Data Analytics Capstone Topic Approval Form

**Student Name:** André Davis ([ada1962@wgu.edu](mailto:ada1962@wgu.edu))

**Student ID:** 010630641

**Capstone Project Name:** Machine Learning [SPAM](https://en.wikipedia.org/wiki/Spamming) Detection Powered by Enron/TREC Public Spam Corpus

**Project Topic**: This initiative aims to develop a proficient model capable of accurately classifying unlabeled content as either SPAM or HAM (Not Spam). To achieve this, the project will utilize the TREC Public Spam Corpus and Enron Emails dataset from 2007, available at <https://www.kaggle.com/datasets/purusinghvi/email-spam-classification-dataset>.

**This project does not involve human subjects research and is exempt from** [**WGU**](https://www.wgu.edu/) **IRB review.**

**Research Question:**   
Is it possible to develop a machine learning model that can accurately classify new content as SPAM or non-SPAM (HAM), using a dataset comprising known SPAM and regular content from the 2007 TREC Public Spam Corpus and Enron emails?

***Hypothesis****:*

**Null hypothesis (HΦ)**- The machine learning model, developed using the 2007 TREC Public Spam Corpus and Enron emails dataset, cannot accurately classify new content as SPAM or non-SPAM (HAM) beyond what would be expected by chance. This implies that any observed accuracy in classification is due to random variation in the data.  
 **Alternate Hypothesis (Ha)**- The machine learning model, developed using the 2007 TREC Public Spam Corpus and Enron emails dataset, can accurately classify new content as SPAM or non-SPAM (HAM) better than what would be expected by chance. This suggests that the model has learned discernible patterns within the dataset that enable it to effectively differentiate between SPAM and non-SPAM content.

**Context:** *Explain why the situation or question would benefit from a data analysis in less than 500 words.*  
The proposed question, focusing on the development of a machine learning model for SPAM and HAM classification, is an excellent candidate for data analysis due to several reasons:

1. **Practical Relevance and Application**: Spam detection is a significant issue in digital communication, impacting both individuals and organizations. Effective spam filters can save time, protect against malware and phishing attacks, and improve overall user experience. By analyzing this data, we can create models that are not only academically interesting but also have practical applications.
2. **Complexity of SPAM and HAM differentiation**: The distinction between SPAM and HAM is not always clear-cut. Spam messages have evolved to be more sophisticated, often mimicking legitimate content. A data-driven approach can uncover subtle patterns and characteristics of spam and non-spam content that may not be immediately apparent to human observers.
3. **Model Evaluation and Improvement**: Data analysis allows for quantitative evaluation of the model's performance. Metrics such as accuracy, precision, recall, and F1-score provide insight into how well the model is performing and where it might be failing. This feedback loop is crucial for refining and improving the model.
4. **Machine Learning Model Selection and Optimization**: Data analysis is not just about processing data; it's also about choosing the right algorithm for the task. Different machine learning models have their strengths and weaknesses. Through analysis, we can determine which model (like Naïve Bayes, SVM, Neural Networks) best suits our data characteristics and requirements.
5. **Adaptability to New Spam Techniques**: Spam tactics evolve, and a model trained on historical data might become less effective over time. Regular analysis of new data sets can help in updating and tuning the model to adapt to new spamming techniques.

In summary, the application of data analysis to this problem is not only apt but necessary. It provides the tools and methodologies to extract insights from complex and large datasets, build and refine predictive models, and ultimately, contribute to solving a problem with significant real-world implications.

**Data:** *Identify data you will need to collect that is relevant to the situation or question.*

The data that will be needed to be collected is a collection of email contents which has been correctly labeled as SPAM or NOT (HAM).

*If an existing data set will be used, describe the data set.* **(Dataset Exists):**

|  |  |  |
| --- | --- | --- |
| Feature | Datatype | Description |
| label | Qualitative | This is the feature that is the label of the dataset. It is a Boolean represented as 1 or 0.  1 represents the associated text is SPAM  0 represents the associated text is HAM (Not SPAM) |
| text | Qualitative | This is just text content (unstructured data). In the case of this dataset the contents are an Email. This is the data that will be used to train and is pre-labeled for the Machine Learning model. |

***Notes:*** *Within the code these will be renamed. label -> IsSpam and text -> EmailContent for readability.*

*Explain who owns the data and why you are allowed to use this data for your capstone project.*

**Dataset Hosting:** [www.kaggle.com](http://www.kaggle.com)

**Dataset Name:** ‘*Spam Email Classification Dataset*’.

**Created by:** Puru Singhvi  
 **Dataset License:** [MIT](https://www.mit.edu/~amini/LICENSE.md)

*Note: If you are using restricted information, please have the Third-Party Authorization Form signed by an authorized agent on behalf of the data owner. The data owner’s legal name is required on the form.*

**Data Gathering:** *Describe the data-gathering methodology you will use to collect data.*The dataset required for this project is a pre-labeled set of training data, readily available for download. It can be accessed as a **.csv** file from Kaggle at [[https://www.kaggle.com/datasets/purusinghvi/email-spam-classification-dataset]](https://www.kaggle.com/datasets/purusinghvi/email-spam-classification-dataset).

However, there are additional considerations regarding data management due to GitHub's file size constraints. Specifically, GitHub imposes a limit of 100MB per file, while the dataset in question is 133MB. To address this, the original dataset has been divided into two separate **.csv** files with index added in case we need to maintain order. These files are stored in a private repository associated with my academic coursework. In the Jupyter Notebook environment, these two files will be merged back into a single dataset for analysis and model training. This approach is a practical solution to the repository's file size limitations and does not reflect any inherent constraints of the data itself.

**Data Analytics Tools and Techniques**: *Identify the appropriate data-analysis technique you will use to analyze this data. (****Justifications are in-line with the chosen items****)*

Exploratory Data Analysis (EDA):

The Exploratory Data Analysis (EDA) will be conducted utilizing the [Polars](https://www.pola.rs/) Python package. This approach will enable us to reassemble the dataset, taking into account the limitation that “GitHub blocks files larger than 100 MiB” (GitHub, n.d.). The process will involve meticulous checks for missing values, analysis of data distributions, identification of any unusual characters, and computation of summary statistics for both the original dataset and the character lengths relevant to the neural networks. Subsequently, the data will be systematically divided into training, validation, and test datasets, ensuring a comprehensive and robust preparation for further modeling stages.

Principal Statistical Method (Regression Analysis in form of Logistical Regression):

I will be using Logistical Regression as per Wikipedia it’s a great option to test if our ‘label’ (IsSpam) has a good relationship with our dependent variable ‘text’ (EmailContent)  
  
“In statistical modeling, regression analysis is a set of statistical processes for estimating the relationships between a dependent variable (often called the 'outcome' or 'response' variable, or a 'label' in machine learning parlance) and one or more independent variables (often called 'predictors', 'covariates', 'explanatory variables' or 'features').” (“Regression Analysis”, Last Update: 10/31/2023)

Machine Learning Models (Neural Network):

I will be using Neural Network (*NN*) to generate a model for spam detection. Because [SPAM](https://en.wikipedia.org/wiki/Spamming) itself is evolving and can cost a business a lot of money if a particular SPAM in the form of [PHISHING](https://en.wikipedia.org/wiki/Phishing) compromises a company’s network.

**Reasons:**

* NN are adept at Pattern Recognition and Learning on complex patterns.
* NN can evolve as SPAM itself evolves and is feed to the NN Model
* NN can handle unstructured data well and because SPAM consists of emails they can come in **any** format.
  + Layers
  + Text-Processing
  + Transformers
* NN are resistant to data-noise, which means irrelevant or misleading information doesn’t damage the model.
* NN are scalable meaning they can handle large amounts of data. In the corporate word SPAM is an onslaught.

Tools:

* [Jupyter](https://jupyter.org/)Notebook
  + Jupyter notebook is an experimentation and documentation platform. This lends itself well to providing out concepts and performance assessments.
* [Python](https://www.python.org/) Programming Language
  + A bigger chunk of the data science and data analytics community build libraries around Python. So, I have chosen to use the simpler and more community supported language.
* Python Packages:
  + [Polars](https://www.pola.rs/) (Was not taught during [WGU](https://www.wgu.edu/) courses)
    - [Polars](https://www.pola.rs/) is a data manipulation library akin to [Pandas](https://pandas.pydata.org/), renowned for its efficiency with small to medium-sized datasets. However, Polars distinguishes itself by being developed in [Rust](https://www.rust-lang.org/), a language that combines Python's simplicity with the high performance of [C](https://en.wikipedia.org/wiki/C_(programming_language)). This project will employ Polars not only as a practical example of its application but also to address the common occurrence of large datasets in real-life scenarios.
  + [Matplotlib](https://matplotlib.org/)
    - This is a community standard visualization library and performed well for all other performance assessments. Although, it has heavy integration with [Pandas](https://pandas.pydata.org/), it is not a requirement to use the visualization library.
  + [Emoji](https://github.com/carpedm20/emoji/) (Was not taught during [WGU](https://www.wgu.edu/) courses)
    - During the pre-processing process before the model is created this library makes it easier to identify non-standard characters that come in the form of emojis and will aid in removing them before modeling.
  + [Unidecode](https://pypi.org/project/Unidecode/) (Was not taught during [WGU](https://www.wgu.edu/) courses)
    - During the pre-processing process before the model is created, this helps remove Unicode ASCII type characters that my be introduced and cause issues. These will be removed.
  + [Insert NN Library]
    - [Insert reasons]

**Project Outcomes**:

The expected project outcomes include the development of a highly refined and effective model designed specifically for business use. This model will play a crucial role in combating spam within email systems. By efficiently filtering out spam, the model will not only eliminate irrelevant or nuisance communications but also significantly reduce the risks associated with phishing attempts. These phishing attempts often target sensitive company data and finances. As such, the implementation of this model will enhance the overall security and efficiency of the business's email communications.

**Projected Project End Date**: 02/29/24 (End of Final Term)

**Sources**:

* PURU SINGHVI. (n.d.). Spam Email Classification Dataset [Data set]. Kaggle.  
   Retrieved from <https://www.kaggle.com/datasets/purusinghvi/email-spam-classification-dataset>
* Pola-rs GitHub Organization. (n.d.). Polars.   
   Retrieved from <https://www.pola.rs/>
* Pola-rs GitHub Organization.. Benchmarks. In Polars.   
   Retrieved from <https://www.pola.rs/benchmarks.html>
* GitHub. (n.d.). About large files on GitHub. GitHub Docs.   
   Retrieved from <https://docs.github.com/en/repositories/working-with-files/managing-large-files/about-large-files-on-github>
* Wikipedia contributors. (n.d). Regression analysis. In Wikipedia. Retrieved 11/16/2023, from <https://en.wikipedia.org/wiki/Regression_analysis>

**Course Instructor Signature/Date:**

The research is exempt from an IRB Review.

An IRB approval is in place (provide proof in appendix B).

Course Instructor’s Approval Status: Approved

Date: Click here to enter a date.

Reviewed by:

Comments: Click here to enter text.