HITU 12

SarcasmDetection_Edited_03-11-23.pdf

PAPERS

Document Details

Submission ID

trn:oid:::1:2740421255

Submission Date

Nov 5, 2023, 9:25 PM GMT+5:30

Download Date

Nov 5, 2023, 9:40 PM GMT+5:30

File Name

SarcasmDetection_Edited_03-11-23.pdf

File Size

1.1 MB

14 Pages

5,590 Words

33,269 Characters

How much of this submission has been generated by AI?

24%

of qualifying text in this submission has been determined to be generated by AI.

Caution: Percentage may not indicate academic misconduct. Review required.

It is essential to understand the limitations of AI detection before making decisions about a student's work. We encourage you to learn more about Turnitin's AI detection capabilities before using the tool.

Frequently Asked Questions

What does the percentage mean?

The percentage shown in the AI writing detection indicator and in the AI writing report is the amount of qualifying text within the submission that Turnitin's AI writing detection model determines was generated by AI.

Our testing has found that there is a higher incidence of false positives when the percentage is less than 20. In order to reduce the likelihood of misinterpretation, the AI indicator will display an asterisk for percentages less than 20 to call attention to the fact that the score is less reliable.

However, the final decision on whether any misconduct has occurred rests with the reviewer/instructor. They should use the percentage as a means to start a formative conversation with their student and/or use it to examine the submitted assignment in greater detail according to their school's policies.

How does Turnitin's indicator address false positives?

Our model only processes qualifying text in the form of long-form writing. Long-form writing means individual sentences contained in paragraphs that make up a longer piece of written work, such as an essay, a dissertation, or an article, etc. Qualifying text that has been determined to be AI-generated will be highlighted blue on the submission text.

Non-qualifying text, such as bullet points, annotated bibliographies, etc., will not be processed and can create disparity between the submission highlights and the percentage shown.

What does 'qualifying text' mean?

Sometimes false positives (incorrectly flagging human-written text as AI-generated), can include lists without a lot of structural variation, text that literally repeats itself, or text that has been paraphrased without developing new ideas. If our indicator shows a higher amount of AI writing in such text, we advise you to take that into consideration when looking at the percentage indicated.

In a longer document with a mix of authentic writing and AI generated text, it can be difficult to exactly determine where the AI writing begins and original writing ends, but our model should give you a reliable guide to start conversations with the submitting student.

Disclaimer

Our AI writing assessment is designed to help educators identify text that might be prepared by a generative AI tool. Our AI writing assessment may not always be accurate (it may misidentify both human and AI-generated text) so it should not be used as the sole basis for adverse actions against a student. It takes further scrutiny and human judgment in conjunction with an organization's application of its specific academic policies to determine whether any academic misconduct has occurred.

Exploring Machine Learning Models for Sarcasm Detection on Twitter: A Comparative Study

Shrishti Sharma
Department of Electronics and Communication Engineering
Indira Gandhi Delhi Technical University for Women
New Delhi, India
shrishti120bteceai22@igdtuw.ac.in

Sripriya Agarwal
Department of Electronics and Communication Engineering
Indira Gandhi Delhi Technical University for Women
New Delhi, India
sripriya139bteceai22@igdtuw.ac.in

Tavleen Kaur
Department of Electronics and Communication Engineering
Indira Gandhi Delhi Technical University for Women
New Delhi, India
tavleen148bteceai22@igdtuw.ac.in

line 1: 4th Given Name Surname
line 2: *dept. name of organization (of Affiliation)*
line 3: *name of organization (of Affiliation)*
line 4: City, Country
line 5: email address or ORCId

***Abstract*-In the contemporary digital world, characterized by the widespread impact of social media platforms such as Twitter, the precise identification of sarcasm has emerged as a pressing and complex issue. Sarcasm, often deeply integrated into the essence of language, can be difficult to identify yet carries significant complications. Its impact extends apart from simple sentiment analysis. It can shape public opinion, influence decision-making, and affect the general atmosphere of online conversations. This research paper addresses sarcasm detection through a thorough investigation and comparative analysis of various models. Different approaches, which are Random Forest, XG Boost, and LSTM-based transformers, have been explored, each representing different approaches to address this complex issue. This research begins with data collection followed by preprocessing of a diverse dataset of Twitter comments classified as Sarcastic and Non-sarcastic. Then, through rigorous evaluation and experimentation, the advantages and drawbacks of each model are identified. The study aims to shed light on the suitability of these models for sarcasm detection tasks across varying data characteristics by subjecting them to testing. These findings are positioned to provide substantial insights into sarcasm detection. They offer valuable insights into in-depth interaction among the nuances between model performance and the specific characteristics of the data they encounter. In particular, our research highlights the promising potential of a proposed model: the ADASYN- TF-IDF - LSTM-based transformers. These state-of-the-art models demonstrate a remarkable level of refinement in handling the complexities of Twitter sarcasm, offering a more effective and accurate detection in the future. In conclusion, this research strives to enhance the comprehension of sarcasm detection within the prevailing social media era context. It aims to empower researchers and professionals with the knowledge and tools necessary to navigate the ever-evolving landscape of online communication, ultimately contributing to more insightful and informed analyses of digital discourse.**

Keywords— sarcasm, sarcasm detection, bagging, boosting, Twitter, tweets, transformers, ADASYN, data balancing

I. Introduction

In an era governed by the omnipresent influence of social media platforms, interpreting textual content transcends conventional boundaries, introducing novel challenges and opportunities. Among these challenges, detecting sarcasm is a significant problem, holding profound implications for sentiment analysis, public opinion assessment, and human-computer interaction. The complexity of sarcasm, characterized by its divergence from literal meaning, presents an enduring problem for automated systems in the digital age. This research paper offers a detailed exploration of the sarcasm detection problem within the dynamic landscape of Twitter.

The two main types of sarcasm detection are:

A. Sentiment-based Sarcasm Detection: This approach focuses on analyzing a text's sentiment or emotional tone to detect sarcasm and is often called "Sentiment-based Sarcasm Detection." It looks for variances between the expressed sentiment and the text's meaning. For example, if a statement appears positive in sentiment but is meant sarcastically, the model would identify it as sarcasm.

B. Pattern-based Sarcasm Detection: Pattern-based detection depends on recognizing particular language cues, patterns, or features commonly associated with sarcasm. This approach doesn't necessarily focus on sentiment while searching for distinctive word combinations, negation, or incongruity between words and context to flag potential sarcasm.

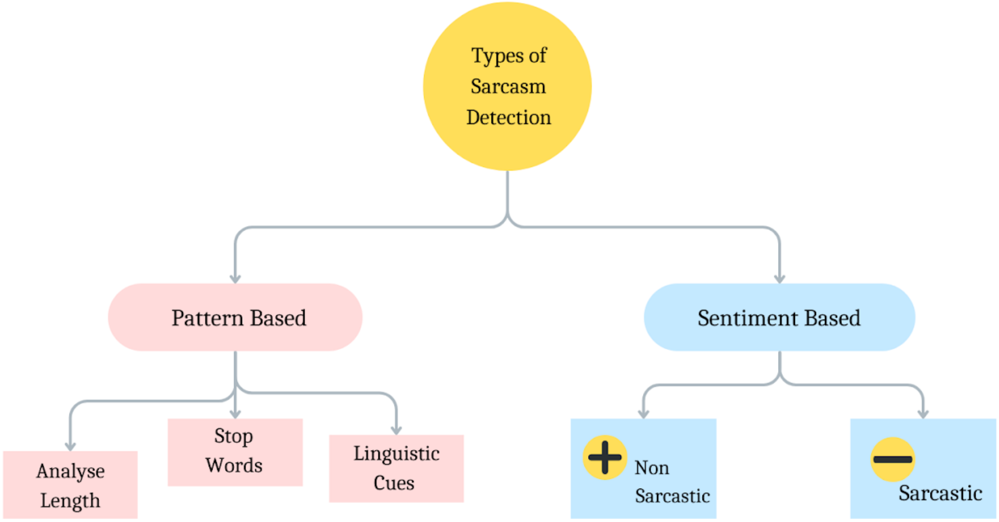

Fig. 1. Types of sarcasm detection techniques

It scrutinizes the efficacy of diverse machine learning paradigms, encompassing ensemble-based models like Random Forest, gradient boosting algorithms as exemplified by XGBoost, and state-of-the-art transformer-based architectures, particularly LSTM networks. The central objective of this study is to elucidate the unique strengths and limitations while presenting a novel perspective on sarcasm detection within this intricate context. The vitality of this research endeavor lies in its contribution to the field of sarcasm detection. It offers nuanced insights into selecting optimal techniques, informed by the contextual considerations defining the Twitter ecosystem. The comparative analysis of these models sheds light on their performance and serves as a compass for practitioners seeking to navigate the complexities of sarcasm identification.

As the subsequent sections unfold, this paper will delve into the intricacies of dataset acquisition and preprocessing, detail the methodologies employed, present the experimental results, and discuss the implications of our findings. Within this journey, it becomes evident that the dynamic nature of sarcasm, intertwined with the rapid evolution of online discourse, necessitates a refined approach to computational understanding. This approach informs our comprehension of language and shapes how we interact with digital communication. In an era where the lines between human communication and automated analysis blur continuously, the pursuit of accurate sarcasm detection emerges as an indispensable endeavor. By comparing diverse models and presenting a novel perspective on sarcasm detection, this research seeks to empower information consumers, enhance sentiment analysis, and make a lasting contribution to natural language processing.

Effects: It's important to note that while sarcasm on Twitter can be fascinating, it can also have serious consequences. The impact of misinterpreted or severe irony can lead to misunderstandings and conflicts and even contribute to a negative online atmosphere. The ratio of deaths caused by suicide determining the precise number of cases arising directly from online sarcasm or cyberbullying is challenging to measure precisely. The relationship between online interactions and mental health issues is complex, and it is crucial to promote online civility and empathy to reduce the possible damage caused by sarcasm and other negative online behaviors.

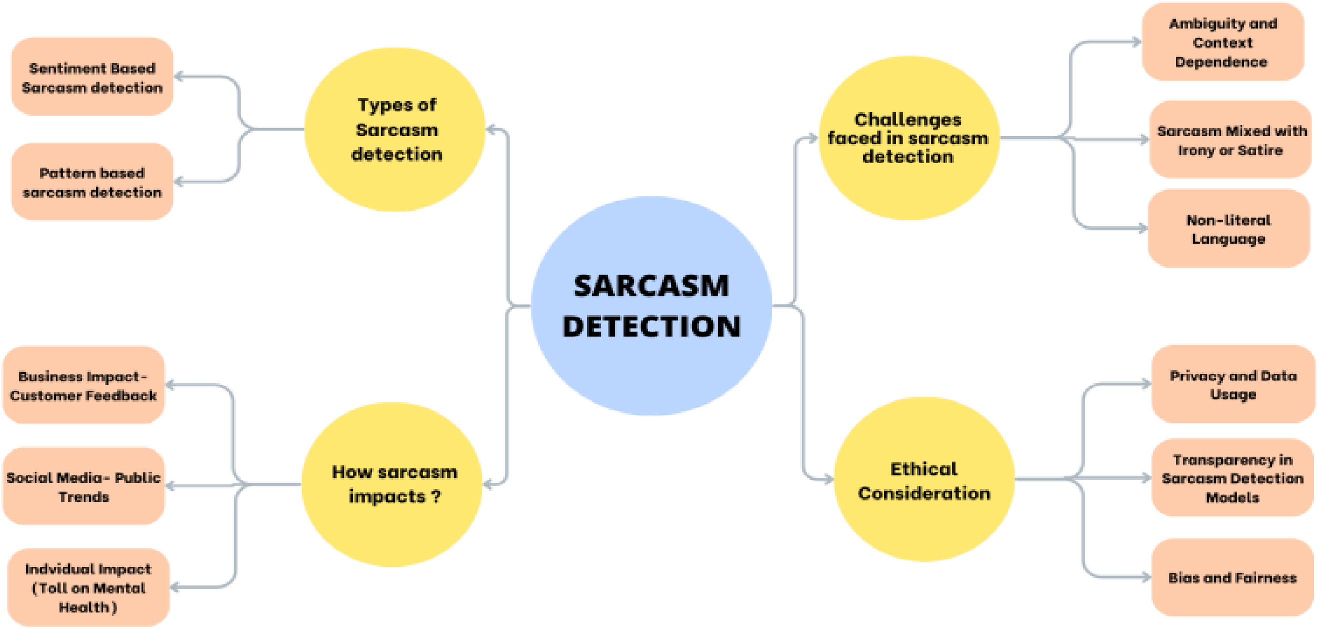
Sentiment Based Sarcasm detection
Pattern based sarcasm detection
Types of Sarcasm detection
SARCASM DETECTION
Challenges faced in sarcasm detection
Ambiguity and Context Dependence
Sarcasm Mixed with Irony or Satire
Non-literal Language
Business Impact- Customer Feedback
Social Media- Public Trends
Indvidual Impact (Toll on Mental Health)
How sarcasm impacts ?
Ethical Consideration
Privacy and Data Usage
Transparency in Sarcasm Detection Models
Bias and Fairness

Fig. 2. Sarcasm Detection Overview

II. Literature Review

In an era defined by the supremacy of social media, detecting sarcasm within platforms like Twitter has emerged as a problematic challenge that echoes profoundly within the domains of natural language processing and sentiment analysis. This literature review offers a comprehensive analysis of the progress made in the field of sarcasm detection.

The inception of sarcasm detection in social media discourse can be traced back to the pioneering work of Davidov et al. [1]. In their study, the authors introduced a semi-supervised approach for identifying sarcastic sentences in Twitter and Amazon reviews. Leveraging linguistic and contextual cues, this work spread the foundation for the following research, highlighting the feasibility of sarcasm identification in online text.

The examination of sarcasm detection unfolds with the work of Reyes et al. [2], who introduced a multidimensional approach on Twitter. Their scrutiny of linguistic, contextual, and behavioral features broadened the understanding of sarcasm detection by acknowledging the complex nature of sarcastic expressions.

Ptáček et al. [3] expanded the application of sarcasm detection to the Czech language, emphasizing the importance of cross-linguistic approaches. This analysis into diverse linguistic patterns highlighted the common relevance of sarcasm detection techniques.

Kreuz and Caucci [4] shed light on the influential role of word choices in shaping sarcasm comprehension. Their work emphasized language cues critical for finfing the subtle nuances of sarcastic expressions.

A pivotal milestone in the journey was the creation of self-annotated corpus, with Khodak et al. [5] contributing significantly. This development laid the groundwork for research, facilitating precise model comparisons and enhancing the understanding of sarcasm detection.

Continuing the exploration, Derczynski et al. [6] conducted a comprehensive analysis on Twitter, considering both behavioral and linguistic dimensions. Their work brought into focus the cues and contextual factors influencing sarcasm detection.

Liebrecht et al. [7] provided a novel perspective by exploring the contrast between positive sentiments and negative situations for sarcasm detection.

Mishra and Mourya [8] offered a comprehensive overview of methodologies and techniques in sarcasm detection, contributing to a retrospective understanding of the field's evolution.

In the midst of the exploration, Riloff et al. [9] directed attention to the realm of language patterns, aiming to extract insights for sentiment analysis and sarcasm detection. This pathway uncovered the potential sarcasm within the subtleties of language.

Gonçalves et al. [10] seamlessly integrated various sentiment analysis methods, proving the efficacy of their approach in sarcasm detection models. This convergence emphasized the interconnectedness of linguistic phenomena.

Buschmeier et al. [11] guided the examination of the effects of different feature types on irony and sarcasm classification. This exploration unveiled linguistic features serving as beacons, illuminating the way forward in identifying sarcasm and irony within context.

Bamman et al. [12] posed a crucial question in the expedition—whether sarcasm detection is more effectively approached as a global or local task. This inquiry prompted a reevaluation of strategies and a refined understanding of the dynamics at play in sarcasm detection.

In the final leg of the exploration, Joshi et al. [13] drew attention to the significance of text inconsistencies as valuable cues for sarcasm detection. Their insights seamlessly wove into the narrative, providing a nuanced perspective on identifying sarcasm within textual content.

Table I. Summary of Literature Review

PAPER ID	PUBLICATI-ON YEAR	DATA SET USED	APPROACH USED	CONCLUSION
RILOFF ET AL	2003	SELF-CRETED CORPUS	Extraction Pattern Learning	The paper proposes a method for identifying long subjective phrases in text, which can be useful for sarcasm detection by identifying patterns associated with subjectivity and sentiment.
KREUZ AND CAUCCI	2007	SELF-CRETED CORPUS	Lexical Influences	The study investigates how lexical cues influence sarcasm perception, shedding light on the linguistic aspects of identifying sarcasm.
DAVIDOV ET AL	2010	TWITTER AND AMAZON	Semisupervised Recognition	The paper presents a semi-supervised approach for recognizing sarcastic sentences in Twitter and Amazon. It describes the effectiveness of semi-supervised learning in improving sarcasm detection, especially when labeled data is limited.
BUSCHMEIER ET AL	2012	SELF-CRETED CORPUS	Feature-Based Classification	This research investigates the impact of various feature types on the classification of sarcasm, contributing to the understanding of which features are most effective for detection.
BAMMAN ET AL	2012	SELF-CRETED CORPUS	Local And Global Sarcasm Detection	The paper examines whether sarcasm detection in tweets is best approached as a local or global task, detecting sarcasm in short text messages
GONÇALVES ET AL	2013	SELF-CRETED CORPUS	Sentiment Analysis Methods Comparison And Combination	The paper explores different sentiment analysis methods, which are foundational for sarcasm detection, and discusses their strengths and weaknesses.
LIEBRECHT ET AL	2013	TWITTER DATA	Comparitive Anallyisis	The conclusions likely discuss the challenges and limitations in detecting sarcasm on Twitter rather than presenting a perfect solution.
DERCZYNSKI ET AL	2013	TWITTER DATA	Behavioral And Linguistic Analysis	The paper analyzes behavioral and linguistic markers of sarcasm in Twitter, providing insights into how sarcasm is expressed on the platform.
REYES ET AL.	2013	TWITTER	Multidimensional Approach	The research introduces a multidimensional approach to Twitter irony detection, emphasizing the significance of considering lexical, semantic, and pragmatic aspects for enhanced accuracy.
PTÁČEK ET AL	2014	CZECH TWITTER	Sarcasm Detection	The paper addresses sarcasm detection in Czech Twitter data, contributing to research beyond English-language contexts.
JOSHI ET AL	2015	SELF-CRETED CORPUS	Context Incongruity-Based Detection	Explores context incongruity as a method for sarcasm detection, the disparity between the expected and actual context to identify sarcastic statements.
KHODAK ET AL	2017	SELF ANNOTATED CORPUS	Corpus-Based Approach	The research presents a valuable self-annotated sarcasm detection corpus, emphasizing its potential for training models and driving future research in the field
MISHRA AND MOURYA	2019	SELF-CRETED CORPUS	Survey	Summarizes key findings, trends, and challenges in the field of sarcasm detection, serving as a valuable resource for researchers and practitioners.

III. Methodology

This section outlines the method used to address the primary issue of detecting sarcasm in Twitter data. It explains the methodical strategy employed to address the difficulties of identifying sarcastic remarks in the vast realm of online conversations. The discussion primarily focuses on the dataset utilized, how it was selected and prepared, and the essential consideration of managing class imbalances. Furthermore, it delves into the ensemble methods such as Random Forest and boosting models like XGBoost, in tandem with the transformative power of LSTM, which constitute the arsenal of techniques harnessed for sarcasm detection.

A. Dataset Description

Twitter, as a prolific source of real-time, user-generated content, stands as an invaluable resource for showcasing the real-world impact of sarcasm detection models. In the era of digital communication, Twitter stands as a lively representation of a diverse range of feelings, perspectives, and interactions. Its concise format and instant nature symbolize the rapid dissemination of information., making it an impactful platform to illustrate the real-world implications of accurate sarcasm detection. By analyzing Twitter data, this study not only pushes the boundaries of natural language processing but also highlights its importance in maneuvering the complexities of contemporary digital discussions.. For this research, we utilized a Kaggle dataset, which featured 4,500 entries, including 3,500 non-sarcastic and 1,000 sarcastic expressions. The dataset's source, Kaggle, is celebrated for its repository of diverse datasets and data-driven challenges. The data consists of integral labels, 0(non-sarcastic) and 1(sarcastic).

B. Data Balancing and normalization

In the quest for diverse and comprehensive data., we've utilized the Adaptive Synthetic Sampling (ADASYN) technique. This resampling approach, a vital element of data preparation, serves as a key factor in addressing the issue of class imbalance. In this research, we observed a noticeable class imbalance, with a significant majority of non-sarcastic expressions compared to the fewer instances of sarcastic ones. ADASYN is specifically designed to dynamically generate synthetic data points within the minority class to rectify this imbalance.

This is a technique particularly used for addressing class imbalance in binary classification problems. Class imbalance occurs when one class in the dataset has lesser samples than the other class, leading to a biased llearning process where the model may perform incorrectly on the minority class. ADASYN addresses this issue by generating synthetic samples for the minority class to make it more balanced.

Adaptive Synthetic Sampling (ADASYN) Algorithm

The working of this model isiinitialised with an unbalanced data, where there are majority samples, m_l, and minority samples m_s.
The first step is to calculate the imbalance ratio,d, of the dataset

$$d=m_s/m_l$$

Then, the requirement of synthetic sample creation is assessed by computing the total number of synthetic minority data points (G) to be generated:

$$G = (m_l - m_s) \times \beta$$

Here, β represents the required ratio of minority to majority data after balancing, where β = 1 means a perfect balance.

Then it evaluates the factor with which a neighbouring data point dominates the minority data point Neighborhood Dominance. Thus, for any minority data point, x_i , the value r_i determines the the how dominant a majority data point is in its respect.

$$ri = (Number\ of\ majoritydata\ points\ among\ k\text{-}nearest\ neighbors) / k$$

In this step it normalises the r_i values in such a way that

$$\sum r_i = 1$$

After this, the number of synthetic examples (Gi) to be generated for each neighborhood is determined:

$$Gi = G \times r_i$$

Them, for each neighborhood associated with x_i another minority class, x_z is selected and a new synthetic sample, s_i is created using :

$$si = xi + \lambda \times (xz - xi)$$

λ is a random number between 0 and 1.

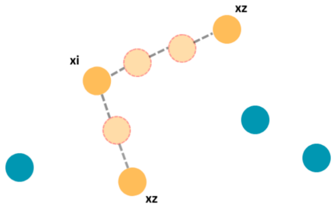

Fig. 3. Generation of synthetic data points in ADASYN

By doing so, it equalizes the class distribution, fostering fairness in subsequent model training and evaluation. This technique ensures that both non-sarcastic and sarcastic expressions are equally represented in the dataset, fortifying the research's scientific rigor.

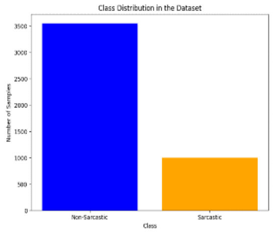

Fig. 4. Unbalanced Data

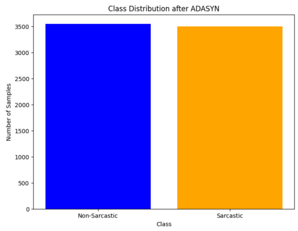

Fig. 5. Balanced Data

C. Feature Extraction

Feature extraction is a fundamental step in preparing textual data for machine learning. In this research, the textual content of Twitter expressions underwent Term Frequency-Inverse Document Frequency (TF-IDF) vectorization. TF-IDF is a widely used technique that quantifies the significance of words and phrases within text documents. In this context, the TF-IDF vectorizer was configured to consider a maximum of 5,000 features while excluding common English stop words and considering word combinations up to bi-grams (n-grams with a range of 1 to 2). This transformation rendered the textual data into a numerical format that could be processed by machine learning models. TF-IDF encapsulated the essence of the expressions, empowering subsequent analyses with rich and informative features extracted from the text.

D. Model Selection

This research embarks on a comparative exploration of three distinct types of machine learning models, each offering its unique approach to unraveling the enigma of sarcasm detection within Twitter data. These model types, Bagging (Random Forest), Boosting (XGBoost), and Transformer (LSTM), stand as representative pillars, poised for a comparative evaluation.

Bagging Model- Ensemble Random Forest : Bagging, embodied by the Random Forest model, thrives on the principle of ensemble learning. It assembles an array of decision tree classifiers, each trained on a random subset of the dataset. Combining the predictions of these diverse trees, it creates a resilient ensemble model. Random Forest excels in capturing complex relationships within data, making it a suitable candidate for the intricate task of sarcasm detection. Its ensemble nature mitigates overfitting and enhances robustness, demonstrating a holistic understanding of language nuances.

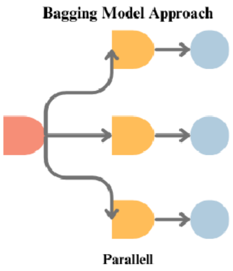

Fig. 6. Bagging Model

Boosting Model: XGBoost, a quintessential boosting algorithm, amplifies the predictive prowess of weak learners into a formidable model. By sequentially training a series of decision trees, it emphasizes the misclassified data points from preceding trees, progressively refining its predictive accuracy. XGBoost's adaptability and ability to handle complex, non-linear relationships are assets that shine in the intricate landscape of sarcasm detection. Its capability to handle imbalanced datasets, coupled with a commitment to precision, renders it a potent tool in our analytical arsenal.

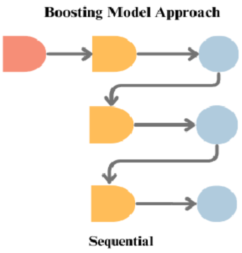

Fig. 7. Boosting Model

Transformer- LSTM: Long Short Term Memory (LSTM) model is a neural network architecture. Renowned for its sequence-to-sequence capabilities and contextual understanding, LSTM incorporates a Transformer layer to grasp the nuances of language patterns. This model excels in capturing long-range dependencies within textual data, vital in the context of sarcasm detection. With its attention mechanisms and memory retention, Transformer LSTM promises to unearth the subtle linguistic cues that underpin sarcastic expressions within the Twitter dataset.

Each of these models contributes distinct advantages to this research, fostering a holistic approach to sarcasm detection. As explored in the experimental setup, their individual roles and functionalities will come to the fore, revealing the synergy that propels our investigation forward.

E. Experimental Setup

The experimental setup forms the bedrock of this research, orchestrating the deployment of three diverse models—Bagging Random Forest, Boosting XGBoost, and Transformer LSTM. This section delineates the systematic process that encompasses dataset splitting, feature extraction, model initialization, training, evaluation, and visualization, harmonizing with the code provided.

1) Dataset Splitting:

The initial phase of the experimental setup entails the segregation of the dataset into two distinct subsets: the training set and the testing set. This division is meticulously orchestrated, adhering to an eighty-twenty ratio allocation. In this allocation, the training set encompasses 80% of the dataset, leaving the remaining 20% designated for the testing set. This partitioning strategy serves a crucial purpose. It ensures that the subsequent model evaluations are conducted on a firm foundation devoid of bias or undue influence. The training set functions as the arena in which the models acquire their proficiency through learning, while the testing set operates as a rigorous assessment platform.

2) Bagging Random Forest Approach:

With the dataset effectively partitioned, model training is embarked on, commencing with the Bagging Random Forest approach. This method involves the iterative training of a Random Forest classifier, a robust ensemble of decision trees, on the features derived from the training data. The ensemble was comprised of 100 individual decision trees.
In each iteration, this ensemble of decision trees underwent training, imbibing patterns and subtleties in the data. Following training, predictions were generated for the test data. Subsequently, predictions from each decision tree were accumulated for a comprehensive analysis. This ensemble approach, akin to seeking wisdom from multiple perspectives, contributed to a more robust model. To gauge the efficacy of the Bagging Random Forest model, crucial performance metrics, including precision, recall, and F1-score were calculated. These metrics provided valuable insights into this model's capacity to accurately classify sarcasm, distinguishing it from non-sarcasm within the Twitter dataset.

3) Boosting XGBoost Approach:

Following the Bagging Random Forest approach, the focus was shifted to the Boosting XGBoost model. This approach is grounded in the training of an XGBoost classifier, a gradient-boosting algorithm, using the TF-IDF features extracted from the training data. The training process of XGBoost is iterative and adaptive, emphasizing previously misclassified data points. This adaptability contributes to enhanced accuracy over successive iterations. To critically assess the performance of the Boosting XGBoost model, key evaluation metrics were analyzed. These metrics included accuracy, allowing us to gauge overall model correctness. A comprehensive classification report offered a detailed breakdown of model accuracy, precision, recall, and F1-score.

4) Transformer LSTM Approach:

The research journey then led to the Transformer LSTM (Long Short-Term Memory) model. This deep learning approach is tailored to capture intricate linguistic patterns within the Twitter dataset. The process began with tokenization, transforming the text data into manageable units. Subsequently, sequences were padded to a fixed length, preparing the data for neural network input. The model architecture itself consisted of an embedding layer, an LSTM layer for sequential analysis, and a dense layer. These layers enabled the model to grasp sequential dependencies and linguistic subtleties within the dataset.
Training the model ensued, with multiple epochs permitting iterative learning. After training, the model was used to generate predictions for the test data. Evaluation metrics, such as accuracy, precision, recall, and F1-score, were meticulously computed to assess our model's capability to classify sarcasm accurately.

Long Short-Term Memory (LSTM) Networks

LSTM is a type of recurrent neural network (RNN).
It has three major components:

- Cell State (C_t) : it acts as a memory unit that can store and retrieve information of long words.
- Hidden State (Ht_t): It is responsible for recording and passing information.
- Gate Mechanisms: LSTMs use three gate mechanisms to control information: the input gate (i), forget gate (f), and output gate (o).

The first step is to calculate the forget gate (f_t) that decides what information from the previous cell state should be deleted and what should be retained.
It uses hidden state (H_t-1) and the current input (X_t).

$$f_t= \sigma(W_f . [H_t\text{-}1, Xt] + b_f)$$

Then in a similar manner, it calculates the input gate (i_t) that decides what new information should be added to the cell state.

$$i_t= \sigma(W_i . [H_t\text{-}1, Xt] + b_i)$$

Then, the updated cell state is calculated, (C~t) , hat stores new information.

$$C'_t= tanh(W_c . [H_t\text{-}1, Xt] + b_c)$$

And the current cell state (Ct) is updated by combining the information from the input gate (i_t) and the candidate cell state (C~t).

$$C_t = f_t * C_t\text{-}1 + it * C'_t$$

Now the output gate (o_t) that determines what information from the current cell state should be passed to the next hidden state is calculated,

$$o_t= \sigma(W_o . [H_t\text{-}1, Xt] + b_o)$$

and the current hidden state (Ht) is changed by applying the output gate (o_t) to the cell state (C_t).

$$(H_t = ot *tanh(C_t)$$

this updated hidden state (Ht) is used as the output for the current step, and is used for prediction of further steps.

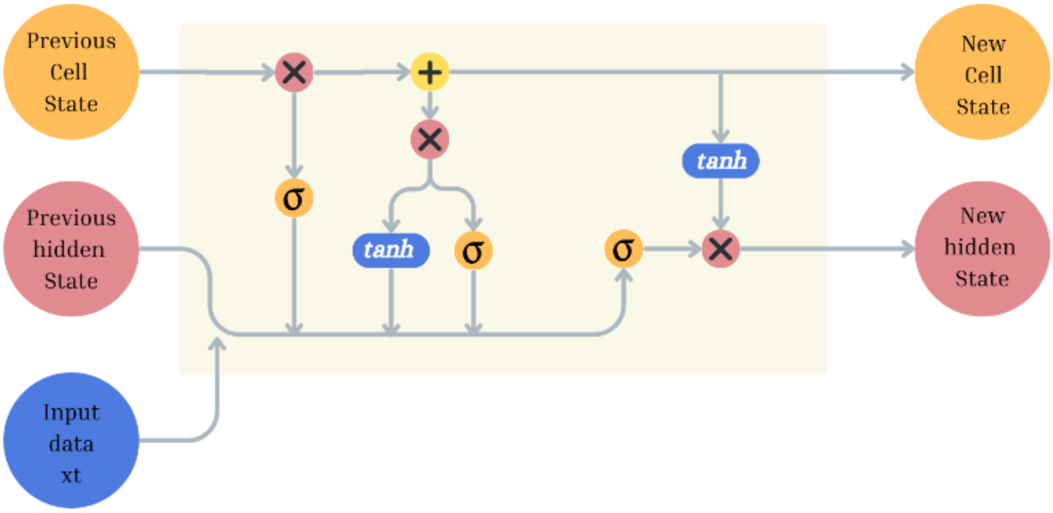

Fig. 8. Gates and Algorithm of LSTM

Furthermore, The ADASYN-TF-IDF-LSTM model proposes an innovative approach to sarcasm detection, underpinned by a combination of strategic techniques. At its core, ADASYN plays a pivotal role by addressing the class imbalance which is commonly observed in sarcasm detection datasets. Through intelligent oversampling of the minority class, this model ensures balanced familiarization with both sarcastic and non-sarcastic instances during training, mitigating biases and enhancing the model's accuracy.

Another crucial feature of this model is the incorporation of TF-IDF vectorization. Beyond just the transformation of text into numerical feature vectors, this approach stands out in capturing the semantic importance of individual words within each document, as compared to the entire dataset. This comprehension allows the model to recognize specific words or phrases critical to conveying sarcastic intent, thereby enhancing its ability to identify sarcasm.

Combining these components is the utilization of LSTM (Long Short-Term Memory) as the model's core architecture. LSTM's sequential data processing capabilities are ideal for capturing the intricate merging of words and their contextual dependence, which is a fundamental aspect of sarcasm detection. Its contextual sensitivity, and consideration of word order and context, prove invaluable in distinguishing between sarcastic and non-sarcastic content, particularly when sarcasm relies on subtle linguistic patterns and contextual cues.

The ADASYN-TF-IDF-LSTM model's impact is highly elaborate. Its combination of techniques gives a high level of accuracy in sarcasm detection, which is a necessary requirement for applications where precise sentiment analysis is a primary concern. By effectively catering to the class imbalance, it reduces bias and ensures correct predictions, and helps in a deeper analysis of sarcastic as well as non-sarcastic texts.

Moreover, it has contextual sensitivity which enables it to perform well in scenarios where sarcasm is deeply embedded within conversations or is very subtle to observe through computing.

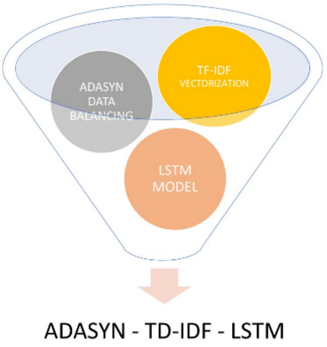

Fig. 9. The ADASYN-LSTM Model

To conclude, the ADASYN-TF-IDF-LSTM model presents an advanced approach in sarcasm detection. It aptly handles class imbalance, maintaining linguistic significance, and understanding the context. It acts as a helpful tool for precise sentiment analysis. The proposed model has the potential to guide sentiment analysis and natural language processing by offering a variety of applications and solving complex sarcasm problems in texts.

IV. Results and Discussions

This study compares and analyses three different models used for sarcasm detection: Bagging Random Forest, Boosting XGBoost, and Transformer LSTM. Each of these models has its own set of advantages and drawbacks, hence, providing insights into various aspects of textual sarcasm analysis.

Table II. Comparison of Performance Metrics

Model	**Model Metrics Comparison**		
	Model Accuracy	***Precision***	***F1 Score***
Random Forest Classifier	0.93	0.94	0.82
XGBoost Model	0.93	0.91	0.88
LSTM Transformer	0.93	0.93	0.93

Bagging Random Forest: Bagging Random Forest exhibits notable strengths, particularly in precision. It achieves the highest precision among the three models, indicating its ability to accurately classify tweets as either sarcastic or non-sarcastic. This model is particularly beneficial when minimizing false positives is critical, making it suitable for applications where accurate identification of sarcasm is paramount. However, a drawback is that it attains a slightly lower recall, which implies that it may miss some actual sarcastic tweets. Additionally, Bagging Random Forest is an ensemble model, which means it would require more computing and more time to train the model.

The ROC (Reciever Operator charastic) of the model (see Fig. 10) is depictive of how well the model can classify false and true data points. In the ROC curve, the AUC represents the area under the curve. A model with a higher AUC ROC score generally indicates a higher ability to correctly classify instances. The boosting model shows an exceptional AUC ROC of 0.94, which means it has a 94 percent precision in classifying tweets.

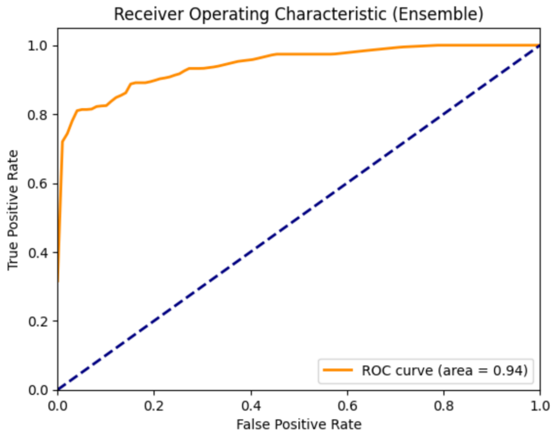

Fig. 10. ROC for bagging Random Forest Model

Boosting XGBoost: Boosting XGBoost offers a balanced set of advantages. Similar to Bagging Random Model Forest, it achieves strong precision, ensuring accurate identification of sarcastic tweets. It also maintains a good balance between precision and recall, resulting in a high F1-Score, which signifies effective overall performance. One notable advantage of XGBoost is its computational efficiency and speed, making it suitable for real-time applications where resources are limited. However, it achieves a slightly lower recall compared to Bagging Random Forest, implying that it may also miss some sarcastic tweets. Fine-tuning of hyperparameters is often required to achieve optimal performance, which can be time-consuming.

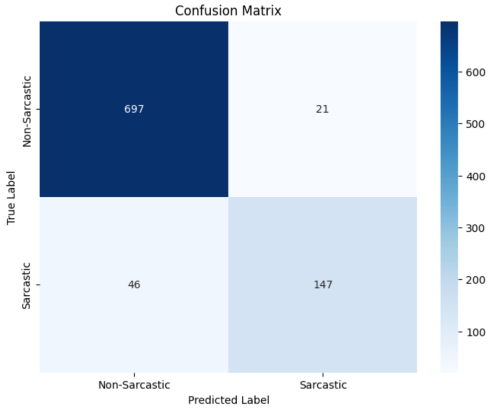

Fig. 11. Confusion Matrix for Boosting Model

The confusion matrix is used to evaluate the performance of a classification model. The matrix is particularly useful when dealing with binary classification problems (two classes: positive and negative). The confusion matrix consists of four main components:

True Positive (TP): Instances that are actually positive and are correctly predicted as positive by the model.

True Negative (TN): Instances that are actually negative and are correctly predicted as negative by the model.

False Positive (FP): Instances that are actually negative but are incorrectly predicted as positive by the model.

False Negative (FN): Instances that are actually positive but are incorrectly predicted as negative by the model.

The conclusion matrix of the boosting model (see Fig. 11) depicts a good performance by accurately identifying sarcastic and non-sarcastic tweets 93% of the time.

Transformer LSTM: Transformer LSTM has its bag of merits. It is commendable in both precision and recall, making it effective at both identifying sarcasm and reducing false positives. This stability is seen in its high F1-Score, indicating a partial mixture of precision and recall. Transformer LSTM is preferable for processing sequential texts, which particularly gives it an upper edge in the detection of sarcasm in text. Yet, it comes with some disadvantages, such as hiked model complexity and more computational burden. Training models like LSTM may also take longer time, depending on the size of the dataset and the build of the model.

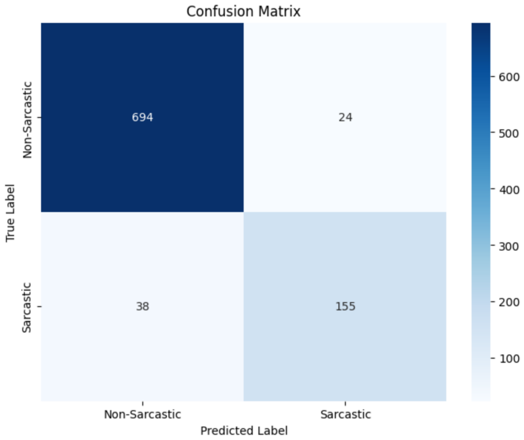

Fig. 12. Confusion Matrix for Transformer Model

The conclusion matrix of LSTM (see Fig, 12) shows exceptional performance in classifying the true positives and false positives. The ADASYN-LSTM model shows excellent model results, with perfectly balanced Accuracy, Precision and F1 Score which means, that the model not only classifies sarcastic tweets accurately but also displays precision in results, which means it can identify correctly most of the time it is tested.

These results present a comprehensive evidence of the performance of the models: Bagging Random Forest, Boosting XGBoost and finally the proposed ADASYN - TF-IDF - LSTM.

V. Future Work and Conclusion

Within the world of sarcasm detection, this research opens up avenues for future exploration. One of the directions is the development of models that can aptly handle multilingual sarcasm detection. As the world becomes interconnected more and more every day, the ability to detect sarcasm in various languages is crucial for global sentiment analysis. Cross-lingual models can enable us to understand and respond to sentiment in diverse cultural contexts and solve the global problem of the negative impacts of sarcasm. Furthermore, integrating real-time sarcasm detection in communication platforms could revolutionize how we engage online. The ability to detect sarcasm in the heat of a live chat or social media discussion has strong potential, both for individuals and businesses.

The research can have a pronounced impact in the future. The correct detection of sarcasm can enhance decision-making processes in various domains. The novel ADASYN-TF-IDF-LSTM model proposed in the paper is a new paradigm in sarcasm detection. This combination of oversampling using ADASYN, TF-IDF, and LSTM techniques enables it to achieve accuracy in context-rich scenarios, where sarcasm is based on linguistic cues and is subtle. By bridging the gap between class imbalance and an in-depth understanding of language, this model could aid future research in sentiment analysis and natural language processing. Its potential impact not only extends to sarcasm detection, but also to other related fields.

In conclusion, the research paper focusses on the intricate issues that arise during sarcasm detection. The study identifies the fundamental problems of class imbalance in the sarcasm detection dataset, which require robust methods for addressing the issue. This innovative approach that combines ADASYN oversampling, TF-IDF vectorization, and LSTM architecture, introduces the ADASYN-TF-IDF-LSTM model. This novel model demonstrates the combination of class balance mitigation and advanced linguistic understanding, setting new standards in sarcasm detection and sentiment analysis.

References

[1] I Davidov, D., Tsur, O., & Rappoport, A. (2010). "Semi-supervised Recognition of Sarcastic Sentences in Twitter and Amazon." In Proceedings of the Fourteenth Conference on Computational Natural Language Learning (CoNLL).

[2] Reyes, A., Rosso, P., & Veale, T. (2013). "A Multidimensional Approach for Detecting Irony in Twitter." Language Resources and Evaluation, 47(1), 239-268

[3] Ptáček, T., Otrusina, L., & Smrž, P. (2014). "Sarcasm Detection on Czech Twitter Data." In Proceedings of the 16th International Conference on Text, Speech and Dialogue (TSD).

[4] Kreuz, R. J., & Caucci, G. M. (2007). "Lexical Influences on the Perception of Sarcasm." In Proceedings of the Annual Meeting of the Cognitive Science Society.

[5] Khodak, M., Saunshi, N., & Vodrahalli, K. (2017). "A Large Self-Annotated Corpus for Sarcasm." In Proceedings of the First Workshop on Sarcasm Detection.

[6] Derczynski, L., Bontcheva, K., & Liakata, M. (2013). "Sarcasm Detection on Twitter: A Behavioral and Linguistic Analysis." In Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (ACL).

[7] Liebrecht, C., Kunneman, F., & van den Bosch, A. (2013). "The Perfect Solution for Detecting Sarcasm in Tweets #not." In Proceedings of the International Workshop on Semantic Evaluation (SemEval).

[8] Mishra, A., & Mourya, D. (2019). "A Survey on Sarcasm Detection in Text." In Proceedings of the International Conference on Machine Learning and Data Science (ICMLDS).

[9] Riloff, E., & Wiebe, J. (2003). Learning extraction patterns for subjective expressions. In Proceedings of the conference on empirical methods in natural language processing (EMNLP).

[10] Gonçalves, P., Araújo, M., Benevenuto, F., & Cha, M. (2013). Comparing and combining sentiment analysis methods. In Proceedings of the first ACM conference on Online social networks (COSN).

[11] Buschmeier, H., Kopp, S., & Baresel, A. (2012). Investigating the influence of different feature types on the classification of irony and sarcasm. In Proceedings of the 13th European Workshop on Natural Language Generation (ENLG).

[12] Bamman, D., Eisenstein, J., & Schnoebelen, T. (2012). Learning to detect sarcasm in tweets: A local or global task? In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-HLT).

[13] Joshi, A., Sharma, A., & Bhattacharyya, P. (2015). Harnessing context incongruity for sarcasm detection.In Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (ACL-IJCNLP).