SARCASAM DETECTION

Dudla Velangani Joshita Lavanya
Yamini Durga Loya
Manogna Vennela Ramireddy

INTRODUCTION

Objective

- This study focuses on detecting sarcasm in text using various NLP models: BiLSTM, DistilBERT, and RoBERTa. Each model was trained and evaluated on three different datasets to assess performance across diverse text characteristics.

Models Overview:

- BiLSTM: Captures temporal dependencies in text using recurrent neural networks. Suitable for smaller datasets but computationally slower.
- DistilBERT: A lightweight transformer model with faster training and inference. Offers a balance of efficiency and performance.
- RoBERTa: An advanced transformer-based model, robustly optimized for state-of-the-art performance on NLP tasks.

DATASET DETAILS

1. Sarcasm Headlines Dataset :

"Sarcasm_Headlines_Dataset.json"

2. Sarcasm on Reddit Dataset

Sarcasm on Reddit

3. Combined Dataset from hugging face
"https://huggingface.co/datasets/SarcasmNet/sarcasm/tree/main"

DATA PROCESSING

The below are the steps applied to all the models:

- Tokenization (specific to each model: GloVe for BiLSTM, tokenizer for DistilBERT and RoBERTa).
- Padding for uniform sequence lengths.
- Train-validation-test splits.

Specific Differences

- BiLSTM: Used GloVe embeddings for vector initialization.
- DistilBERT and RoBERTa: Pre-trained tokenizers from Transformers library.

MODEL ARCHITECTURE

BiLSTM

•Embedding layer initialized with GloVe.
•Bidirectional LSTM layers (64 and 32 units).
•Dense layers for binary classification.

DistilBERT

•Pre-trained transformer model with fewer parameters.
•Fine-tuned with Adam optimizer and a learning rate of 5e-5.
•Custom tokenization using DistilBERT tokenizer.

RoBERTa

•Pre-trained transformer model with robust optimization.
•Fine-tuned using Hugging Face Trainer API.
•Key parameters: learning_rate=2e-5, batch_size=8, epochs=10.

MODEL ARCHITECTURE

- BiLSTM:
  - Trained for 10 epochs with early stopping.
  - Optimized using Adam optimizer and binary_crossentropy loss
  .
- DistilBERT:
  - Trained for 3 epochs with a batch size of 16.
  - Validation after every epoch.
  - SparseCategoricalCrossentropy with from_logits=True, this function is used because, the model outputs logits (unnormalized scores) instead of probabilities.

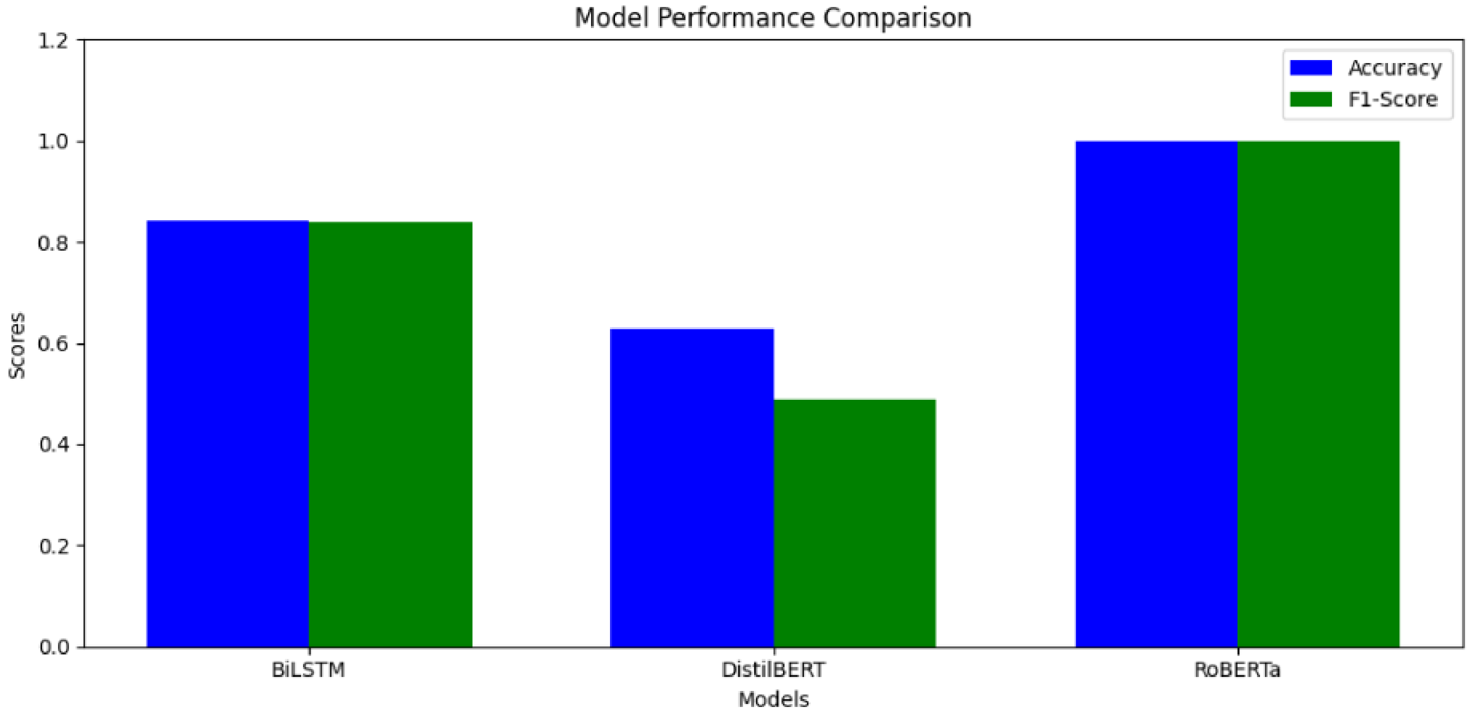
- RoBERTa:
  - Fine-tuned for 10 epochs using the Trainer API.
  - Evaluated with accuracy during training, cross entropy loss us used.

EVALUATION

Metric	BiLSTM	DistilBERT	RoBERTa
Accuracy	0.842	0.6295	1.00
Precision			
Not Sarcastic	0.83	0.63	1.00
Sarcastic	0.86	0.00	1.00
Recall			
Not Sarcastic	0.90	1.00	1.00
Sarcastic	0.76	0.00	1.00
F1-Score			
Not Sarcastic	0.87	0.77	1.00
Sarcastic	0.81	0.00	1.00
Macro Average			
Precision	0.85	0.31	1.00
Recall	0.83	0.50	1.00
F1-Score	0.84	0.39	1.00
Weighted Average			
Precision	0.84	0.40	1.00
Recall	0.84	0.63	1.00
F1-Score	0.84	0.49	1.00

EVALUATION

BiLSTM: Suitable for smaller datasets; slower but interpretable.
DistilBERT: Lightweight, fast, and efficient for low-resource environments.
RoBERTa: Best performance overall; requires more computational resources.



CONCLUSION

Summary of findings:
•**RoBERTa**: Outperformed others in all metrics but is computationally expensive.
•**DistilBERT**: Offers a balance between speed and accuracy.
•**BiLSTM**: A good baseline for smaller datasets or interpretability-focused tasks.

Recommendations:
•Use **DistilBERT** for real-time applications.
•Use **RoBERTa** for high-stakes tasks requiring maximum accuracy.

THANKYOU