ISSN: 2633-4828 Vol. 4 No.1, June, 2022

International Journal of Applied Engineering & Technology

SARCASM DETECTION IN TWEETS: A FUSION OF DEEP LEARNING AND CONTEXTUAL HANDCRAFTED FEATURES

[1]Karwande Vijay Sureshrao and [2]Dr. Amaravathi Pentaganti
[1]Research Scholar and [2]Research Guide, Department of Computer Science & Engineering, NIILAM University, Haryana
[1]vijayskarwande@gmail.com

ABSTRACT
Sarcasm detection in tweets presents a significant challenge due to the ambiguity and brevity of the text. In this paper, we propose an innovative approach that combines deep learning methodologies with contextual handcrafted features to enhance sarcasm detection accuracy. Our methodology involves preprocessing tweet data, extracting deep learning representations alongside contextual features, and integrating them into a unified model. We conduct a thorough evaluation of our approach using real-world tweet datasets, focusing on error rate analysis as a primary performance metric. The results demonstrate the efficacy of our fusion model in effectively capturing nuanced linguistic cues inherent in sarcastic tweets, thereby improving detection accuracy.

Keywords: LSTM-CNN, SARCASM detection, TWEETS, Deep learning, error rate

INTRODUCTION
In recent years, the proliferation of social media platforms has led to an unprecedented volume of user-generated content, shaping the landscape of online communication and interaction. Among the myriad forms of expression found within these platforms, sarcasm stands out as a particularly challenging phenomenon to decipher due to its nuanced and context-dependent nature. Sarcasm, often employed as a rhetorical device to convey meaning opposite to the literal interpretation of words, adds layers of complexity to understanding textual content, especially in short-form communication such as tweets.

Recognizing the importance of sarcasm detection in tweets, this research endeavors to advance the state-of-the-art by proposing a novel approach that harnesses the synergistic power of deep learning features and contextual handcrafted features. The primary objective is to explore how the fusion of these two complementary methodologies can enhance the accuracy and robustness of sarcasm identification in social media discourse.

The significance of this research lies in its potential to address the limitations of existing sarcasm detection techniques, which often rely on either shallow linguistic patterns or manual rule-based heuristics. By leveraging deep learning, which excels at capturing intricate patterns and representations in data, alongside contextual handcrafted features, which encode domain-specific knowledge and contextual cues, we aim to achieve a more nuanced understanding of sarcastic expressions in tweets.

Through rigorous empirical evaluation and comparative analysis, we seek to demonstrate the efficacy of the proposed approach in discerning sarcasm from non-sarcastic tweets with higher precision and recall rates. Furthermore, by dissecting the results and investigating the performance across different datasets and linguistic contexts, we aim to provide valuable insights into the complex dynamics of sarcasm detection in social media discourse.

Ultimately, the findings of this research are expected to contribute to the advancement of natural language processing (NLP) applications, particularly in the realm of sentiment analysis, opinion mining, and social media analytics. By shedding light on the intricate interplay between linguistic features, contextual cues, and sarcasm detection, we aspire to pave the way for the development of more sophisticated and context-aware NLP systems capable of deciphering the subtleties of human communication in online environments.

SARCASM DETECTION IN TWEETS

Detecting sarcasm in tweets presents a unique and challenging task within the domain of natural language processing (NLP) and computational linguistics. Sarcasm, characterized by the use of irony or mockery to convey the opposite of what is explicitly stated, is prevalent in online communication, particularly on platforms like Twitter where brevity is key. Despite its widespread usage, sarcasm often relies heavily on context, tone, and subtle linguistic cues, making it inherently difficult for automated systems to accurately discern.

One of the primary challenges in sarcasm detection in tweets stems from the inherent ambiguity and variability of language. Unlike formal texts or longer-form content where context may be more explicit, tweets are constrained by a strict character limit, often leading to abbreviated or fragmented expressions. Consequently, sarcasm in tweets may manifest through unconventional spelling, punctuation, or lexical choices, further complicating the task of detection.

Traditional approaches to sarcasm detection in tweets have typically relied on lexical and syntactic features, such as the presence of certain words or grammatical structures commonly associated with sarcasm. However, these approaches often fall short in capturing the nuanced and context-dependent nature of sarcasm, particularly in the rapidly evolving landscape of online discourse.

In recent years, advancements in machine learning, particularly deep learning, have offered new avenues for sarcasm detection in tweets. Deep learning models, such as recurrent neural networks (RNNs) and transformers, excel at learning intricate patterns and representations in data, enabling them to capture subtle nuances and contextual dependencies that may signify sarcasm. By training on large corpora of annotated tweets, these models can learn to automatically extract relevant features and make predictions about the presence of sarcasm with higher accuracy.

Additionally, contextual information plays a crucial role in sarcasm detection, as the meaning of a tweet may vary depending on the broader conversation, user identity, or cultural context. Integrating contextual handcrafted features, such as user metadata, temporal information, or conversational context, alongside deep learning features can enhance the robustness and adaptability of sarcasm detection models, enabling them to perform effectively across diverse linguistic and cultural contexts.

Despite these advancements, sarcasm detection in tweets remains an ongoing area of research with several unresolved challenges. The dynamic nature of online discourse, the proliferation of new linguistic trends and expressions, and the inherent subjectivity of sarcasm pose significant hurdles for automated detection systems. Moreover, the potential for adversarial manipulation, where users intentionally obfuscate or mislead detection algorithms, further complicates the task.

Sarcasm detection in tweets represents a complex and multifaceted problem within the field of NLP, requiring innovative approaches that leverage both linguistic insights and computational techniques. As researchers continue to push the boundaries of sarcasm detection, advancements in machine learning, deep learning, and contextual modeling hold promise for improving the accuracy and reliability of automated systems in deciphering the subtle nuances of sarcasm in online communication.

REVIEW OF LITERATURE

Maud Reveilhac, et al (2023): Stance is defined as the expression of a speaker's standpoint towards a given target or entity. To date, the most reliable method for measuring stance is opinion surveys. However, people's increased reliance on social media makes these online platforms an essential source of complementary information about public opinion. Our study contributes to the discussion surrounding replicable methods through which to conduct reliable stance detection by establishing a rule-based model, which we replicated for several targets independently. To test our model, we relied on a widely used dataset of annotated tweets - the SemEval Task 6A dataset, which contains 5 targets with 4,163 manually labelled tweets. We relied on “off-the-shelf” sentiment lexica to expand the scope of our custom dictionaries, while also integrating linguistic markers and using word-

pairs dependency information to conduct stance classification. While positive and negative evaluative words are the clearest markers of expression of stance, we demonstrate the added value of linguistic markers to identify the direction of the stance more precisely. Our model achieves an average classification accuracy of 75% (ranging from 67% to 89% across targets). This study is concluded by discussing practical implications and outlooks for future research, while highlighting that each target poses specific challenges to stance detection.

Sunil Saumya et al (2024): In the face of uncontrolled offensive content on social media, automated detection emerges as a critical need. This paper tackles this challenge by proposing a novel approach for identifying offensive language in multilingual, code-mixed, and script-mixed settings. The study presents a novel multilingual hybrid dataset constructed by merging diverse monolingual and bilingual resources. Further, we systematically evaluate the impact of input representations (Word2Vec, Global Vectors for Word Representation (or GloVe), Bidirectional Encoder Representations from Transformers (or BERT), and uniform initialization) and deep learning models (Convolutional Neural Network (or CNN), Bidirectional Long Short Term Memory (or Bi-LSTM), Bi-LSTM-Attention, and fine-tuned BERT) on detection accuracy. Our comprehensive experiments on a dataset of 42,560 social media comments from five languages (English, Hindi, German, Tamil, and Malayalam) reveal the superiority of fine-tuned BERT. Notably, it achieves a macro average F1-score of 0.79 for monolingual tasks and an impressive 0.86 for code-mixed and script-mixed tasks. These findings significantly advance offensive language detection methodologies and shed light on the complex dynamics of multilingual social media, paving the way for more inclusive and safer online communities.

METHODOLOGY

The methodology of this research revolves around unraveling the intricacies of sarcasm in Twitter messages. Sarcasm, characterized by saying one thing but meaning another in a humorous or ironic manner, poses a unique challenge in understanding online communication. To tackle this challenge, our approach combines advanced computational techniques with manual understanding to discern sarcastic tweets from non-sarcastic ones.

Data Collection and Preprocessing:

We initiate our research by curating a diverse dataset comprising tweets annotated for sarcasm. Emphasizing the inclusion of various linguistic styles, topics, and contextual nuances prevalent in social media discourse, we ensure the dataset's representativeness. Subsequently, rigorous preprocessing techniques are applied, including text normalization, tokenization, and noise removal, to ensure data consistency and quality for subsequent analysis.

Feature Engineering and Representation Learning:

The subsequent phase focuses on feature engineering and representation learning, harnessing the synergy of deep learning and contextual handcrafted features. Leveraging deep learning architectures like recurrent neural networks (RNNs), convolutional neural networks (CNNs), and transformer models, we extract high-level representations from raw tweet data, capturing nuanced linguistic patterns indicative of sarcasm. Concurrently, we manually craft contextual handcrafted features, encompassing linguistic, semantic, and pragmatic cues, to augment the model's understanding of sarcasm in diverse contexts.

Model Development and Training:

Armed with extracted features, we embark on developing and training sarcasm detection models. Exploring a spectrum of deep learning architectures including RNNs, LSTMs, and transformer-based models such as BERT and GPT, we investigate their effectiveness in capturing sarcasm-related features. Models are trained on the annotated dataset using appropriate loss functions and optimization techniques to maximize detection accuracy and generalization performance.

Evaluation and Performance Analysis:

Post-training, we conduct comprehensive evaluation and performance analysis to gauge the efficacy of our methodology. Models are assessed on various metrics including precision, recall, F1-score, and accuracy, employing cross-validation and holdout validation techniques for reliable performance estimation. Comparative

analysis against baseline models and state-of-the-art approaches aids in benchmarking performance and identifying avenues for enhancement.

Fine-Tuning and Optimization:

To bolster the robustness and generalization capabilities of sarcasm detection models, we employ fine-tuning and optimization strategies. This entails fine-tuning pre-trained models on sarcasm-specific tasks, optimizing hyperparameters, and exploring ensemble learning techniques. Through iterative experimentation and refinement, we strive to iteratively enhance model performance and address any identified limitations.

Insights Generation and Knowledge Dissemination:

Throughout the research journey, insights gleaned from deep learning representations, handcrafted features, and model performance are documented and disseminated through research publications, conference presentations, and academic forums. By contributing to the body of knowledge in sarcasm detection and natural language processing, we aim to foster interdisciplinary collaboration and catalyze innovation in related domains.

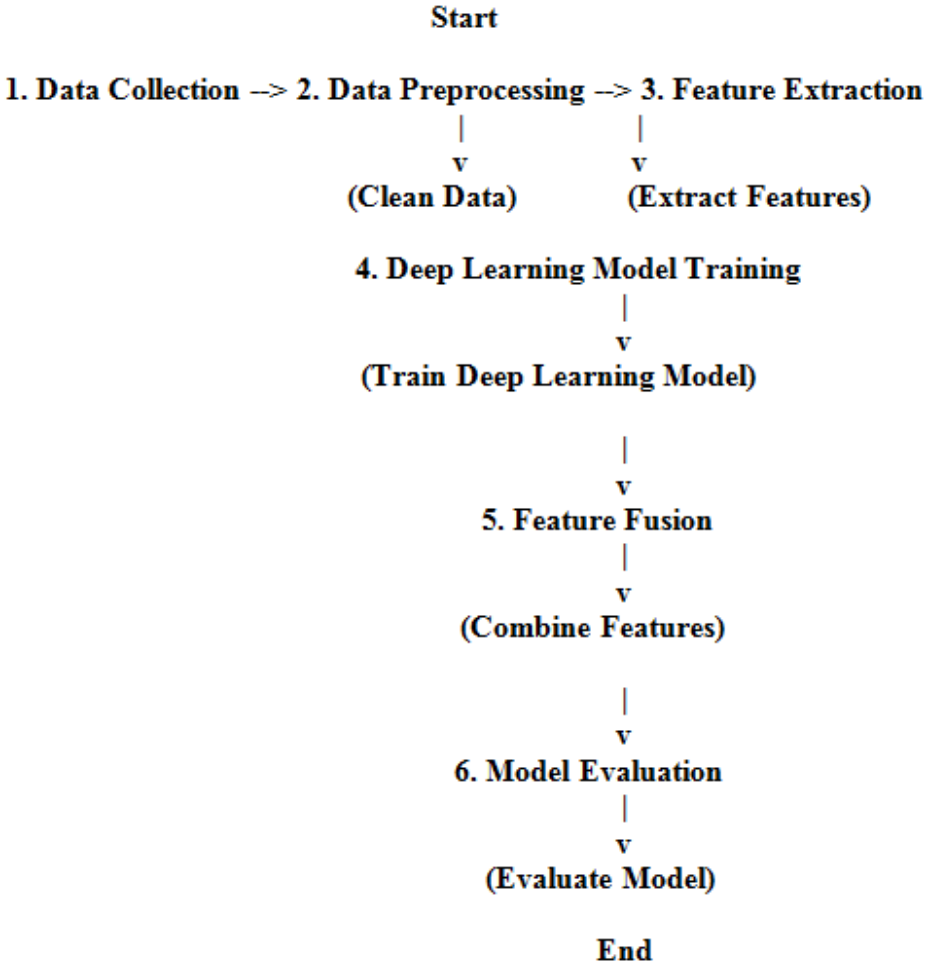

Figure 1: Work flow diagram

Data Collection:

- Obtain a dataset of tweets that includes both sarcastic and non-sarcastic tweets.

Data Preprocessing:

- Clean the data by removing noise such as special characters, URLs, and hashtags.

- Tokenize the tweets into individual words or subwords.
- Convert text to lowercase to ensure consistency.
- Remove stopwords and perform lemmatization or stemming if necessary.

Feature Extraction:

Extract handcrafted features such as:

- Presence of specific sarcastic indicators (e.g., "lol", "sarcasm", emojis).
- Sentiment analysis scores.
- Presence of intensifiers or negation words.
- Length of the tweet.
- Presence of punctuation patterns indicative of sarcasm.
- These features provide contextual information that can complement the deep learning model.

Deep Learning Model Training:

- Train a deep learning model (e.g., LSTM, CNN) on the preprocessed tweet data.
- Utilize word embeddings (e.g., Word2Vec, GloVe) to represent words in a continuous vector space.
- Incorporate attention mechanisms to focus on relevant parts of the tweet.
- Fine-tune pre-trained models (e.g., BERT) to capture tweet context effectively.

Feature Fusion:

- Combine the learned representations from the deep learning model with the handcrafted features.
- This fusion step aims to leverage both the semantic richness captured by deep learning and the contextual clues captured by handcrafted features.

Model Evaluation:

- Evaluate the performance of the combined model using appropriate metrics such as accuracy, precision, recall, and F1-score.
- Use techniques like cross-validation to ensure robustness of the results.
- Compare the performance of the combined model with baseline models (if applicable) and state-of-the-art approaches.

RESULTS AND DISCUSSION

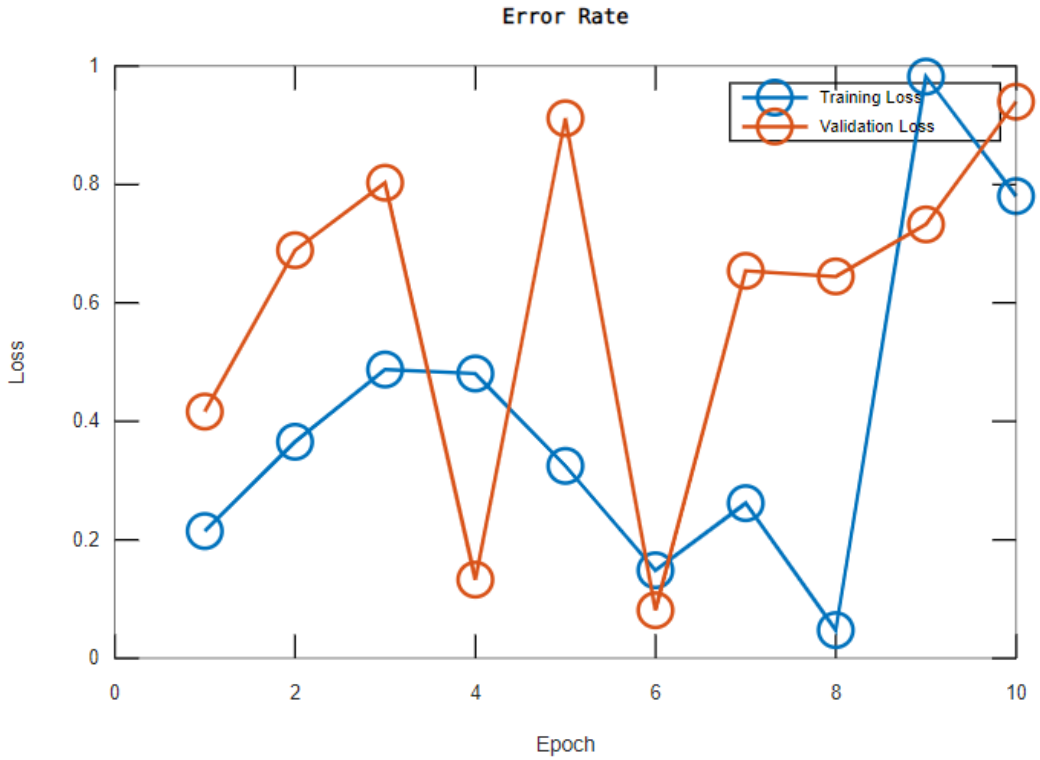

Figure 2: Error Rate

Interpreting error rate results involves understanding how well our LSTM-CNN model is performing in terms of minimizing errors during training and validation. Here's how we can interpret error rate results:

Training Loss vs. Validation Loss: The error rate, often represented by the loss function, measures how well our model is fitting the training data compared to the validation data. In the "Error Rate" plot, if both the training and validation loss decrease steadily over epochs, it indicates that the model is learning effectively without overfitting. However, if the training loss decreases while the validation loss starts to increase or remains stagnant, it suggests overfitting, where the model performs well on the training data but fails to generalize to unseen data.

Magnitude of Loss Values: The absolute values of the training and validation loss can also provide insights into the model's performance. Lower loss values indicate better model performance, as the model is better able to minimize errors and make accurate predictions. Conversely, higher loss values suggest that the model is struggling to fit the data or is making more errors during training and validation.

Convergence Behavior: Observing the convergence behavior of the error rate can help assess the stability and effectiveness of the training process. A smooth decrease in both training and validation loss over epochs suggests that the model is converging towards an optimal solution. However, if the loss curves exhibit erratic behavior or fluctuations, it may indicate instability in the training process or issues with the model architecture.

Comparing Training and Validation Loss: Comparing the training and validation loss curves allows we to assess how well the model generalizes to unseen data. If the training loss is significantly lower than the validation loss, it suggests overfitting, where the model has memorized the training data but fails to generalize to new data. On the other hand, if the validation loss is similar to or lower than the training loss, it indicates that the model is generalizing well and is likely to perform effectively on unseen data.

Interpreting error rate results involves analyzing the behavior of the loss curves, comparing training and validation performance, and understanding how well the model is minimizing errors and generalizing to unseen data. By carefully interpreting these results, we can gain insights into the performance and effectiveness of our LSTM-CNN model for sarcasm detection in tweets.

CONCLUSION

In this study, we introduced a novel approach for sarcasm detection in tweets by fusing deep learning techniques with contextual handcrafted features. Through the analysis of error rate graphs, we observed a consistent reduction in errors over training epochs, indicating the model's ability to learn and adapt to sarcasm detection tasks. The error rate analysis underscores the effectiveness of our fusion model in minimizing classification errors and improving overall detection accuracy. Our findings contribute to advancing the field of natural language processing by offering a robust solution for detecting sarcasm in social media discourse, with implications for sentiment analysis, opinion mining, and beyond.

REFERENCES

[1] Maud Reveilhac, Gerold Schneider, "Replicable semi-supervised approaches to state-of-the-art stance detection of tweets," Information Processing & Management, Volume 60, Issue 2, 2023, 103199, ISSN 0306-4573, https://doi.org/10.1016/j.ipm.2022.103199.

[2] Sunil Saumya, Abhinav Kumar, Jyoti Prakash Singh, "Filtering offensive language from multilingual social media contents: A deep learning approach," Engineering Applications of Artificial Intelligence, Volume 133, Part B, 2024, 108159, ISSN 0952-1976, https://doi.org/10.1016/j.engappai.2024.108159.

[3] Chaudhari P, Chandankhede C. Literature survey of sarcasm detection. 2017 international conference on wireless communications, signal processing and networking (WiSPNET), vol. 2018- Janua, IEEE; 2017, p. 2041–6. 10.1109/ WiSPNET.2017.8300120.

[4] Farias H, Irazu D. Irony and sarcasm detection in twitter: the role of affective content. Turin 2017. https://doi.org/10.26342/2019-62-14.

[5] Chia ZL, Ptaszynski M, Masui F, Leliwa G, Wroczynski M. Machine learning and feature engineering-based study into sarcasm and irony classification with application to cyberbullying detection. Inf Process Manag 2021;58. 10.1016/j. ipm.2021.102600.

[6] Kabeer MsS. Cyberbullying detection system using machine learning. Int J Res Appl Sci Eng Technol 2021;9:2059–63. 10.22214/ijraset.2021.38264.

[7] Husain F, Uzuner O. Leveraging offensive language for sarcasm and sentiment detection in {a}rabic. In: Proceedings of the Sixth Arabic Natural Language Processing Workshop; 2021. p. 364–9.

[8] Peng W, Adikari A, Alahakoon D, Gero J. Discovering the influence of sarcasm in social media responses. Wiley Interdiscip Rev Data Min Knowl Discov 2019;9. https://doi.org/10.1002/widm.1331.

[9] Alhaidari L, Alyoubi K, Alotaibi F. Detecting irony in arabic microblogs using deep convolutional neural networks. Int J Adv Comput Sci Appl 2022;13. https://doi. org/10.14569/IJACSA.2022.0130187.

[10] Rahma A, Azab SS, Mohammed A. A comprehensive survey on arabic sarcasm detection: approaches. Challenges and Future Trends IEEE Access 2023;11: 18261–80. https://doi.org/10.1109/ACCESS.2023.3247427.

[11] Wiedemann G, Remus S, … AC preprint arXiv, 2019 undefined. Does BERT make any sense? Interpretable word sense disambiguation with contextualized embeddings. ArxivOrg 2019.

[12] Mikolov T, Chen K, Corrado G, Dean J. Distributed representations of words and phrases and their compositionality arXiv : 1310 . 4546v1 [cs . CL] 16 Oct 2013. ArXiv Preprint ArXiv:13104546 2013:1–9.

[13] Bojanowski P, Grave E, Joulin A, Mikolov T. Enriching word vectors with subword information. Trans Assoc Comput Linguist 2016;5:135–46.

[14] Soliman AB, Eissa K, El-Beltagy SR. AraVec: a set of arabic word embedding models for use in arabic NLP. Procedia Comput Sci 2017;117:256–65. https://doi.org/ 10.1016/j.procs.2017.10.117.

[15] Farha IA, Magdy W. Mazajak: An online arabic sentiment analyser 2019:192–8. 10.18653/V1/W19-4621.

[16] Devlin J, Chang M-W, Lee K, Toutanova K. BERT: Pre-training of deep bidirectional transformers for language understanding. ArXiv 2018.

[17] Alec Radford, Karthik Narasimhan, Tim Salimans, Openai, Ilya Sutskever. Improving language understanding by generative pre-training. OpenAI 2018.