iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 12 **Issue:** III **Month of publication:** March 2024

DOI: https://doi.org/10.22214/ijraset.2024.59532

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 12 Issue III Mar 2024- Available at www.ijraset.com

Integrating TextBlob and VADER for Dynamic Sentiment Analysis: A GUI-Based Approach with Emotion Visualization and Confidence Assessment

Mala Das[1], Dr. Mohd. Shahnawaz Ansari[2]
[1]*Research Scholar, Dept. of computer science and engineering*
[2]*Professor, Dept. of computer science and engineering*

Abstract: ***This research presents a Graphical User Interface (GUI) application developed in Python using the Tkinter library for sentiment analysis. The GUI integrates two distinct sentiment analysis libraries, TextBlob and VADER, to provide a comprehensive analysis of text sentiment. Each library's sentiment analysis results are displayed along with confidence scores, allowing users to gauge the reliability of the analysis. Additionally, the GUI features a visual indicator in the form of a lamp that dynamically changes color based on the detected emotion (positive, negative, or neutral) inferred from the sentiment analysis results. Moreover, the GUI includes a combined sentiment score, calculated by averaging the sentiment scores obtained from both TextBlob and VADER, enhancing the overall accuracy and reliability of the sentiment analysis. This GUI provides users with an intuitive and interactive tool for analyzing text sentiment, with the ability to visualize emotions and assess sentiment confidence.***

Keywords: Open source, Python, Sentiment analysis

I. INTRODUCTION

Sentiment analysis, also known as opinion mining, is a computational technique employed to discern and quantify sentiment or opinion expressed within text data. With the exponential growth of digital content on the internet, social media platforms, and various other sources, sentiment analysis has become a crucial tool for understanding public opinion, consumer sentiment, and user feedback. By analyzing text data, sentiment analysis enables organizations to extract valuable insights, make informed decisions, and tailor products or services to meet customer needs effectively.

At its core, sentiment analysis aims to categorize text into predefined sentiment categories such as positive, negative, or neutral. This process involves employing a variety of natural language processing (NLP) techniques, machine learning algorithms, and lexicons to analyze linguistic patterns, sentiment expressions, and contextual clues within the text. Sentiment analysis models can range from rule-based systems like VADER (Valence Aware Dictionary and sEntiment Reasoner) to more sophisticated machine learning models such as deep learning-based approaches.

One of the primary challenges in sentiment analysis is handling the inherent ambiguity, context-dependency, and subjectivity of human language. Textual expressions of sentiment can vary greatly in complexity, tone, and nuance, making accurate sentiment analysis a non-trivial task. Researchers and practitioners continuously strive to develop more robust sentiment analysis techniques that can effectively capture subtle nuances, sarcasm, irony, and cultural variations present in text data.

Despite its challenges, sentiment analysis holds immense potential across various domains and applications. From marketing and brand management to customer service and product development, sentiment analysis empowers businesses to gain actionable insights from vast amounts of textual data. Furthermore, sentiment analysis finds applications in social media monitoring, political analysis, healthcare, and beyond, demonstrating its versatility and importance in understanding human sentiment and opinion in the digital age.

II. ADVANTAGES OF SENTIMENT ANALYSIS:

A. Business Intelligence and Brand Monitoring

Analyzing sentiment is crucial for businesses to grasp the perception of their brand within the market [1]. Through the examination of customer reviews, social media mentions, and other textual feedback, companies can acquire valuable insights..

B. Customer Feedback Analysis

For companies inundated with a high volume of customer feedback, sentiment analysis streamlines the extraction of significant insights [2]. This capability empowers organizations to pinpoint areas for improvement, promptly address customer concerns, and elevate overall customer satisfaction levels.

C. Market Research

Sentiment analysis is pivotal in market research as it furnishes a quantitative gauge of public opinion. Researchers leverage sentiment analysis to scrutinize sentiments expressed across forums, blogs, and social media platforms, enabling them to assess the reception of new products, track trends, and comprehend consumer preferences [3]. By harnessing this real-time data, businesses can remain agile and responsive to market dynamics.

D. Social Media Monitoring

The widespread use of social media renders it a treasure trove of opinions and discussions. Sentiment analysis tools empower businesses and individuals to monitor social media platforms for mentions of their brand, products, or pertinent topics [4]. This proactive monitoring facilitates prompt responses to customer queries or concerns and aids in the management of online reputation.Political and

E. Public Opinion Analysis

In the domain of politics and public affairs, sentiment analysis assists in assessing the public's sentiment towards policies, political figures, and societal issues [5]. This data is indispensable for policymakers, political strategists, and researchers to grasp public sentiment, pinpoint crucial concerns, and adapt communication strategies accordingly.

F. Fraud Detection and Risk Management

Sentiment analysis finds application in financial and legal sectors for fraud detection and risk management purposes [6]. Through the analysis of textual data from sources like financial news or social media, organizations can detect early indicators of potential risks, market trends, or fraudulent activities, enabling proactive measures to be taken.

G. Human Resources and Employee Feedback

Within corporate settings, sentiment analysis holds applicability in scrutinizing employee feedback, evaluations, and sentiments conveyed via internal communication platforms. This equips HR departments with valuable insights into employee contentment levels, facilitates the identification of potential concerns, and contributes to cultivating a conducive and uplifting work atmosphere.

H. Healthcare and Patient Feedback

Within the healthcare sector, sentiment analysis is utilized for assessing patient feedback sourced from surveys, reviews, and social media platforms. This feedback serves as vital information for healthcare providers, aiding them in comprehending patient experiences, refining service quality, and advancing overall healthcare provision.

III. PROPOSED MODEL

In this research work we have made a GUI using the python for doing the sentiment analysis. In this work, we have used two different python libraries names a text blob and vader to do the sentiment analysis. A user can input any sentence in the label given in the GUI and can get the emotion of the sentence via both the libraries. The confidence score from both the libraries are shown over the GUI. The GUI also shows the combined confidence score. The GUI is equipped with a dashboard lamp. This dashboard lamp changes color based on the emotion. In this section first the sentiment analysis using python is discussed and the details of both the textblob and vader libraries is discussed.

A. Sentiment Analysis using Python

The application of natural language processing (NLP) techniques for sentiment analysis using Python has become increasingly prevalent, thanks to the flexibility and strength of Python's programming environment. Specialized libraries such as NLTK (Natural Language Toolkit), spaCy, and TextBlob enable efficient execution of sentiment analysis tasks, catering to both beginners and experienced developers.

In Python, the process typically kicks off with text preprocessing, encompassing the cleaning, tokenization, and normalization of raw textual data to ensure uniformity. Following this, the chosen sentiment analysis library is employed to assess the sentiment conveyed in the text. These libraries often utilize pre-trained machine learning models or lexicon-based approaches to categorize sentiment as positive, negative, or neutral. One widely used sentiment analysis library in Python is TextBlob, which provides a user-friendly API for text processing and sentiment extraction. Additionally, spaCy offers a comprehensive NLP framework inclusive of sentiment analysis functionalities. The NLTK library is also popular due to its extensive toolkit covering various linguistic resources, making it adaptable to diverse sentiment analysis tasks. Python's versatility extends beyond sentiment analysis algorithms to encompass the development of sophisticated sentiment analysis applications. Integrating sentiment analysis into web applications, chatbots, or social media analytics tools can be accomplished using Python frameworks such as Flask or Django. Moreover, visualization libraries like Matplotlib or Seaborn can be utilized to generate insightful graphical representations of sentiment analysis outcomes. The expansive Python ecosystem, coupled with its readability and simplicity, makes it an optimal choice for sentiment analysis projects. Whether utilized for business intelligence, customer feedback analysis, or social media monitoring, Python's plethora of libraries and tools empowers developers and data scientists to extract valuable insights from the extensive landscape of textual data, thereby enhancing decision-making processes across various domains.

B. Sentiment analysis using textblob

Sentiment analysis, powered by TextBlob, offers a versatile and accessible solution for extracting sentiment from textual data. One of its key advantages lies in its simplicity and ease of use, making it an ideal choice for both beginners and experienced developers. TextBlob provides a straightforward API that simplifies the process of text processing and sentiment extraction, enabling users to quickly implement sentiment analysis tasks without the need for extensive coding knowledge. Additionally, TextBlob incorporates pre-trained models for sentiment analysis, which enhances efficiency and accuracy in identifying sentiments as positive, negative, or neutral. Furthermore, TextBlob's integration with Python's rich ecosystem of libraries and tools facilitates seamless integration into various projects and applications, ranging from social media monitoring to customer feedback analysis. Overall, TextBlob streamlines the sentiment analysis process, offering an accessible and effective solution for deriving valuable insights from textual data.

C. Sentiment Analysis using Vader

Sentiment analysis using VADER (Valence Aware Dictionary and sEntiment Reasoner) is a powerful tool in natural language processing that excels in analyzing the sentiment of text, particularly in social media contexts. VADER is pre-trained on a lexicon specifically designed to capture sentiment from social media posts, making it adept at handling informal language, slang, and emoticons commonly found in online conversations. One of the key advantages of VADER is its speed and simplicity of use, making it accessible even to those without extensive programming knowledge. Additionally, VADER provides not only polarity scores (indicating whether the sentiment is positive, negative, or neutral) but also intensity scores, allowing for a nuanced understanding of the strength of sentiment expressed in the text. Its ability to accurately handle both polarity and intensity of sentiment makes VADER a valuable tool for various applications, including social media monitoring, customer feedback analysis, and brand reputation management.

IV. BLOCK DIAGRAM OF THE PROPOSED MODEL

Given below is the block diagram of the proposed model

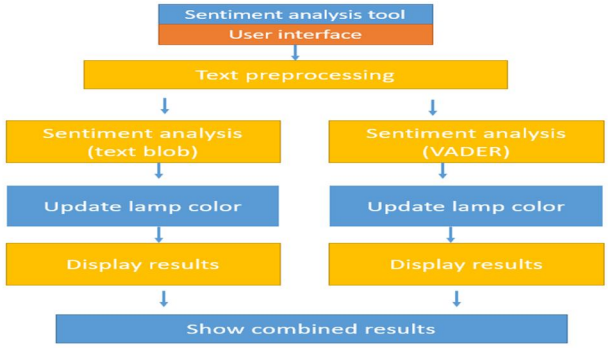


Figure 1 Block diagram of the proposed model

A. Explanation of Components

1) *User Interface:* Provides a graphical interface for user interaction.
2) *Text Preprocessing:* Cleans the input text by removing unnecessary characters or whitespace.
3) *Lamp Color:* Represents the sentiment polarity of the analyzed text (green for positive, red for negative, yellow for neutral).
4) *Sentiment Label & Confidence:* Displays the sentiment label (positive, negative, or neutral) along with confidence scores for each sentiment analysis method.
5) *Compound Score:* Represents the overall sentiment polarity of the text obtained from VADER.
6) *Combined Score:* Represents the average sentiment score obtained by combining scores from TextBlob and VADER.
7) *User Interaction:* Enables users to input text, trigger sentiment analysis, and view results in real-time.
8) *Error Handling:* Handles cases where no text is entered for analysis by displaying an error message.

V. ALGORITHM USED IN THE PROPOSED MODEL

Given below is the outline of the algorithm used in the sentiment analysis tool:

A. Input Text

The user inputs text into the provided text entry field in the GUI.

B. Text Preprocessing

The input text is retrieved from the text entry field and trimmed of leading and trailing spaces.

C. Sentiment Analysis with TextBlob

The *analyze_sentiment_textblob()* function is called when the user clicks the "Analyze Sentiment (TextBlob)" button.
The TextBlob library is used to perform sentiment analysis on the input text.
TextBlob's *sentiment.polarity* attribute is used to determine the sentiment polarity of the text.
The *get_sentiment_label()* function is called to convert the polarity score into a sentiment label (positive, negative, or neutral) and confidence score.
The sentiment label and confidence score are displayed in the GUI using the *result_label_textblob label*.
The color of the lamp graphic is updated based on the sentiment polarity using the *update_lamp_color()* function.

D. Sentiment Analysis with VADER

The *analyze_sentiment_vader()* function is called when the user clicks the "Analyze Sentiment (VADER)" button.
The VADER (Valence Aware Dictionary and sEntiment Reasoner) library is used to perform sentiment analysis on the input text.
VADER's *SentimentIntensityAnalyzer()* is used to obtain the sentiment scores.
The compound score from VADER is retrieved and passed to the *get_sentiment_label()* function to determine the sentiment label and confidence score.
The sentiment label and confidence score are displayed in the GUI using the *result_label_vader* label.
The color of the lamp graphic is updated based on the compound score using the *update_lamp_color()* function.

E. Combined Score Calculation

The *update_combined_score()* function is called after either sentiment analysis task is performed.
TextBlob and VADER are both used to obtain sentiment scores for the input text.
The average of the sentiment scores from TextBlob and VADER is calculated as the combined score.
The combined score is displayed in the GUI using the combined_label label.

F. User Feedback

If no text is entered when the user attempts to perform sentiment analysis, an error message is displayed using the *messagebox.showinfo()* function.

G. GUI Interaction

The GUI is created using Tkinter, with labels, buttons, text entry fields, and a canvas for the lamp graphic.

User interactions with the GUI elements trigger the corresponding functions to perform sentiment analysis and update the display accordingly. This algorithm provides a step-by-step process for analyzing sentiment using both TextBlob and VADER libraries and updating the GUI interface with the results

VI.EXPERIMENTAL RESULTS

A. User Interface of the Developed GUI

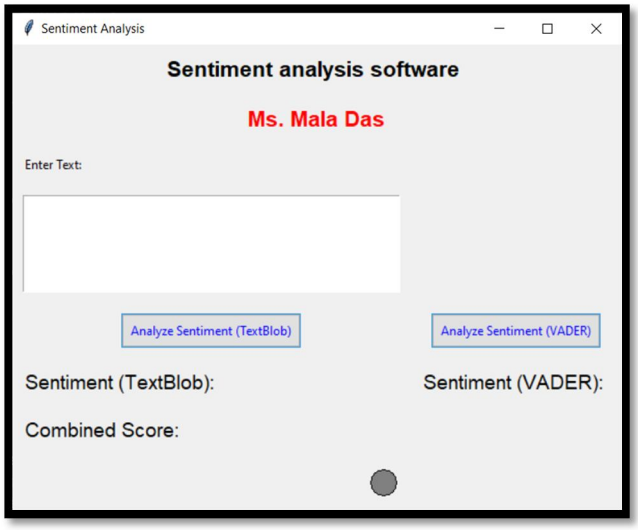


Figure 2 User interface of developed code

As displayed on the GUI is an entry field where users can input the sentiment they wish to analyze. Upon clicking the "Analyze Document" button, the computation process is initiated, delivering the result. This outcome is reflected by a lamp graphic, which dynamically alters its color in accordance with the analyzed sentiment.

1) Case study 1; Analysing Positive Sentiments

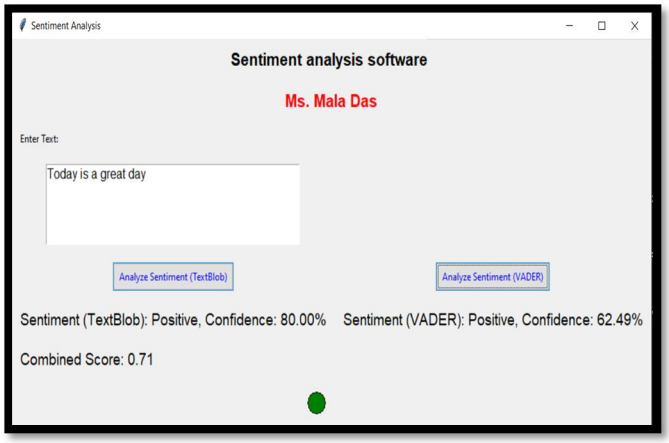


Figure 3 Software analysing positive sentiments

As it can be seen in the above gui that the software is resulting a positive sentiment and also the color of the lamp turns green. The input sentence was ‘today is a great day’, the confidence score from the textblob, vader and combined is resulted out to be 80, 62.49 and 71%.

2) Case study 2. Analyzing negative comments

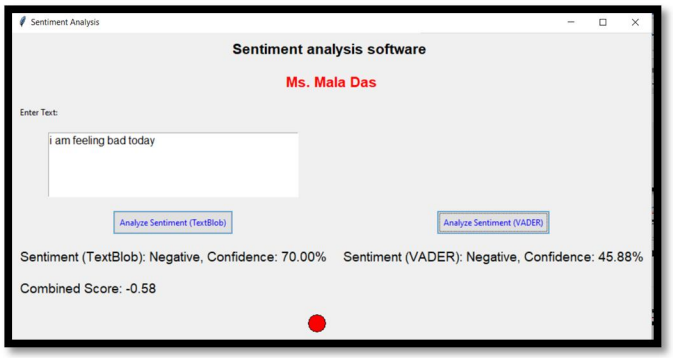


Figure 4 Software analysing negative sentiments

We can see that the software is recognizing the negative sentiments also and lamp color now changes to red. Here the inputted sentence was 'I am feeling bad today'. The confidence score from the textblob, vader and combination is found out to be 70, 45.88 and 58% respectively

3) Case Study 3: Analyzing the software with a neutral sentiment.

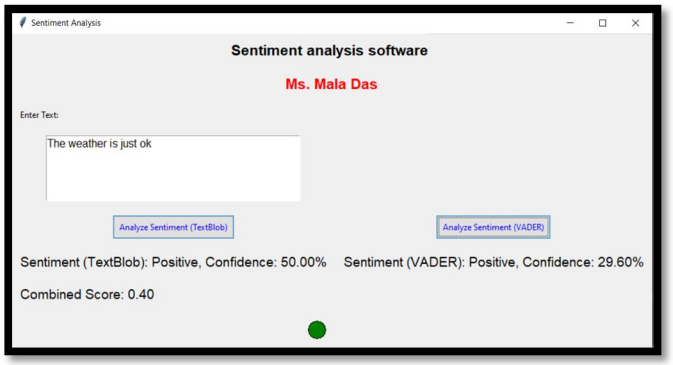


Figure 5 Software analysing neutral sentiments

As shown above the software is recognizing the neutral comment also and the lamp color now turn to yellow. Here the input sentence was 'The weather is just okay'. The confidence score for textblob, vader and combination is found out to be 50, 29.60 and 40%.

B. Given Below is the Summarized Result

Input sentence	Confidence Score in % (textblob)	Confidence Score in % (vader)	Confidence Score in % (combination)	Sentiment analysed
Today is a great day	80	62.49	71	Positive
I am feeling bad today	70	45.88	58	Negative
The weather is just ok	50	29.60	40	Neutral

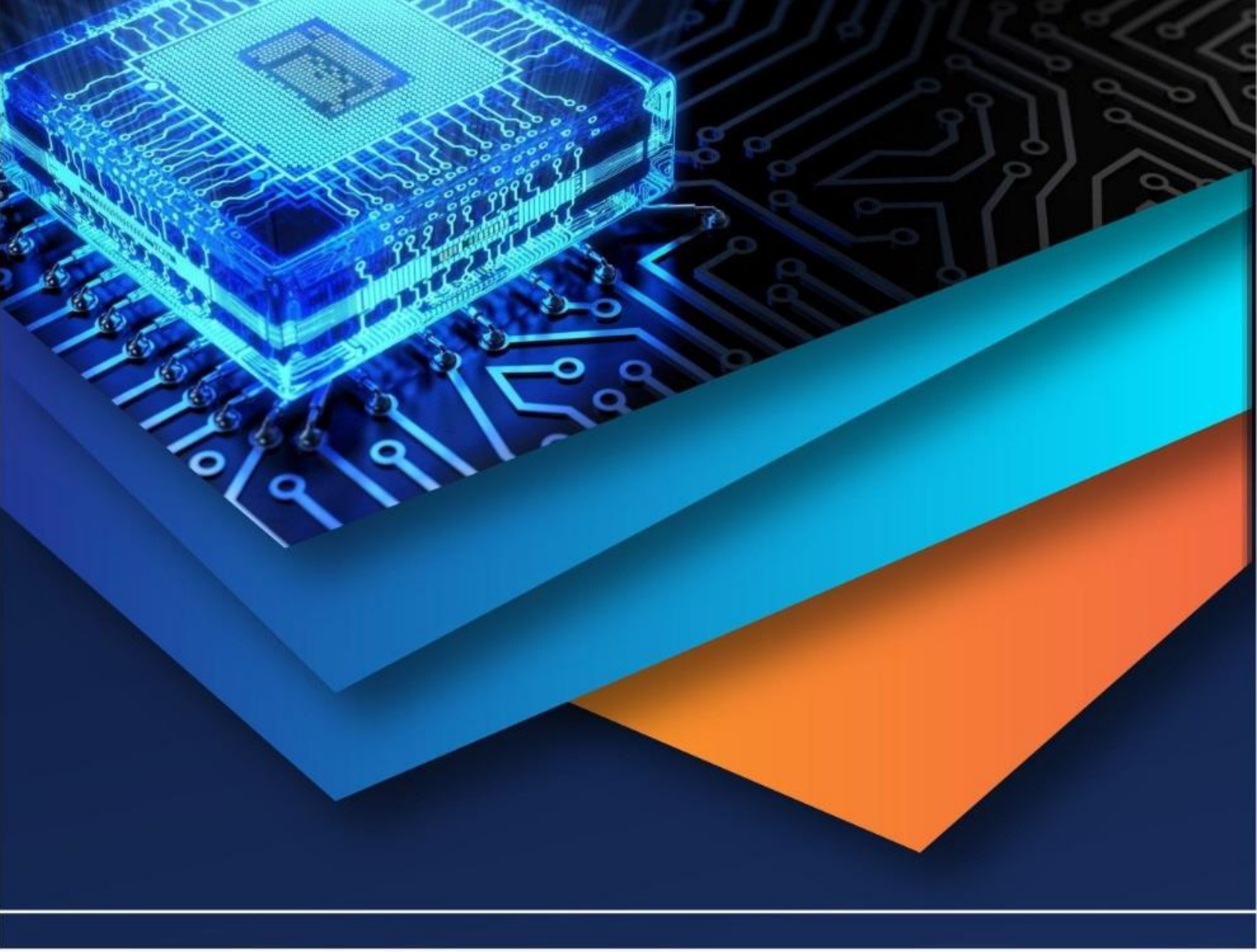
Figure 6 Comparative Analysis

VII. CONCLUSION AND FUTURE SCOPE

This research introduces a Graphical User Interface (GUI) application developed using Python's Tkinter library for sentiment analysis. The GUI seamlessly integrates two prominent sentiment analysis libraries, TextBlob and VADER, to offer users a comprehensive understanding of text sentiment. By presenting sentiment analysis results alongside confidence scores, users can better evaluate the reliability of the analysis. The inclusion of a visual indicator in the form of a dynamically changing lamp enhances user interaction, allowing for quick interpretation of emotions inferred from the sentiment analysis. Furthermore, the GUI incorporates a combined sentiment score derived from averaging the outputs of TextBlob and VADER, thereby improving the overall accuracy and robustness of sentiment analysis. Overall, this GUI serves as an intuitive and interactive tool for analyzing text sentiment, enabling users to visualize emotions and assess sentiment confidence with ease. Looking forward, there are several promising directions for expanding the functionality of the sentiment analysis GUI application. One potential avenue involves integrating advanced sentiment analysis techniques to improve the accuracy of results. Additionally, extending support for sentiment analysis in multiple languages could broaden the user base. Real-time sentiment monitoring is another area for future development, allowing users to track sentiment trends as they unfold. Customization options and integration with external applications could further enhance usability and flexibility. Incorporating mechanisms for user feedback and providing educational resources within the interface would contribute to continuous improvement and user understanding. By exploring these future scope areas, the sentiment analysis GUI application can evolve into a more versatile tool for analyzing text sentiment across various domains.

REFERENCES

[1] M. M. Altawaier& S. Tiun “Comparison of Machine Learning Approaches on Arabic Twitter Sentiment Analysis”. International Journal on Advanced Science, Engineering and Information Technology, Vol. 6 (2016) No. 6, pages: 1067-1073.
[2] K. Dave, S. Lawrence and D. M. Pennock. "Mining the peanut gallery: Opinion extraction and semantic classification of product reviews". In Proceedings of the 12th international conference on World Wide Web, pp. 519-528.
[3] A. Balahur, R. Mihalcea and A. Montoyo "Computational approaches to subjectivity and sentiment analysis": Present and envisaged methods and applications. Computer Speech & Language, 28(1): 1-6, 2014.
[4] E. Boiy and M. F. Moens "A machine learning approach to sentiment analysis in multilingual Web texts". Information retrieval, 12(5): 526- 558, 2009.
[5] M. Ghiassi, J. Skinner and D. Zimbra. "Twitter brand sentiment analysis: A hybrid system using n-gram analysis and dynamic artificial neural network". Expert Systems with applications, 40(16): 6266-6282, 2013.
[6] B. Liu, M. Hu and J. Cheng. "Opinion observer: analyzing and comparing opinions on the web". In Proceedings of the 14th international conference on World Wide Web, pp. 342-351.
[7] J. Brooke, M. Tofiloski and M. Taboada "Cross-Linguistic Sentiment Analysis: From English to Spanish". In RANLP, pp. 50-54.
[8] P. Arora, A. Bakliwal and V. Varma "Hindi subjective lexicon generation using WordNet graph traversal". International Journal of Computational Linguistics and Applications, 3(1): 25-39, 2012.
[9] R. Feldman."Techniques and applications for sentiment analysis". Communications of the ACM, 56(4): 82-89, 2013.
[10] M. Taboada, J. Brooke, M. Tofiloski, K. Voll and M. StedeLexiconbased methods for sentiment analysis. Computational linguistics, 37(2): 267-307, 2011.
[11] R. Xia, C. Zong and S. Li. "Ensemble of feature sets and classification algorithms for sentiment classification". Information Sciences, 181(6): 1138-1152, 2011.
[12] J. Kamps, M. Marx, R. J. Mokken and M. d. Rijke. "Using WordNet to measure semantic orientations of adjectives". 2004.
[13] ZulfadzliDrus, HaliyanaKhalid ,The Fifth Information Systems International Conference 2019 Sentiment Analysis in Social Media and Its Application: Systematic Literature Review . Elseveir
[14] Zhou Gui, “Research on Sentiment Analysis Model of Short Text Based on Deep Learning”, Scientific ProgrammingHindawi 2022.
[15] Liaqat MI, Awais Hassan M, Shoaib M, Khurshid SK, Shamseldin MA. Sentiment analysis techniques, challenges, and opportunities: Urdu language-based analytical study. PeerJComput Sci. 2022 Aug 31;8:e1032. doi: 10.7717/peerj-cs.1032. PMID: 36091980; PMCID: PMC9454799.
[16] Y. Chen and Z. Zhang, "Research on text sentiment analysis based on CNNs and SVM," 2018 13th IEEE Conference on Industrial Electronics and Applications (ICIEA), Wuhan, China, 2018, pp. 2731-2734, doi: 10.1109/ICIEA.2018.8398173.
[17] Altrabsheh N. Sentiment analysis towards a tool for analysis real time students feedback. In: IEEE 26th International Conference on tools with artifcial intelligence IEEE, 2014. SN Computer Science (2022) 3:469 Page 9 of 9 469 SN Computer Science
[18] Yadav SK. Multimodal sentiment analysis: sentiment analysis using audiovisual format. In: 2nd International Conference on Computing for Sustainable Global Development. (INDIACom) 2015.
[19] Kandhro IA. Student feedback sentiment analysis model using various Machine Learning schemes: a review. Indian J Sci Technol. 2019; 12(14).
[20] El-Sayad A, Ewis A, Abdel Rauof MM, Ghoneim O. A new approach in identifying the psychological impact of COVID-19 on university Students’ academic performance. Alexandria Eng J. 2022. https://doi.org/10.1016/j.aej.2021.10.046.
[21] Tarik A, Aissa H, Yousef F. Artifcial Intelligence and Machine Learning to predict Student performance during COVID-19. In: The 3rd International workshop on Big Data and Business Intelligennce(BDBI 2021) March 23–26, 2021



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY