word like, 'is' will appear very often. So, for such a word, the importance can be reduced. This can be done using TF-IDF (Term Frequency-Inverse Document Frequency). It uses the following formulae:

$$\text{tf-idf}(t, d) = \text{tf}(t,d) \times \text{idf}(t, d)$$

$$\text{idf}(t, d) = \log \frac{n_d}{1 + \text{df}(d, t)},$$

Where n_d is the total number of documents, and df(d,t) is the number of documents d that contain the term t. Here, one sentence is considered as one document. The bag array shown in the previous task, can be converted into a relevance matrix as below:

```
In [5]: np.set_printoptions(precision=2)

In [6]: from sklearn.feature_extraction.text import TfidfTransformer

        tfidf = TfidfTransformer(use_idf=True, norm='l2', smooth_idf=True)
        print(tfidf.fit_transform(count.fit_transform(docs)).toarray())

        [[0.   0.43 0.   0.56 0.56 0.   0.43 0.   0.  ]
         [0.   0.43 0.   0.   0.   0.56 0.43 0.   0.56]
         [0.5  0.45 0.5  0.19 0.19 0.19 0.3  0.25 0.19]]
```

3.5 Calculating the tf-idf of each term:

The scikit-learn library used here, calculates tf-idf as follows:

$$\text{tf-idf}(t, d) = \text{tf}(t, d) \times (\text{idf}(t, d) + 1)$$

$$v_{\text{norm}} = \frac{v}{||v||_2} = \frac{v}{\sqrt{v_1^2 + v_2^2 + \cdots + v_n^2}} = \frac{v}{\left(\sum_{i=1}^{n} v_i^2\right)^{\frac{1}{2}}}$$

Using scikit learn, tf-idf is calculated for each term in the following manner:

```
In [8]: tfidf = TfidfTransformer(use_idf=True, norm=None, smooth_idf=True)
        raw_tfidf = tfidf.fit_transform(count.fit_transform(docs)).toarray()[-1]
        raw_tfidf

Out[8]: array([3.39, 3.  , 3.39, 1.29, 1.29, 1.29, 2.  , 1.69, 1.29])

In [9]: l2_tfidf = raw_tfidf / np.sqrt(np.sum(raw_tfidf**2))
        l2_tfidf

Out[9]: array([0.5 , 0.45, 0.5 , 0.19, 0.19, 0.19, 0.3 , 0.25, 0.19])
```

3.6 Data Preparation

In this step, all sorts of punctuations are removed from the text. Hyphens are replaced with a single space and all characters are converted to lower case. Additionally, all emojis are moved to the end of the sentence. The following function uses a regular expression to do all of these things:

```
In [11]: import re
         def preprocessor(text):
             text = re.sub('<[^>]*>', '', text)
             emoticons = re.findall('(?::|;|=)(?:-)?(?:\)|\(|D|P)', text)
             text = re.sub('[\W]+', ' ', text.lower()) +\
                 ' '.join(emoticons).replace('-', '')
             return text
```

To test this function, the following code is used:

```
In [13]: preprocessor("</a>This :) is :( a test :-)!")

Out[13]: 'this is a test :) :( :)'
```

3.7 Tokenization of documents

Tokenization is used to convert split sentences into tokens. Here, I have used NLTK stem porter which not only tokenizes the sentences but also stems the words, i.e., reduces each word to its root word. Using root words makes the Sentiment Categorization process faster. Along with this, stop words which do not help in the analysis are removed using the nltk's list of English stop words. The following code shows this process being applied on a sentence: a runner likes running and runs a lot.

```
In [15]: from nltk.stem.porter import PorterStemmer

def tokenizer_porter(text):
    return [porter.stem(word) for word in text.split()]
```

```
In [19]: from nltk.corpus import stopwords

         stop = stopwords.words('english')
         [w for w in tokenizer_porter('a runner likes running and runs a lot')[-10:]
         if w not in stop]

Out[19]: ['runner', 'like', 'run', 'run', 'lot']
```

3.8 Document classification via a logistic regression model and load saved model from the disk:

In this step, the logistic regression algorithm is applied for sentiment categorization. Some of the characteristics of logistic regression are as follows:

- Linear classification model
- Can handle sparse data
- Fast to train
- Weights can be interpreted

To use this algorithm, the data is split into a 50-50 split where in 50% of the data is used for training and 50% of the data is used to testing. As this is a large dataset and takes quite a bit of time to learn, once the model has finished learning, it is saved in the disk for future use. This is done to avoid having to run all the code done so far again. The saved model can be loaded from the disk using pickle library as and when required.

3.9 Model accuracy:

In this final step, the accuracy of the model is calculated on the test data set which is found to be 0.899. This accuracy is more than the accuracy provided by the comparison table earlier.

REFERENCES

[1] Ummara Ahmed Chauhan, Muhammad Tanvir Afzal, Abdul Shahid, Moloud Abdar, Mohammad Ehsan Basiri, Xujuan Zhou, "A comprehensive analysis of adverb types for mining user sentiments on amazon product reviews" Journal of World Wide Web 2020 1811:1829

[2] Xing Fang and Justin Zhan, "Sentiment analysis using product review data" Journal of Big Data 2015 2:5

[3] Farkhund Iqbal, Jahanzeb Maqbool Hashmi, Benjamin C.M. Fung, Rabia Batool, Asad Masood Khattak, Saiqa Aleem and Patrick C.K. Hung, "A Hybrid Framework for Sentiment Analysis Using Genetic Algorithm Based Feature Reduction" IEEE 2019

[4] Kashfia Sailunaz, Reda Alhajj, "Emotion and sentiment analysis from Twitter text" Journal of Computational Science 36 (2019) 101003

[5] Nadeem Akhtar, Nashez Zubair, Abhishek Kumar and Tameem Ahmad, "Aspect based Sentiment Oriented Summarization of Hotel Reviews" Science Direct 7th International Conference on Advances in Computing & Communications, ICACC-2017

[6] Ekaterina O., Jukka T. and Hannu K., "Conceptualizing big social data" Jounal of Big Data 2017 4:3

[7] Ruths D. and Pfeffer J., "Social media for large studies of behavior" Science. 2014 346

[8] Ramanathan V. and Meyyapan T., "Survey of text mining" International Conference on Technology and Business and Management 2013 pp. 508-514

[9] Benamara F., Cesarano C., Picariello A., Recupero DR and Subrahmanian VS, "Sentiment Analysis: adjectives and adverbs are better than adjectives alone" Proceedings of ICWSM conference 2007

[10] Yu H and Hatzivassiloglou V., "Towards answering opinion questions: separating facts from opinions and identifying the polarity of opinion sentences" Proceedings of the 2003 conference on empirical methods in natural language processing, EMNLP'03. Stroudsburg: Association for Computational Linguistics; 2003 pp. 129-36

[11] Chesley P., Vincent B., Xu L and Srihari RK, "Using verbs and adjectives to automatically classify blog sentiment" AAAI symposium on computational approaches to analyzing weblogs (AAAI-CAAW) 2006 pp. 27-9.

[12] Vasant Dhar, "Data science and prediction" Communications of the ACM. 2013 56 (12): pp. 63-73

[13] Jeff Leek, "The key word in "Data Science" is not Data, it is Science" Simply Statistics 2014

[14] Abhilasha Singh Rathor, Amit Agrawal, Preeti Dimri, "Comparative Study of Machine Learning Approaches for Amazon Reviews" Procedia Computer Science 132 (2018) pp. 1552-1561

[15] Shahan P. Ha., Bini Ommanb, "Evaluation of Features on Sentimental Analysis" ICICT (2015) pp. 1585-1592

[16] Charu Gupta, Amita Jain, Nisheeth Joshi, "A Novel Approach to Feature Hierarchy in Aspect Based Sentiment Analysis Using OWA Operator" Proceedings of ICCCN Springer (2019) pp. 661-667

[17] J. Ramteke, S. Shah, D. Godhia and A. Shaikh, "Election result prediction using Twitter sentiment analysis," 2016 International Conference on Inventive Computation Technologies (ICICT), Coimbatore, 2016, pp. 1-5.