# **Natural language assignment**

**Assignment**

**Victor Lees**

# Pitch

Organizations in sectors like telecommunications, e-commerce, and financial services can significantly benefit from funding a spam detection application. The perceived value includes increased customer satisfaction by reducing unwanted messages, enhanced security by filtering potential phishing attempts, and improved brand image by protecting users from spam. Financial institutions, for instance, can mitigate fraud risks, while e-commerce platforms can maintain a cleaner communication channel with their customers.

**Key Benefits:**

1. **Enhanced Customer Experience**:

* **Reduction in Spam**: Effectively filters spam messages, reducing unwanted and potentially harmful content, leading to a more secure and pleasant user experience.
* **Improved Trust**: Builds customer trust by ensuring their communication channels are protected from spam and fraudulent messages.

**2.** **Operational Efficiency**:

* **Automated Filtering**: Automates the identification and filtering of spam messages, reducing manual effort and allowing customer service teams to focus on more critical tasks.

**3**. **Financial Savings**:

* **Cost Reduction**: Reduces operational costs by automating spam detection, minimizing the need for manual intervention.
* **Fraud Prevention**: Protects against financial losses by blocking phishing and fraudulent messages, safeguarding customers.

**4**. **Security Enhancement**:

* **Phishing Protection**: Identifies and blocks phishing attempts, protecting sensitive user information and reducing the risk of security breaches.

**5**. **Regulatory Compliance**:

* **Adherence to Regulations**: Helps organizations comply with regulations regarding unsolicited communications, reducing the risk of fines and legal issues.

# 2. Data source:

The data for this project is sourced from the publicly available [SMS Spam Collection Dataset](https://www.kaggle.com/datasets/uciml/sms-spam-collection-dataset) from Kaggle. This dataset is a collection of SMS messages tagged as "spam" or "ham" and is commonly used for text classification tasks. For those interested in using the dataset, download it from the UCI Machine Learning Repository using the provided link

3. Model and Data Justification

The model **bert-base-cased** from [Hugging Face](https://huggingface.co/google-bert/bert-base-cased) is selected for its superior natural language understanding capabilities, leveraging a pre-trained transformer architecture that captures contextual nuances in text. This BERT model is fine-tuned on the SMS Spam Collection dataset, which provides a robust framework for distinguishing between spam and legitimate messages by utilizing its pre-trained linguistic knowledge and adjusting to the specific task of spam detection. The combination of a well-documented dataset and a state-of-the-art model ensures high accuracy and generalization in identifying spam in diverse SMS messages..

# 4. Commented examples:

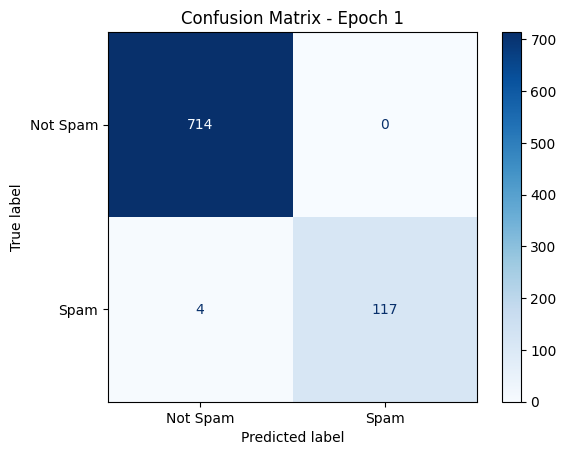
|  |  |  |
| --- | --- | --- |
| Input | Output | Observations |
| "Go until jurong point, crazy.. Available only in bugis n great world la e buffet... Cine there got amore wat..." | **Predicted Label**: Not Spam  **Expected/Actual Label**: Not Spam | The message is casual and informational, typical of non-spam content. The model correctly identifies it as Not Spam.  **Analysis**: The prediction aligns with the context of the message being a friendly, non-promotional SMS. |
| "Free entry in 2 a wkly comp to win FA Cup final tkts 21st May 2005. Text FA to 87121 to receive entry question(std txt rate)T&C's apply 08452810075over18's" | **Predicted Label**: Spam  **Expected/Actual Label**: Spam | The message promotes a contest with a call-to-action to text a number, which is typical spam behavior.  **Analysis**: The model correctly identifies promotional language and the structure characteristic of spam messages. |
| "FreeMsg Hey there darling it's been 3 week's now and no word back! I'd like some fun you up for it still? Tb ok! XxX std chgs to send, £1.50 to rcv" | **Predicted Label**: Spam  **Expected/Actual Label**: Spam | The message suggests a personal engagement and mentions charges, common in spam.  **Analysis**: The model successfully identifies the spam nature due to the solicitation for interaction and the mention of charges. |
| "As per your request 'Melle Melle (Oru Minnaminunginte Nurungu Vettam)' has been set as your callertune for all Callers. Press \*9 to copy your friends Callertune" | **Predicted Label**: Not Spam  **Expected/Actual Label**: Not Spam | This message appears to be a notification about a service setting, which is typically not considered spam.  **Analysis**: The model correctly classifies this as non-spam due to the informational nature and lack of promotional content. |

# 5. Testing

# Training and Validation Summary

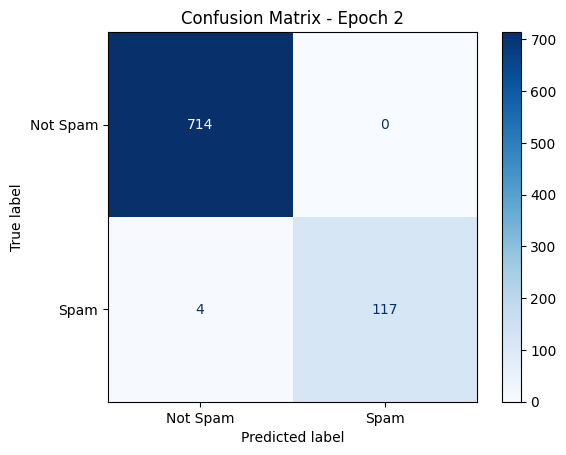
**Epoch 1**: Training Loss = 0.1121

**Observation**: The training loss consistently decreased across epochs, indicating that the model is learning effectively and the fit to the training data improved with each epoch.



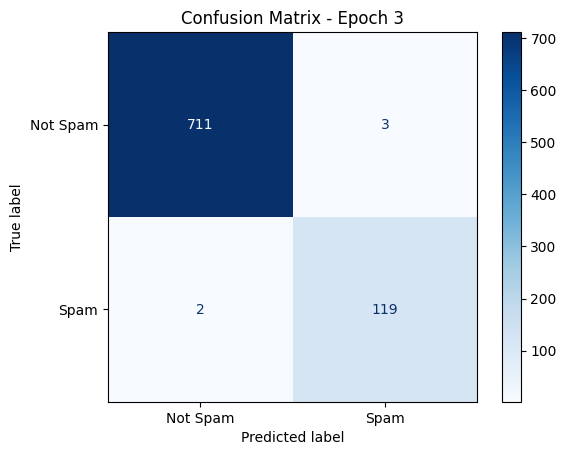
#### **Validation Metrics**

* **Accuracy**: 1.00
* **Precision**: 1.00
* **Recall**: 0.97
* **F1 Score**: 0.98
* **Validation Loss**: 0.0354

**Epoch 2**: Training Loss = 0.0548

#### **Validation Metrics**

* **Accuracy**: 1.00
* **Precision**: 1.00
* **Recall**: 0.97
* **F1 Score**: 0.98
* **Validation Loss**: 0.0207

**Epoch 3**: Training Loss = 0.0278

#### **Validation Metrics**

* **Accuracy**: 0.99
* **Precision**: 0.98
* **Recall**: 0.98
* **F1 Score**: 0.98
* **Validation Loss**: 0.0173

### **Conclusion**

The model is highly effective at distinguishing between spam and non-spam messages, showing excellent performance across training, validation, and test datasets. The high metrics and low confusion matrix errors suggest the model is robust and reliable for practical spam detection applications. Further, fine-tuning and testing with larger and more diverse datasets can ensure continued high performance and generalization.

6. Code and instructions to run it:

This is the link to Google Colab [Google Colab BERT Model Spam Detection](https://colab.research.google.com/drive/1bo1h1IqeLkokYvmtee7mJBhouXkXCYa_?usp=sharing). I am running this model on an L4 GPU, leveraging CUDA drivers to significantly enhance performance. Initially, the BERT model took over 60 minutes to converge, using approximately 22GB of GPU memory. However, after further fine-tuning, I reduced the convergence time to 20 minutes across 3 epochs, with the model now using only 4.7GB of GPU resources.

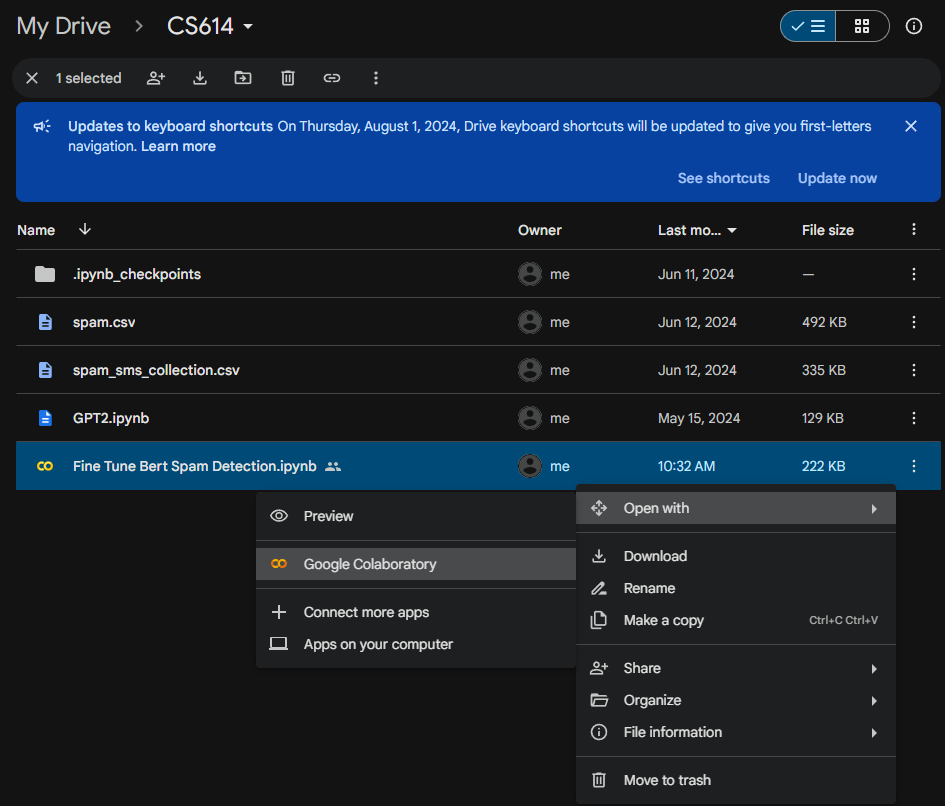
The link above should give you full access to run my BERT Model, however if you have issues, here is how to perform a setup on Google Colab

**Google Drive Setup:**

* 1. Download Kaggle Dataset from [SMS Spam Collection Dataset](https://www.kaggle.com/datasets/uciml/sms-spam-collection-dataset),
     + <https://www.kaggle.com/datasets/uciml/sms-spam-collection-dataset>
  2. Goto Google Drive and Create a Folder like MyDrive\BERT
     + Copy and or move the Kaggle dataset to this folder
  3. Copy “Fine Tune Bert Spam Detection.ipynb “into your Google Drive Path
     + <https://colab.research.google.com/drive/1bo1h1IqeLkokYvmtee7mJBhouXkXCYa_?usp=sharing>

**Google Colab Setup:**

* 1. Goto you MyDrive\BERT and open “Fine Tune Bert Spam Detection.ipynb” using Google Colab



* 1. Correct file paths in Google Colab Notebook example

Change: df = pd.read\_csv('/content/drive/MyDrive/**CS614**/spam.csv', encoding = "ISO-8859-1")

To: df = pd.read\_csv('/content/drive/MyDrive/**YOURPATH**/spam.csv', encoding = "ISO-8859-1")

* 1. To Execute Press CTRL + F9 or under Runtime select Run all