**SMS Spam Detection Report**

Kankana Ghosh

**Introduction:**

In today's interconnected world, mobile phones have become an integral part of our lives, serving as a primary means of communication. However, this convenience also brings challenges, one of which is the prevalence of SMS spam. Unwanted and potentially harmful messages flood our inboxes, making it crucial to develop effective methods for detecting and mitigating spam. This report explores the use of deep learning models for SMS spam detection, focusing on methodologies, data preprocessing, model evaluation, and performance analysis.

**Methodology:**

The project involves the development of various deep learning models using TensorFlow for SMS spam detection. The goal is to create models that can distinguish between legitimate (ham) messages and spam messages. The dataset contains 5572 observations with five columns. Columns with NaN values are removed. The 'v1' column is renamed as 'Label,' and the 'v2' column is renamed as 'text.' An additional column, 'label\_enc,' is introduced to encode labels as 0 for 'ham' and 1 for 'spam.'As the data doesnot contain any missing data in rows so the final dataset contains 5572 observations on 3 variates. The dataset is explored to understand the distribution of classes (ham vs. spam). From the bar graph having the count of ham and spam depict that the number of ham sms is more compared to spam SMS which is natural. The average number of words per sentence is 15 and the total number of unique words in the dataset is 15585.A baseline model is constructed, and its performance is evaluated. Multinomial Naive Bayes classifier is selected as the baseline model due to its effectiveness for text classification with discrete features. Now to beat the performance of the baseline model using deep learning models like Custom-Vector-Embedding Model, Bidirectional-LSTM Model, USE-Transfer Learning Model.

**Model Evaluation:**

Baseline Model Metrics:

The model's performance is evaluated using precision, recall, F1-score, support, accuracy, macro average, and weighted average metrics. For class 0 (ham), the precision is 0.96, indicating that 96% of predicted ham instances were accurate. For class 1 (spam), the precision is 1.00, implying perfect accuracy. The model achieved a recall of 1.00 for ham, correctly identifying all instances, but 0.72 for spam, suggesting 28% were missed. The F1-scores are 0.98 for ham and 0.84 for spam, representing a balance between precision and recall. With 965 instances of ham and 150 of spam, the model's overall accuracy is 96%. The macro average yields 0.98 precision, 0.86 recall, and 0.91 F1-score. The weighted average shows 0.96 precision, 0.96 recall, and 0.96 F1-score, considering class imbalances. Although the model excels in ham detection, its lower spam recall implies challenges in identifying spam instances.

Other Model Metrics:

USE-Transfer Learning Model:

The model achieved an accuracy of approximately 98.30%, indicating it made correct predictions for the majority of instances in the dataset. The precision for this model is around 95.17%. Out of the instances predicted as positive, about 95.17% were truly positive, while the remaining 4.83% were false positives. The recall is approximately 92.00%. The model correctly identified 92.00% of the actual positive instances. The F1-score is approximately 93.56%, which is a balanced measure of precision and recall.

Bidirectional-LSTM Model:

This model achieved an accuracy of approximately 98.12%, similar to the first model. Precision: The precision is around 96.40%. This indicates that out of the instances predicted as positive, about 96.40% were truly positive. The recall is approximately 89.33%. The model identified 89.33% of the actual positive instances. The F1-score is approximately 92.73%, indicating the balance between precision and recall.

Custom-Vec-Embedding Model:

Similar to the previous models, this model achieved an accuracy of approximately 98.12%. Precision: The precision is quite high at around 97.78%, indicating accurate positive predictions. The recall is approximately 88.00%. The model captured 88.00% of the actual positive instances. The F1-score is approximately 92.63%, again reflecting a balance between precision and recall.

Multinomial Naive Bayes Model:

This model achieved an accuracy of approximately 96.23%, which is slightly lower than the other models. Precision: The precision is perfect at 100.00%, indicating that all instances predicted as positive were indeed positive. The recall is around 72.00%. The model captured only 72.00% of the actual positive instances, indicating some difficulty in identifying positives. The F1-score is approximately 83.72%, which reflects the trade-off between precision and recall.

In summary, all models have high accuracy rates, but they differ in precision, recall, and F1-scores. The "MultinomialNB Model" has the highest precision but struggles with recall. The "USE-Transfer learning Model," "Bidirectional-LSTM Model," and "Custom-Vec-Embedding Model" exhibit good trade-offs between precision and recall. The choice of model may depend on the specific requirements of the task, such as the importance of precision and recall in the context of the application.

**Conclusion:**

The SMS spam detection project has demonstrated the effectiveness of various deep learning models in accurately identifying spam messages. Despite the challenges posed by unbalanced data, the models exhibited strong performance across multiple evaluation metrics. The deep learning models, including the USE-Transfer Learning Model, Bidirectional-LSTM Model, and Custom-Vec-Embedding Model, achieved impressive accuracy, precision, recall, and F1-scores.It is important to note that the choice of model should depend on the specific requirements of the application. For instance, the "Multinomial Naive Bayes Model" exhibited perfect precision but struggled with recall. In contrast, the deep learning models showed balanced trade-offs between precision and recall. Overall, the project underscores the significance of precision, recall, and F1-score in evaluating the performance of SMS spam detection models.