Subtask 3: Model Analysis, Enhancement & Comparative Study
Sarcasm Detection in English–Hindi Code-Mixed Tweets

Jash Shah
202201016
Dhirubhai Ambani University

November 2025

Abstract

This report presents a comprehensive analysis of baseline sarcasm detection models, followed by systematic enhancements targeting improved generalization, robustness, and semantic understanding. Building upon Subtask 1 and Subtask 2, the study expands the dataset, refines preprocessing, and evaluates multiple modeling strategies including TF-IDF-based classical models, FastText-based embedding models, and a neural BiLSTM architecture. The findings highlight clear improvements from the enhanced pipeline and provide insights into design choices that influence performance.

1 Introduction

Sarcasm detection in code-mixed English–Hindi (Hinglish) social media content is challenging due to transliteration, inconsistent grammar, and domain-specific expressions. Previous subtasks established the dataset foundations and implemented a TF-IDF Random Forest baseline. This subtask aims to:

- analyze strengths and limitations of the baseline
- introduce a substantive enhancement to the modeling pipeline

- compare baseline and enhanced models quantitatively and qualitatively
- provide insights guiding future improvements

2 Dataset Expansion and Preprocessing Enhancements

2.1 Dataset Combination

Three datasets were merged into a unified corpus:

- Akshita Aggarwal dataset (9 samples)
- HackArena dataset (9840 samples)
- Swami et al. dataset (5250 samples) – used in Subtask 2

After cleaning and standardizing columns:

Total samples: 15099

Label distribution:

Not sarcastic: 8091, Sarcastic: 7008

2.2 Improved Preprocessing Pipeline

The enhanced preprocessing includes:

- normalization to lowercase
- removing punctuation
- removing English stopwords
- consistent text tokenization

These improvements help reduce vocabulary sparsity and noise, enhancing the signal for both sparse and dense representations.

3 Baseline Models

3.1 TF-IDF + Random Forest (Baseline)

Using the combined dataset:

$$\text{Train size: } 12079, \quad \text{Test size: } 3020$$

Results:

- Test Accuracy: 0.9755
- Test F1 Score: 0.9755

Confusion matrix:

$$\begin{bmatrix} 1611 & 7 \\ 67 & 1335 \end{bmatrix}$$

3.2 TF-IDF + Linear SVM (Baseline)

Results:

- Test Accuracy: 0.9762
- Test F1 Score: 0.9762

Confusion matrix:

$$\begin{bmatrix} 1587 & 31 \\ 41 & 1361 \end{bmatrix}$$

4 Enhanced Models

The enhancement introduced in this subtask is the transition from sparse vectorization (TF-IDF) to dense **FastText embeddings**, providing semantic and subword-level representation that is better suited for code-mixed Hinglish text.

Additionally, a neural **BiLSTM** classifier was implemented to capture contextual and sequential patterns that classical models cannot represent.

4.1 FastText Embedding Training

The FastText model was trained with:

- dimension: 100
- min_count: 1
- epochs: 10

Vocabulary size: 28176 words.
The embeddings were averaged to produce 100-dimensional sentence vectors.

4.2 FastText + Random Forest (Enhanced)

Results:

- Test Accuracy: 0.9772
- Test F1 Score: 0.9771

Confusion matrix:

$$\begin{bmatrix} 1615 & 3 \\ 66 & 1336 \end{bmatrix}$$

Notable improvements:

- false positives reduced from 7 to 3
- maintains strong recall for the sarcastic class

4.3 FastText + SVM (Enhanced)

Results:

- Test Accuracy: 0.9493
- Test F1 Score: 0.9492

Confusion matrix:

$$\begin{bmatrix} 1589 & 29 \\ 124 & 1278 \end{bmatrix}$$

The drop in performance reflects that linear SVM benefits more from sparse TF-IDF features than dense embeddings.

4.4 BiLSTM Model

Using FastText embeddings as input:

- Test Accuracy: 0.9705
- Test F1 Score: 0.9705

Confusion matrix:

$$\begin{bmatrix} 1598 & 20 \\ 69 & 1333 \end{bmatrix}$$

BiLSTM captures sequential and contextual signals better than classical models but does not surpass the best classical baseline.

5 Comparative Analysis

5.1 Quantitative Comparison

Model	**Accuracy**	**F1 Score**
TF-IDF + Random Forest	0.9755	0.9755
TF-IDF + SVM	0.9762	0.9762
FastText + Random Forest	**0.9772**	**0.9771**
FastText + SVM	0.9493	0.9492
BiLSTM (FastText)	0.9705	0.9705

Table 1: Model Performance Comparison

5.2 Qualitative Observations

- **FastText embeddings** significantly improve robustness to spelling variation and Hinglish transliteration.
- **Random Forest consistently benefits** from both TF-IDF and FastText features.
- **SVM performs best with sparse TF-IDF** and struggles with dense embeddings.
- **BiLSTM models perform well**, capturing sequential dependencies but require more data to surpass classical methods.

6 Discussion of Design Choices

6.1 Impact of Dataset Expansion

Merging datasets increased both diversity and coverage of sarcasm patterns, yielding significant improvements across all models.

6.2 Effect of Enhanced Preprocessing

Stopword removal and punctuation cleaning helped reduce sparsity, improving classifier stability, especially for TF-IDF based models.

6.3 Effectiveness of FastText Upgrade

FastText represents subwords, enabling:

- better handling of Hinglish slang
- improved generalization across tweet topics
- reduced false positives for sarcasm detection

7 Lessons Learned

- Dense embeddings outperform sparse features for noisy, code-mixed data.
- Classical models (RF, SVM) remain highly competitive, especially with strong features.
- Neural networks need larger datasets and fine-tuning to outperform classical models.
- Preprocessing quality significantly influences model performance in sarcasm detection.

8 Future Work

Future improvements may include:

- hierarchical subword embeddings
- transformer-based models such as mBERT or IndicBERT
- augmentation via back-translation or paraphrasing
- sarcasm-specific contrastive learning

9 Conclusion

This subtask successfully demonstrates how dataset expansion, improved preprocessing, and enhanced semantic modeling contribute to better sarcasm detection performance. FastText embeddings emerged as the most effective enhancement, improving robustness while maintaining interpretability. The comparative analysis provides a clear roadmap for further advances in multilingual, code-mixed sarcasm detection.

References

1. Swami, S., Khandelwal, A., Singh, V., Akhtar, S. S., & Shrivastava, M. *A Corpus of English-Hindi Code-Mixed Tweets for Sarcasm Detection*. Available at: `https://arxiv.org/abs/1805.11869`

2. Aggarwal, A., Wadhawan, A., Chaudhary, A., & Maurya, K. *"Did you really mean what you said?" : Sarcasm Detection in Hindi-English Code-Mixed Data using Bilingual Word Embeddings*. Available at: `https://arxiv.org/abs/2010.00310`

3. Kaggle Dataset: *HackArena Theme 2 – Multilingual Sarcasm Detection*. Available at: `https://www.kaggle.com/datasets/divyanshu134/hackarena-theme-2-multilingual-sarcasm-detection/data`

4. Bedi, M., Kumar, S., Akhtar, M. S., & Chakraborty, T. *Multi-modal Sarcasm Detection and Humor Classification in Code-mixed Conversations*. Available at: `https://arxiv.org/abs/2105.09984`

5. GitHub Repository (Codebase): *sarcasm-lens*. Available at: `https://github.com/jash0803/sarcasm-lens`