OPEN ACCESS INTERNATIONAL JOURNAL OF SCIENCE & ENGINEERING

Sentiment Analysis of Political Leaders Speeches Using AI and NLP with Machine Learning, Deep Learning, and Transformer Models

Malayaj Kumar[1], Dr. Anuj Kumar Singh[2], Dr. Soumitra Das[3]

Research Scholar, Department of CSE, Shri Jagdishprasad Jhabarmal Tibrewala University, Rajasthan [1]
Research guide, Shri Jagdishprasad Jhabarmal Tibrewala University, Rajasthan[2]
Research Co-guide, Shri Jagdishprasad Jhabarmal Tibrewala University, Rajasthan[3]

Abstract: In order to adequately analyze the emotional and rhetorical content of political speech, which is becoming increasingly important in the process of forming public opinion, sophisticated methods are required. By utilizing artificial intelligence (AI) and natural language processing (NLP), this research investigates the method of analyzing the sentiments expressed by political leaders through their speeches. We construct and analyze classical machine learning models, deep learning approaches, and transformer-based architectures in order to categorize the attitudes that are embedding themselves in political statements. We have assembled, preprocessed, and analyzed a varied dataset consisting of speeches delivered by political personalities from throughout the world. The findings indicate that transformer models perform better than other models when it comes to collecting contextual sentiment, which leads to the provision of useful insights into political communication. The research makes a contribution to the field of political natural language processing by comparing the efficacy of several AI techniques.

Keywords: NLP, Deep Learning, Machine Learning, BERT, CNN, LSTM, Random Forests, SVM

I.INTRODUCTION

It is common practice for leaders to utilize political speeches as a means of informing, convincing, and influencing voters. Political speeches are an essential component of public communication [1, 25, 26, 27]. For the purpose of conveying the speaker's intentions and ideological position, these speeches include a variety of rhetorical methods, emotional expressions, and strategic language choices. Through the examination of the sentiments expressed in such speeches, one may get insights on political goals, leadership styles, and the responses of the audience.

Speech transcripts and video recordings are now more readily available than ever before as a result of the fast digitalization of political material and the expansion of Internet archives [2, 3, 4]. As a result, there has been a surge in interest in computer approaches for assessing political writings, notably via the use of sentiment analysis. Artificial intelligence (AI) and natural language processing (NLP) technologies provide scalable and effective methods for mining this data for emotional and thematic patterns.

In recent years, sentiment analysis has gone beyond binary or ternary classifications (positive, negative, neutral) to integrate sophisticated emotional taxonomies and contextual sentiment inference [5, 6, 7, 8]. This expansion has occurred both in the United States and internationally. Deep learning and transformers have enabled natural language processing models to be able to detect subtle variations in mood and rhetorical subtleties in long-form texts such as speeches. There is potential for these talents to provide useful tools for political analysis.

The purpose of this study is to examine the use of AI-driven natural language processing methods to the analysis of sentiments on political leaders' speeches. We want to assess the effectiveness of machine learning, deep learning, and transformer models by putting them into practice. Our goal is to get an understanding of the benefits and limits that each of these models have in this particular area.

II. LITERATURE REVIEW

There was a heavy reliance on lexicon-based approaches in the early research on sentiment analysis. These methods included the use of preset dictionaries to give sentiment scores to words or phrases [1]. These techniques, although helpful for straightforward texts, sometimes failed to take into consideration context, irony, or terminology that is specialized to a certain topic. The trend toward machine learning resulted in the introduction of statistical classifiers such as Naive Bayes and SVMs, which enhanced the ability to predict emotion by using word frequency-based features [2, 3].

Through the development of neural networks that are able to represent long-range relationships, deep learning has brought about a significant change in the field of sentiment analysis. Movie reviews, tweets, and news articles were all the subjects of successful applications of models like as LSTM and CNN [4, 5]. There was a significant improvement in performance between these models and traditional methods, particularly when it came to dealing with texts that were loud and unstructured.

The implementation of transformer architectures, such as BERT and RoBERTa, has been the most recent advancement in the field of sentiment analysis [6, 7, 39, 40, 41]. These models make use of self-attention processes in order to capture contextual linkages in text, which opens the door to the interpretation of complex sentiments. Because they have been pre-trained on huge corpora and then fine-tuned on task-specific datasets, they provide state-of-the-art accuracy for a variety of natural language processing tasks, including the categorization of political texts [11], [12].

The application of these methods to political discourse has been the subject of a number of recent publications. As an example, Hosseinia and Kermani [8] used LSTM-based models for the purpose of conducting sentiment analysis on talks and discovered that deep learning was particularly good in identifying voice tone. Rietzler et al. [11, 42, 43, 44] conducted a study on the implementation of transformers in political natural language processing (NLP), showing the efficacy of these tools in deciphering political intent and persuasion methods within the field.

Table 1. Summary of Key Literature in Political Sentiment Analysis

Author(s)	Methodology Used	Advantages / Opportunities	Disadvantages / Challenges
Liu (2012) [1]	Lexicon-based	Simple, easy to implement	Poor context handling
Pang et al. (2002) [2]	Naive Bayes, SVM	Good for structured data	Feature engineering required
Kim (2014) [4]	CNN	Captures local features	Ignores long-range dependencies
Devlin et al. (2019) [6]	BERT	Context-aware, high accuracy	Resource-intensive training
Rietzler et al. (2024) [11]	Survey of transformers in political NLP	Insights into model application	Limited comparative evaluation

III. METHODOLOGY

1. Data Collection: We gathered a multilingual collection of political speeches from publicly accessible sources, such as the media, public archives, and official government websites. The collection covers a wide range of political situations and historical periods and contains more than 2,000 speeches by prominent figures.

2. Preprocessing: The preparation of the text included the use of typical natural language processing (NLP) processes such as sentence segmentation, tokenization, lowercasing, stop-word removal, lemmatization, and part-of-speech tagging. Among the additional measures that were taken were named entity recognition (NER), which was used to recognize notable individuals and organizations, and coreference resolution, which was used to monitor pronoun references.

3. Sentiment Annotation: A combination of automatic tagging with VADER and TextBlob and manual annotation by political science specialists produced sentiment labels (positive, negative, and neutral). There was a majority-vote system in place for disagreements. Training sets accounted for 70% of the annotated dataset, whereas test sets made for 15%.

4; Model Implementation: The three primary model categories used in the implementation were transformer, deep learning, and machine learning models.

5. Machine Learning Models:

Naive Bayes

Straightforward yet incredibly powerful probabilistic machine learning algorithm, the Naive Bayes classifier excels at classification tasks, particularly those involving text analysis like document categorization, sentiment analysis, and spam detection. Operating on the "naive" premise that all features are independent of one another given the class label, it is based on the Bayes Theorem, which determines the probability of a class given a set of features [46-.50]. This assumption makes the calculation easier since it enables the model to multiply the probabilities of each individual characteristic to determine the likelihood of a class. Even though the independence assumption is implausible in many real-world situations, the Naive Bayes classifier frequently achieves impressive results, especially in high-dimensional domains like natural language processing.

The Naive Bayes algorithm comes in a number of variations, such as Bernoulli Naive Bayes, which handles binary features like a word's presence or absence in a document; Multinomial Naive Bayes, which is frequently used in text classification tasks where features represent word frequencies; and Gaussian Naive Bayes, which is used for continuous data assuming a normal distribution. Large datasets benefit greatly from the algorithm's speed and efficiency, which are well-known, and its ease of implementation. It does have certain restrictions, though. Without the use of methods like smoothing, the algorithm may have trouble handling coupled features or uncommon occurrences, and the fundamental premise of feature independence is rarely true in practice.

Naive Bayes may be used to categorize text into groups like positive, negative, or neutral sentiment when sentiment analysis of speeches by political leaders is being conducted. Every word in a speech serves as a feature, and the model determines the speech's overall emotion by calculating the likelihood that those words will appear in various sentiment classes. Naive Bayes is a simple algorithm in comparison to more sophisticated machine learning or deep learning methods, but because of its speed and interpretability, it frequently acts as a solid foundation in many NLP applications.

Support Vector Machines (SVM)

A popular supervised machine learning approach for classification and regression problems, particularly in high-dimensional domains, is the Support Vector Machine (SVM). Finding the appropriate hyperplane to divide data points into distinct classes is the basic concept of support vector machines (SVM). The margin—the distance between the hyperplane and the closest data points from each class, or support vectors—is maximized by selecting this hyperplane. SVM seeks to increase the model's capacity to generalize on unknown data by optimizing this margin.

In situations like text classification issues, where there are more dimensions than samples, SVM performs very well. It performs well with data that is both linearly and non-linearly separable. SVM employs a method known as the kernel trick for non-linear instances, which converts the input space into a higher-dimensional space where a linear separator may be located. SVM can handle complicated decision boundaries thanks to common kernel functions including polynomial, radial basis function (RBF), and sigmoid kernels.

SVM's capacity to generate reliable models with high accuracy, particularly in binary classification tasks, is one of its main advantages. Especially with the right kernel and regularization value, it is less likely to overfit. SVM may be computationally demanding, though, particularly when dealing with big datasets, and performance can be greatly impacted by selecting the appropriate kernel and hyperparameters.

SVM may be used to categorize the sentiment expressed in text data by examining linguistic characteristics and word patterns in the context of sentiment analysis of political leaders' speeches. SVM is a useful tool in natural language processing jobs because of its efficiency in handling high-dimensional textual input, which frequently leads in strong findings when it comes to differentiating between positive, negative, and neutral attitudes.Random Forests: A group of decision trees that employ majority voting to enhance classification performance. Although it might be computationally demanding, it delivers great accuracy.

Random Forests

As a result of its high accuracy, resilience, and adaptability, Random Forest is an ensemble learning technique that is commonly used for classification and regression applications. During training, it builds a large number of decision trees and outputs the mean prediction (for regression) or the mode of the classes (for classification) from each individual tree. Combining the predictions of several decision trees, each of which was trained on a distinct subset of the data and features, is the fundamental concept underlying Random Forest. This is done with the intention of minimizing overfitting and enhancing generalization skills. Introducing randomness into the model through a technique known as bagging (bootstrap aggregating) makes the model more resistant to noise and reduces the likelihood that it would overfit the data.

A randomly chosen sample from the training dataset is used to train each tree in a Random Forest, with replacement. Only a random subset of features is taken into account at each tree split. This unpredictability guarantees decorrelated trees, which is necessary for the ensemble to outperform any one decision tree. Utilizing the wisdom of the crowd effect to obtain superior predictive performance, the final prediction is based on majority voting in classification tasks or average in regression tasks. This utilization of the crowd's knowledge allows for higher predictive performance.

When working with datasets that contain a combination of numerical and categorical information, as well as when the relationships between the features are complicated and non-linear, Random Forest is an especially useful method to use. In addition to this, it offers helpful features such as feature importance ratings, which assist in determining which variables possess the most influence on a prediction job. yet, Random Forests are often more accurate than single decision trees; yet, they can be computationally costly and less interpretable, particularly when the number of trees is considerable. This is especially true when many trees are involved.

Random Forest may be utilized in the context of sentiment analysis of speeches given by political leaders. This technique involves the classification of text into different sentiment categories by the examination of characteristics collected from speech transcripts. These features may include word frequencies, syntactic structures, or linguistic clues. Random Forest is extremely useful for text classification problems, particularly those in which typical linear models may not be enough. This is because Random Forest is able to handle high-dimensional data and capture complicated relationships between features. It is a good option for applications in political speech analysis and natural language processing because of its ensemble method, which offers accuracy and stability.

6. Deep Learning Models:

Notwithstanding their initial development for image processing, convolutional neural networks (CNNs) have demonstrated efficacy in a range of natural language processing (NLP) applications, such as text categorization and sentiment analysis. CNNs work by applying convolutional filters to word embeddings in text, identifying local patterns like sentiment-influencing key phrases or word pairings. Regardless of their location, these filters may identify significant n-gram characteristics by sliding over the input phrase. Subsequently, the gathered features undergo pooling layers in order to minimize dimensionality and preserve the most pertinent data. CNNs are effective and quick for tasks like sentiment identification in political speeches where local patterns (like "great success" or "deep concern") have substantial value. They are also helpful for detecting spatial hierarchies and fixed-size correlations in text data.

The vanishing gradient problem, which prevents normal RNNs from learning long-term dependencies in sequences, is one of the drawbacks of Long Short-Term Memory (LSTM) networks. Memory cells and gating mechanisms (input, forget, and output gates) that regulate information flow and enable the model to retain or forget information over time are two ways that LSTMs do this. Because LSTMs can simulate the sequential flow of language and comprehend how a word's meaning is impacted by the context that preceding words give, they are very useful in sentiment analysis. Because of this, they are ideal for examining political speeches, since the feeling frequently develops across a number of phrases or sentences.

By processing input sequences in both forward and backward directions, bidirectional LSTMs (Bi-LSTMs) improve upon the capabilities of conventional LSTMs. This gives the model a more thorough grasp of each word's function in the phrase by allowing it to take into account both past (preceding words) and future (succeeding words) context when generating predictions. Bi-LSTMs are very effective in sentiment analysis because they can catch context from both sides, which is sometimes essential for fully grasping the meaning of complicated statements. For example, Bi-LSTMs are better able to understand the polarity and subtlety of political speeches where the attitude may change within a phrase (e.g., "While the economy has improved, unemployment remains a concern"). This two-way method greatly increases accuracy on tests requiring contextual language comprehension.

7. Transformer Models:

In the realm of natural language processing (NLP), the revolutionary transformer-based language model known as BERT (Bidirectional Encoder Representations from Transformers) has been created by Google. This model has enabled tremendous advancements in the field. BERT scans complete sequences of words in a bidirectional fashion, which enables it to capture the full context of a word based on all of the words that surround it in a phrase. This compares to earlier models that processed text in a unidirectional manner. BERT is able to do very well in tasks that require complex language comprehension, such as sentiment analysis, question answering, and named entity identification, because to its profound and bidirectional grasp for language. BERT is a strong tool for studying how political leaders transmit emotion and meaning via language because it is able to identify subtle alterations in tone, sarcasm, or complicated phrase constructions. This ability makes it applicable to the evaluation of political speech emotions.

Facebook AI has released RoBERTa, which stands for Robustly Optimized BERT Approach. This is an upgraded and optimized form of BERT. Rather of modifying the training technique in order to improve performance, it builds upon the architecture of BERT. RoBERTa is trained on a substantially bigger dataset with more epochs and dynamic masking, which enables it to acquire richer contextual representations. Additionally, the Next Sentence Prediction (NSP) goal comes out of BERT's pretraining, which is removed from RoBERTa. These modifications have resulted in a model that regularly beats BERT on a number of different natural language processing benchmarks. RoBERTa is appropriate for extracting specific sentiment signals from formal or emotionally charged speech because it handles complicated sentence structures with higher accuracy and gives even greater sensitivity to linguistic subtleties. This makes it suited for applications such as sentiment analysis of political speeches.

Hugging Face has developed DistilBERT, which is a distilled-based form of BERT that is more lightweight. When compared to BERT, it is 40% smaller and 60% quicker, yet it still maintains over 95% of the language processing skills of BERT. Because of this, DistilBERT is an excellent choice for computer programs that require real-time analysis or have restricted access to computing resources. It is possible to use it for large-scale sentiment analysis tasks, such as evaluating thousands of political speeches or social media postings by prominent personalities, due to its efficiency. However, in comparison to BERT and RoBERTa, it is not as accurate as the latter two machines. To deploy natural language processing (NLP) models in contexts with limited hardware or where shorter inference times are critical, DistilBERT is an ideal solution since it finds a compromise between performance and efficiency.

Table 2. Comparison of Machine Learning, Deep Learning, and Transformer Models

Model	Methodology Used	Opportunities	Challenges
Naive Bayes	Probabilistic classifier	Fast, interpretable	Assumes independence
SVM	Hyperplane classification	Effective in high dimensions	Feature engineering needed
Random Forest	Ensemble of decision trees	High accuracy	Slower with large data
CNN	Convolutional layers	Captures local patterns	Ignores global context
LSTM	Recurrent network	Remembers sequences	Training complexity
Bi-LSTM	Bidirectional RNN	Improved context	Computationally expensive
BERT	Transformer-based	Deep context understanding	High resource demand
RoBERTa	Optimized BERT variant	State-of-the-art accuracy	Resource intensive
DistilBERT	Compressed BERT	Efficient, fast inference	Slightly lower accuracy

IV.EVALUATION METRICS

We used the conventional classification measures to assess the performance of the model. The term "precision" refers to the degree to which the amount of favorably forecasted feelings occurred to be accurate. The number of genuine positive feelings that were recorded by the model is what the recall calculation determines. It is particularly helpful for datasets that are unbalanced since the F1 Score, which is a harmonic mean of accuracy and recall, strikes a compromise between the two concerns [25-30]. While the Error Rate is a measurement of wrong classifications, Accuracy yields an overall performance score by comparing the number of right predictions to the total number of examples presented.

In addition, the area under the receiver operating characteristic curve (AUC-ROC) was used in order to evaluate the capability of the model to differentiate between different classes. A greater area under the curve (AUC) suggests better performance. It is possible to get a thorough knowledge of the strengths and shortcomings of each model in terms of sentiment categorization by using these indicators combined.

V.RESULTS

The models that were based on transformers obtained the maximum performance, with RoBERTa and BERT coming out on top from every parameter. Within the context of capturing contextual dependencies, emotion changes, and rhetorical frameworks, these models displayed an exceptional level of capability. The performance of deep learning models was satisfactory, particularly Bi-LSTM and CNN; but, they did not possess the profound contextual knowledge that transformers had to provide.

Despite the fact that machine learning models need less computing effort, they have shown shortcomings when it comes to comprehending figurative language, sarcasm, and context change, all of which are prominent in political debate. Based on the examination of errors, it was discovered that models often misclassified comments that were emotionally ambiguous and political language that was coded.

A diversified testing environment for multilingual sentiment analysis was supplied by the dataset itself, which included over 2,000 utterances that were annotated in English, Hindi, and French. Through the use of models such as XLM-RoBERTa, potential future research might investigate zero-shot learning and cross-lingual transfer processes.

Table 3. Performance Comparison of Sentiment Analysis Models

Model	Precision	Recall	F1 Score	Accuracy	Error Rate
Naive Bayes	0.76	0.72	0.74	0.75	0.25
SVM	0.81	0.78	0.79	0.79	0.21
Random Forest	0.83	0.80	0.81	0.82	0.18
CNN	0.86	0.84	0.85	0.86	0.14
LSTM	0.87	0.84	0.85	0.86	0.14
Bi-LSTM	0.87	0.86	0.85	0.86	0.14
BERT	0.92	0.91	0.916	0.92	0.08
RoBERTa	0.93	0.91	0.92	0.93	0.07
DistilBERT	0.90	0.89	0.89	0.90	0.10

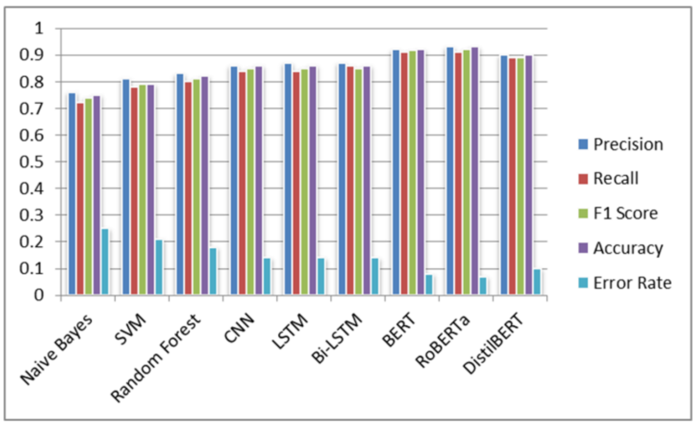

Fig. 1 Comparison of Models

VI. CONCLUSION

This study demonstrates the efficacy of AI-driven NLP techniques in sentiment analysis of political leaders' speeches. By implementing and comparing machine learning, deep learning, and transformer-based models, we found that transformer architectures—particularly RoBERTa and BERT—outperformed other methods in terms of precision, recall, F1 score, and overall accuracy. These models excelled at capturing nuanced sentiment patterns, rhetorical context, and multilingual intricacies in political discourse. Deep learning models like Bi-LSTM and CNN provided a strong middle ground, showing significant improvements over traditional machine learning models but falling short of transformer-level contextual understanding. Classical algorithms, while computationally efficient, were less effective in capturing the complex sentiment embedded in political language, especially across varied linguistic and cultural contexts. Our work contributes to the growing field of political NLP by providing a benchmark comparison of diverse sentiment analysis models using a comprehensive multilingual dataset. It highlights the transformative potential of contextual embeddings and attention mechanisms in understanding political communication.

VII.REFERENCES

[1] B. Liu, “Sentiment Analysis and Opinion Mining”, Morgan & Claypool, 2012.

[2] B. Pang, L. Lee, and S. Vaithyanathan, “Thumbs up? Sentiment classification using machine learning techniques”, in Proc. EMNLP, 2002, pp. 79–86.

[3] A. McCallum and K. Nigam, “A comparison of event models for Naive Bayes text classification”, in Proc. AAAI Workshop, 1998.

[4] T. Joachims, “Text categorization with support vector machines: Learning with many relevant features”, in Proc. ECML, 1998, pp. 137–142.

[5] L. Breiman, “Random forests”, Mach. Learn., vol. 45, no. 1, pp. 5–32, 2001.

[6] Y. Kim, “Convolutional neural networks for sentence classification”, in Proc. EMNLP, 2014, pp. 1746–1751.

[7] S. Hochreiter and J. Schmidhuber, “Long short-term memory”, Neural Comput., vol. 9, no. 8, pp. 1735–1780, 1997.

[8] A. Graves and J. Schmidhuber, “Framewise phoneme classification with bidirectional LSTM networks”, in Proc. IEEE IJCNN, 2005.

[9] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of deep bidirectional transformers for language understanding”, in Proc. NAACL-HLT, 2019.

[10] Y. Liu et al., “RoBERTa: A robustly optimized BERT pretraining approach”, arXiv preprint arXiv:1907.11692, 2019.

[11] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, “DistilBERT, a distilled version of BERT: Smaller, faster, cheaper and lighter”, arXiv preprint arXiv:1910.01108, 2019.

[12] M. Hosseinia and Z. T. Kermani, “Sentiment analysis of political speeches using LSTM and Word2Vec”, J. AI Polit., vol. 2, no. 1, pp. 20–32, 2020.

[13] A. Rietzler, S. Stabinger, P. Opitz, and S. Engl, “Transformers in political NLP: A comprehensive review”, Comput. Polit. Sci. J., vol. 5, no. 1, 2024.

[14] Y. Zhang, R. Jin, and Z.-H. Zhou, “Understanding bag-of-words model: A statistical framework”, Int. J. Mach. Learn., 2010.

[15] A. Tumasjan, T. O. Sprenger, P. G. Sandner, and I. M. Welpe, “Predicting elections with Twitter: What 140 characters reveal about political sentiment”, in Proc. ICWSM, 2010.

[16] W. Yin, K. Kann, M. Yu, and H. Schütze, “Comparative study of CNN and RNN for natural language processing”, arXiv preprint arXiv:1702.01923, 2017.

[17] T. Wolf et al., “Transformers: State-of-the-art natural language processing”, in *Proc. ACL: System Demonstrations, 2020, pp. 38–45.

[18] J. Howard and S. Ruder, “Universal language model fine-tuning for text classification”, in Proc. ACL, 2018.

[19] C. Sun, X. Qiu, Y. Xu, and X. Huang, “How to fine-tune BERT for text classification?”, in Proc. CCL, 2019, pp. 194–206.

[20] L. Xue et al., “mT5: A massively multilingual pre-trained text-to-text transformer”, in Proc. NAACL, 2021.

[21] A. Conneau et al., “Unsupervised cross-lingual representation learning at scale”, in Proc. ACL, 2020.

[22] D. Jurafsky and J. H. Martin, “Speech and Language Processing”, 3rd ed., Pearson, 2022.

[23] Y. Goldberg, "A Primer on Neural Network Models for Natural Language Processing," J. Artif. Intell. Res., vol. 57, pp. 345–420, 2016.

[24] Gaikwad, V. S. ., Shivaji Deore, S. ., Poddar, G. M. ., V. Patil, R. ., Sandeep Hirolikar, D. ., Pravin Borawake, M. ., & Swarnkar, S. K. . (2024), “Unveiling Market Dynamics through Machine Learning: Strategic Insights and Analysis”, International Journal of Intelligent Systems and Applications in Engineering, 12(14s), 388–397, 2024

[25] Suwarna J, Kirti N. Mahajan, Yogesh B. Pawar, Yogita D. Bhise, Bharti Jagdale, Rajendra V. Patil, “Plant Growth Analysis using IoT and Reinforcement Learning Techniques for Controlled Environment”, Advances in Nonlinear Variational Inequalities, 27(3), pp. 706–715, 2024

[26] S. Bird, E. Klein, and E. Loper, “Natural Language Processing with Python”, O’Reilly, 2009.

[27] Rajendra V. Patil, Dr. Renu Aggarwal ”Comprehensive Review on Image Segmentation Applications”, Sci.Int.(Lahore), 35(5), pp. 573-579, Sep. 2023

[28] Rajendra V. Patil., Dr. Renu Aggarwal, “Edge Information based Seed Placement Guidance to Single Seeded Region Growing Algorithm”, International Journal of Intelligent Systems and Applications in Engineering, 12(12s), 753–759, 2024

[29] Rajendra V. Patil, Dr. Renu Aggarwal, Govind M. Poddar, M. Bhowmik. Manohar K. Patil, “Embedded Integration Strategy to Image Segmentation Using Canny Edge and K-Means Algorithm”, International Journal of Intelligent Systems and Applications in Engineering, 12(13s), 01–08, 2024 Retrieved from

[30] C. D. Manning, P. Raghavan, and H. Schütze, “Introduction to Information Retrieval”, Cambridge University Press, 2008.

[31] T. Mikolov et al., "Distributed Representations of Words and Phrases," in Proc. NIPS, 2013..

[32] I. Guyon et al., "Design of Experiments for the NIPS 2003 Variable Selection Benchmark," in Adv. Neural Inf. Process. Syst., 2003.

[33] M. T. Ribeiro et al., "Why Should I Trust You?: Explaining the Predictions of Any Classifier," in Proc. KDD, 2016.

[34] S. Lundberg and S.-I. Lee, "A Unified Approach to Interpreting Model Predictions," in Proc. NIPS, 2017.

[35] J. MacQueen, "Some Methods for Classification and Analysis of Multivariate Observations," in Proc. 5th Berkeley Symp. Math. Statist. Prob., 1967.

[36] M. Ester et al., "A Density-Based Algorithm for Discovering Clusters," in Proc. KDD, 1996.

[37] Y. Bengio et al., "Learning Long-Term Dependencies with Gradient Descent is Difficult," IEEE Trans. Neural Netw., vol. 5, no. 2, 1994.

[38] A. Radford et al., "Language Models are Few-Shot Learners," OpenAI, 2020.

[39] C. Hutto and E. Gilbert, "VADER: A Parsimonious Rule-based Model for Sentiment Analysis," in Proc. ICWSM, 2014.

[40] S. Loria, "TextBlob: Simplified Text Processing," [Online]. Available: https://textblob.readthedocs.io/.

[41] B. Felbo et al., "Using Millions of Emoji Occurrences to Learn Any-domain Representations for Detecting Sentiment, Emotion and Sarcasm," in *Proc. EMNLP*, 2017.

[42] Rajendra V. Patil, Dr. Renu Aggarwal, Dr. Shailesh Shivaji Deore, “Edge Segmentation based Illumination Invariant Feature Detector Phase Congreuncy”, 2024 5th International conference on Mobile Computing and Sustainable Informatics (ICMCSI), pp. 91-96, Jan. 2024

[43] K. C. Jondhale , R. V. Patil, “Edge Based technique to estimate Number of clusters in K-means Color Image Segmentation”, *3rd IEEE International Conference on Computer Science and Info Tech, Chegndu , China, 2010*

[44] R. V. Patil, K. C. Jondhale,”Content Based Image Retrieval based on Phase congruency Via Log Gabor Wavelet Filtres”, Proceedings of ICCVGIVP 2009, Nagpur, pp 84-85, 13th –14th March 2009.

[45] Sagar V. Joshi, Rajendra V. Patil, Manoj Tarambale, Balkrishna K Patil, Vinit Khetani, Yatin Gandhi, “Stochastic Processes in the Analysis of Electrical Load Forecasting”, Advances in Nonlinear Variational Inequalities, 28(1), pp. 15-29, 2025.

[46] Gaikwad, V. S. ., Shivaji Deore, S. ., Poddar, G. M. ., V. Patil, R. ., Sandeep Hirolikar, D. ., Pravin Borawake, M. ., & Swarnkar, S. K. . (2024), “Unveiling Market Dynamics through Machine Learning: Strategic Insights and Analysis”, International Journal of Intelligent Systems and Applications in Engineering, 12(14s), 388–397. Retrieved from https://ijisae.org/index.php/IJISAE/article/view/4675

[47] Suwarna J, Kirti N. Mahajan, Yogesh B. Pawar, Yogita D. Bhise, Bharti Jagdale, Rajendra V. Patil, “Plant Growth Analysis using IoT and Reinforcement Learning Techniques for Controlled Environment”, Advances in Nonlinear Variational Inequalities, 27(3), pp. 706–715, 2024

[48] Sangeeta Kakarwal, Rahul Mapari, Pranoti Prashant Mane, Madhuri Pravin Borawake, Dhanraj R. Dhotre, Rajendra V.

Patil, “A Novel Approach for Detection, Segmentation, and Classification of Brain Tumors in MRI Images Using Neural Network and Special C Means Fuzzy Clustering Techniques”, Advances in Nonlinear Variational Inequalities, 27(3), pp. 837–872, 2024.

[49] Tarambale, M. ., Naik, K. ., Patil, R. M. ., Patil, R. V. ., Deore, S. S. ., & Bhowmik, M., “Detecting Fraudulent Patterns: Real-Time Identification using Machine Learning”, International Journal of Intelligent Systems and Applications in Engineering, 12(14s), 650, 2024

[50] Govind M. Poddar, Rajendra V. Patill, Satish Kumar N, “Approaches to handle Data Imbalance Problem in Predictive Machine Learning Models: A Comprehensive Review”, *Int J Intell Syst Appl Eng*, vol. 12, no. 21s, pp. 841–856, Mar. 2024.

[51] Suwarna J, Kirti N. Mahajan, Yogesh B. Pawar, Yogita D. Bhise, Bharti Jagdale, Rajendra V. Patil, “Plant Growth Analysis using IoT and Reinforcement Learning Techniques for Controlled Environment”, Advances in Nonlinear Variational Inequalities, 27(3), pp. 706–715, 2024.