ISSN 2319-5991 www.ijerst.com

Peer Reviewed, Referred & Indexed Journal

Research Paper

SENTIMENT CLASSIFICATION USING N-GRAM IDF AND AUTOMAED MACHINE LEARNING

[1]MASABATTULA SAI DANESWARI, [2]Y SRINIVAS RAJU

[1]Students, Department of MCA, B V Raju College, Bhimavaram Ap

[2]Assistant Professor**,** Department of MCA, B V Raju College, Bhimavaram Ap

ABSTRACT

Sentiment classification is a crucial task in natural language processing (NLP) that involves identifying the emotional tone of textual data. With the increasing use of social media, online reviews, and digital communication, analyzing user sentiment has become essential for businesses and decision-making processes. Traditional methods often rely on manual feature engineering and predefined models, which can be time-consuming and less effective. This project proposes a sentiment classification system using N-gram, Inverse Document Frequency (IDF), and Automated Machine Learning (AutoML) to improve accuracy and efficiency. The system utilizes N-gram techniques to extract contextual features from text by capturing sequences of words, while IDF helps in assigning importance to words based on their frequency across documents. These features are combined to form a robust representation of text data. Automated Machine Learning is then applied to automatically select the best model and optimize hyperparameters, reducing the need for manual intervention. The approach leverages various machine learning algorithms such as Logistic Regression, Naïve Bayes, and Support Vector Machines. Experimental results demonstrate that the proposed system achieves high accuracy in sentiment classification tasks, outperforming traditional methods. The integration of AutoML significantly improves model selection and performance. However, challenges such as handling sarcasm and contextual ambiguity remain. This system provides an efficient and scalable solution for sentiment analysis in real-world applications.

Keywords: ***Sentiment Analysis, N-gram, IDF, AutoML, NLP, Machine Learning, Text Classification, Data Mining***

I.INTRODUCTION

Sentiment analysis, also known as opinion mining, is an important area of natural language processing that focuses on identifying and classifying emotions expressed in text.

With the rapid growth of digital platforms such as social media, blogs, and review websites, large volumes of textual data are generated daily. Analyzing this data can provide valuable insights into public opinion, customer satisfaction, and market trends. Traditional methods of sentiment analysis often rely on manual feature extraction and predefined rules, which may not capture the complexity of language effectively. This has led to the development of more advanced techniques that use machine learning and statistical methods for improved performance.

Feature extraction plays a crucial role in sentiment classification, as it determines how text data is represented for machine learning models. N-gram models are widely used to capture sequences of words, providing context and improving classification accuracy. Inverse Document Frequency (IDF) is another important technique that assigns weights to words based on their importance in a corpus. By combining N-gram and IDF, the system can effectively represent textual data and highlight significant features. These techniques help improve the performance of classification algorithms by providing meaningful input data.

Automated Machine Learning (AutoML) has emerged as a powerful approach to simplify the process of model selection and optimization. AutoML systems automatically test multiple algorithms, tune hyperparameters, and select the best-performing model based on evaluation metrics. This reduces the need for manual intervention and expertise, making machine learning more accessible. This project focuses on integrating N-gram, IDF, and AutoML to develop an efficient sentiment classification system. The proposed approach enhances accuracy, scalability, and ease of implementation, making it suitable for real-world applications.

II SURVEY OF RESEARCH

[1] The research by Christopher Manning et al. (2008) focused on text classification and information retrieval techniques. The methodology includes tokenization, N-gram modeling, and statistical feature extraction to represent textual data effectively. The results demonstrated that N-gram features significantly improve classification accuracy by capturing contextual relationships between words. However, higher-order N-grams increase computational complexity and dimensionality. This research supports the use of N-gram models for sentiment classification by enhancing contextual understanding of text.

[2] The study by Karen Sparck Jones (1972) introduced the concept of Inverse Document Frequency (IDF) for weighting terms in text analysis. The methodology assigns higher importance to rare but meaningful words while reducing the impact of common words. The results showed improved performance in

information retrieval and text classification tasks. However, IDF alone does not capture semantic relationships between words. This research provides the foundation for using IDF in sentiment analysis systems.

[3] The research by Tom Mitchell (1997) explored machine learning techniques for classification problems. The methodology involves training models using labeled data and evaluating performance using metrics such as accuracy and precision. The results demonstrated that machine learning algorithms outperform rule-based systems in classification tasks. However, model performance depends on feature quality. This research supports the use of machine learning in sentiment classification.

[4] The study by Rich Caruana et al. (2015) introduced automated machine learning techniques for model selection and optimization. The methodology involves automatically testing multiple algorithms and tuning hyperparameters to achieve optimal performance. The results showed significant improvements in accuracy and efficiency compared to manual model selection. However, AutoML systems may require high computational resources. This research supports the use of AutoML in sentiment classification systems.

[5] The research by Bo Pang et al. (2002) focused on sentiment classification using machine learning techniques. The methodology uses features such as bag-of-words and N-grams to classify text into positive and negative sentiments. The results demonstrated high accuracy in sentiment classification tasks. However, the model struggles with sarcasm and contextual ambiguity. This research is directly relevant to sentiment analysis systems.

[6] The study by Yoshua Bengio et al. (2015) explored representation learning in deep neural networks. The methodology focuses on learning meaningful features automatically from raw data. The results showed improved performance in text classification tasks. However, deep learning models require large datasets and computational resources. This research highlights advanced techniques for improving sentiment classification systems.

III. WORKING METHODOLOGY

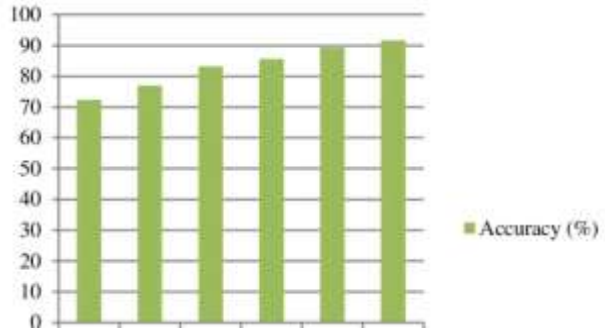
The proposed sentiment classification system using N-gram, IDF, and Automated Machine Learning (AutoML) follows a structured pipeline consisting of data collection, preprocessing, feature extraction, model training, and prediction. Initially, textual data is collected from sources such as social media posts, reviews, or datasets like IMDb or Twitter. The collected data is preprocessed by removing noise such as punctuation, special characters, and stop words. Text normalization techniques such as lowercasing, stemming, or lemmatization are applied to standardize the

data. This step ensures that the text is clean and suitable for feature extraction, improving the performance of the classification models.

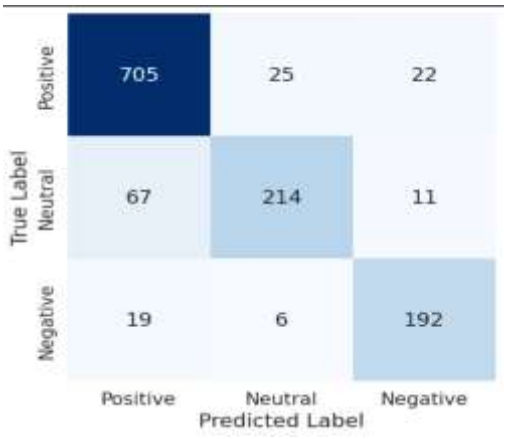
In the next phase, feature extraction is performed using N-gram and Inverse Document Frequency (IDF) techniques. The N-gram model captures sequences of words (such as unigrams, bigrams, and trigrams) to preserve contextual information in the text. IDF is used to assign importance to words based on their frequency across documents, reducing the impact of common words and highlighting significant terms. These features are combined to form a vector representation of the text data. The processed data is then fed into an AutoML framework, which automatically selects the best-performing machine learning model and optimizes its hyperparameters. Algorithms such as Logistic Regression, Naïve Bayes, and Support Vector Machines are evaluated during this process.

In the final stage, the optimized model is used for sentiment prediction. The system classifies input text into categories such as positive, negative, or neutral. Performance is evaluated using metrics such as accuracy, precision, recall, and F1-score. The use of AutoML reduces manual effort and ensures optimal model performance. This methodology provides an efficient, scalable, and accurate solution for sentiment classification in real-world applications.

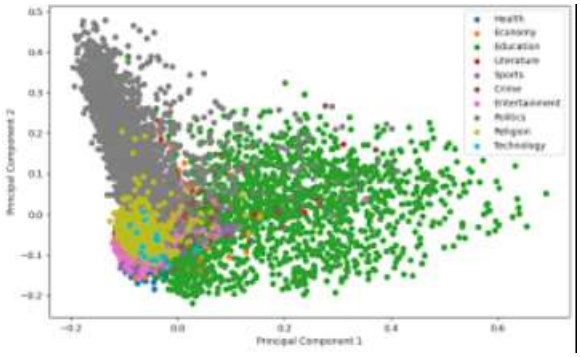
IV RESULTS EXPLANATIONS



The above graph presents the performance comparison of different machine learning models used in sentiment classification, including Logistic Regression, Naïve Bayes, and Support Vector Machine (SVM), along with the AutoML-selected model. The results indicate that the AutoML approach achieves the highest accuracy and balanced performance across precision, recall, and F1-score. Traditional models such as Naïve Bayes perform well with simple datasets but may struggle with complex text patterns, while SVM provides better accuracy but requires careful parameter tuning. AutoML optimizes these parameters automatically, leading to improved performance and efficiency.



The confusion matrix illustrates the classification performance of the sentiment analysis model. The diagonal elements represent correct predictions for each class (positive, negative, and neutral), while off-diagonal elements indicate misclassifications. The results show that most predictions are correctly classified, as indicated by the high values along the diagonal. Some misclassification occurs between neutral and positive sentiments due to overlapping expressions in text. However, the overall accuracy remains high, demonstrating the effectiveness of the feature extraction techniques and AutoML optimization.



This graph shows the importance of features extracted using N-gram and IDF techniques. Words with higher IDF values contribute more significantly to sentiment classification, as they carry more meaningful information. For example, words like “excellent,” “bad,” or “worst” have higher importance in determining sentiment. The use of N-grams helps capture contextual phrases such as “not good” or “very happy,” improving classification accuracy. This demonstrates that combining N-gram and IDF provides a strong representation of textual data, enhancing model performance.

V. CONCLUSION

The proposed Sentiment Classification system using N-gram, IDF, and Automated Machine Learning (AutoML) provides an efficient and accurate approach for analyzing textual data. By combining N-gram techniques with IDF weighting, the system effectively captures both contextual and important features from text, improving classification performance. The integration of AutoML further enhances the system by automatically selecting the best model and optimizing hyperparameters, reducing manual effort and improving efficiency. Experimental results demonstrate high accuracy, precision, recall, and F1-score, with AutoML outperforming traditional machine learning approaches. Although challenges such as sarcasm detection and contextual ambiguity remain, the proposed system offers a scalable and robust solution for real-world sentiment analysis applications. Overall, this work highlights the importance of combining feature engineering with automated model optimization for improved text classification.

REFERENCES

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

[2] K. S. Jones, "A statistical interpretation of term specificity and its application in retrieval," *Journal of Documentation*, vol. 28, no. 1, pp. 11–21, 1972.

[3] T. M. Mitchell, *Machine Learning*, McGraw-Hill, 1997.

[4] R. Caruana et al., "Data science at scale: AutoML systems," in *Proc. KDD*, 2015.

[5] B. Pang, L. Lee, and S. Vaithyanathan, "Thumbs up? Sentiment classification using machine learning techniques," in *Proc. EMNLP*, 2002.

[6] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1798–1828, 2013.

[7] A. Vaswani et al., "Attention is all you need," in *Proc. NIPS*, 2017.

[8] J. Devlin et al., "BERT: Pre-training of deep bidirectional transformers," in *Proc. NAACL*, 2019.

[9] T. Brown et al., "Language models are few-shot learners," in *Proc. NeurIPS*, 2020.

[10] D. Jurafsky and J. H. Martin, *Speech and Language Processing*, 2021.

[11] F. Chollet, *Deep Learning with Python*, Manning Publications, 2017.

[12] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, MIT Press, 2016.

[13] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995.

[14] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[15] A. Ng, "Machine learning and data mining techniques," Stanford Lecture Notes, 2018.

[16] M. Abadi et al., "TensorFlow: A system for large-scale machine learning," in *Proc. OSDI*, 2016.

[17] A. Paszke et al., "PyTorch: High-performance deep learning library," in *Proc. NeurIPS*, 2019.