Assignment 1 - NLP using DL techniques (83374)

Dana Gibor Ido Sar Shalom Natalya Sigal

1 Introduction

This report details the development and evaluation of deep learning models for classifying text into six emotion categories: *Sadness, Joy, Love, Anger, Fear,* and *Surprise*. We implemented and compared two RNN architectures: **Gated Recurrent Unit (GRU)** and **Bidirectional Long Short-Term Memory (BiLSTM)**. The project involved extensive data preprocessing, distinct embedding strategies, systematic hyperparameter tuning, and a comparative analysis to identify the optimal approach.

2 Exploratory Data Analysis

Before proceeding to data preparation, we conducted an analysis of the class balance within the dataset.

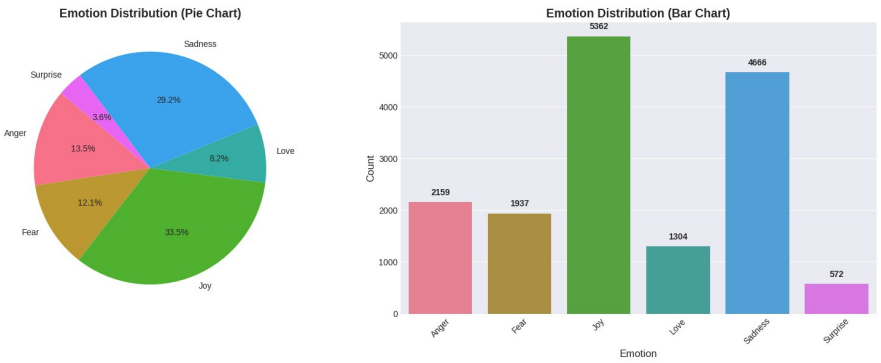


Figure 1: Distribution of Emotion Classes.

The distribution, as illustrated in Figure 1, reveals a significant class imbalance. The dataset is dominated by *Joy* (33.5%) and *Sadness* (29.2%), which together constitute over 60% of the samples. In contrast, classes such as *Surprise* (3.6%) and *Love* (8.2%) are heavily underrepresented. This disparity suggests a potential bias where the model may favor majority classes.

3 Data Preparation

3.1 Data Preprocessing

A robust pipeline was implemented to clean and prepare the data. Key steps included:

- **Cleaning & Normalization:** Removing URLs, special characters, numbers, and converting text to lowercase.
- **Filtering:** Removing standard stopwords and non-alphanumeric characters.
- **Duplicate Handling:** Strictly removed from training to prevent overfitting, but retained in validation/test sets.

3.2 Embedding Strategies

Distinct pre-trained embeddings were chosen to complement each network's strengths:

- **Word2Vec (Google News 300d) with GRU:** A predictive model capturing local context patterns. This matches the GRU's efficiency with local temporal patterns.

- **GloVe (Twitter 200d) with BiLSTM:** A count-based model leveraging global co-occurrence statistics. This complements the LSTM's ability to handle long-term dependencies and aligns with the informal dataset nature.

4 Model Architectures & Experiments

Both models utilized a sequential architecture: Embedding → Bidirectional RNN → BatchNorm → Dropout → Dense → Output.

4.1 Hyperparameter Tuning Methodology

We adopted a systematic approach, running controlled experiments for each hyperparameter:

- **Learning Rate:** Tested $0.0001 - 0.01$ to find the convergence "sweet spot".
- **Batch Size:** Tested $16, 32, 64, 128$. Smaller batches provided better regularization.
- **Hidden Units:** Tested $64-256$ to determine the minimum capacity needed to avoid overfitting.
- **Dropout Rate:** Tested $0.2 - 0.6$ to prevent memorization.

To manage computational constraints effectively, we employed a pragmatic **One-Factor-at-a-Time (OFAT)** tuning strategy instead of an exhaustive Grid Search. This heuristic approach involved establishing a baseline configuration and systematically varying one hyperparameter (e.g., Learning Rate) while holding all others constant.
The final configurations were selected based strictly on maximizing **Validation Accuracy** while minimizing **Validation Loss**.

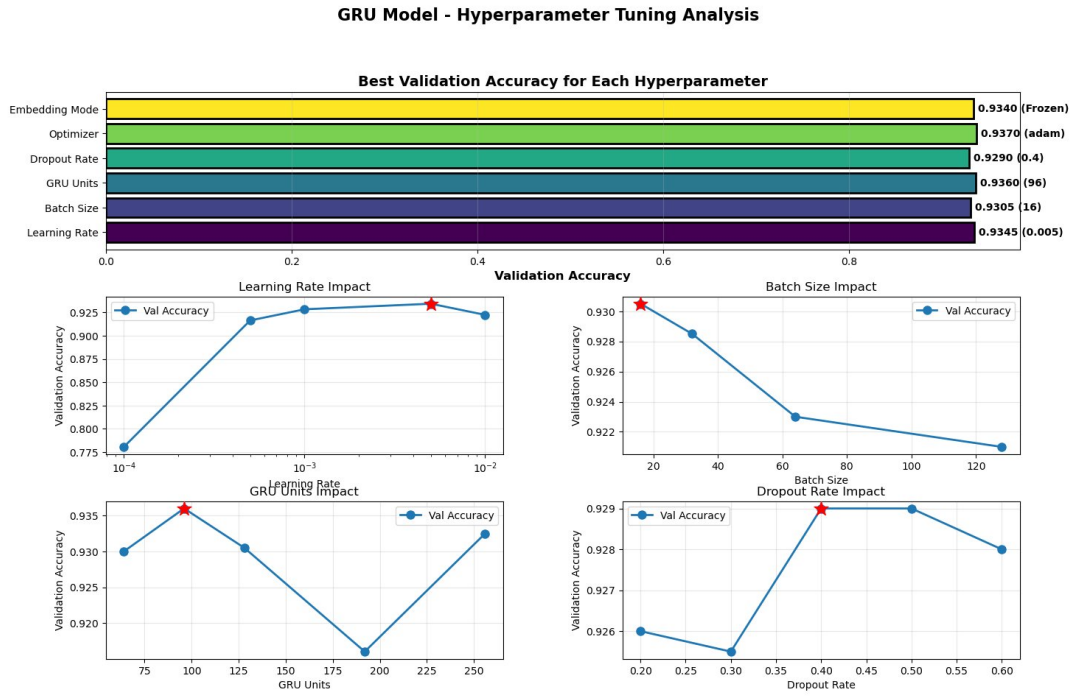


Figure 2: Impact of various hyperparameters on GRU model performance.

4.2 Optimal Configurations

Systematic tuning yielded the following optimal configurations:
GRU Architecture (Word2Vec):

- **Config:** LR: 0.005, Batch: 16, Units: 96, Dropout: 0.4, Optimizer: Adam, Embeddings: Frozen.
- **Result:** Highest accuracy (0.9345). Frozen embeddings prevented overfitting.

BiLSTM Architecture (GloVe):

- **Config:** LR: 0.001, Batch: 32, Units: 192, Dropout: 0.3, Optimizer: Adam.
- **Result:** Required lower learning rate and significantly more capacity (192 vs 96 units) than GRU.

5 Results

Both models exceeded the 75% target accuracy. Early stopping (patience=5) prevented overfitting.

Table 1: Performance Comparison Summary

Metric	GRU Model	BiLSTM Model
Embedding	Word2Vec (300d)	GloVe (200d)
Validation Accuracy	**92.80%**	91.65%
Validation Loss	**0.1344**	0.1998
Optimal Units	96	192
Converged Epochs	∼10	∼15

5.1 Error Analysis

To gain deeper insights into model performance, we examined the confusion matrix of the GRU network.

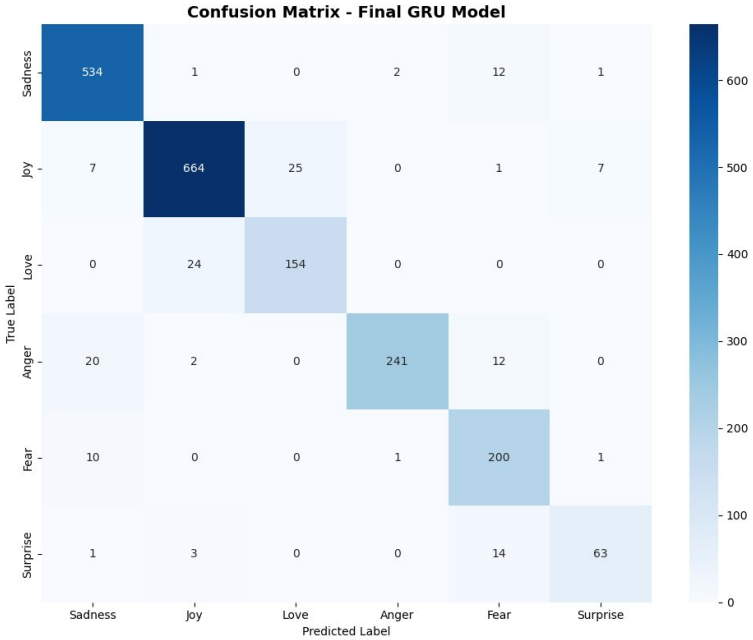
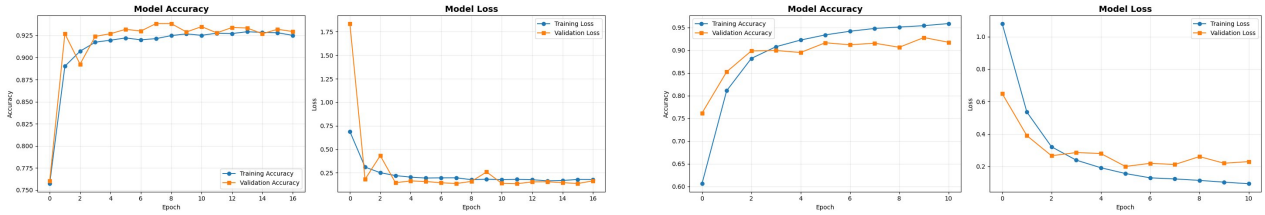


Figure 3: Confusion Matrix for the GRU model.

The analysis reveals that while the model generally performs well, there are notable shared misclassifications. In particular, we observe confusion between similar emotions like *Fear* and *Sadness* or *Love* and *Joy*. This overlap is likely due to the shared vocabulary and semantic proximity of these emotions, a pattern that was observed to affect the BiLSTM model as well.

5.2 Convergence Analysis

The GRU model converged faster (epoch 10-11) with stable loss reduction. The BiLSTM model required more epochs and higher capacity (units) to reach comparable performance.



(a) GRU Training History

(b) BiLSTM Training History

Figure 4: Training Accuracy and Loss over epochs.

6 Conclusion

Based on the comparative analysis of the validation set, the **GRU model with Word2Vec embeddings** appears to be the optimal configuration for this emotion classification task.

While these results strongly favor the GRU architecture, it is important to note that performance on the unseen test set may vary. However, given the consistent superiority in validation metrics and computational efficiency, the GRU model is our recommended candidate for deployment.