Comparative Analysis Report

---

1. Introduction

This report presents a comparative evaluation of various classical RNN and LSTM-based models applied to sentiment analysis on multiple Hinglish (Hindi + English) datasets. The primary objective is to establish a baseline for model performance, measured primarily through classification accuracy, against which quantum models can later be compared. The evaluation spans a range of configurations, including architectural variations, embedding strategies, hidden dimensions, and training durations.

The datasets include:

Dataset	Description
fb_hi_cg_train2	Facebook-based Hindi code-switched dataset
hi_dataset/CR/{TrainX/TestX}	curated versions of Hindi review datasets for controlled training and testing

2. Model Architectures

A total of 18 configurations were evaluated across two model types - RNN and LSTM, including both standard and manually implemented variants, as well as enhancements such as multi-layer and bidirectional architectures.

Model Variant	Description
Vanilla RNN	Basic recurrent model with standard RNN cells
Manual RNN	Manually implemented RNN cell
Manual LSTM	Manually implemented LSTM units
Two-layered RNN	RNN with stacked recurrent layers for hierarchical representation
Bidirectional RNN	Incorporates both forward and backward temporal dependencies
LSTM (API)	Framework-supported LSTM model
Complex Model	Enhanced version of RNN with increased parameters and complexity
model_Dx	Custom RNN variants trained on partitioned datasets (Train1–Train4)

3. Embedding and Configuration Settings

Most models used **FastText embeddings** (100 or 200 dimensions), except the baseline Vanilla RNN, which used randomly initialized embeddings from a normal distribution.

Other key configurations:

- **Hidden Dimensions:** 32, 64, or 128
- **Epochs:** Varied across models; mostly 90–100, with longer runs (up to 800) for manually implemented RNNs and LSTMs
- **Class Weights:** Enabled in later models to address class imbalance

4. Training Performance

The models showed a broad spectrum of learning behaviour depending on architecture complexity and data configuration.

Model	Epochs	Train Accuracy
Vanilla RNN	200	23.84% – 37.15%
Complex RNN (V1)	90	22.91% – 36.13%
Manual RNN	90	Up to 45.93%
Manual LSTM	**800**	**55.78% – 59.83%**
Two-layered RNN	90	24.27%
Bidirectional RNN	**500**	**60.41% – 62.42%**

5. Test Performance

Test accuracy generally followed the training performance trend:

Model	Test Accuracy
Vanilla RNN	26.95% – 39.76%
modelV1 Series	~ 23% – 37%
Manual RNN	~ 47.06%
Manual LSTM	**~ 61.13%**
Two-layered RNN	~ 25.07%
Bidirectional RNN	**~ 63.59%**

6. Model Behavior Observations

- **Weight Initialization:** Random embeddings underperformed compared to FastText-initialized models.
- **Class Weights:** Notably improved generalization, especially for manual LSTM and bidirectional variants.
- **Complexity vs Performance:** Deeper or bidirectional models achieved better performance, though gains saturated beyond certain epochs.

7. Conclusion

- **Bidirectional RNNs** achieved the highest accuracy (~63.6%) on test data, leveraging context from both past and future.
- **Manual LSTMs**, especially with class weighting, proved highly effective with prolonged training.
- Simpler models like **Vanilla RNN** stagnated early and failed to exceed random baseline (~ 50%) significantly.
- The **choice of embeddings and training epochs** was critical for convergence and generalization.

Name: Ashutosh Rai
Scholar No: 2422308
Guide: Dr. Partha Pakray