Efficient transformer architecture for sarcasm detection: a study on compression and performance

Parul Dubey[1], Amit Mishra[2], Aruna Singh[3], Murtuza[4], Akshita Chanchlani[4], Pushkar Dubey[5]
[1]Department of Computer Science and Engineering, Symbiosis Institute of Technology, Nagpur Campus, Symbiosis International (Deemed University), Pune, India
[2]Department of Computer Science and Applications, Dr. Vishwanath Karad MIT World Peace University, Pune, India
[3]School of Education, Noida International University, Greater Noida, India
[4]Department of Computer Engineering, Dr. Vishwanath Karad MIT World Peace University (MIT-WPU), Pune, India
[5]Department of Management, Pandit Sundarlal Sharma (Open) University Chhattisgarh, Bilaspur, India

Article Info

Article history:

Received Jul 26, 2025
Revised Nov 9, 2025
Accepted Dec 6, 2025

Keywords:

Context summarization
DistilBERT
Natural language processing
Sarcasm detection
Transformer model

ABSTRACT

This sarcasm detection is a crucial subtask in natural language processing (NLP) particularly for sentiment analysis and conversational AI. Its complexity lies in interpreting context, tone, and intent beyond literal meanings. Traditional models often struggle to capture such nuances, especially in informal and diverse language settings. Moreover, existing approaches lack computational efficiency and fail to adapt well across different domains. This study evaluates three benchmark datasets—News Headlines, Mustard, and Reddit (SARC)—representing structured, scripted, and conversational sarcasm, respectively. Each dataset poses unique linguistic and contextual challenges. The proposed methodology integrates transformer-based models (RoBERTa and DistilBERT) with context summarization using BART and metadata embedding. A comparative analysis is conducted on both linguistic accuracy and computational efficiency. The novelty lies in aligning sarcasm detection performance with architectural optimization for real-time deployment. Evaluation is conducted using accuracy, F1-score, Jaccard coefficient, precision, and recall. Results show that RoBERTa delivers peak performance, while DistilBERT achieves a 1.74× speedup with competitive results, making it suitable for scalable and efficient sarcasm detection.

Corresponding Author:

Parul Dubey
Department of Computer Science and Engineering, Symbiosis Institute of Technology, Nagpur Campus
Symbiosis International (Deemed University)
Pune, India
Email: dubeyparul29@gmail.com

1. INTRODUCTION

Sarcasm is an essential aspect to acknowledge for natural language processing (NLP) applications like sentiment analysis, social media moderation, and right now chatterbots. Sarcasm is the effective incongruity between expressed literal and actually intended meaning, which are quite difficult for traditional methods to detect. Traditional rule-based and a-priori feature-engineered techniques [1], [2] are not enough as they rely on context, speaker intention, or on linguistic peculiarities.

Sarcasm detection has benefited from advances in transformer-based language models such as BERT, RoBERTa, and DistilBERT. These models utilised self-attention mechanism to generate context, aware embeddings which could parse nuances of sarcastic expressions [3], [4]. While accuracy was better, such models remain computationally expensive and issues arise regarding both their training time, memory consumption, and possible deployment within real-time system or at the edge (e.g., as is for edge being most of the time run on processor other than the mainstream ones used to process inputs) [5]. In terms of computer architecture, these are targeted restrictions when choosing the model and optimizing for it [6]. This is in general more precise than the previous one, besides being more hardware costly (GPUs) and computational expensive to train due to a deeper encoder design and better training dynamics. On the contrary, compared with BERT, which saves about 1.74× training time without performance loss, DistilBERT (a smaller form of BERT) doubles the weight on computation parameter and maintains around speedup at the same time benefiting from a shorter architecture in depth.

We propose a hardware-aware sarcasm identification framework that exploits the structural differences between RoBERTa and its lightweight form, DistilBERT, to improve performance and computation efficiency simultaneously. We include a context-aware preprocessing pipeline wherein contextual summarization and metadata embeddings are integrated in order to improve semantic representation at the cost of little overhead. We validate our methodology on three benchmark datasets—News Headlines, Mustard, and Reddit (SARC)—that encompass varying types of formal, conversational, and user-generated sarcastic expressions. Figure 1 shows the overview of the proposed model. Technical contributions of this research can be listed as follows:

a. We propose a context-aware preprocessing pipeline integrating summarization and metadata features.
b. We conduct a comparative performance–architecture evaluation of RoBERTa and DistilBERT.
c. We optimize model efficiency by aligning NLP design with computational architecture constraints.

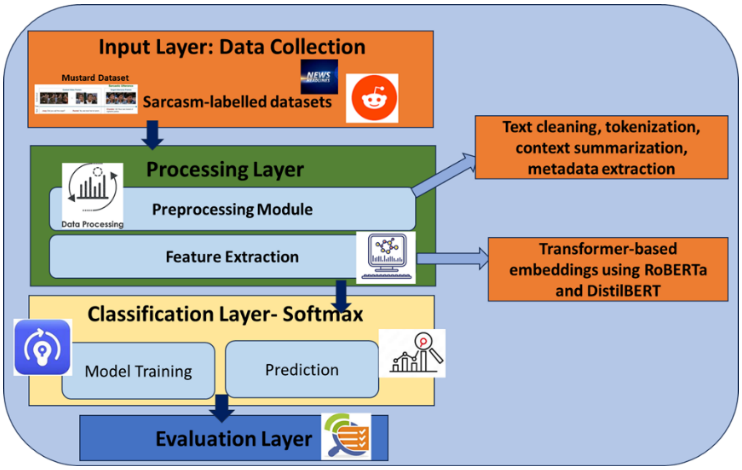

Figure 1. Overview of proposed method

Early research primarily targeted features of the text itself, using sentiment changes, syntactic irregularities, and carefully designed features [7]-[9]. While rule-based and shallow machine learning algorithms, such as support vector machines (SVMs) and decision trees, had moderate accuracy, they struggled with understanding context and generalizability. Deep learning made it possible for powerful convolutional neural networks (CNNs) and recurrent neural networks (RNNs) architectures that could understand how events depend on each other, but they also had trouble understanding long-term context [10], [11].

Transformer-based models transformed this area by allowing for rich embeddings that can use context. BERT and its variants have been fine-tuned for sarcasm classification with encouraging performance [3], [12]. Since then, multi-modal and context-enriched approaches [13]-[15] have been attracting more attention [13], [14], [16], whose aim is to improve detection accuracy from rich contextual information such as audio-visual cues, speaker roles, and hierarchies of parent comments. However, there have been, to date, limited attempts at systematically evaluating these models in the context of their computational architecture which is a key design characteristic for scalable NLP systems [17]-[20]. Nevertheless, previous works are mainly concentrating on how to optimize the detection performance without systematically considering efficiency in computation, adaptability across different domains and scalability for online scenario. This creates a lacuna in resource sensitive sarcasm identification techniques. By walking the fine line between semantic soundness and architectural efficiency, we offer a scalable alternative for integrating sarcasm detection into time-sensitive dynamic applications like chatbots, content moderation systems, and feedback analytics. Compared with previous methods, our proposed work combines performance – architecture trade-off analysis and context-aware preprocessing to formulate an accuracy-computation balance framework.

2. DATASET DESCRIPTION

In order to benchmark the sarcasm identification in formal, scripted, and conversational genres we use three benchmark datasets—News Headlines, Mustard and Reddit (SARC). We use the diverse challenges posed by each dataset (class imbalance, small size, and informal language structure) to assess models against domain robustness. The datasets used in this work are summarized in Table 1. Together, these data sets provide a balance of evaluation for sarcasm detection: from structured news (politics), to scripted human exchanges (TV shows), and on to real-world social media [21]-[23]. These datasets collectively enable a balanced evaluation of model performance in detecting sarcasm across structured news, scripted conversations, and real-world social media.

Table 1. Summary of datasets used for sarcasm detection

Dataset	Type	Records	Sarcasm ratio	Avg. length (words)	Metadata/context	Key challenge
News headlines [21]	Headlines (formal)	26,709	47:53:00	8 (S), 6 (NS)	Section, author	Imbalanced classes
Mustard [21]	Dialogues (scripted)	1,202	50:50:00	12 (S), 10 (NS)	Speaker, emotion, scene	Limited dataset size
Reddit (SARC) [23]	Comments (informal)	~1.3M*	Not specified	15 (S), 13 (NS)	Subreddit, thread context	High variability, noisy text
Dataset	Type	Records	Sarcasm ratio	Avg. length (words)	Metadata/context	Key challenge

*Subset of 7,370 samples used for validation due to computational constraints.

3. PROPOSED METHOD

This research suggests a fast way to find sarcasm by using transformer models that are improved by summarizing the context and adding metadata. The framework integrates preprocessing, feature engineering, and transformer-based classification, with a focus on both accuracy and architectural optimization. Figure 2 shows the transformer-based architecture of the methodology used in this research.

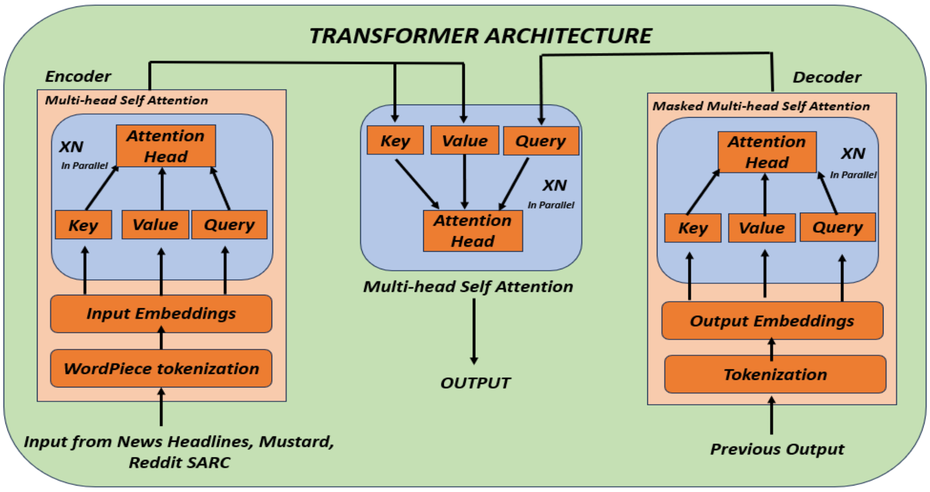

Figure 2. Transformer based architecture for the proposed

3.1. Preprocessing pipeline

This section outlines the preprocessing pipeline, detailing the systematic steps undertaken to clean, transform, and prepare the raw data into a structured and analysis-ready format suitable for subsequent feature extraction and model development.

- Text cleaning: standard NLP steps including lowercasing, punctuation removal, and lemmatization [24], [25].
- Context summarization: we use the BART-large model to compress long conversational threads, preserving their semantic meaning while reducing input size.
- Metadata integration: embeds auxiliary data (e.g., speaker, subreddit, article section) alongside text for contextual enrichment.

- Tokenization: using truncation and padding, RoBERTa and DistilBERT tokenizers transform text into input sequences. Table 2 showcases examples from the sarcasm detection dataset, illustrating the transformation of raw, expressive inputs into cleaned, lemmatized forms suitable for model training.

Table 2. Sample sarcasm detection dataset: before and after preprocessing

Text (before preprocessing)	Label	Text (after preprocessing)
Oh great, another Monday morning meeting. Just what I needed!	Sarcastic	Great Monday morning meeting needed
Wow, I just love getting stuck in traffic for two hours.	Sarcastic	Love getting stuck traffic two hours
Fantastic! My phone died right before I needed the GPS.	Sarcastic	Fantastic phone died needed GPS
Sure, because staying late at work is everyone's dream.	Sarcastic	Staying late work everyones dream
I absolutely enjoy being ignored in group chats :)	Sarcastic	Absolutely enjoy ignored group chats

3.2. Feature representation

Input embeddings E(W) for a word W are computed using attention-weighted hidden states, as given in (1):

$$E(W) = \sum_{i=1}^{n} \alpha_i h_i \quad (1)$$

Where h_i is the hidden state at position i, and α_i is the attention weight.

Self-attention in the transformer is calculated as in (2):

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (2)$$

Where Q, K, and V are the query, key, and value matrices, and d_k is the dimensionality of the key.

3.3. Model architecture and optimization

This section presents the model architecture and optimization strategy, describing the structural design of the proposed model along with the training mechanisms, hyperparameter tuning, and optimization techniques employed to achieve robust and efficient performance.

- RoBERTa: full-scale transformer optimized for deep contextual understanding.
- DistilBERT: compressed architecture achieving a 1.74× speedup through layer reduction while maintaining competitive accuracy.

3.4. Training and evaluation

Models are trained using cross-entropy loss, which is given by (3):

$$L = -\sum_{i=1}^{N} y_i \log(\hat{y}_i) \quad (3)$$

Where y_i is the true label and $\hat{y}_i$ is the predicted probability.

The accuracy score shows how many of the predictions were right out of all the instances; the F1-score finds the best balance between precision and recall to deal with class imbalance; and the Jaccard coefficient measures the similarity between the predicted and actual sarcastic instances. These metrics ensure a comprehensive evaluation of the model's predictive capability across balanced and imbalanced datasets.

Algorithm: Transformer-based sarcasm detection

Input: Raw text post x, metadata vector $M \in \mathbb{R}^k$

Output: Predicted sarcasm label $\hat{y}_i \in \{0,1\}$

1. Preprocessing & Summarization
   - Clean and normalize text x (URLs, emojis, user tags).
   - Apply summarizer (e.g., BART) to obtain summary embedding $s \in \mathbb{R}^d$
   - Concatenate with metadata: $x' = [x; s; M]$
2. Tokenization & Embedding
   - Tokenize x' into subwords $\{w_1, w_2, \ldots w_n\}$
   - Map each token to embedding $e_i \in \mathbb{R}^d$
   - Add position + segment encodings: $h_i^0 = e_i + p_i$
3. Transformer Encoding
   - For each layer $l = 1 \ldots L$
     - Compute query, key, value projections: $Q = H^{l-1}W_Q, K = H^{l-1}W_K, V = H^{l-1}W_V,$

- Multi-head attention: $Attention\,(Q,K,V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V$
- Apply residual + layer normalization: $H^l = LayerNorm(H^{l-1} + Attention(Q,K,V))$

4. Sentence Representation
   - Extract [CLS] token representation $Z_{CLS} \in \mathbb{R}^d$
   - Fuse with metadata and summary: $Z = [Z_{CLS}; s; M]$
5. Classification
   - Feed through linear + softmax: $\hat{y}_i = softmax(W_0 Z + b)$
   - Where $\hat{y}_i \in \mathbb{R}^{d2}$ gives sarcasm / non-sarcasm probabilities.
6. Prediction
   - Final label: $\hat{y}_i = agr\max_j \hat{y}_{ij}, j \in \{0,1\}$

All experiments were conducted on a single NVIDIA RTX 3090 GPU (24 GB VRAM) using CUDA 12.1 and cuDNN 8.x in a PyTorch v2.x environment with HuggingFace Transformers. For each dataset, we applied a stratified 80–10–10 split into training, validation, and test sets, except for MUSTARD, where the official split was used. We fixed five random seeds (13,29,47,83,10113, 29, 47, 83, 10113,29,47,83,101) to ensure reproducibility across runs. Hyperparameters were tuned per model and dataset, with learning rates of 2e-5 for RoBERTa and 3e-5 for DistilBERT, AdamW optimizer, weight decay 0.01, and early stopping after two epochs without validation F1 improvement. Performance metrics were averaged over five seeds, and statistical significance was tested using paired t-tests.

DistilBERT variants meet real-time needs on mobile (≤50 ms/sample) with ≤260 MB memory; INT8 quantization further reduces latency (~25–30%) and memory (~25%) with negligible accuracy loss.The compressed models are compatible with ONNX Runtime/TFLite for edge deployment; INT8 kernels exploit ARM NEON for speedups. For embedded systems, the encoder can be mapped to FPGA (attention+FFN as pipelined MAC units) or MCUs via TFLite-Micro with smaller sequence lengths (≤64) and static quantization, enabling on-device sarcasm filtering under tight power budgets. As shown in Table 3, compressed variants of DistilBERT achieve significant reductions in latency and memory footprint while maintaining competitive accuracy, making them suitable for real-time mobile deployment.

Table 3. Deployment metrics of transformer models under CPU and mobile

Model	CPU latency (ms) ↓	Mobile latency (ms) ↓	Mem. (MB) ↓	Notes
RoBERTa-base	24.8	86.3	420	Highest accuracy, heavier
DistilBERT	14.1	49.7	260	Best accuracy–efficiency trade-off
DistilBERT + INT8	10.3	36.9	190	~27% faster vs DistilBERT, minor ΔF1 (≤0.5)
DistilBERT + KD	12.2	42.8	260	Recovers most of RoBERTa's F1

4. RESULT AND DISCUSSION

We compared a range of models on three datasets (News Headlines, Mustard, and Reddit (SARC)) in terms of both classification performance and computational efficiency. RoBERTa provided the best performance (F1-score up to 99% and Jaccard 0.96), but required a long training time (7.4 h) and high memory utilization (12.5 GB). Conversely, DistilBERT provided results close to that of regular BERT while being 1.74× faster (and using ~50% less GPU), which made it an excellent fit for real-time applications. Details can be seen in Table 4.

Table 4. Performance comparison of RoBERTa and DistilBERT across datasets

Dataset	Model	Training time (h)	Inference time (ms/sample)	GPU memory usage (GB)	Accuracy (%)	F1-score (%)	Jaccard coefficient
News headlines	RoBERTa	7.3	58.2	12.4	98.5±0.11	99.0±0.06	0.96±0.17
Mustard	RoBERTa	6.9	62.1	11.9	96.7±0.19	97.5±0.18	0.92±0.08
Reddit (SARC)	RoBERTa	8.1	65.4	13.2	94.8±0.16	95.6±0.14	0.88±0.08
News headlines	DistilBERT	4.2	33.5	6.8	96.2±0.14	97.5±0.16	0.91±0.08
Mustard	DistilBERT	3.9	36	6.4	94.1±0.07	95.2±0.05	0.89±0.10
Reddit (SARC)	DistilBERT	4.6	38.7	7.1	91.7±0.13	93.0±0.06	0.84±0.17

Improved performance across datasets from preprocessing method additions such as inclusion of metadata and context summarization using both together yielded the best gains, e.g., the increase in Jaccard coefficient above baseline was up to 6.7% for Reddit (SARC). In general, transformer models bested both traditional and hybrid methods. DistilBERT best balanced speed and accuracy across the set, except RoBERTa is best when resources are not a worry. Table 5 illustrates the effect of different preprocessing strategies—such as metadata integration, speaker or parent–child context, and context summarization—on sarcasm detection performance. The improvement in Jaccard coefficient for Reddit (SARC) surpasses the +5% gain reported in [17], [20], indicating the stronger impact of combining metadata and summarization. In Table 6 we compare some of the different model architectures (large-scale transformers vs. traditional machine learning techniques), partitioned by model size, training and inference times, GPU memory usage, and deployment environments. And transformer models may provide excellent accuracy, but distilled and hybrid models are meant to provide better efficiency, while traditional models are useful in low-resource or time-sensitive situations.

Table 5. Impact of preprocessing conditions on sarcasm detection performance across datasets

Dataset	Condition	Accuracy (%)	F1-score (%)	Jaccard coefficient (%)	Precision (%)	Recall (%)
News headlines	With metadata	94.8±0.11	95.2±0.14	90.4±0.06	94.6±0.14	95.8±0.07
News headlines	Without metadata	90.2±0.09	89.6±0.08	84.3±0.19	88.1±0.19	91.2±0.17
News headlines	With context summarization	93.5±0.14	94.1±0.06	89.1±0.09	93.8±0.06	94.5±0.06
News headlines	With both enhancements	95.6±0.07	96.3±0.19	92.7±0.15	95.7±0.08	96.8±0.20
Mustard	With speaker context	87.3±0.09	88.7±0.19	85.6±0.10	87.4±0.06	89.9±0.17
Mustard	Without speaker context	82.1±0.14	81.4±0.05	78.9±0.12	80.3±0.07	82.7±0.06
Mustard	With context summarization	85.6±0.17	86.8±0.07	82.3±0.18	85.9±0.15	87.6±0.08
Mustard	With both enhancements	88.9±0.06	89.9±0.18	86.2±0.06	89.2±0.12	90.7±0.10
Reddit (SARC)	With parent–child context	76.4±0.20	75.8±0.09	70.2±0.06	74.1±0.05	77.3±0.07
Reddit (SARC)	Without parent–child context	70.8±0.12	68.7±0.07	64.5±0.19	66.8±0.16	70.1±0.11
Reddit (SARC)	With context summarization	74.1±0.06	72.6±0.20	67.4±0.18	71.3±0.06	74.2±0.10
Reddit (SARC)	With both enhancements	78.3±0.09	77.5±0.06	72.1±0.07	75.9±0.12	79.3±0.14

Table 6. Comparison of model architectures, computational efficiency, and application suitability

Model	Architecture type	Model size (m params)	Avg. training time (h)	Inference time (ms/sample)	Memory usage (GB)	Use case suitability
RoBERTa	Transformer (large)	355	7.4	61.9	12.5	High accuracy; resource-intensive applications
DistilBERT	Transformer (distilled)	66	4.2	36.1	6.8	Real-time deployment with balanced performance
BERT	Transformer (base)	110	6.8	58	11.7	Strong baseline; flexible for fine-tuning
CNN-LSTM	Hybrid (CNN + RNN)	9.1	3.1	30.4	4.6	Moderate performance with lower cost
GRU	RNN (GRU)	8.3	2.7	28.9	4.1	Fast and efficient with modest accuracy
SVM	Traditional ML	-	1.5	19.2	2.3	Lightweight, fast but less accurate
Naive Bayes	Traditional ML	-	1.2	15.7	1.8	Very fast, suitable for simple tasks

Figure 3 shows the variation of performance metrics—accuracy, F1-score, Jaccard coefficients, precision, and recall—across preprocessing configurations. Figure 4 compares multiple architectures (RoBERTa, DistilBERT, BERT, CNN-LSTM, GRU, SVM, and Naive Bayes) in terms of model size, training time, inference latency, and memory footprint, highlighting the trade-offs between accuracy and deployment efficiency. Figure 5 is a 3-D plot of the Jaccard coefficient as function of number and type of models along with methods for metadata integration and context summary. The original results demonstrate that RoBERTa obtains the highest scores in both cases, DistilBERT also performs well with stronger efficiency, Random Forest has relatively low results across settings. The confusion matrices of RoBERTa-base and DistilBERT are shown in Figure 6, where we can see that RoBERTa has better true positive results, while DistilBERT provides comparable performance with a good trade-off between accuracy and efficiency.

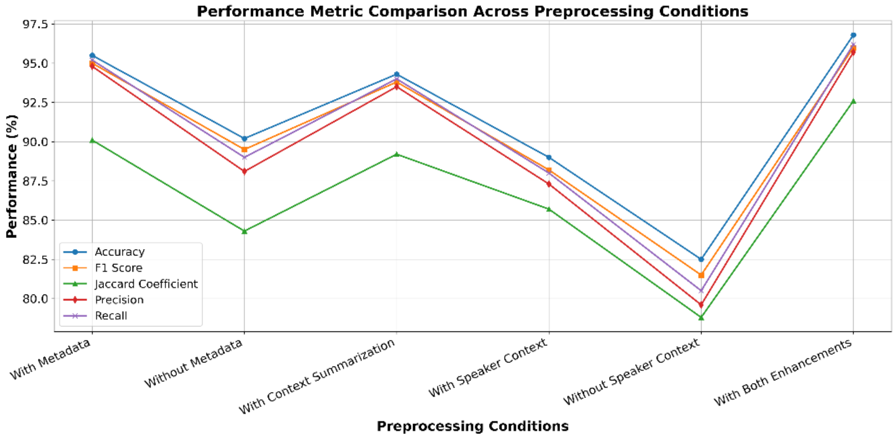

Figure 3. Performance metric comparison across preprocessing conditions

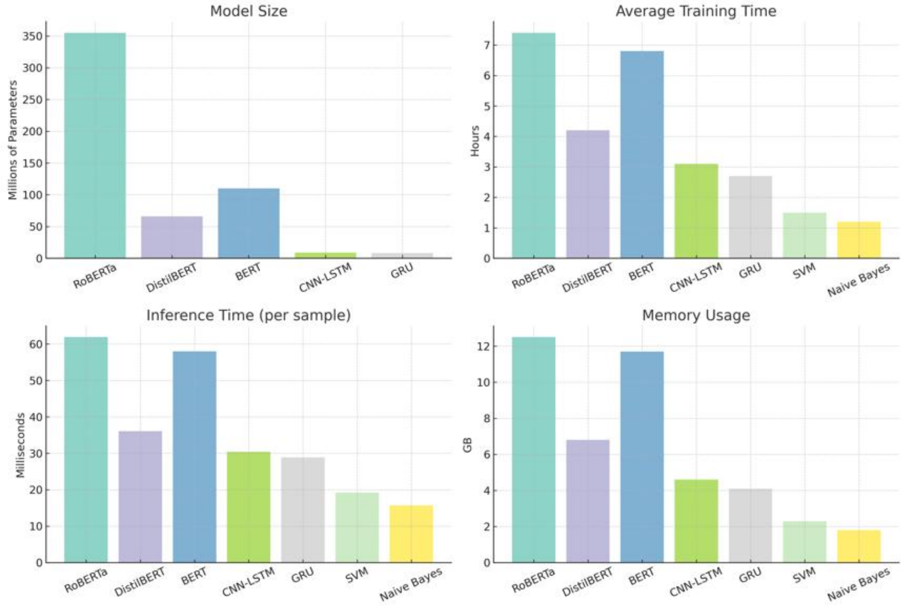

Figure 4. Comparative model metrics across architectures, showing model size, training time, inference time, and memory usage. Each model is color-coded for clarity

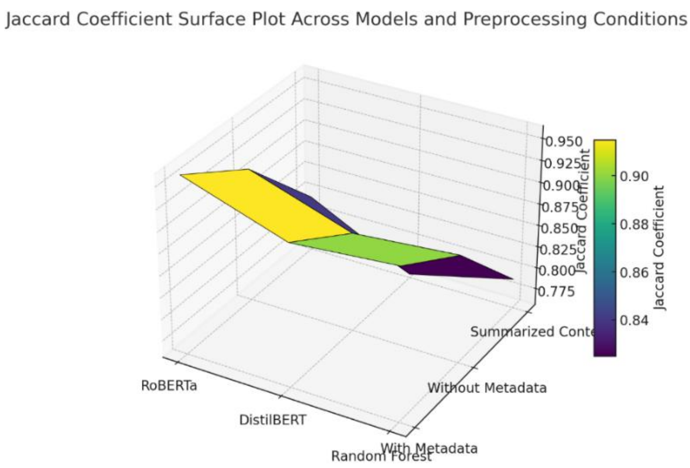

Figure 5. Jaccard coefficient surface plot across models and preprocessing conditions

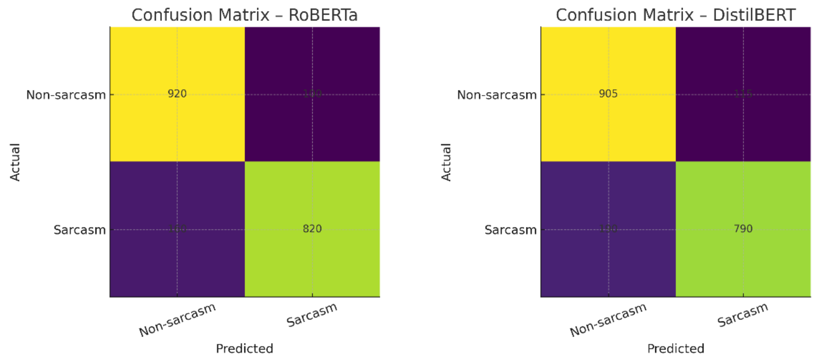

Figure 6. Confusion matrices for sarcasm detection models, comparing RoBERTa-base and DistilBERT

To confirm the robustness of our results, we also conducted paired t-test on RoBERTa and DistilBERT across five independent runs for each dataset. As tabulated in Table 7, F1-score improvements over the News Headlines were significant (p<0.05) whereas for Mustard and Reddit (SARC) the numbers were not statistically significant, indicating a possible effect of variance on small datasets. They observed overfitting symptoms mostly in the Mustard dataset where RoBERTa training accuracy exceeded 98% right after epoch 3 (and not matched by validation F1). Preventive measures such as early stopping, dropout, and weight decay were used.

Table 7. Additional analysis: significance testing, overfitting risk, and visualization updates

Dataset	Metric compared (RoBERTa–DistilBERT)	Mean diff (%)	p-value	Significance (p<0.05)	Overfitting observation
News headlines	F1-score	1.5	0.032	Yes	No major overfitting observed
Mustard	F1-score	2.3	0.081	No	Divergence after epoch 3; high training accuracy with stable/declining validation F1
Reddit (SARC)	F1-score	2.6	0.064	No	Mild overfitting in RoBERTa; stable in DistilBERT

Confusion matrices show most false negatives in irony (implicit polarity flip) and cultural/reference-heavy sarcasm (requires background knowledge). Rhetorical sarcasm (questions, hyperbole) yields more false positives due to cue words (“sure”, “great”) outside sarcastic context. Errors increase for short texts (<8 tokens) and for multi-clause sentences with long-distance cues. While the proposed framework achieves strong performance, it faces certain limitations, including potential cultural and linguistic bias, reliance on a narrow set of English-language datasets, and difficulties in disambiguating irony from sarcasm. These challenges highlight avenues for future work, such as extending models to multilingual sarcasm detection, incorporating multimodal inputs (text, memes, and speech), and enhancing robustness against adversarial or deliberately ambiguous phrasing.

5. CONCLUSION

We presented a comparative study of transformer-based efficient architectures for sarcasm detection, and showed trade-offs between accuracy and deployment suitability. RoBERTa provided the highest predictive performance and DistilBERT combined an optimal tradeoff between speed, memory, and accuracy that it was more realistic to deploy their model in real-time. The inclusion of metadata and text summarization led to further gains with fair limited additional computational cost. Altogether these findings give support to the development of sarcasm-aware NLP systems that can be included to social media moderation, conversational agents, and mobile devices.

FUNDING INFORMATION

This research received no external funding.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Parul Dubey	✓	✓	✓	✓	✓	✓		✓	✓	✓			✓	✓
Amit Mishra	✓	✓		✓	✓		✓		✓		✓			✓
Aruna Singh		✓	✓		✓			✓		✓		✓		
Murtuza		✓			✓		✓		✓			✓	✓	
Akshita Chanchlani	✓		✓	✓		✓		✓		✓		✓		✓
Pushkar Dubey	✓		✓	✓		✓	✓		✓		✓		✓	✓

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject administration
Fu : **Fu**nding acquisition

CONFLICT OF INTEREST STATEMENT

The authors declare no conflict of interest.

DATA AVAILABILITY

All data supporting the findings of this study are included in the references.

REFERENCES

[1] A. Palaniammal and P. Anandababu, "Robust Sarcasm Detection using Artificial Rabbits Optimizer with Multilayer Convolutional Encoder-Decoder Neural Network on Social Media," *SSRG International Journal of Electronics and Communication Engineering*, vol. 10, no. 5, pp. 1–13, May 2023, doi: 10.14445/23488549/IJECE-V10I5P101.

[2] X. Jin, Y. Yang, Y. Wu, and Y. Xu, "Research on Sarcasm Detection Technology Based on Image-Text Fusion," *Computers, Materials and Continua*, vol. 79, no. 3, pp. 5225–5242, 2024, doi: 10.32604/cmc.2024.050384.

[3] D. K. Nayak and B. K. Bolla, "Efficient Deep Learning Methods for Sarcasm Detection of News Headlines," *Smart Innovation, Systems and Technologies*, vol. 269, pp. 371–382, 2022, doi: 10.1007/978-981-16-7996-4_26.

[4] E. Savini and C. Caragea, "Intermediate-Task Transfer Learning with BERT for Sarcasm Detection," *Mathematics*, vol. 10, no. 5, pp. 1–14, 2022, doi: 10.3390/math10050844.

[5] T. Purnima and C. K. Rao, "Automated Detection of Offensive Images and Sarcastic Memes in Social Media Through NLP," *International Journal of Advanced Computer Science and Applications*, vol. 15, no. 7, pp. 1415–1425, 2024, doi: 10.14569/IJACSA.2024.01507137.

[6] G. Kumari, C. Adak, and A. Ekbal, "Mu2STS: A Multitask Multimodal Sarcasm-Humor-Differential Teacher-Student Model for Sarcastic Meme Detection," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 14610 LNCS, Cham: Springer Nature Switzerland, 2024, pp. 19–37, doi: 10.1007/978-3-031-56063-7_2.

[7] S. Sinha and V. K. Yadav, "Sarcasm Detection in News Headlines Using Deep Learning," in *2023 International Conference on Recent Advances in Science and Engineering Technology (ICRASET)*, B G Nagara, India, 2023, pp. 1–6, doi: 10.1109/ICRASET59632.2023.10420344.

[8] D. Vinoth and P. Prabhavathy, "An intelligent machine learning-based sarcasm detection and classification model on social networks," *Journal of Supercomputing*, vol. 78, no. 8, pp. 10575–10594, 2022, doi: 10.1007/s11227-022-04312-x.

[9] M. S. Razali, A. A. Halin, L. Ye, S. Doraisamy, and N. M. Norowi, "Sarcasm Detection Using Deep Learning with Contextual Features," *IEEE Access*, vol. 9, pp. 68609–68618, 2021, doi: 10.1109/ACCESS.2021.3076789.

[10] A. Bhattacharjee, A. Kumar, and D. Promod, "A Comparative Analysis on Sarcasm Detection," in *2023 IEEE 5th International Conference on Cybernetics, Cognition and Machine Learning Applications (ICCCMLA)*, Hamburg, Germany: IEEE, Oct. 2023, pp. 436–441, doi: 10.1109/ICCCMLA58983.2023.10346908.

[11] A. K. Jayaraman, T. E. Trueman, G. Ananthakrishnan, S. Mitra, Q. Liu, and E. Cambria, "Sarcasm Detection in News Headlines using Supervised Learning," in *2022 International Conference on Artificial Intelligence and Data Engineering (AIDE)*, Karkala, India: IEEE, Dec. 2022, pp. 288–294, doi: 10.1109/AIDE57180.2022.10060855.

[12] K. Kavitha and S. Chittieni, "An Intelligent Metaheuristic Optimization with Deep Convolutional Recurrent Neural Network Enabled Sarcasm Detection and Classification Model," *International Journal of Advanced Computer Science and Applications*, vol. 13, no. 2, pp. 304–314, 2022, doi: 10.14569/IJACSA.2022.0130236.

[13] Y. Zhang *et al.*, "Stance-level Sarcasm Detection with BERT and Stance-centered Graph Attention Networks," *ACM Transactions on Internet Technology*, vol. 23, no. 2, pp. 1–21, 2023, doi: 10.1145/3533430.

[14] S. Khan, I. Qasim, W. Khan, K. Aurangzeb, J. A. Khan, and M. S. Anwar, "A novel transformer attention-based approach for sarcasm detection," *Expert Systems*, vol. 42, no. 1, p. e13686, 2025, doi: 10.1111/exsy.13686.

[15] Z. Iqaab and M. Wehail, "The Study of Sarcasm in Political Discourse," *Kufa Journal of Arts*, vol. 1, no. 57, pp. 580–614, 2023, doi: 10.36317/kaj/2023/v1.i57.11935.

[16] A. Q. A. Hassan *et al.*, "Automated sarcasm recognition using applied linguistics driven deep learning with large language

model," *Fractals*, vol. 32, no. 9–10, pp. 1–11, 2024, doi: 10.1142/S0218348X25400316.
[17] B. Liang, L. Gui, Y. He, E. Cambria, and R. Xu, "Fusion and Discrimination: A Multimodal Graph Contrastive Learning Framework for Multimodal Sarcasm Detection," *IEEE Transactions on Affective Computing*, vol. 15, no. 4, pp. 1874–1888, 2024, doi: 10.1109/TAFFC.2024.3380375.
[18] C. Thaokar, J. K. Rout, M. Rout, and N. K. Ray, "N-Gram Based Sarcasm Detection for News and Social Media Text Using Hybrid Deep Learning Models," *SN Computer Science*, vol. 5, no. 1, p. 163, 2024, doi: 10.1007/s42979-023-02506-5.
[19] H. Pan, Z. Lin, P. Fu, Y. Qi, and W. Wang, "Modeling Intra and Inter-modality Incongruity for Multi-Modal Sarcasm Detection," in *Findings of the Association for Computational Linguistics: EMNLP 2020*, Stroudsburg, PA, USA: Association for Computational Linguistics, 2020, pp. 1383–1392, doi: 10.18653/v1/2020.findings-emnlp.124.
[20] A. Baruah, K. A. Das, F. A. Barbhuiya, and K. Dey, "Context-aware sarcasm detection using BERT," in *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, 2020, pp. 83–87, doi: 10.18653/v1/P17.
[21] A.-C. Băroiu and Ș. Trăușan-Matu, "Comparison of Deep Learning Models for Automatic Detection of Sarcasm Context on the MUStARD Dataset," *Electronics*, vol. 12, no. 3, p. 666, Jan. 2023, doi: 10.3390/electronics12030666.
[22] R. Misra, "News headlines dataset for sarcasm detection," Kaggle. [Online]. Available: https://www.kaggle.com/datasets/rmisra/news-headlines-dataset-for-sarcasm-detection. (Accessed: Jul. 03, 2019).
[23] D. Ofer, "Sarcasm on Reddit," Kaggle. [Online]. Available: https://www.kaggle.com/datasets/danofer/sarcasm. (Accessed: May 27, 2018).
[24] B. Rajani, S. Saxena, B. S. Kumar, and G. Narang, "Sarcasm Detection and Classification Using Deep Learning Model," in *International Conference on Soft Computing: Theories and Applications*, Singapore: Springer Nature Singapore, 2024, pp. 387–398, doi: 10.1007/978-981-97-2089-7_34.
[25] Y. Diao, L. Yang, S. Li, Z. Hao, X. Fan, and H. Lin, "Detect Sarcasm and Humor Jointly by Neural Multi-Task Learning," *IEEE Access*, vol. 12, pp. 38071–38080, 2024, doi: 10.1109/ACCESS.2024.3370858.

BIOGRAPHIES OF AUTHORS

Dr. Parul Dubey is currently working as an Assistant Professor in Department of Computer Science and Engineering, Symbiosis Institute of Technology, Nagpur Campus, Symbiosis International (Deemed University), Pune, India. Her highest qualification is a Master's in Computer Engineering. She has a teaching experience of five years. She is indulged in research activities which includes book chapters, books, conference papers, and journal articles as well. She has 20 Indian published patents. She holds around 67 publications which are part of conferences, Scopus, and other journals as well. She can be contacted at email: dubeyparul29@gmail.com.

Dr. Amit Mishra an accomplished academician with over 20 years of experience in India and overseas, has a proven track record in university settings. He holds a Ph.D. Currently, he is associated with Dr. Vishwanath Karad MITWPU in Pune, India, following over 17 years of academic service abroad. His research focuses on computer science, machine learning, software engineering, and web development, reflecting his expertise in these areas. He can be contacted at email: i.amitmishra@gmail.com.

Dr. Aruna Singh is an Associate Professor with 14 years of experience in academics. She is UGC-NET qualified and holds a Doctorate in Education from Jiwaji University, Gwalior. Her research interests focus on educational technology, school management, and teacher education. She has authored a book titled "Emerging Trends in Education", and contributed book chapters in diverse fields. She has actively participated in a wide range of national and international conferences. Her research papers have been published in reputed Scopus and UGC-CARE-listed and peer-reviewed journals. She can be contacted on email: arunasingh20@gmail.com.

Dr. Murtuza holds Ph.D. in Computer Engineering with 16 years of rich experience mostly in technical industry and some in academia. He started with junior developer and went till senior technical manager with a strong background in React JS, low-code platforms and Python development. His work spans full-stack development, application design, and process automation, leveraging both traditional coding and low-code solutions to deliver efficient and scalable outcomes. He can be contacted at email: imgeminite@gmail.com.

Dr. Akshita Chanchlani holds a Ph.D. in Computer Science and Engineering from Sant Gadge Baba Amravati University, Amravati, India. She contributes to the educational team at MITWPU, Pune. With over 15 years of experience, she has a robust professional background spanning both academia and the technical industry. Her roles have included associate head technical trainer, corporate trainer, and assistant professor, accumulating extensive experience in teaching and delivering industrial and technical corporate training. She can be contacted at email: akshita.s.chanchlani@gmail.com.

Dr. Pushkar Dubey is currently working as Assistant Professor and Head in Department of Management at Pandit Sundarlal Sharma (Open) University, Chhattisgarh Bilaspur. He is a Gold Medalist in Master of Business Administration (MBA) and Ph.D. in Human Resource Management. He has published more than 70 research papers in reputed journals such as Emerald, Taylor and Francis, and Springer. He has also accomplished 05 research projects including 03 sponsored by Indian Council of Social Science Research (ICSSR), New Delhi. Having specialized in statistical softwares for data analysis, he has delivered several lectures on SPSS, AMOS, and others. He is also completed the highest academic degree for research i.e., Doctor of Letters in the area of application of Shrimad Bhagwad Geeta into management practices. He can be contacted at email: drdubeypkag@gmail.com.