Sarcasm Detection

Team Members :

Vishesh Paka
Jahnavi Chowdary Tumati
Srikar Rambhatla

INTRODUCTION

The subtle nature of language and contextual dependencies makes sarcasm detection in textual data especially news headlines extremely difficult.

In this work, we introduce a novel method for sarcasm detection based on a carefully selected dataset from reliable news sources, including HuffPost and The Onion. By utilizing expertly crafted news headlines, we hope to improve the precision and consistency of sarcasm detection systems while circumventing the drawbacks of the existing Twitter-based datasets.

Our approach starts with a thorough pre-processing of the data to guarantee its consistency and quality.
Using Word2vec and Glove embeddings, we train machine learning models specifically designed for sarcasm detection tasks.

OBJECTIVE

- The performance of sarcasm detection algorithms is impacted by the limitations of existing datasets sourced from social media platforms such as Twitter which are often noisy and lack contextual information.
- Our objective is to create a model that can detect sarcasm in news headlines that have been expertly crafted, thus improving accuracy and expanding the model's usefulness in real world scenarios.

DATASET

The dataset comprises approximately 28,000 news headlines categorized as Sarcastic or Not Sarcastic which offers unrestricted access.

Each headline is associated with a label indicating its sarcasm status.

The dataset can be accessed through the following links:

- Sciencedirect Article
- GitHub Repository

This dataset acquisition process underscores the importance of utilizing credible and well-labeled datasets for robust and meaningful research outcomes in sarcasm detection.

METHODOLOGY

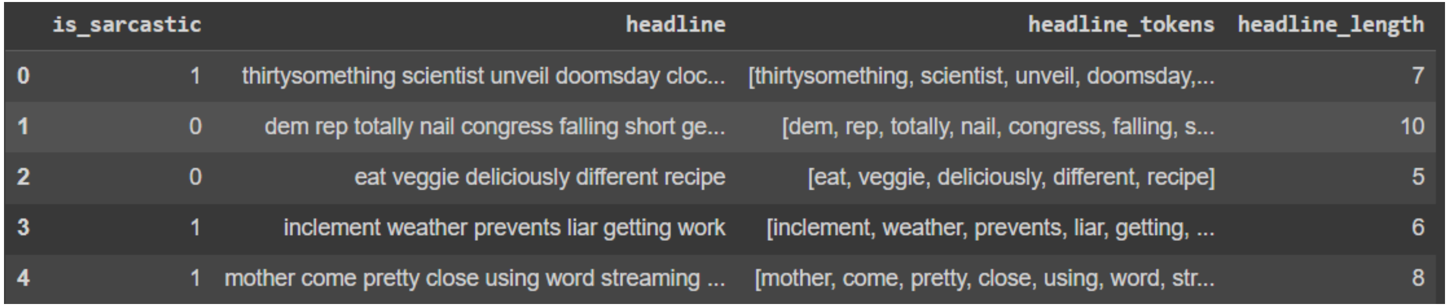
- Data collection and Preprocessing
- Model selection and comparison
- Training and evaluation
- Analysis and interpretation
- Documentation and reporting

APPROACH

- Our approach integrates data collection, preprocessing, model selection, training, and evaluation for sarcasm detection in news headlines.
- We utilize curated datasets from TheOnion and HuffPost known for professional and sarcastic headlines.
- After data preprocessing, we employ Word2Vec and GloVe word embedding models for vector representation.
- Model selection is followed by training and performance assessment using standard metrics.
- Key variables influencing accuracy are identified through evaluation.
- Comprehensive documentation summarizes findings and suggests further research in sarcasm detection

DATA PRE-PROCESSING

- Convert to Lowercase
- Remove Punctuation
- Remove Numbers
- Strip HTML Tags
- Remove Content in Square Brackets
- Remove URLs
- Remove Stopwords
- Lemmatization
- Our final processed sample data:

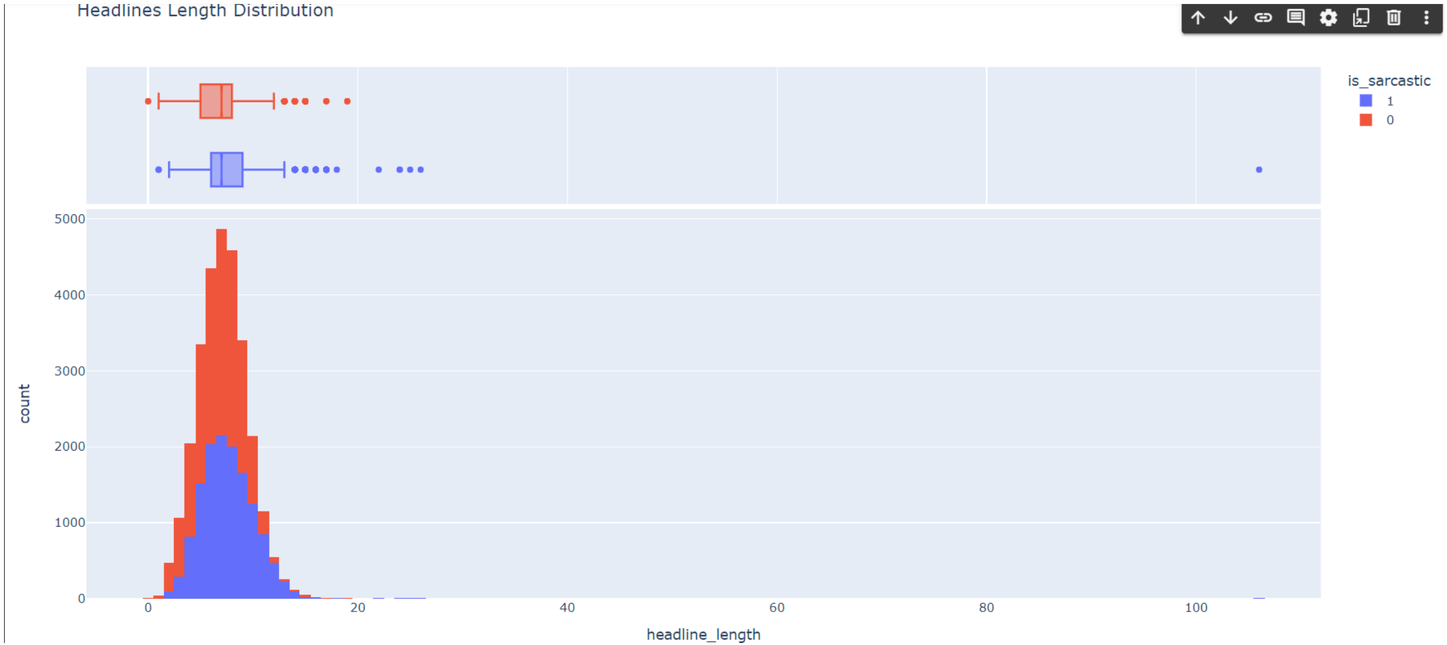


	is_sarcastic	headline	headline_tokens	headline_length
0	1	thirtysomething scientist unveil doomsday cloc...	[thirtysomething, scientist, unveil, doomsday,...	7
1	0	dem rep totally nail congress falling short ge...	[dem, rep, totally, nail, congress, falling, s...	10
2	0	eat veggie deliciously different recipe	[eat, veggie, deliciously, different, recipe]	5
3	1	inclement weather prevents liar getting work	[inclement, weather, prevents, liar, getting, ...	6
4	1	mother come pretty close using word streaming ...	[mother, come, pretty, close, using, word, str...	8

Headline Length Distribution

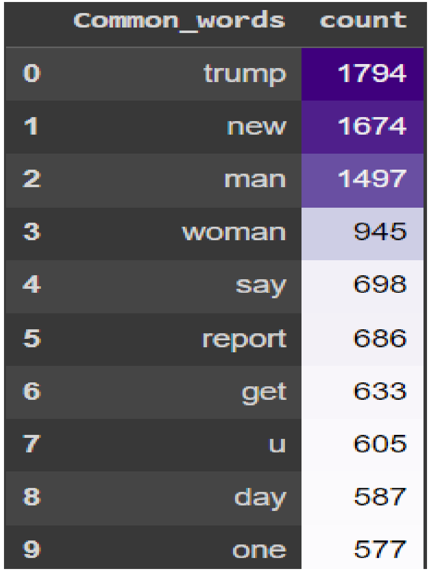
Understanding headline length distribution is crucial in NLP, especially for sarcasm detection. We analyze headline lengths, focusing on outliers to ensure concise inputs for models.

Using Plotly Express, we visualize headline length distribution, aiming to identify variations in structure. Ideal headlines for NLP tasks range between 20 to 30 words.

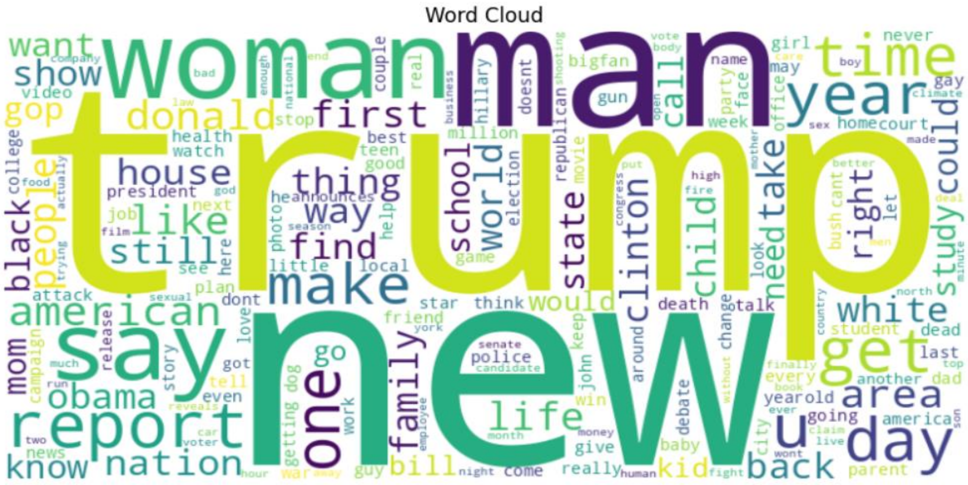


Word Frequency Analysis

Word frequency analysis is essential for uncovering prevalent terms and patterns in news headlines. Using NLTK, we preprocess data and employ the Counter module to quantify word occurrences. The code generates word clouds, visually representing word frequencies in the dataset and distinguishing between sarcastic and non-sarcastic headlines. These visualizations aid in feature identification for subsequent analysis.

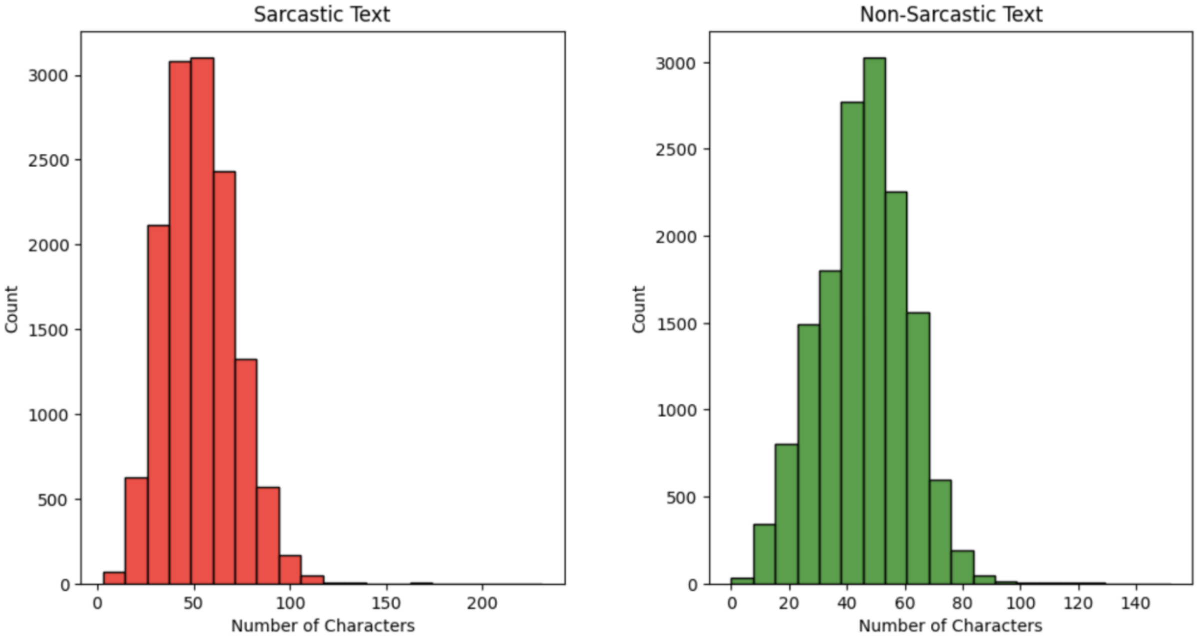


	Common_words	count
0	trump	1794
1	new	1674
2	man	1497
3	woman	945
4	say	698
5	report	686
6	get	633
7	u	605
8	day	587
9	one	577

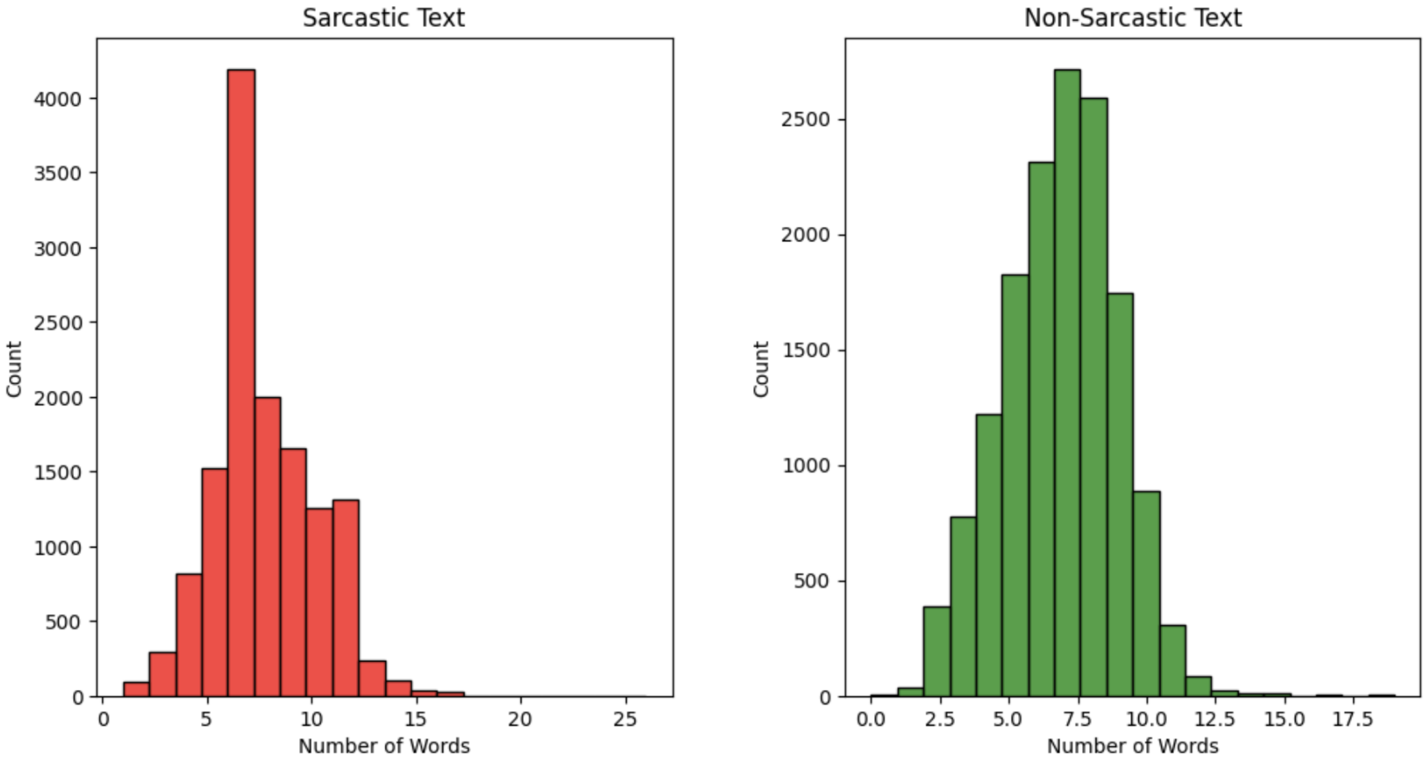


Character length distribution in headlines

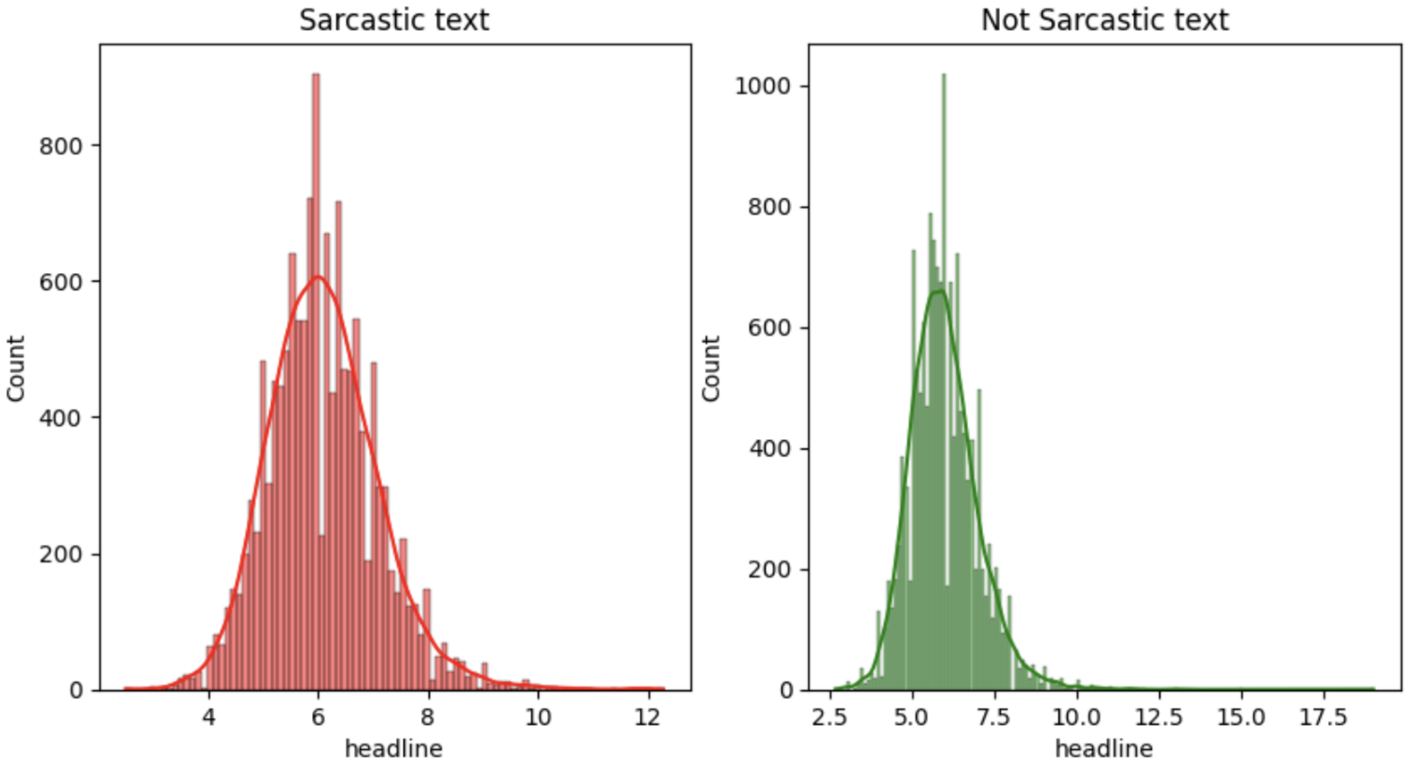
.



Words distribution in headlines



Average word length in each headline

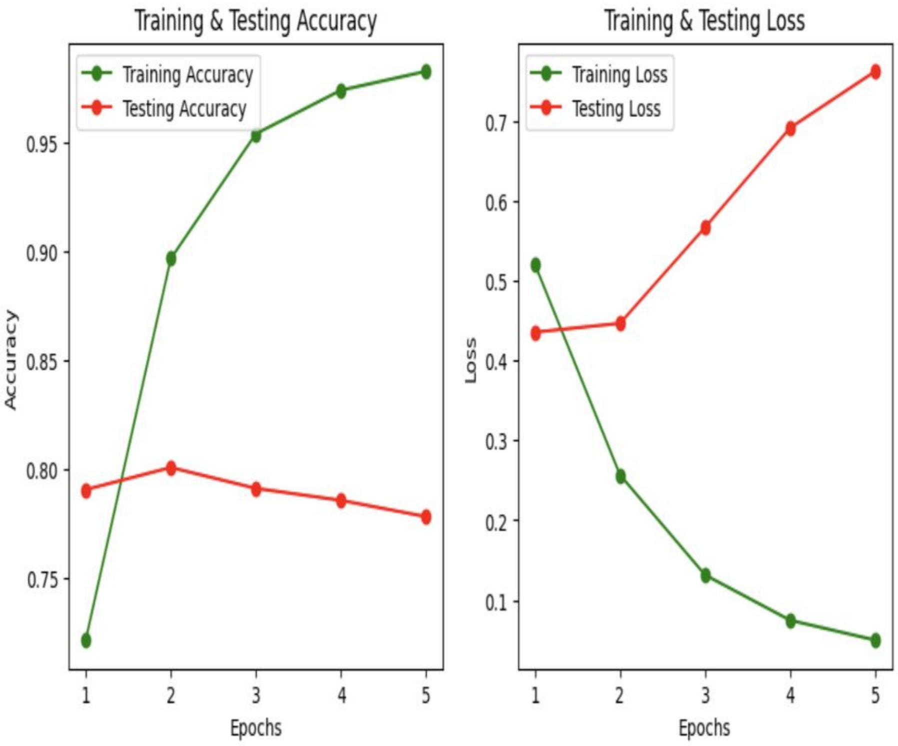


LSTM Models

Our study employs a neural network architecture for sarcasm detection in news headlines. It features:

- Embedding layer (200 dimensions)
- Bidirectional LSTM and GRU units with dropout regularization
- Dense layer with sigmoid activation for binary classification
- Adam optimizer (learning rate: 0.001)

We evaluate model performance using accuracy and loss metrics across epochs. Plots illustrate training/testing accuracy and loss, offering insights into learning progress and generalization ability. Critical for NLP sarcasm detection assessment.

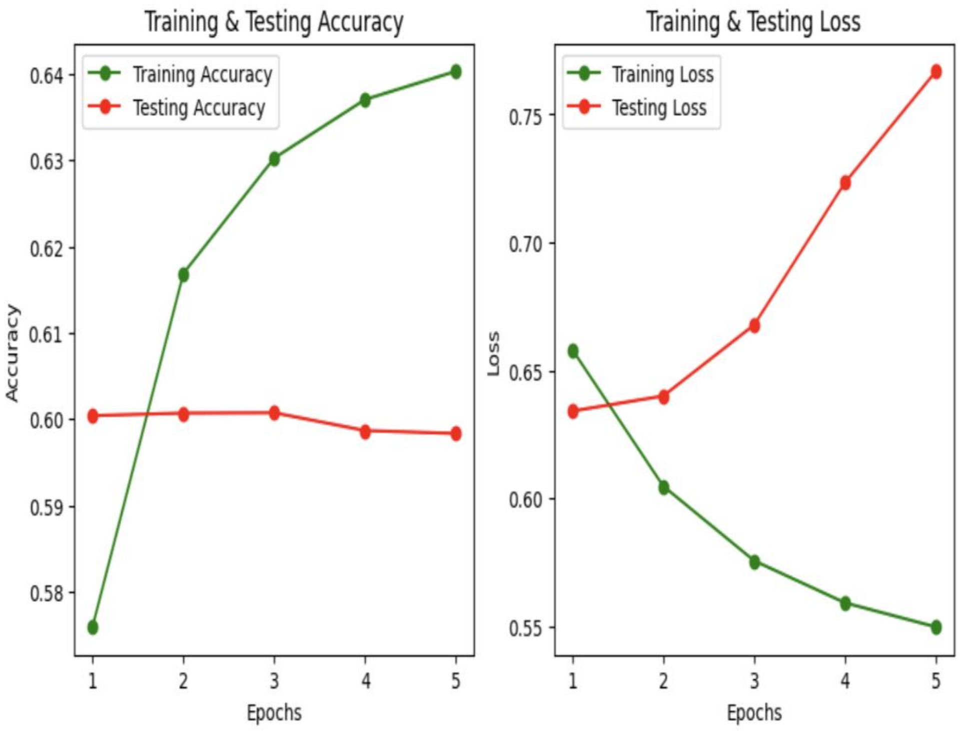


Analysis of Bidirectional LSTM, GRU model with accuracy and loss :

The training accuracy of the model is exceptionally high at 99.42%, indicating that it performs very well on the training data. However, when evaluated on the testing data, the accuracy drops to 78.87%.

This significant drop suggests that the model may be overfitting to the training data, meaning it is learning to memorize the training examples rather than generalize well to unseen data.

One possible reason for this overfitting could be the high complexity of the model architecture, which may lead to excessive parameter tuning and capturing of noise in the training data.



Analysis of LSTM model with accuracy and loss

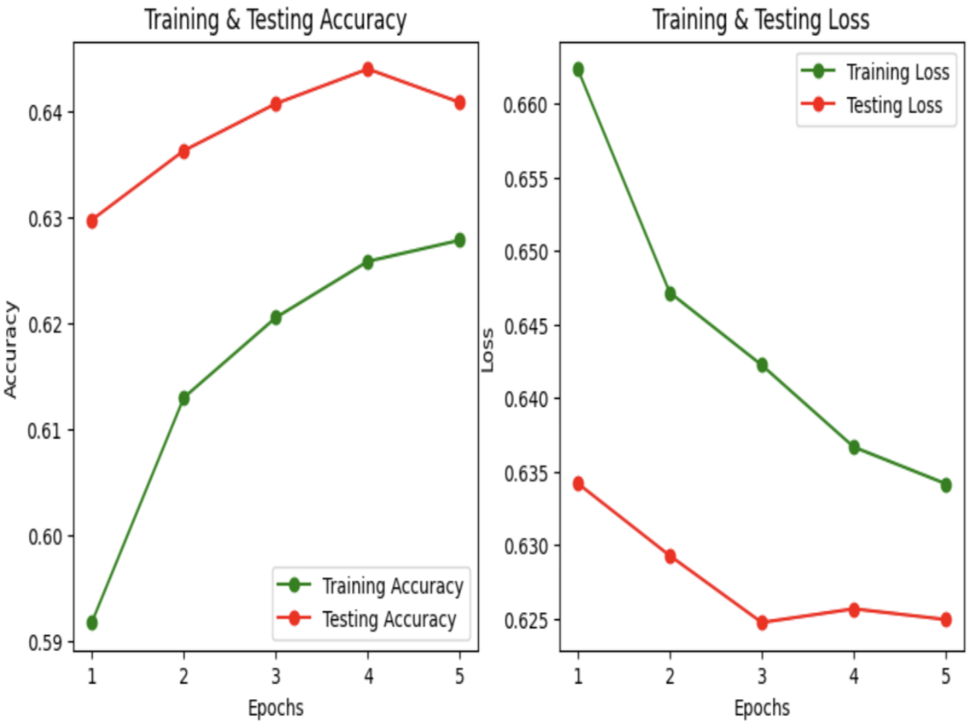
The simpler model with fewer layers is not overfitting as much as the more complex model. However, its performance is poor compared to the complex model. Specifically, the accuracy of the simpler model on the training data is 64.45%, while on the testing data, it is 59.80%.

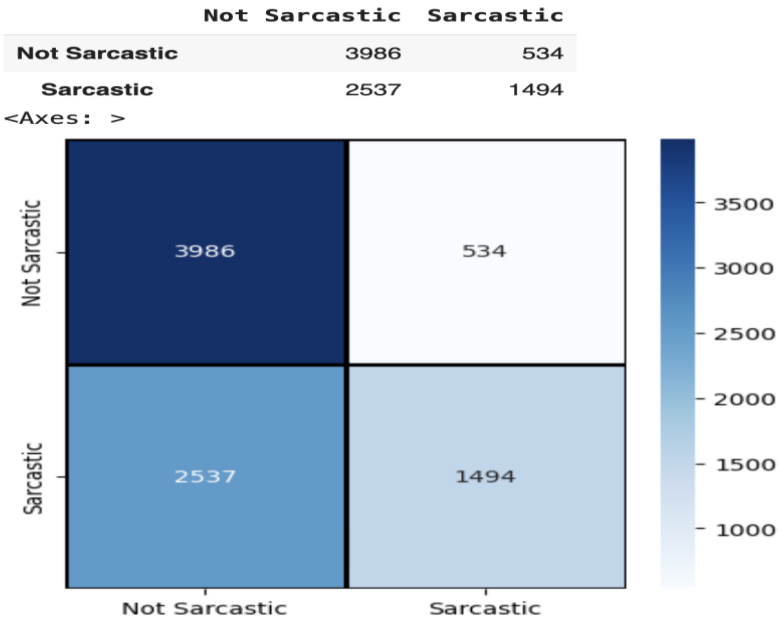
Word2Vec

Word2Vec is a popular technique in natural language processing (NLP) used to learn distributed representations of words in a continuous vector space. The main idea behind Word2Vec is to represent words as dense, fixed-length vectors, often called word embeddings, in such a way that similar words have similar vector representations.

Word2Vec learns from vast text data to predict target words or surrounding context. Post-training, embeddings capture semantic word relationships, aiding tasks like sentiment analysis and machine translation. Pre-trained embeddings are used in various NLP tasks.

Analysis of Word2Vec model with accuracy and loss





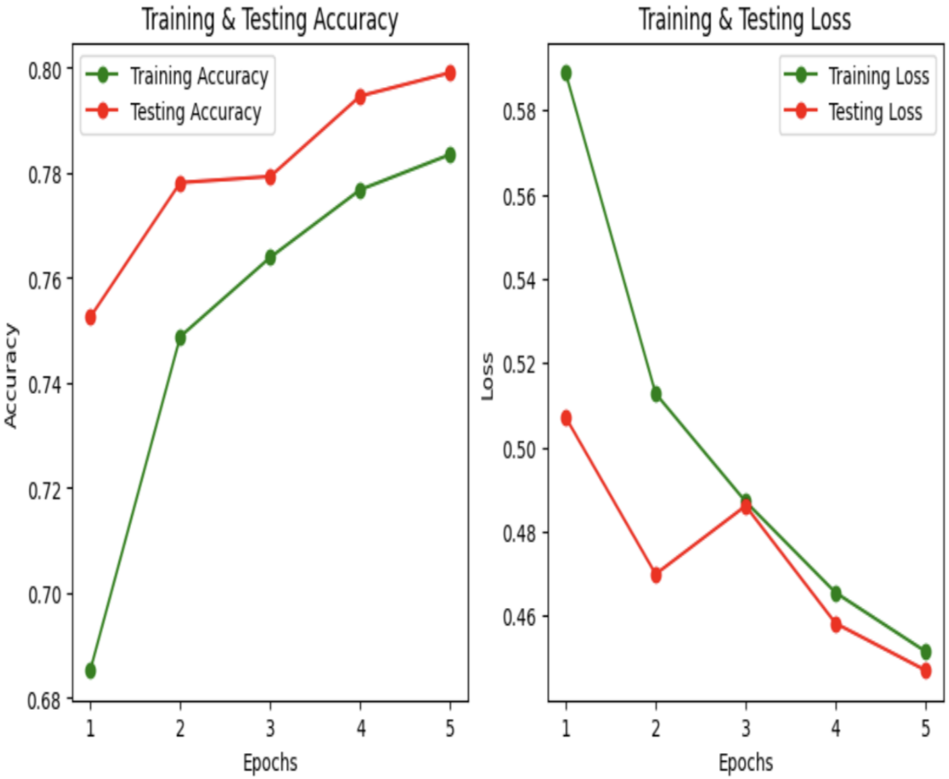
```
                Not Sarcastic  Sarcastic
 Not Sarcastic           3986        534
     Sarcastic           2537       1494
<Axes: >
```

GLOVE

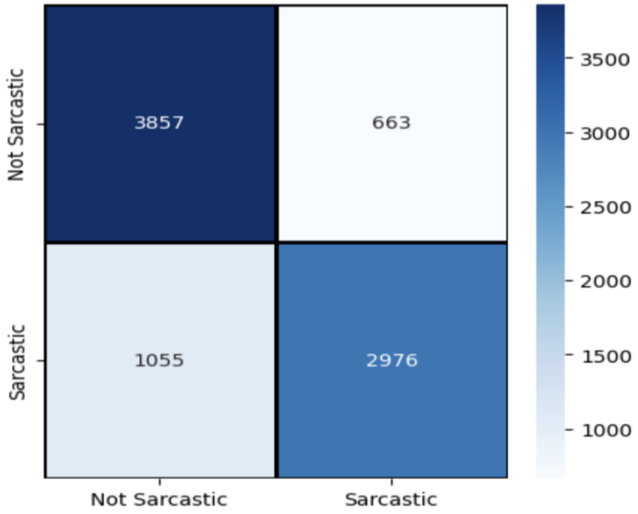
GloVe Integration in Neural Network:

- Embedding Matrix Creation**:**
  - Constructs embedding matrix (embedding_matrix) to store pre-trained GloVe embeddings (output_dim=100).
- Neural Network Definition**:**
  - Embedding Layer: Utilizes non-trainable embedding layer with pre-computed embedding_matrix.
  - Bidirectional LSTM Layer: Incorporates Bidirectional LSTM layer with dropout regularization.
  - Dense Classification Layer: Appends dense layer with sigmoid activation.
- Model Compilation:
  - Compiles model_glove using Adam optimizer and binary cross-entropy loss

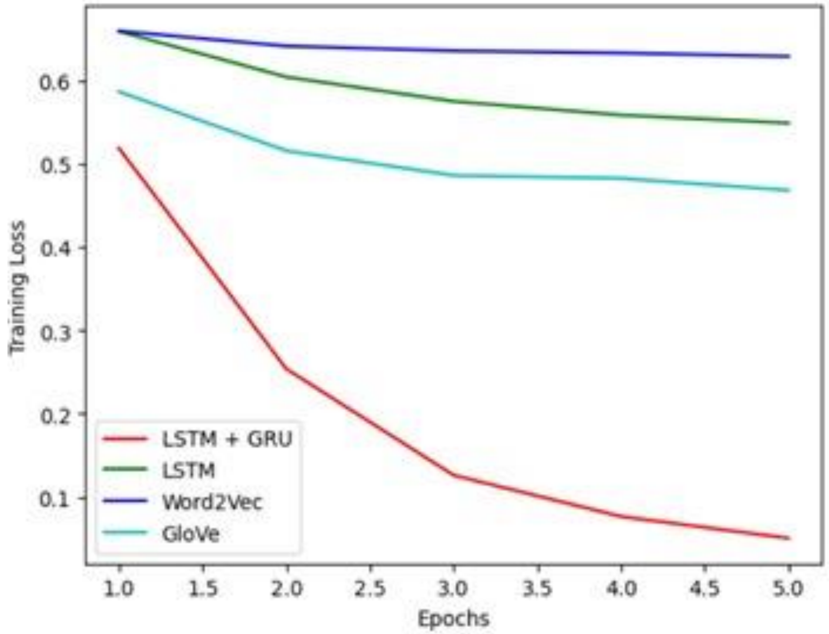
Analysis of Glove model with accuracy and loss

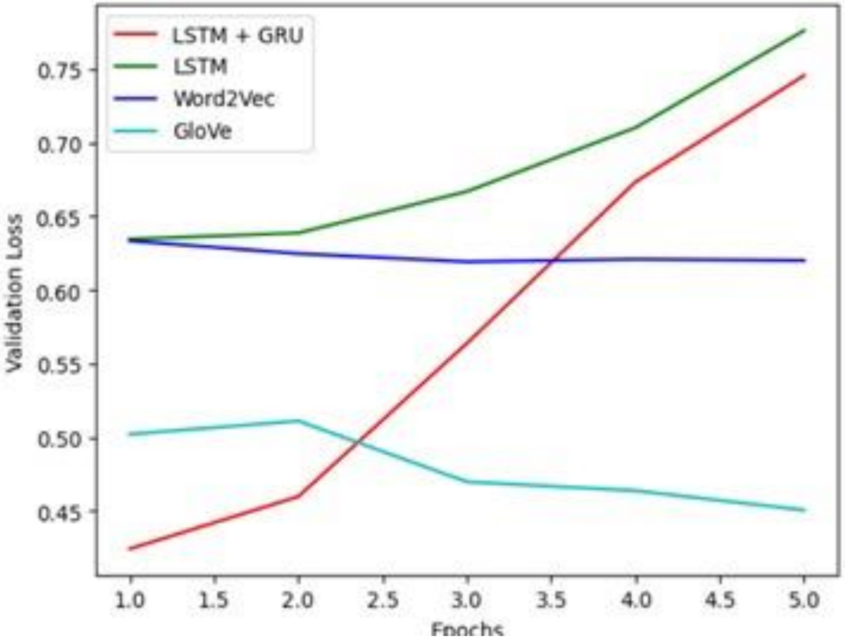


```
               Not Sarcastic  Sarcastic
Not Sarcastic           3857        663
    Sarcastic           1055       2976
<Axes: >
```

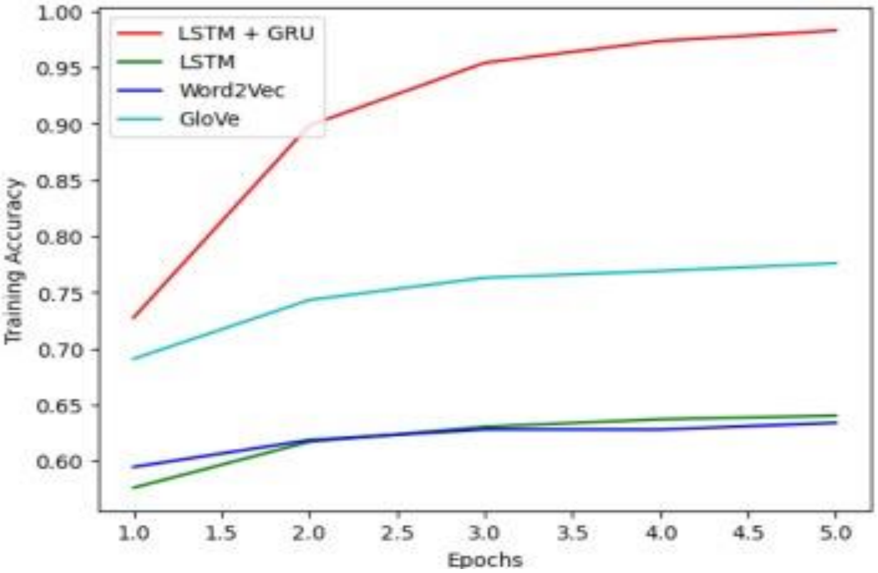


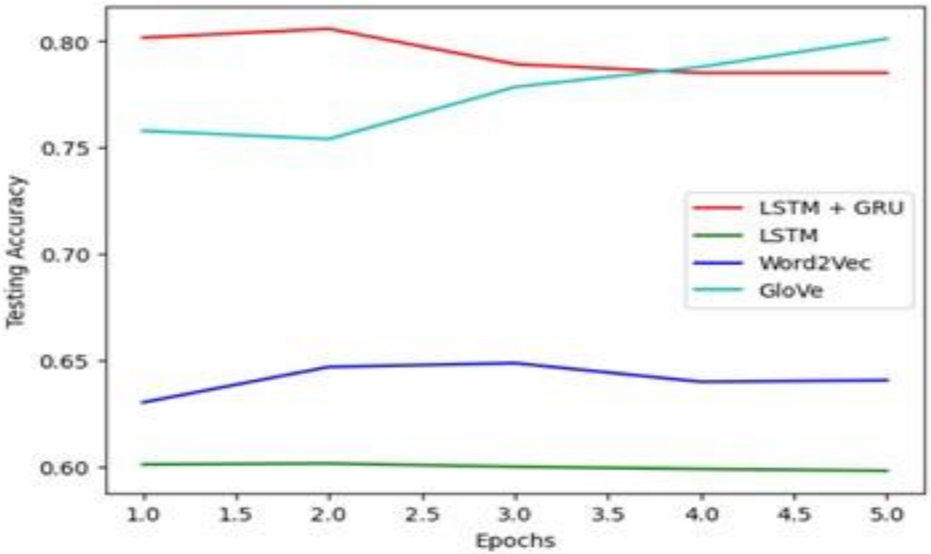
Model Comparision





Model Comparision





Conclusions and Future Discussions

In conclusion, this project demonstrates the effectiveness of neural network architectures for sarcasm detection in textual data. Leveraging pre-trained embeddings, particularly GloVe embeddings, significantly enhances model performance.

Future research directions may include exploring ensemble methods, fine-tuning hyperparameters, and incorporating attention mechanisms to further improve model accuracy and robustness in sarcasm detection tasks.

Further optimization and exploration of different architectures, including advanced models like BERT, may yield even better results. BERT, with its powerful bidirectional contextual understanding, has demonstrated remarkable performance in various NLP tasks and holds promise for enhancing sarcasm detection accuracy further.

THANK YOU