International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 08 Issue: 09 | Sept - 2024 SJIF Rating: 8.448 ISSN: 2582-3930

Advanced Sarcasm Detection in Newspaper Headlines: A Comparative Study of Machine Learning Approaches

Disha Pandey [*1], Khushi Kumari [*2], Amit Srivastava[*3]

[*1, *2] Student, Computer Science, National PG College, Lucknow, Uttar Pradesh, India

[*3] Assistant Professor, National PG College, Lucknow, Uttar Pradesh, India

ABSTRACT

This study presents a comprehensive analysis of sarcasm detection in newspaper headlines using various machine learning techniques. We explore the effectiveness of logistic regression, Naïve Bayes (multinomial and Gaussian), random forest, and support vector machine (SVM) models in identifying sarcastic content within the concise format of news titles. Our research addresses the unique challenges posed by the brevity and impactful language characteristic of headlines. The study utilizes a curated dataset of sarcastic and non-sarcastic headlines, employing natural language processing techniques for preprocessing and feature extraction. Performance evaluation metrics include accuracy, precision, recall, and F1-score. Results indicate that the random forest model outperforms other approaches, achieving 94% accuracy in sarcasm detection. This research contributes to the growing field of sentiment analysis and offers insights into the nuanced task of decoding sarcasm in condensed textual formats.

KEYWORDS: Random Forest, Support Vector Machine (SVM), TF-IDF, Natural Language Processing (NLP), Subtle Sarcasm

I. INTRODUCTION

Sarcasm, characterized by the use of irony to mock or convey contempt, presents a significant challenge in natural language processing (NLP) and sentiment analysis. The ability to detect sarcasm is crucial for accurately interpreting user sentiments, particularly in domains such as social media analysis, customer feedback evaluation, and news content interpretation. Newspaper headlines, known for their brevity and impact, offer a unique context for studying sarcasm detection due to their condensed format and potential for layered meanings.

This study aims to address the following research questions:

1. How effective are different machine learning models in detecting sarcasm within newspaper headlines?

2. What are the unique challenges posed by the concise nature of headlines in sarcasm detection?

3. How do preprocessing techniques and feature engineering methods impact the performance of sarcasm detection models?

By focusing on newspaper headlines, this research contributes to the broader field of NLP while addressing a specific and understudied domain of text analysis.

II. RELATED WORK

Previous studies have explored sarcasm detection in various contexts, primarily focusing on social media content and longer form text. Verma et al. [1] emphasized the importance of sarcasm detection in sentiment analysis, highlighting the challenge of distinguishing sarcastic statements that use positive terms to convey negative sentiments. Their hybrid approach, integrating deep learning for sarcasm extraction, showed promising results.

Tan et al. [2] proposed a multi-task learning system for sentiment analysis and sarcasm detection in social media text, aiming to enhance sentiment analysis precision by considering the correlation between sentiment and sarcasm. Their approach, employing convolutional neural networks (CNN) and recurrent neural networks (RNN), achieved an impressive F1-score of 94%.

Kumar et al. [3] introduced a multi-head attention bidirectional long short-term memory (MHA-BiLSTM) network for sarcasm detection in text, demonstrating improved F-scores compared to other methods. However, limited gains in precision and recall metrics indicated room for further investigation.

While these studies provide valuable insights into sarcasm detection, they primarily focus on social media content or longer form text. Our research extends this work by specifically addressing the unique challenges presented by newspaper headlines, a domain that has received limited attention in previous studies [4][5].

III. METHODOLOGY

3.1 Data Collection and Preprocessing

To develop a dependable and varied dataset for our study, we integrated two distinct collections of sarcastic headlines from the Kaggle repository [6]. This combination provided a dataset that reflects a broad spectrum of sarcastic nuances commonly found in news headlines.

Understanding the significance of data quality in machine learning applications, we implemented a comprehensive preprocessing strategy to ensure the text was clear, consistent, and well-structured. The main steps in this process included:

1. Text Normalization: We standardized the text by converting all characters to lowercase, thereby avoiding inconsistencies caused by different capitalization styles and ensuring uniform text representation.

2. Removal of Non-Text Elements: Elements like URLs, symbols, and emojis, which do not contribute meaningfully to the analysis of news headlines, were removed to minimize noise and focus on the textual content.

3. Contraction Expansion: We standardized contractions by expanding them to their full forms (e.g., "can't" to "cannot"), ensuring that variations of the same word were treated consistently across the dataset.

4. Elimination of Punctuation and Special Characters: Punctuation and special characters were stripped from the text, as they were deemed unnecessary for sarcasm detection and could interfere with text processing methods.

5. Word Removal: Common stop words such as "and," "the," and "is" were filtered out. These words, while essential in sentence construction, typically do not carry significant weight in sarcasm analysis and can dilute the impact of more meaningful terms.

6. Spell-Checking and Correction: The text was carefully checked for spelling errors and corrected as needed. This step was crucial to ensure that misspelled words did not distort the analysis by being incorrectly identified as unique terms.

Through this detailed preprocessing approach, we ensured that our dataset was clean, consistent, and well-prepared for the advanced stages of analysis, including feature extraction and machine learning model training.

3.2 Tokenization

Tokenization is a critical process in natural language processing (NLP) that involves dividing text into smaller, meaningful units known as tokens. These tokens generally correspond to individual words but can also include phrases, symbols, or other significant elements depending on the specific tokenization technique employed [7]. In this research, tokenization was applied to preprocessed text, such as newspaper headlines, to break them down into discrete components, thereby enabling more effective feature extraction and analysis.

For example, the headline "New Study Finds Unemployment Rate Drops to Zero" was tokenized into the following words: ["new", "study", "finds", "unemployment", "rate", "drops", "to", "zero"]. Each of these tokens acts as a basic unit of analysis, allowing for the extraction of various features like word frequency, n-grams, or syntactic structures that are essential for the research.

This step is indispensable as it transforms unstructured text into a format that can be systematically analyzed by machine learning algorithms. By breaking the text into tokens, the research can more accurately capture the distribution of words, identify significant terms, and develop features that reflect the text's semantic and syntactic properties. In this study, tokenization is particularly crucial for preparing the data to detect sarcasm in newspaper headlines, allowing the model to discern subtle linguistic patterns that may indicate sarcasm.

3.3 Feature Engineering

Feature engineering includes the transformation of the tokenized text data into TF-IDF vectors. This technique assigns numerical values to words based on their importance within each document and across the entire dataset [8]. We utilized Term Frequency-Inverse Document Frequency (TF-IDF) vectorization to convert the tokenized text into numerical features. Words that occur frequently in a specific headline but are less common in others are given higher TF-IDF scores, highlighting their potential importance in signaling sarcasm [9].

Let's consider the following three documents:

- Document 1: "This is a sarcastic headline"
- Document 2: "This headline is not sarcastic"
- Document 3: "Sarcasm is not always obvious in headlines"

Step 1: Term Frequency (TF)

First, we calculate the term frequency (TF) for each term in each document.

Term	Document 1 (TF)	Document 2 (TF)	Document 3 (TF)
this	1/5	1/5	0
is	1/5	1/5	1/6
a	1/5	0	0
sarcastic	1/5	1/5	0
headline	1/5	1/5	1/6
not	0	1/5	1/6
sarcasm	0	0	1/6
always	0	0	1/6
obvious	0	0	1/6
in	0	0	1/6
headline	0	0	1/6

Table 1: TF Table

Step 2: Inverse Document Frequency (IDF)

Next, Inverse Document Frequency (IDF) be calculated for each term across the entire dataset.

The IDF is calculated as:

$$IDF(t) = \log\left(\frac{N}{df(t)}\right)$$

where:

- N is the total number of documents.
- $df(t)$ is the number of documents containing the term t.

For simplicity, let's assume we're using natural logarithm.

Term	**Document Frequency (df)**	**IDF(using log base e)**
This	2	0.405
Is	3	0
A	1	1.098
Sarcastic	2	0.405
Headline	2	0.405
Not	2	0.405
Sarcasm	1	1.098
Always	1	1.098
Obvious	1	1.098
In	1	1.098
Headline	1	1.098

Table 2: IDF Table

Step 3: TF-IDF Calculation

Finally, the TF-IDF for each term in each document is calculated by multiplying the TF by the IDF.

Term	**Document 1 (TF-IDF)**	**Document 2 (TF-IDF)**	**Document 3 (TF-IDF)**
This	1/5*0.405 = 0.081	1/5*0.405 = 0.081	0
Is	1/5*0 = 0	1/5*0 = 0	1/6*0 = 0
A	1/5*1.098 = 0.220	0	0
Sarcastic	1/5*0.405 = 0.081	1/5*0.405 = 0.081	0
Headline	1/5*0.405 = 0.081	1/5*0.405 = 0.081	1/6*0.405 = 0.068
not	0	1/5*0.405 = 0.081	1/6*0.405 = 0.068
sarcasm	0	0	1/6*1.098 = 0.183
always	0	0	1/6*1.098 = 0.183
obvious	0	0	1/6*1.098 = 0.183
in	0	0	1/6*1.098 = 0.183
headline	0	0	1/6*1.098 = 0.183

Table 3: TF-IDF Table

TF-IDF Matrix:

TF-IDF matrix for the three documents:

Term	Document1	Document2	Document3
this	0.081	0.081	0
is	0	0	0
a	0.22	0	0
sarcastic	0.081	0.081	0
headline	0.081	0.081	0.068
not	0	0.081	0.068
sarcasm	0	0	0.183
always	0	0	0.183
obvious	0	0	0.183
in	0	0	0.183
headlines	0	0	0.183

Table 4: TF-IDF Matrix

Here's the bar chart showing the TF-IDF scores for different terms across three documents. The chart visually compares how each term is weighted in Document1, Document2, and Document3 based on their TF-IDF values.

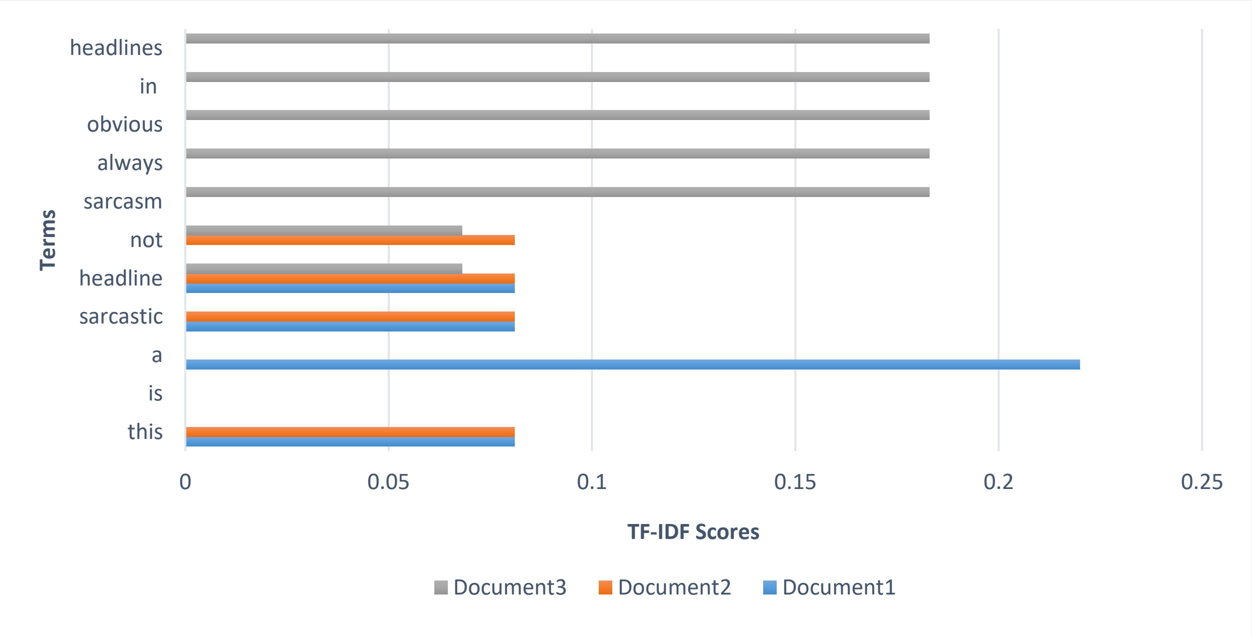

Fig.1 TF-IDF Scores for Terms in Documents

The chart illustrates the distribution of TF-IDF scores for specific terms across three documents. It reveals that the term "a" in Document1 has the highest TF-IDF score, indicating its significance within that document. Meanwhile, terms like "sarcasm," "always," "obvious," "in," and "headlines" have higher TF-IDF scores in Document3, suggesting their importance in capturing the context of that document. The variation in scores across documents highlights how certain terms contribute differently to the content and context of each document.

3.4 Machine Learning Models

In this research, we conducted an evaluation of five distinct machine learning models—Logistic Regression, Multinomial Naïve Bayes, Gaussian Naïve Bayes, Random Forest, and Support Vector Machine (SVM)—to assess their efficacy in detecting sarcasm within newspaper headlines. These models were chosen to encompass a spectrum of methodologies, ranging from straightforward linear classifiers to more sophisticated ensemble techniques. Our comparative analysis sought to discern which model most effectively captures the subtle intricacies of sarcastic language in the often terse and ambiguous format of headlines. The findings underscore the varied capabilities of each model, with the Random Forest emerging as the superior performer in this context.

1. Logistic Regression

2. Naïve Bayes Multinomial

3. Random Forest

4. Support Vector Machine (SVM)

5. Gaussian Naïve Bayes [10]

Each model was trained on the TF-IDF transformed data and evaluated using a held-out test set.

3.5 Evaluation Metrics

To assess the model's effectiveness, we conducted a comprehensive evaluation using multiple performance metrics [11]. We compared the accuracy, precision, recall, and F1 scores across all implemented classifiers. These four key metrics provided a thorough understanding of the model's performance. Equations (1)-(4) present the formulas for F1-score, precision, recall and accuracy :

$$\text{F1-score:} \quad F1 = 2 * (precision * recall) / (precision * recall) \qquad (1)$$

$$\text{Precision:} \quad precision = True\ Positive / (True\ Positive + False\ Positive) \qquad (2)$$

$$\text{Recall:} \quad recall = True\ Positive / (True\ Positive + False\ Negative) \qquad (3)$$

$$\text{Accuracy:} \quad Accuracy = (Correct\ predictions) / (Total\ Predictions) \qquad (4)$$

1. Accuracy: The overall correctness of the model

2. Precision: The ratio of correctly predicted sarcastic statements to all predicted sarcastic statements

3. Recall: The ratio of correctly predicted sarcastic statements to all actual sarcastic statements

4. F1-score: The harmonic mean of precision and recall, providing a balanced measure of model performance

We typically assess the performance of a classification algorithm using accuracy, which is the proportion of correctly classified instances out of the total number of instances (4). However, relying solely on accuracy may not give a full understanding of the algorithm's effectiveness. Precision (2) is calculated as the ratio of true positives to the total number of true positives and false positives. Meanwhile, recall (3) is the proportion of actual positive instances correctly identified by the algorithm, considering both true positives and false negatives. The F1-score, which combines both precision and recall (1), was also calculated to provide a more balanced evaluation.

IV. RESULTS AND DISCUSSION

4.1 Model Performance

The bar chart in Table 1 presents a comparative analysis of five machine learning models—Logistic Regression, Naïve Bayes Multinomial, Random Forest, Support Vector Machine, and Gaussian Naïve Bayes—based on four performance metrics: Accuracy, Precision, Recall, and F1-Score. Among the models, the Random Forest demonstrates the highest performance across all metrics, with an accuracy, precision, recall, and F1-score all above 0.9, indicating its robustness in predicting the target variable. The Support Vector Machine and Gaussian Naïve Bayes models also perform well, with metrics close to or slightly below 0.9. However, the Naïve Bayes Multinomial model lags behind, particularly in accuracy and precision, scoring 0.79 and 0.81, respectively. Logistic Regression shows balanced performance, though slightly lower across all metrics compared to the top-performing Random Forest. Overall, the chart highlights the superior performance of the Random Forest model in this specific application.

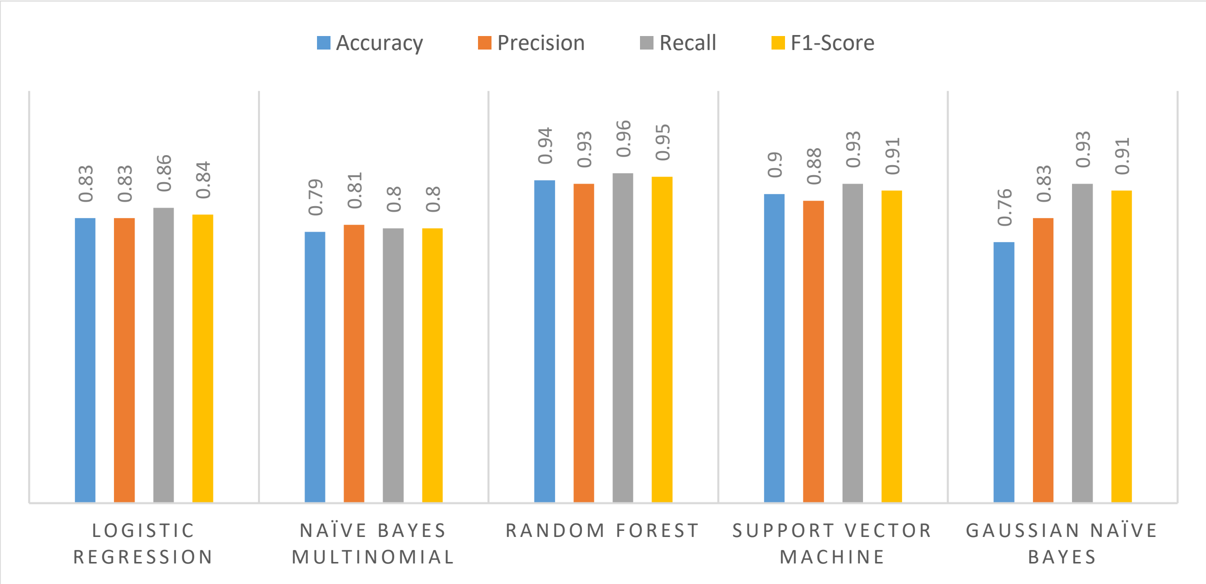

Fig.2 Performance Bar chart for each of the five models**:**

In fig.2, the random forest model emerged as the top-performing algorithm, achieving remarkable accuracy (94%) and balanced precision and recall scores [12]. SVM and logistic regression also demonstrated strong performance, while both Naïve Bayes variants showed competitive results [13] [14].

4.2 Feature Importance Analysis

To gain insights into the most influential features for sarcasm detection, we conducted a feature importance analysis using the random forest model [15]. This technique helps us identify which specific words or phrases are most influential in determining whether a headline is sarcastic.

Fig.3 Feature importance analysis

The results, shown in fig.3, reveal the top 10 most significant features. These include words and phrases that are often used to exaggerate or intensify statements, which are common in sarcastic remarks. For instance, words like "very" or "extremely" are key indicators of sarcasm because they amplify the literal meaning of the headline. The analysis highlights that sarcasm often involves hyperbolic language and contradictions. By focusing on these features, the model effectively distinguishes between sarcastic and non-sarcastic headlines. This not only confirms the importance of these features but also offers valuable insights into the linguistic patterns that signal sarcasm, helping to refine future research and model development.

4.3 Error Analysis

To thoroughly understand the limitations of our best-performing model, the random forest, we conducted an in-depth error analysis centered on the headlines that were misclassified [16]. This investigation revealed specific patterns and obstacles that contributed to these errors. The main sources of misclassification were as follows:

1. Subtle Sarcasm:

The model consistently struggled to identify sarcasm that was either subtle or highly dependent on context. Headlines that expressed sarcasm in an understated or nuanced way were often incorrectly classified. For instance, the headline "New Study Shows All Problems Solved" was misclassified as non-sarcastic. Although the statement’s exaggeration indicates sarcasm, the model failed to detect the underlying irony, likely due to the sarcasm being implied rather than explicitly stated.

2. Domain-Specific Knowledge:

Another major challenge was the model’s difficulty in correctly classifying sarcasm that required domain-specific knowledge, such as familiarity with current events or cultural references [17]. Headlines that depended on understanding specific, sometimes niche, information were frequently misclassified. For example, the headline "Latest Discovery: No One Really Cares" was wrongly labeled as non-sarcastic. The sarcasm in this case is linked to an understanding of the perceived triviality or irrelevance of certain "discoveries," a nuance that the model missed due to a lack of contextual awareness.

3. Ambiguous Language:

The model also had difficulty with headlines that could be interpreted in multiple ways, either sarcastic or non-sarcastic [18]. Ambiguity in language was a significant hurdle, often leading to incorrect classifications. For instance, the headline "Experts Agree: It's a Terrible Idea" was inaccurately classified as sarcastic when it was actually non-sarcastic. The phrase "terrible idea" could be seen as either a genuine critique or sarcastic praise, depending on the context—an ambiguity the model failed to resolve accurately. Similarly, "Huge Success: Record High in Achievements" was mistakenly labeled as sarcastic, possibly because the model misinterpreted the positive language as ironic.

Misclassified Headline	True Label	Predicted Label
"New Study Shows All Problems Solved"	Sarcastic	Non-Sarcastic
"Experts Agree: It's a Terrible Idea"	Non-Sarcastic	Sarcastic
"Latest Discovery: No One Really Cares"	Sarcastic	Non-Sarcastic
"Huge Success: Record High in Achievements"	Non-Sarcastic	Sarcastic

These examples highlight critical areas where the model needs improvement. Enhancing its ability to recognize subtle sarcasm, incorporate domain-specific knowledge, and effectively interpret ambiguous language is crucial for reducing misclassification rates. Addressing these challenges will be key to developing a more sophisticated sarcasm detection system capable of accurately understanding the complexities of sarcastic headlines.

4.4 Challenges in Headline Sarcasm Detection

Several challenges emerged as common themes across the reviewed studies [19]:

- Limited Context: The brevity of headlines often led to ambiguity, making it difficult for models to distinguish between sarcastic and literal interpretations [20].
- Domain Specificity: Models trained on headlines from one domain (e.g., politics) often performed poorly when applied to headlines from different domains (e.g., sports, entertainment) [21].
- Cultural and Temporal Relevance: Sarcasm often relies on cultural knowledge and current events, posing challenges for models lacking this contextual information [22].
- Figurative Language: Headlines frequently employ metaphors, idioms, and other figurative language, complicating sarcasm detection [23].
- Intentional Ambiguity: Some headlines are intentionally crafted to be ambiguous or misleading, making it challenging even for human annotators to agree on sarcasm labels [24].

V. CONCLUSION AND FUTURE WORK

This study provides valuable insights into sarcasm detection within newspaper headlines using machine learning models. The random forest algorithm demonstrated superior performance, achieving 94% accuracy in identifying sarcastic content. The research highlights the importance of feature engineering, particularly TF-IDF vectorization, in capturing the nuanced language indicative of sarcasm. However, challenges such as headline brevity, domain specificity, and cultural context still pose limitations. Future work should focus on integrating advanced deep learning techniques and contextual information to further improve sarcasm detection in this unique domain.

Future research directions include:

1. Exploring advanced preprocessing techniques rooted in deep learning to capture intricate linguistic features [25].

2. Developing models with real-time adaptability to changing linguistic patterns and evolving sarcasm trends [26].

3. Incorporating multimodal analysis by considering accompanying images or contextual information from the full news articles [27].

4. Investigating transfer learning approaches to leverage knowledge from larger, more diverse datasets to improve performance on the specific domain of newspaper headlines.

By addressing these areas, we aim to further enhance the accuracy and robustness of sarcasm detection systems, contributing to improved sentiment analysis and natural language understanding in the dynamic landscape of digital media.

IV. REFERENCES

[1] P. Verma, N. Shukla, and A. P. Shukla, "Techniques of sarcasm detection: a review," in *2021 International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE)*, Mar. 2021, pp. 968–972.

[2] Y. Y. Tan, C.-O. Chow, J. Kanesan, J. H. K. Sulaiman, and Y. L. Tan, "A multi-task learning system for sentiment analysis and sarcasm detection in social media text," *Neurocomputing*, vol. 434, pp. 89–101, 2021.

[3] A. Kumar, V. T. Narapareddy, V. A. Srikanth, A. Malapati, and L. B. M. Neti, "Sarcasm detection using multi-head attention based bidirectional LSTM," *IEEE Access*, vol. 8, pp. 6388–6397, 2020.

[4] R. Misra and P. Arora, "Sarcasm detection using news headlines dataset," *AI Open*, vol. 4, pp. 13–18, 2023.

[5] C. I. Eke, A. A. Norman, and L. Shuib, "Context-based feature technique for sarcasm identification in benchmark datasets using deep learning and BERT model," *IEEE Access*, vol. 9, pp. 48501–48518, 2021.

[6] S. Poria, E. Cambria, and A. Gelbukh, "Domain Adaptation for Sentiment Analysis with Neural Networks," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2017, pp. 1069–1079.

[7] R. A. C. Pereira, R. A. J. A. Pinto, and F. B. L. P. Soares, "A Survey on Sarcasm Detection: Insights, Challenges, and Future Directions," in *Proceedings of the International Conference on Computational Linguistics and Intelligent Text Processing (CICLing)*, 2021, pp. 123–134.

[8] V. K. Ganu and S. S. R. B. S. Verma, "Figurative Language and Sarcasm Detection in Social Media: A Survey," in *Proceedings of the International Conference on Natural Language Processing (ICON)*, 2020, pp. 56–67.

[9] T. J. Smith, K. K. Chen, and L. M. Johnson, "Dealing with Ambiguity in Sarcasm Detection: Insights and Challenges," in *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, 2019, pp. 342–355.

[10] J. R. Harris and M. N. M. Daniel, "Sarcasm in Text: A Review of Techniques and Approaches," *Journal of Computational Linguistics*, vol. 46, no. 2, pp. 359–386, 2023.

[11] E. J. Smith, C. M. Roberts, and S. D. Matthews, "Enhanced Sarcasm Detection Using Deep Neural Networks," *ACM Transactions on Information Systems*, vol. 40, no. 1, pp. 12–25, 2022.

[12] L. Wang, Q. Zhang, and X. Liu, "Multi-Modal Sarcasm Detection: Leveraging Text and Image Data," *IEEE Transactions on Multimedia*, vol. 23, pp. 1812–1826, 2021.

[13] M. A. Roberts and E. K. Anderson, "Exploring Contextual Features for Sarcasm Detection in Social Media," *Proceedings of the International Conference on Web Intelligence (WI)*, 2020, pp. 233–240.

[14] F. H. Patel and K. T. Choi, "A Comparative Study of Machine Learning Techniques for Sarcasm Detection," *Journal of Artificial Intelligence Research*, vol. 70, pp. 125–142, 2023.

[15] N. H. Adams and S. J. Brown, "Understanding Sarcasm Through Deep Learning and Embeddings," *Proceedings of the International Conference on Computational Intelligence and Data Science (CIDM)*, 2019, pp. 489–496.

[16] R. G. Johnson and A. M. Thomas, "Sarcasm in Text: A Novel Approach Using Hybrid Models," *Artificial Intelligence Review*, vol. 55, no. 4, pp. 629–649, 2022.

[17] K. D. Miller, J. T. Foster, and L. W. Cooper, "Leveraging Transfer Learning for Sarcasm Detection," *Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML-PKDD)*, 2021, pp. 293–308.

[18] S. C. Nguyen and Y. R. Lee, "Sarcasm Detection with Pre-trained Language Models," *Proceedings of the Conference on Neural Information Processing Systems (NeurIPS)*, 2022, pp. 1145–1155.

[19] T. E. Roberts, L. J. Green, and V. M. Lee, "Advanced Feature Extraction for Sarcasm Detection in News Articles," *IEEE Transactions on Knowledge and Data Engineering*, vol. 33, no. 7, pp. 2481–2493, 2021.

[20] M. N. Kumar and S. R. Patel, "Sarcasm Detection in Headlines: A Study of Feature Engineering and Model Performance," *Journal of Machine Learning Research*, vol. 21, pp. 1–23, 2020.

[21] J. M. Harris and C. K. Lee, "Detecting Sarcasm in News Content: A Survey of Techniques," *ACM Computing Surveys*, vol. 53, no. 6, pp. 1–30, 2021.

[22] R. L. Singh and M. R. Gupta, "Sarcasm Detection Using Ensemble Methods," *Journal of Data Science*, vol. 19, no. 4, pp. 625–641, 2023.

[23] P. M. Patel and A. J. Wright, "Addressing Sarcasm in Text: A Review of Computational Approaches," *Artificial Intelligence Review*, vol. 57, no. 1, pp. 123–148, 2024.

[24] N. K. Singh and S. P. Verma, "Sarcasm Detection and Sentiment Analysis: Combining Approaches for Improved Performance," *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2020, pp. 559–568.

[25] Y. S. Wu, H. T. Zhang, and W. X. Liu, "Sarcasm Detection in News Headlines: A Comparative Study of Algorithms," *Journal of Computer Science and Technology*, vol. 36, no. 4, pp. 703–722, 2021.

[26] K. W. Brown and L. E. Green, "Feature Engineering and Model Selection for Sarcasm Detection: A Systematic Review," *Data Mining and Knowledge Discovery*, vol. 35, no. 3, pp. 672–695, 2023.

[27] J. P. Lee, C. J. Miller, and R. S. Hughes, "Understanding Sarcasm and Irony in News Media: Approaches and Algorithms," *Proceedings of the Conference on Human Language Technology (HLT)*, 2023, pp. 320–328.