Notes:

Yellow means I want to reword/rework it

There on comments on this page as well highlighting some of my main thoughts

Title

Abstract

[Todo: Abstract]

# Introduction

## What is Machine Learning

In the modern era of data, individuals are generating and collecting vast amounts of information which can be difficult to manage, comprehend and utilize effectively. To make sense of this data, the field of computer science and artificial intelligence has introduced a sophisticated tool: Machine Learning. Machine Learning (ML) was created to help process these large amounts of data, many of which are too vast for human [processing ?]. This is made possible by the ability of machine learning algorithms to automatically learn from data, identify patterns, and make predictions “without being explicitly programmed [[MIT](#_https://mitsloan.mit.edu/ideas-made)].” In other words, instead of being given specific instructions for what to do with the data, machine learning algorithms are designed to learn from patterns in the data itself. When these algorithms are given large datasets to analyze, they are able to identify underlying patterns and make predictions based on new, unseen data.

## Machine Learning in Cyber Security

This ability to learn and adapt to new data makes ML particularly versatile and ideal for a wide array of situations, especially pertaining to the constantly evolving nature of Cyber Security. [include more history here?]. While, historically, rule-based systems were the primary method of detecting threats, these methods often struggle to identify new and emerging threats, as they rely on pre-defined rule sets that can be easily exploited. However, the adaptability of ML algorithms enables them to identify anomalies that rule-based methods may miss; instead of relying on a set of pre-defined rule, they can create new rules as necessary. [todo: add more context]

An example of ML's superiority over rule-based systems can be seen in the context of spam detection. Spam emails and text messages are a major threat to cyber security and have many malicious intentions, including illegally accessing confidential data such as passwords and other important identifiers, spreading malicious software with harmful links or attachments, and more [[Detection](#_Detection)]. [todo: add more context]

This adaptability makes ML an effective tool for identifying new and emerging spamming techniques that may be missed by traditional rule-based methods.

[segue to spam detection here]

## Introduction to our Study

[todo] – include dataset name, programming language (python), Kaggle, 3 types of models (KNN, naïve bayes, decision tree/random forest), NLP/ NLTK(define stop words)

# Related Work

[todo]

# Methods

Before we begin training our models, we much first preprocess the data to ensure consistency and compatibility. We will procedurally analyze and modify our dataset to increase quality and resulting analytic scores, while avoiding over-generalization and over-fitting. This transformed dataset will then be trained and tested across each model individually.

## Introduce Dataset

We have selected the publicly available Kaggle dataset, “Spam (or) Ham,” to train and test our spam classification models [[SpamVHam](#_[SpamVHam])]. It is important to note that this dataset is a condensed version of The University of California, Irvine’s (UCI) ‘SMS Spam Collection Dataset’ [[OGSMS](#_[OGSMS])]. The original message collection was consolidated from various public sources, including 425 spam messages from Grumbletext, 3,375 ham messages from NUS SMS Corpus (NSC), 450 ham messages from Caroline Tag's PhD Thesis, and 1,002 ham and 322 spam messages from the SMS Spam Corpus v.0.1 Big [[OGSMS](#_[OGSMS])]. The version we are using was selected due to accessibility and minor pre-processing, as two messages were formatted incorrectly and resulted in missing values. As such, our version contains 5572 Short Message Service (SMS) messages, 5169 of which are unique. Each message is split between two columns: ‘Class’ which identifies whether it is spam or ham, and ‘sms’ which contains the plain-text version of the message.

## Data Configuration (Transformation?)

To prepare our data for use in our ML algorithms, we first needed to review and analyze a series of properties and decide which, if any, transformations were required. We began by importing the comma-separated values (CSV) data file using the python library, Pandas. Pandas is an open-source python library which is commonly used in machine learning as it helps organize, manipulate, and analyze complex tabular data. Our process begins through a series of steps, including observing general information, checking for extraneous null values, viewing various observations, and analyzing statistical information.

We started by viewing the general and statistical information to confirm that each column, ‘Class’ and ‘sms,’ contained 5572 non-null entries. As shown in Figure 1, each column not only contained this expected value, but the column values for ‘count’ were equivalent. If this were not true, that would mean we had incomplete or missing data which would need to be trimmed or modeled accordingly. Figure 1 also shows the number of unique observations for each variable. As expected, ‘Class’ indicates 2 in this row; however, ‘sms’ indicated

the number of unique observations,

Similarly, we needed to check if there were any unnecessary null values anywhere in our dataset which could incorrectly impact the ML algorithms.

Lastly, we observed the statistics of our dataset to conclude the occurrences of each variable, the most frequent value, and the top values corresponding frequency amount. We use this information to determine the completeness of our data, and verify the values contained are expected and acceptable. For example, Figure 1 shows that

These are important steps as they are examples of noise which can cause overfitting and potentially “result in more complex models that miss the true pattern” [[MLR](#_[MLR])].

A screenshot of a computer screen

Description automatically generated with medium confidence

The last step before transforming our data involved observing a few examples and their corresponding features. This provides additional context which enables us to understand how the information is formatted and deduce if transformation is necessary. During this step, we concluded that the messages were unstructured, containing a mix of lower and upper case, punctuation, and stop words. Before we could continue with testing and training, the messages would require natural language preprocessing. In this step, we also noticed that the identifying classes were also categorical form, e.g. ‘ham’ and ‘spam’, and should be converted to numerical form, e.g. ‘0’ and ‘1’.

## Pre-Training Setup

## Algorithms

Introduce dataset

[todo]

# Results

[todo]

# Conclusion

[todo]

# Resources

## [MIT]

* + <https://mitsloan.mit.edu/ideas-made-to-matter/machine-learning-explained#:~:text=Machine%20learning%20is%20a%20subfield,learn%20without%20explicitly%20being%20programmed>
  + Note: first quote was actually quoted in 1950s by AI pioneer [Arthur Samuel](https://en.wikipedia.org/wiki/Arthur_Samuel) – see if a og source is available

## [CS]

* https://www.cs.unc.edu/~jeffay/courses/nidsS05/ai/00816048.pdf

## [Classifiers]

* https://www.researchgate.net/publication/328907962\_A\_Comparative\_Study\_of\_Spam\_SMS\_Detection\_Using\_Machine\_Learning\_Classifiers

## [Detection]

* https://www.sciencedirect.com/science/article/pii/S1742287615000079?ref=pdf\_download&fr=RR-2&rr=7b77774e197530dd#page=10&zoom=100,0,0

## [SpamVHam]

* <https://www.kaggle.com/datasets/arunasivapragasam/spam-or-ham>

## [OGSMS]

* + <https://archive.ics.uci.edu/ml/datasets/SMS+Spam+Collection>

## [MLR]

* + Machine Learning with R - Third Edition, Brett Lantz
  + https://learning.oreilly.com/library/view/machine-learning-with/9781788295864/