National University of Computer & Emerging Sciences

Project Report

Deep Learning for Perception

Section: BSCS-8B

Spam Detection in Emails using LSTM and Transformer-based approach

Group Members

21K4513 Hamza Iqbal

Objective

The primary goal of this project is to develop an effective email spam classifier using deep learning techniques. The focus was not only on achieving high accuracy for English-language spam detection but also on designing a solution adaptable to other languages like Urdu. The classifier should be suitable for practical deployment and scalable to multilingual datasets.

Problem Statement

Spam emails pose a persistent threat to digital communication by cluttering inboxes, wasting time, and potentially introducing phishing or malware risks. While many spam detection systems exist, they often rely on hand-crafted rules or traditional machine learning approaches that may not generalize well to varied linguistic patterns, especially in low-resource languages. This project aims to compare modern neural network approaches — specifically, a Bidirectional LSTM and a Transformer-based BERT model — for building a reliable, multilingual spam detection system.

Methodology

1. Dataset

- A CSV file named `clean_data.csv` was used, containing 50,000 rows of labeled email data with two columns: `text` (email content) and `label` (0 = ham, 1 = spam).
- The dataset was split into training and test sets using an 80-20 ratio.

2. Approach 1: Bidirectional LSTM

- A text preprocessing layer (`TextVectorization`) was used to tokenize and pad input sequences.
- The model architecture included:
    - Input layer
    - Embedding layer
    - Bidirectional LSTM

    - Dense layer with sigmoid activation for binary classification

- Compiled with `binary_crossentropy` loss and `adam` optimizer.
- Trained for 5 epochs with batch size 32.

3. Approach 2: BERT Transformer (Hugging Face)

- Pretrained `bert-base-uncased` model was fine-tuned using `TFBertForSequenceClassification`.
- Used `BertTokenizer` for input preprocessing, converting text to token IDs with attention masks.
- Training was done using TensorFlow's `fit()` API or custom training loops.
- Trained for 3 epochs on the same dataset using a batched TensorFlow Dataset pipeline.

Results

Model	Accuracy (Test Set)	Notes
Bidirectional LSTM	~90%	Performed well on clean English data, struggled with non-English text
BERT Transformer	~94–96%	Strong generalization, better handling of sentence structure

- **LSTM** was faster to train and easier to deploy with low compute resources but showed reduced performance when applied to more complex or non-English data.
- **BERT** took longer to train and required more memory but significantly improved performance and robustness, especially as multilingual support is introduced in future work.