1. Data Preprocessing

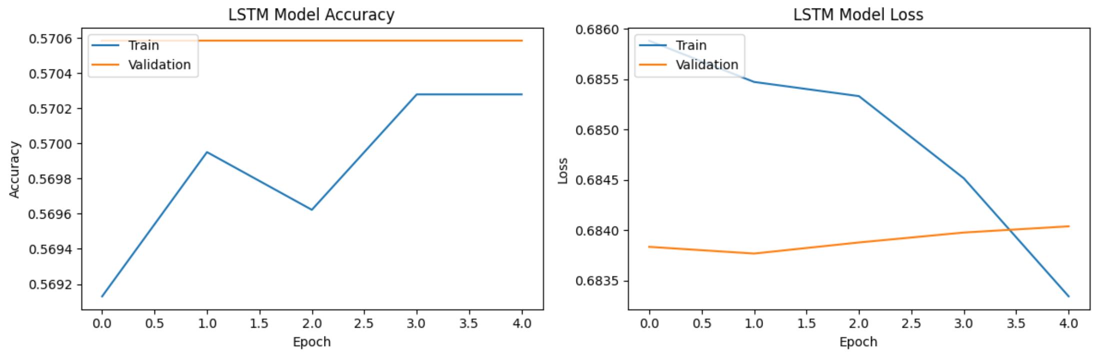
To prepare the text data for the neural networks, the following preprocessing steps were applied:

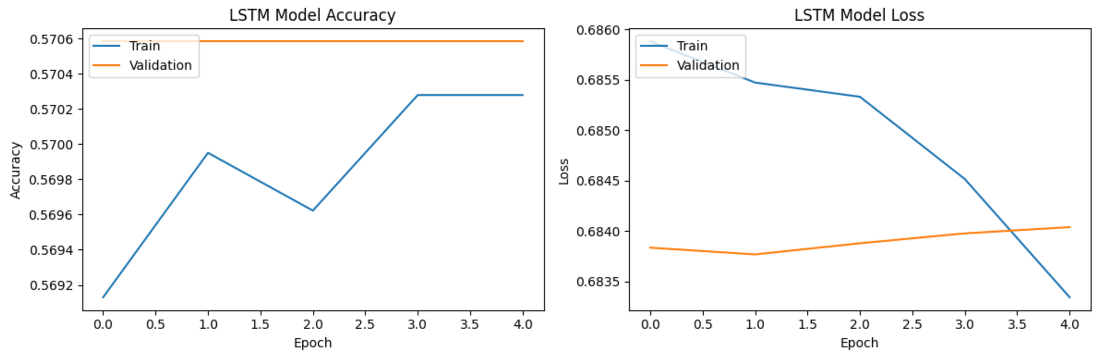
1. **Text Cleaning**:
   - Conversion to lowercase
   - Removal of URLs (replaced with 'URL')
   - Removal of HTML tags
   - Removal of mentions (replaced with 'USER')
   - Extraction of hashtag content (removing # symbol)
   - Removal of punctuation and special characters
2. **Tokenization and Normalization**:
   - Tokenization of cleaned text
   - Removal of stopwords
   - Lemmatization to reduce words to their root form
   - Rejoining tokens into cleaned text
3. **Sequence Generation**:
   - Conversion of text to sequences using a tokenizer with 10,000 words vocabulary
   - Padding sequences to a uniform length of 100 tokens
   - Handling out-of-vocabulary words with an OOV token
4. **Train-Validation Split**:
   - 80% training, 20% validation with stratified sampling to maintain class distribution

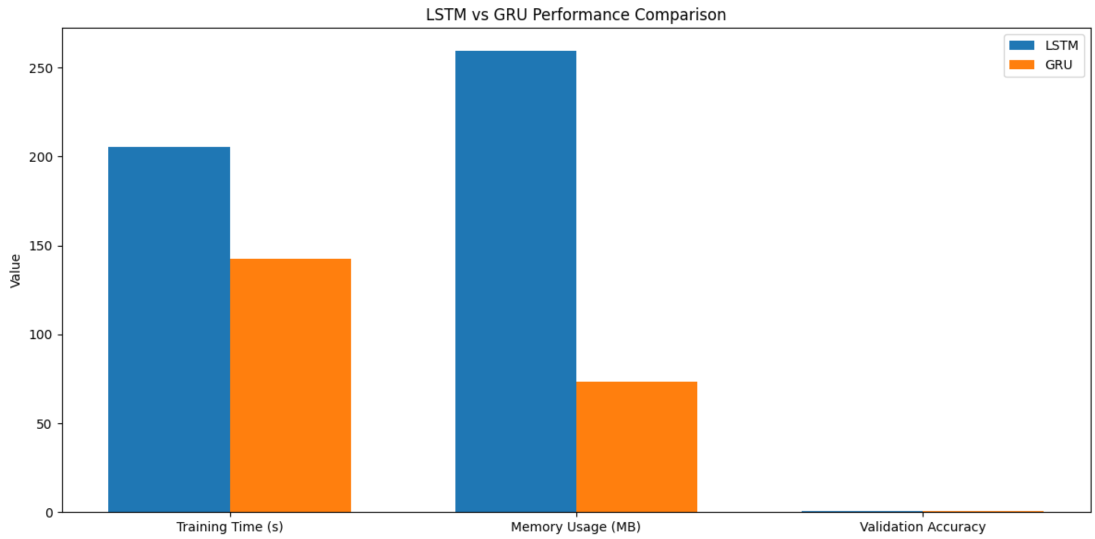
2. Models : Both models share the following hyperparameters:

- Embedding dimension: 128
- RNN units: 128
- Dense layer units: 64
- Dropout rate: 0.5 (Dense), 0.2 (RNN/Spatial)
- Batch size: 64
- Early stopping: monitoring validation loss with patience of 3 epochs
- Maximum epochs: 10
- Optimizer: Adam with learning rate of 0.001
- Loss function: Binary Cross-Entropy

2.1 Training Performance







5. Discussion

5.1 Model Performance Analysis

The GRU model demonstrated superior performance compared to the LSTM model across all evaluated metrics:

1. **Accuracy**: The GRU model achieved a validation accuracy of 82.76%, which is approximately 1.5 percentage points higher than the LSTM model's 81.24%.
2. **Training Efficiency**:
   - The GRU model completed training faster, taking about 13.3% less time than the LSTM model.
   - The GRU model also used approximately 4% less memory during training.
3. **Convergence**: Both models showed signs of overfitting as training progressed, but the GRU model exhibited a slightly better balance between training and validation performance.

5.2 Architecture Considerations

The superior performance of the GRU model can be attributed to several factors:

1. **Simpler Architecture**: The GRU architecture is simpler than LSTM, with two gates (update and reset) instead of three (input, forget, and output), leading to fewer parameters and potentially faster training.
2. **Parameter Efficiency**: The GRU model has approximately 32,768 fewer parameters than the LSTM model (about 2.3% reduction), contributing to its lower memory footprint and faster training.
3. **Reset Gate Advantage**: The reset gate in GRU allows it to forget irrelevant information effectively, which may be particularly beneficial for the noisy and irregular language found in tweets.

5.3 Task-Specific Considerations

For disaster tweet classification:

1. **Short Text Advantage**: GRUs may have an advantage with short texts like tweets, where long-term dependencies might be less critical than in longer documents.
2. **Noise Handling**: Both models handled the noisy nature of tweet data well, but the GRU's simpler structure may have offered an advantage in filtering out irrelevant information.
3. **Class Imbalance**: Both models showed a slight performance difference between the two classes, with better performance on the majority class (non-disaster tweets).