iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 10 **Issue:** VII **Month of publication:** July 2022

DOI: https://doi.org/10.22214/ijraset.2022.45794

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue VII July 2022- Available at www.ijraset.com

An Efficient Book Reading Mechanism using Deep Learning

Anusha Kukreja[1], Ayushi Garg[2], Satya Bhanu Khichi[3]
[1, 2, 3]*School of Computer Science and Engineering (SCOPE), VIT, Vellore, Tamil Nadu, India*

Abstract: ***The importance of reading books and novels has evolved notably with the advancement of information and technology. The relationship among characters in a novel tells an important part of the story. Great novels usually have large and complex character relationship networks due to the number of characters and story complexity. However, with the fast-changing world, people do not get time to read complete books and novels. So, identifying and categorizing the important part of the books, we plan to propose an efficient mechanism based on neural networks, semantic analysis and deep learning which will enable readers to instantly get information about their favourite books.***
Keywords: ***Sentiment, characters, graph, analysis, readers, media, novel.***

I. INTRODUCTION

In this hectic schedule, it is nearly impossible to completely read and understand an entire novel. Nowadays people generally focus on the story rather than on character. Therefore, many people look for a summary of the novel which fails to give insight into the various character. For any story, the relationship of characters plays an important part. Keeping in mind this importance and the amount of free time people have in recent times, this paper introduces a model that helps one understand the relationship between characters using visualized graphs.

.

II. LITERATURE SURVEY

Technology in today's world is driven by techniques like-text mining, NLP, sentimental analysis. Researchers in past have worked on these techniques in different fields which had led to the growth of this domain of technology. The algorithms developed in conclusion from these researches were used further to innovate new technologies. Researches have been carried out in semantic analysis field of natural language processing which lead to better understanding and analysis of the fast-growing data of social media platforms [1] and [2] proposed how sentimental analysis of the enormous data of Twitter can lead to generation of sentiments such as positive, negative and neutral. They used methods like naïve Bayes, Maximum entropy and Support vector machine to apply them to real time tweets which benefited the user to check for the sentiment of the tweet. Researches are conducted on finding the opinions through the sentiment analysis of tweets. In [10] researchers have worked to analyse opinion about festivals and cultural events by automatically detecting polarity in Twitter data by latent Dirichlet analysis (LDA). Alexander et al, 2017 but the dataset used are the tweets related to demonetization in India. Thus, above research papers focused on mainly analysis of extracted data from Twitter social media.

Social media has become an integral part of our lives these days. Since emotions and tone cannot be visible on text, therefore people often use emoticons to express their emotions. Researchers like in [3] and [8], researchers have carried out their prime research in the field of sentiment analysis focusing majorly on extracting the information from emoticons using the normalisation and tokenization techniques. On the other end, in [7] and [4] researchers performed the techniques of sentimental analysis on the data of newspaper articles containing annotations like underline, bold letters etc.

Another domain of economy is tourism which became the key target for the analysis of sentiment analysis. In [5] and [6] have done sentimental analysis on data related to tourism to conclude about the popular monuments, culture, and infrastructure. Probabilistic Latent Semantic Analysis, Word Embedding, and LSTM techniques allowed users to get accurate information. As a result of this, there has been a significant growth in economy. The techniques mentioned above have been used in the field of entertainment. In recent times, sentimental analysis and Natural Language Processing Techniques are serving as a useful tool to extract music from the mythological literature. In [9] they proposed two-layer artificial neural network whose output allows us to represent entire contents of words as a series of numerical values in manageable vector space. Many studies in the domain related to text processing have led to development of highly structured algorithms which make text segmentation and extraction an easy work to do.

In [11] they focus on drawing trends and methodological evidences from previous studies in big data field considering the fact that big data helps individuals, governments and businesses understand the essence of collecting large data and processed it towards effective decisions making having identified many patterns within it.

In [12] researchers presented statistics on the evolution of sentiment analysis. He proposed that the data will be analysed in two different ways: with a statistical keyword analysis and with Latent Dirichlet Allocation. While the keyword analysis looks at the keywords only, LDA analyses the words used in the title and the abstract of the publications as well. The purpose of the LDA analysis is to see what conclusions regarding the topics could be made based on the words used in the title and abstract.

Above recent researches provided modern solutions to the real-life data handling and extraction situations. Such studies helped researchers to perform more analysis in the field of data mining and language processing. New studies are conducted to generate a better and efficient algorithm to extract not only sentiments but a lot more from the real-time generating data.

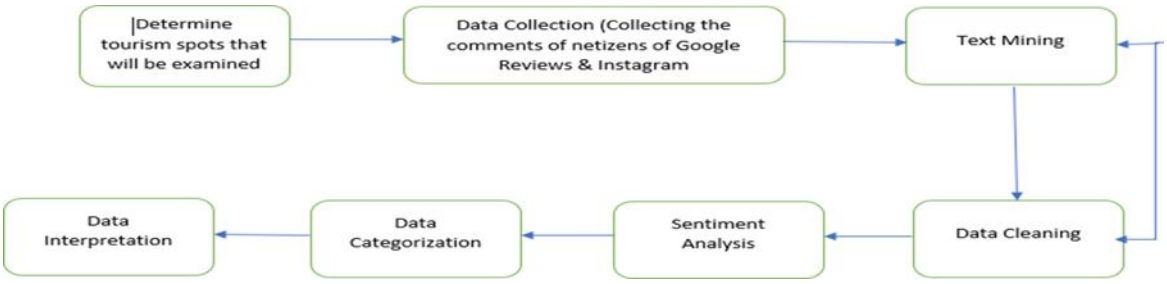


Fig 1: Steps in Sentiment Analysis

III. DETAILED EXPLANATION OF RESEARCH

The proposed project is using network graph, NLP techniques (entity recognition, sentiment analysis) to analyse the relationships among characters in a novel. The relationship among characters in a novel tells an important part of the story. The proposed project automatically analyses complex character networks through computer programs, which takes in novel text only and is able to generate a comprehensive visualized network graph as output.

The output will be a sentiment graph in which each node represents a character in the novel and each edge represents the relationship between the two characters it's connected to. The node size represents the importance of a character, which is measured by the number of occurrences of its name in the novel. In the graph each edge has a different colour, from bright (yellow) to dark (purple), which represents the sentiment relationship between two characters, or in a more human-understandable way, hostile or friendly relationship. The brighter the colour of the edge, the friendlier the relationship is; the darker the colour, the more hostile it is. As the story proceeds the graph's edges because we split the whole novel series by episode itself to generate one set of edge parameters respectively for each episode, so that we can see the relationship changes among characters as story proceeds.

There has to be a novel and common words file which contains the whole story split into sentences and 4000+ English words respectively. To figure out the characters in the novel by name entity recognition (NER) the proposed project uses the pertained Spacy NER classifier and count the occurrence of a character more accurately. From last step each character's importance is calculated, using the Scikit-Learn text processing function CountVectorizer. Next co-occurrence is calculated using a binary occurrence matrix that gives information on whether a name occurs in each sentence, again with function CountVectorizer. While the co-occurrence matrix above gives information on the co-occurrence or interaction intensity, the sentiment matrix gives information on the sentiment intimacy among characters.

IV. MODEL FRAMEWORK

The main objective of the proposed model is to generate the visualization of the characters of the novel by linking them to each other and degerming their sentiment score using their sentiment analysis. Data collection and Preparation- before Processing and analysing the novels are collected and cleaned using a common words file. These novel files contain the novel text of the whole story which is split into sentences for processing. Building the model- This is the actual training step which includes the use of learning algorithms.

Name Entity Recognition- In this step the names of the characters of the novels are extracted from the pre-processed novel files using the learning algorithm which will be used in the latter processes for co-occurrence counting and sentiment score calculation. In this step the model also aggregates the extracted data, filters it to remove the duplicate identities.

Pre-processing is the initial step where the source and suspicious documents are subjected to certain refinements like stop-word removal, tokenization, lowercasing, sentence segmentation, punctuation removal etc. This helps in reducing the size of actual data by removing the irrelevant information with respect to the approach used. The tokenization process is language dependent. Stop words are frequently occurring and insignificant words in a language that help construct sentences but do not represent any content of the documents. Articles, prepositions and conjunctions and some pronouns are natural candidates. Common stop words in English include: a, about, an, are, as, at, be, by, for, from, how, in, is, of, on, or, that, the, these, this, to, was, what, when, where, who, will, with. Such words should be removed before documents are indexed and stored. Stop words in the query are also removed before retrieval is performed.

Analysis of extracted data- Once the names of the characters are extracted, the proposed model calculates the sentiment score of a certain text calculate the sentiment score of a certain text context (a sentence, a fixed length of words, a paragraph etc.), which is the base of character sentiment relationship. Various matrix multiplication are performed to speed up the calculation of the sentiment analysis. Network Visualization- This step generates the network graph from the data obtained from the previous step. Upon visualizing the graph, the model clearly states the relationship among the characters and thus making it easy for the readers.

V. ARCHITECTURE

The flowchart below clearly depicts the different components of the proposed model. Before starting with the processing, the initial text data i.e. the novels and books are converted into their respective sentences using the readnovel() function. These extracted sentences are converted into character names, tokens by passing through the name entity recognition stage. The sentences go through several iterations to get the well-defined character names without any duplications. Sentiment analysis serves as the major component of the architecture of the proposed model. The extracted character names are parsed through the process of sentiment analysis which calculates the sentiment score. This component connects to another major component which functions to generate co-occurrence matrix and sentiment matrix. These matrices are converted into the edges of the graph by using toedge() function. The next component aggregates all the edges of all the characters and generates an overall network graph for each matrix. The above components majorly describes the architecture of the proposed model which is efficient enough to generate the real time network graphs for the books and novels thus making it easy for the readers.

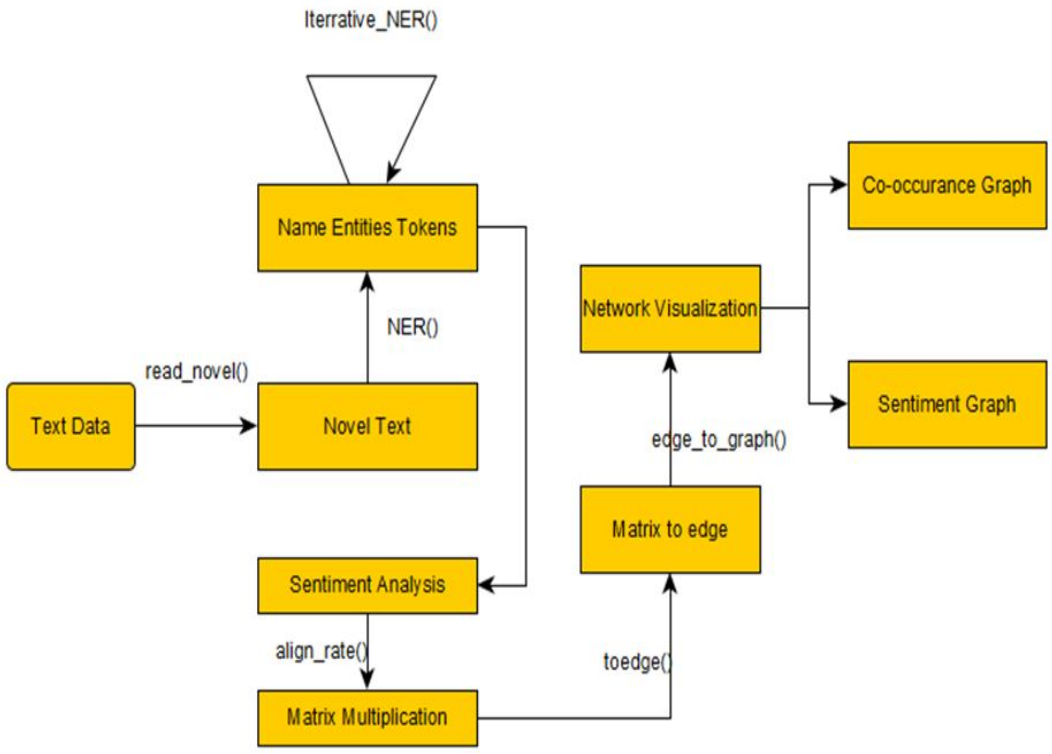


Fig 2: Flow chart of Model

VI. WORKING MODEL

The sentences are extracted from the novel. The proposed project uses python libraries to extract the characters. They further check the occurrence of characters. Also, they check how many times a character has occurred in single line. The two matrices are created – sentiment matrix and co-occurrence matrix. The sentiment matrix is used to comment on the relationship between two characters whereas the co-occurrence matrix calculates which two characters occur in a single sentence. Using python libraries, a network model is generated and the final output is the visualization of sentiment matrix and co-occurrence matrix.

VII. PROPOSED METHODOLOGY

The research conducted during the course of this project lies in the field of Data Science particularly in the domain of Natural Language Processing. Keeping in mind the disadvantages of previously proposed models, this proposed methodology proves to be efficient in terms of analysing the data set. Data plays a major role in any natural language processing research. Choosing a well-defined dataset is important to perform the proposed models and generate the required output. Well oriented and extracted data is necessary for any research process. Data analysis is highly dependent on the dataset chosen. Starting with the collection of data, after extensive search for data set from sources and various books and journals. The research makes use of text files of various novels as the date set. Before processing and analysing the novels, there arises a need to prepare the *common words-file* and the *novel* where the *novel's file* contains the text of the whole story and the *common words file* consists of majorly used 5000+ English words. The sentences of *novel's file* will be split in latter stage. *Common word's file* is used to reduce the errors during the stage of name entity recognition which removes common words that appear in the *novel file.* With no information about the novel, the model expects to figure out the novel's characters by name-entity recognition (NER). The proposed methodology makes use of *pertained* Spacy NER *classifier.* The model runs the NER process sentence by sentence instead of going through whole novel because the spacy NLP class initiation occupies high memory space. For each sentence the name-entities are identified splitting of the names into single words is done for the names that consists of multiple words. The main aim is to count the occurrence of names of characters accurately. After this filtrations of the names that are present in *common words* is performed and at last the names from every sentence are aggregated. From the primer character name list we get from last advance, we can ascertain each character's significance, or all the more explicitly, the event recurrence. This undertaking should be possible effectively with the Scikit-Learn content handling capacity CountVectorizer. From that point forward, we can choose the top characters of intrigue dependent on their significance. The top names work yields the main 20 characters and their frequencies as default, however right now model, we set the number to be 25 to catch a bigger system. In the proposed methodology, the simplest meaning of co-occurrence picked is that a co-occurrence is watched if two character names appear in a similar one sentence. (There may be some different definitions dependent on passages, number of words, a few sentences and so on.) To ascertain co-occurrence, first a binary co-occurrence matrix is required, that gives data on whether a name happens in each sentence using CountVectorizer.

$$X_{cooccur} = X_{occur}^{T} \cdot . X_{occur}$$

$$X_{sentiment} = X_{occur}^{T} . (X_{occur}^{T} X V_{sentiment})^{T} + X_{cooccur} X \theta_{align}$$

At that point, the co-occurrence matrix arises from the dot product result of occurrence matrix and its transpose. As co-occurrence is commonly intuitive, the co-occurrence matrix is rehashed (symmetric) along the corner to corner, so it is triangularize and inclining components are set to zero. While the co-occurrence matrix gives data on the co-occurrence the sentiment matrix gives data on the supposition closeness among characters. The bigger the worth, the more positive relationship (friends, lovers) between two characters, the lower the worth (could be negative), the more negative relationship (foes, rivals and so forth.) between two characters. The score of the sentiments of every sentence in the novel file that is the sentiment score showcases the relationship among two characters that simultaneously occur in the text. This score is calculated by ***affin NLP library*** and the scores are kept together in 1-D array. Various creators and various kinds of books may have various methods for accounts, which will prompt various degrees of feeling depiction. For instance, a ghastliness novel should likely have more negative feeling portrayals than a pixie tail. The diverse portrayal styles will prompt a skewness of the estimation scores on the general character organize. In extraordinary cases, the connections may be all positive or all negative.

VIII. IMPLEMENTATION

The proposed model is implemented on various novels and books to generate character network graphs for better visualization of characters. The implementation makes use of python. Various modules are defined for each process such as flattening of matrices, reading the novel text, name entity recognition and conversion of matrices to edge list.

Python is used for its wide range of libraries for data science. Spacy, Numpy, NLTK packages are imported for handling operations of reading of novel and matrices generation. Networkx is imported for generation of network of characters extracted and stored in matrices. For the results, first 3 books of Harry Porter series is used at dataset. The python code is implemented and executed on Anaconda software which provides the desired output.

IX. RESULTS

When applied to novel text, the model perfectly works and extracts the top characters from the text and stores in matrix. The tokenization process yields accurate tokens. The main character of the novel, i.e. Harry Porter is perfectly highlighted in the graph generated. Similarly, rest all the characters are also shown in the graphs below.

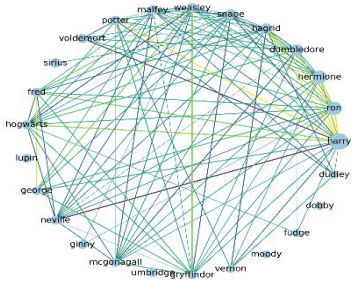


Fig 3: Harry Potter 1(Sentiment Graph)

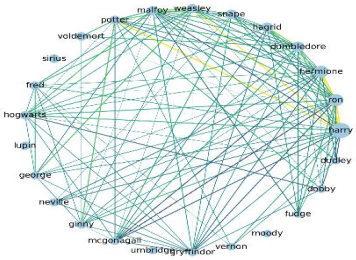


Fig 4: Harry Potter II (Sentiment Graph)

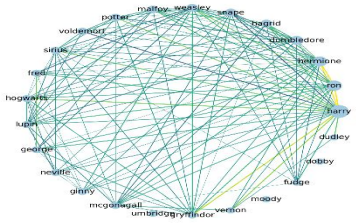


Fig 5: Harry Potter III(Sentiment Graph)

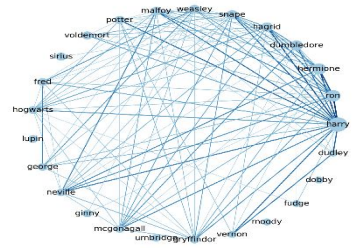


Fig 6: Harry Potter 1 (Co-occurrence Graph)

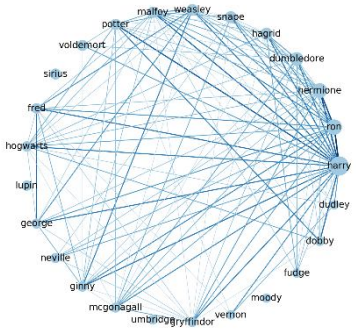


Fig 7: Harry Potter 2 (Co-occurrence Graph)

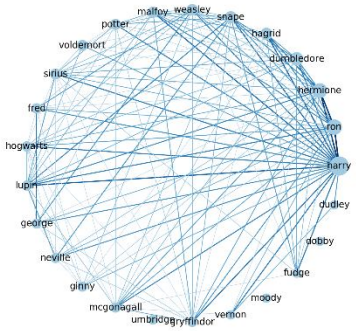


Fig 8: Harry Potter 3 (Co-occurrence Graph)

The sentiment analysis provides the result according to the relations of the characters and is shown accurately in the result obtained. The intensity of edges of the network generated correctly shows their relationship.

The results are obtained with an accuracy of 92% depicting the relationship among characters and the co-occurrence of the different characters. The error rate is around 7-8%. Hence, the results meet with the proposed outcome of the model. The model can be future tested for various novels.

X. CONCLUSIONS

Reading books and novels is the habit of many people of different age groups. The model allows them to visualize their favourite character's relationship with other characters and also helps them to visualize the co-occurrence of the characters which helps in determining the closeness of characters. The model provides the convenient and innovative way to get information about the characters. This model serves as an efficient option for the first-time readers to get acquainted with the nature of characters i.e. positive or negative with the other characters. One of the drawback of this model is that it is time consuming. For testing a complete series of novel, it takes about 25-30 minutes that accounts for the wastage of lot of time of the readers. Hence, in future, a better algorithm can be modelled which reduces the execution time of the generation of graphs. This model can be further incorporated with online applications for better visualization over internet.

REFERENCES

[1] Agarwal, A., Xie, B., Vovsha, I., Rambow, O., & Passonneau, R. (2011, June). Sentiment analysis of twitter data. In Proceedings of the Workshop on Language in Social Media (LSM 2011) (pp. 30-38).

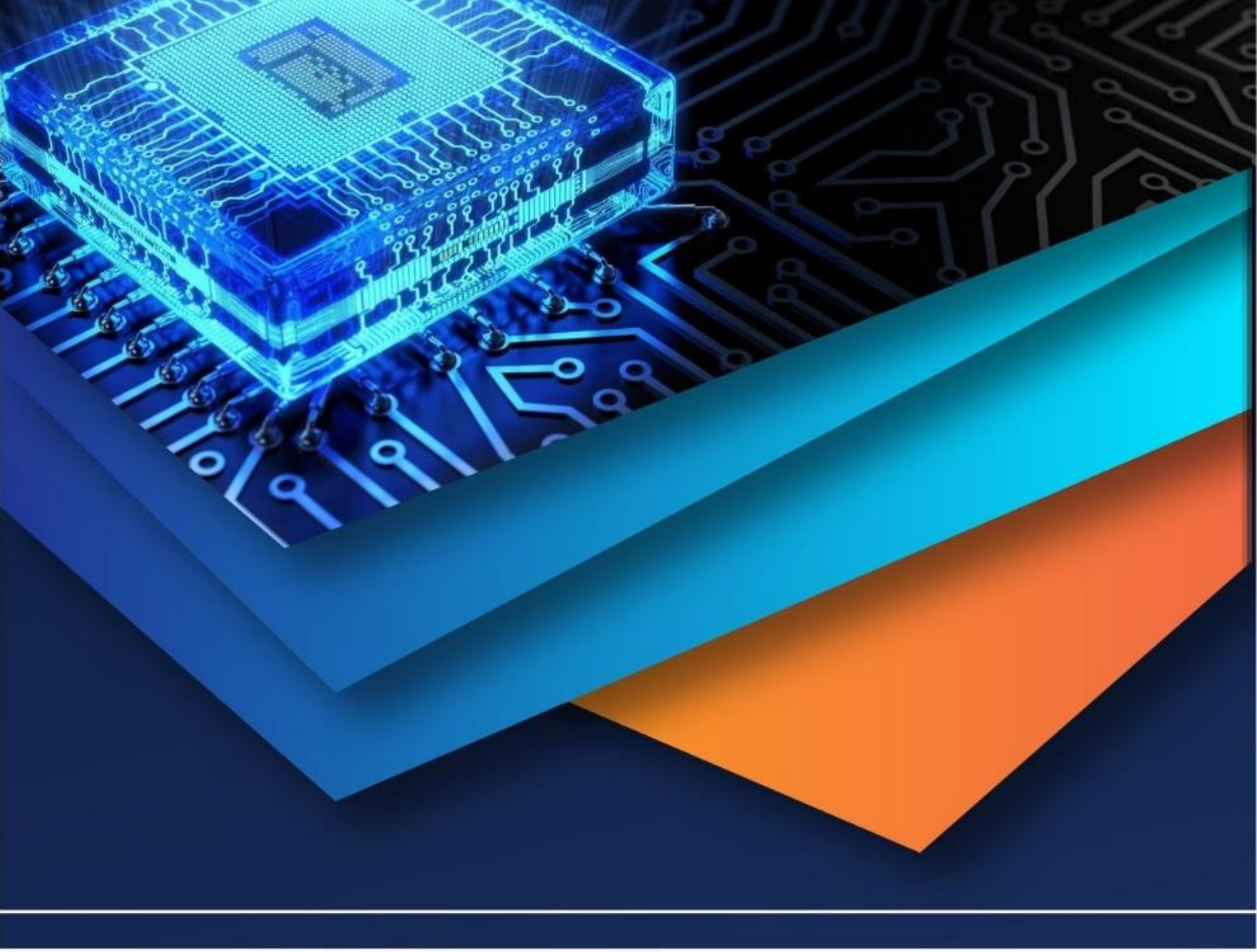
[2] Alsaeedi, A., & Khan, M. Z. A Study on Sentiment Analysis Techniques of Twitter Data.Ethem Alpaydin. Introduction to Machine Learning, second edition.2010.

[3] Wikarsa, L., & Kim, M. AUTOMATIC GENERATION OF WORD-EMOTION LEXICON FOR MULTIPLE SENTIMENT POLARITIES ON SOCIAL MEDIA TEXTS

[4] Shukla, A. (2011). Sentiment analysis of document based on annotation. arXiv preprint arXiv:1111.1648

[5] Alamanda, D. T., Ramdhani, A., Kania, I., & Hadi, E. S. (2019). Sentiment Analysis Using Text Mining of Indonesia Tourism Reviews via Social Media. In GI Social Sciences Forum.

[6] Khotimah, D. A. K., & Sarno, R. (2019). Sentiment Analysis of Hotel Aspect Using Probabilistic Latent Semantic Analysis, Word Embedding and LSTM. International Journal of Intelligent Engineering and Systems, 12(4), 275-290.
[7] Islam, M. U., Ashraf, F. B., Abir, A. I., & Mottalib, M. A. (2017, December). Polarity detection of online news articles based on sentence structure and dynamic dictionary. In 2017 20th International Conference of Computer and Information Technology (ICCIT) (pp. 1-5). IEEE.
[8] Mohammad Shabaz, & Dr. Urvashi (2018) Evaluation and Classification of Positive and Negative Words from Noisy Data, Volume 5, Issue 6, June 2018 ISSN NO: 0972-1347
[9] Chaithra Shree, U.S., Kaur, H., Kumari, M. and CV, M.M., SENTIMENT ANALYSIS AND TOPIC EXTRACTION.
[10] Bisallah, H. I., Olumide, O., & Aminat, A. Big Data Exploration in New Media: Trends and Methodological Approaches.
[11] Kumar P K & Dr S Nandagopalan. (2017) Exploring the opinions of people on demonetization tweets from different cities.
[12] Ahlgren, O. (2016, December). Research on sentiment analysis: the first decade. In 2016 IEEE 16th International Conference on Data Mining Workshops (ICDMW) (pp. 890-899). IEEE.
[13] Pak, A., & Paroubek, P. (2010, May). Twitter as a corpus for sentiment analysis and opinion mining. In LREc (Vol. 10, No. 2010, pp. 1320-1326).



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY