# Determining the Best Email and Human Behavior Features on Phishing Email Classification

Ahmad Fadhil Naswir[1], Lailatul Qadri Zakaria[2]
Faculty of Information Science and Technology
Universiti Kebangsaan Malaysia
Bangi, Selangor, Malaysia

Saidah Saad[3]
Faculty of Information Science and Technology
Universiti Kebangsaan Malaysia
Bangi, Selangor, Malaysia

***Abstract*—There are many email filters that have been developed for classifying spam and phishing email. However, there is still a lack of phishing email filters developed because of the complexity of feature extraction and selection of the data. There are several categories of features for classifying phishing emails, either on the email part or on the human part. The absence of which features are best for helping to classify phishing emails is one of the challenges; in the previous experiment, there was no benchmark for the features to be used for phishing email classification. This research will provide new insight into the feature selection process in the phishing email classification area. Therefore, this work extracts the features based on the category and determines which features have the most impact on classifying email as phishing or not phishing using a machine learning approach. Feature selection is one of the essential parts of getting a good classification result. Therefore, obtaining the best features from email and human behavior will significantly impact phishing classification. This research collects the public phishing email dataset, extracts the features based on category using Python, and determines the feature importance using machine learning approaches with the PyCaret library. The dataset experimented on three different experiments in which each feature category was separated, and one experiment was the combined feature selection. Binary classification is also done with the extracted features. The experiment verified that the proposed method gave a good result in feature importance and the binary classification using selected features in terms of accuracy compared to previous research. The highest result obtained is the classification with combined features with 98% accuracy. The results obtained are better compared to previous studies. Hence, this research proves that the selected features will increase the performance of the classification.**

***Keywords*—Phishing; phishing email classification; features selection; binary classification; email features; human features**

## I. Introduction

In spite of the fact that numerous email filters have been created for spam emails, exceptionally few phishing email filters have been created [1]. Due to the complexity of current phishing attacks, detecting and classifying phishing attacks is a major challenge. Obtaining high-quality training data is one of the biggest problems with machine learning, as labelling data can be tedious and costly [2]. Valuating the dataset is hard because it involves figuring out the limits of the phishing email dataset and whether or not it is the same appropriate dataset as in the previous study. This is done by looking at the dataset that the previous researcher used.

For techniques used in the classification process, machine learning algorithms such as Naïve Bayes, Support Vector Machine, Decision Tree, Random Forest, and others are implemented according to the features used. From previous studies in the classification area, SVM and NB are the most commonly used methods in the phishing email classification area. The accuracy of the result from both algorithms is very satisfying. However, in this case, extraction and selection of features based on the email structures play an essential role in improving the result of email classification on specific content [3]. To improve the classification performance, a feature selection algorithm is presented, and feature selection methods are commonly used to reduce the dimensionality of datasets to improve the classification performance, reduce the processing time, or both [4]. In recent studies, there are several researches that implement word embedding for feature representation, which is one way to solve the text classification problem [5].

Phishing email classification requires clear features so that the classification produces accuracy and good performance evaluation results. The features that have been selected and extracted will represent the identity of an email itself; examples of some of the features used for email classification, especially phishing emails, are the body and URL features. Features will be extracted based on the feature type itself [6].

The relationship between features can also be determined by other fields, such as linguistic features. On the behavior side, the features extracted are classified as text feature extraction, which extracts text information that is used with the aim of representing a text message [7]. Stylometrics is one of the fields of linguistics related to the procedure for writing text, where this feature is used to identify the contents of phishing emails. Stylometric features have several categories, namely lexical, structural, content-specific, syntactic, and idiosyncratic [8]. Each category of features has its own characteristics. Email also has several main parts: a header, body, and URL.

Each corpus will be processed by following the research framework, including feature extraction. The first corpus used in this research is the IWSPA-AP 2018 dataset, which was requested by the committee of the Security and Privacy Analytics workshop [9]. The second is the custom-made corpus that combines 2 email datasets, the Enron CALO dataset and PhishCorpus [10] [11]. The detailed information about each corpus will be explained more in data collection.

Feature selection is an important stage that can affect the results of a classification process. The features to be used must have a significant impact that can make performance more accurate, especially in the machine learning area. Feature selection can be roughly classified into supervised, semi-supervised, and unsupervised methods [12]. There are standard feature selection methods for categorical data, namely Chi-Squared and information gain, but this method has drawbacks for data that has many categories. In phishing emails, some features are classified into categorical data with more than two categories. Therefore, this problem requires a new method to determine the effectiveness of the features in a dataset used. Also, there is no benchmark for the best features in the phishing email classification area [13]. PyCaret is a Python library that is useful for automating machine learning workflows. One of the features of PyCaret is feature importance, which is the process of evaluating the features that contribute the most to predicting target variables using a combination of supervised techniques, including Random Forest, Adaboost, and others [14].

The current issue regarding feature selection and extraction in the email classification area is that there is no benchmark for the feature set and which feature is the best for identifying phishing properly. Thus, it is promising that by using a combination of features on different fields with email features and using PyCaret's feature-critical algorithm can identify which features have the most significant impact on the area of classification of phishing emails. In addition, the list of best features can be produced and used as a benchmark for feature sets to help improve the performance of phishing email classification.

The rest of the paper is organized as follows: Section II discusses the works that are related to determining features. The framework for feature extraction in this experiment is in Section III, and an explanation of the data preparation and feature selection process is in Section IV. Section V will show the results of all the experiments carried out and closed with a conclusion in Section VI.

## II. Related Work

This experiment is to determine the best feature for phishing email classification using feature importance, and several researchers used either stylometric features, email features, or both features for their experiments.

In [15] experiment, they proposed a classification method using the persuasion principle based on content-specific categories in the stylometric area. Several persuasion principles were used, namely: Authority, Reciprocation, and Scarcity were used as one of the feature selections in this experiment. They also used email features such as URL and body features which are included as part of feature selection. The dataset used is from Nazario PhishCorpus.

In [16], used word analysis features to detect spear-phishing emails. In contrast to the above study, this study uses a spear-phishing dataset collected from Enron Corpus because spear phishing has a specific target to attack. In the study, the analysis features used are those on the behavioral aspect, such as gender features, stylometric features, and personality features. The gender features will detect the gender of the email sender based on the choices of words in the email. The stylometric features are from the grammar side, and the personality features are emotion detection due to word selection. These features are classified as stylistic features, which is the study of the interpretation of each individual's text or spoken language in terms of accent, grammar, or word choices (lexicon). For author identification, stylistic features are often used in several journals and articles with different fields and areas, for example, author identification of a book or gender identification of a character in a novel. In this case, stylistic features are used for author identification to detect spear phishing.

Another comparison of the phishing classification using stylometric features is with [17]. The experiment extracted 26 human features more focused on syntactic feature categories. For machine learning, the classifiers used are DT, SVM, Naïve Bayes, Logistic Regression, and Neural Network. IWSPA is used as a dataset for phishing classification.

Features selection has become a crucial part of conducting email classification research. A better result will be given by selecting the best and most convenient features in the experiment. However, there are no such optimized features that can be equally applicable in all domains [18]. In the past years, the researchers tried different feature selection and extraction. The list of features to be tested is obtained based on the literature, which states that there is a lack of human features (stylometric) approaches in different research fields. In this case, stylometric features are combined with email features to detect phishing emails [19].

There are several categories from stylometrics that indicate the email is categorized as phishing or legit email. Table II shows the categories of stylometric features used in this study, namely lexical, syntactic, content-specific, structural, and idiosyncratic. The most commonly used features for phishing email classification are header, body, and URL for the email features. Features extracted are part of the main category of an email, where each category (header, body, and URL) has value in the form of text or numbers that will be analyzed at a later stage. The list of email features extracted is shown in Table I. Based on the literature survey, this research will combine features from two main features, namely human behavior focusing on stylometric features and email behavior features focusing on the structure, content, and metadata of the email itself and evaluating the effectiveness of the features extracted.

The use of PyCaret for feature engineering or classification tasks was also carried out in several experiments. The study [20] used PyCaret to focus on the feature engineering steps in the classification process using the Titanic dataset. Feature importance is used to select the best features to increase the efficiency of the classification model. PyCaret is also used for other areas, such as regression analysis. The author in [21] uses PyCaret to predict the price of a diamond. The dataset used is from the PyCaret repository. The best machine learning approaches for the experiment are Gradient Boosting Machine and Light Gradient Boosting Machine, respectively. Due to this library is newly developed in the area of machine learning

tools, there is still little research that uses PyCaret in other research fields as a supporting library.

In [22], the experiment used PyCaret to compare the selection of Polycystic Ovarian Syndrome (PCOS) attributes. The feature selection method used is GA which is the input for PyCaret. The results obtained from this experiment are accuracy of 87% with the extra tree algorithm that has been provided in PyCaret. However, the feature selection used is using another method (GA) in which Pycaret itself already has a feature for calculating feature importance which can be used to evaluate the feature that has been selected.

In [23], they proposed a classification model to detect cardiovascular disease using Pycaret. The features used have been selected in advance based on previous research in which there is no feature engineering process in this experiment. The research [24] compares 14 machine learning models from PyCaret to predict whether students will drop out or not. The results obtained by experimenting with all the features, with the Decision Tree as the most appropriate model, are pretty good. Feature importance is used to see which features affect the classification results, where feature importance is obtained according to the experimental feature analysis. Finally, experiments using PyCaret as a tool to compare models can be done well and get satisfactory results.

Clustering and classification are performed to analyze employee satisfaction using machine learning. A comparison of the best models was also carried out on 5 models included in PyCaret. The dataset used is the Kaggle-IBM analytics dataset, which consists of 1470 samples. This research uses Principal Component Analysis (PCA) for feature engineering to simplify the model features. PCA has several weaknesses, one of which is that it can eliminate information from these features because the correlation between data can be lost [25]. However, it can be seen that the flexibility of PyCaret can be applied to research in other areas as well.

It can be seen that several previous studies using PyCaret have not used the feature importance provided in PyCaret to evaluate the top features. In this study, PyCaret feature importance is used to determine the best features that can be selected to evaluate each extracted category feature.

## III. Methodology

This section will determine which features significantly influence the phishing email classification process by experimenting with the dataset obtained and performing feature extraction. The results of this experiment will be a combined list of features that have a high impact on the classification of phishing emails from the email behavior part, namely the structure of email and human behavior, stylometric area. The framework for the feature extraction is shown in Fig. 1 below:

### A. Data Collection

Phishing email datasets are very limited in number; there are only a few publicly available sources. In previous studies, the majority of researchers used the same dataset source, and modifications were made according to the needs of the research. In this experiment, two corpora were used to answer the question of which features had the most significant influence on the classification of phishing emails.

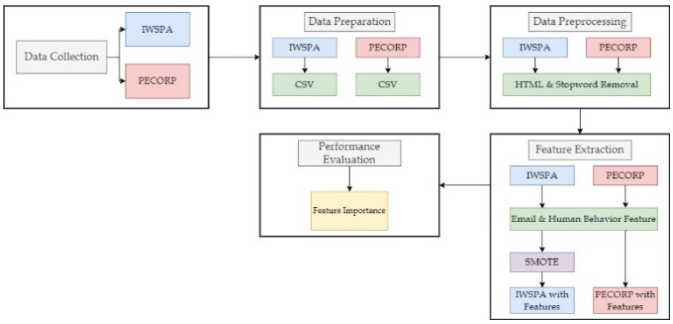


Fig. 1. The Framework for Feature Extraction

#### 1) IWSPA-AP 2018 Corpus (IWSPA)

The first is the IWSPA-AP 2018 corpus, obtained by submitting an application to gain access to the dataset. IWSPA-AP 2018 has two different types of datasets: the IWSPA dataset with full header and no header. The full header IWSPA dataset consists of 4082 legitimate emails and 503 phishing emails, and no header IWSPA dataset consists of 5091 legitimate emails and 628 phishing emails. This corpus is classified as an unbalanced dataset because of the massive difference in the ratio of legitimate and phishing emails.

This corpus is provided in the form of text files in which every email is on a separate text file. In order to work efficiently with this data, combining all the text files into one CSV file is required to do further processing. The data extracted and transformed into CSV files is organized as follows: From, To, Date, Subject, and Body. There are additional columns, namely Label and Label Number, to determine the type of email, where 1 is for Phishing, 0 for Ham or non-phishing email.

#### 2) Phishing Enron Corpus (PECORP)

The second corpus comes from a combination of two publicly available datasets, namely the Phishing and the Enron corpus. These two corpora are combined to create a full phishing email dataset in which the phishing emails from the Phishing corpus and legitimate emails from the Enron corpus.

A total of 2712 phishing emails come from the Online Phishing Corpus by Nazario, and 2801 legitimate emails come from the CALO Enron Email Dataset by Carnegie Mello University (CMU). This dataset was collected and prepared by the CALO Project (A Cognitive Assistant that Learns and Organizes). The CALO project dataset is the most widely used and is publicly available for download, as well as the Online Phishing Corpus by Nazario. Both corpora are combined into one CSV file. In this research, this combined corpus is called PECORP (Phishing Enron Corpus).

### B. Data Preparation

Both corpora have a different format, IWSPA corpus is the text file type, and PECORP corpus is the "mbox" file type. To make the two corpora can be used as experimental material, a process is carried out to convert the two corpora into CSV format so that further processes can be carried out smoothly. Each corpus has different content of email and fields, either

from header or body that needs to be processed so that it can produce a good corpus. The corpora are converted from their original type to CSV format with the extraction of some column/fields, namely "FROM," "TO," "DATE," "SUBJECT," "BODY," and "LABEL." Both corpora's conversion and field extraction are done with Python using the PANDAS library.

The IWSPA corpus has an unbalanced amount of data ratio for the amount of data. In contrast, PECORP has been sorted for the same amount of data on phishing and legitimate emails, which was already explained in the previous section. To overcome unbalanced datasets, data processing stages are carried out so that the data in training can show good performance evaluation results. After the data preparation is complete, it will proceed to the next stage, namely data preprocessing.

### C. Data Preprocessing

The preprocessing step is basically a data cleansing before the data is ready to move into the next classification process. It is crucial to preprocess the data with machine learning approaches [26]. Some preprocessing data is carried out so that the results obtained can be evaluated and meet the requirements of a good experiment. Several fields need to be preprocessed before performing feature extraction: punctuation removal for “FROM” fields, HTML checker and removal for “BODY,” and tokenization for each part of the email. The preprocessing results will be continued with the extraction of features that are in accordance with the research objective, namely human features and email features. For each corpus used, both IWSPA and PECORP will go through the preprocessing stages individually, which are carried out using the Python programming language. After this process, the data are technically feasible to pass the next stage, namely feature selection.

### D. Feature Selection and Extraction

Feature selection is one stage for determining which features on the email and human side significantly affect phishing email classification. In this research, the list of features to be tested is obtained based on the literature, which states a lack of human features (stylometric) approaches in different research fields [19]. Both corpora will go through a feature extraction process after the preprocessing process has been carried out on them. The features will be extracted based on their respective categories, namely email and human behavior features. The extraction process is carried out using the Python programming language using various supporting libraries. “Pandas” library for the data frame, “re” library for regular expressions, BeautifulSoup4 for HTML file text usage, Spellchecker library for misspelt words and NLTK library for stopwords and tokenize usage. All features extracted will be placed in a new column with the appropriate data rows with the help of the Pandas library. The extracted features are as follows:

#### 1) Email Features

Based on the observations in the literature review, the most commonly used email features for phishing email classification are header, body, and URL. Features extracted are part of the main category of an email, where each category (header, body, and URL) has value in the form of text or numbers that will be analyzed at a later stage. The process is done using Python, where each part of the email feature extraction is done in a separate function. Hence, the feature is obtained according to its category (header, body, URL).

Email feature extraction uses several Python libraries, with Jupyter Notebook as the tool. For the data frame, the Pandas library is used as the initial frame for the data analysis and manipulation. NLTK and BeautifulSoup4 libraries are used to tokenize the email field and detect HTML elements (URL and JavaScript), respectively. A regular expression is used to obtain the time from the email.

The dataset obtained and analyzed is in the form of full text, which means that the features that can be extracted are features that are in accordance with the type of data itself, for example, the number of several parts of the email, such as character length, token length, URL length, body shape, and others. Some of the URL features were extracted based on previous experiments [29]. There are several additional URL features that were extracted. The list of email features extracted is shown in Table I below:

TABLE I. LIST OF EMAIL FEATURES

| Feature | Observed Field | Value | Description |
|---|---|---|---|
| Header | FROM | Char Length | Total number of characters in the “FROM” field |
| | SUBJECT | Token Length | Total number of tokens in the “SUBJECT” field |
| | TIME | Time | Time stamp when the email is received in “hour:minute” format |
| Body | BODY TEXT | Body Format | Boolean value that represents email body is an HTML format or non-HTML format |
| | | JavaScript Presence | Boolean value that represents there are <script> tag in the HTML format |
| URL | BODY TEXT | URL Flag | Boolean value that represents the presence of URL in an email by detecting <a> tag |
| | | URL Length | A total number of URLs character length |
| | | URL Count | A total number of URLs found in the body text |
| | | Presence of IP address | Boolean value that checks if the URL is on the IP address form. |

#### 2) Human Features

In terms of human features, there are 29 features from five categories that were observed and extracted. These features were extracted in each corpus, namely IWSPA and PECORP. The extracted features are based on the stylometric area, which has five categories: lexical, syntactic, structural, content-

specific, and idiosyncratic. Each category has its own characteristics which can be extracted into a feature in the body of the email. Thus, each stylometrics category produces features described descriptively in the following Table II.

TABLE II. LIST OF HUMAN FEATURES

| Feature | Observed Field | Value | Description |
|---|---|---|---|
| Lexical | Full email | Character Count | A total number of characters in the email text |
| | | Token Count | A total number of tokens in the email text |
| | | Average Word Length | Difference between character count and token count |
| | | Lexical Diversity | The ratio of different unique word stems (types) to the total number of words (tokens) |
| Syntactic | Body Text | Presence of function word | Function words include determiners, conjunctions, prepositions, pronouns, auxiliary verbs, modals, qualifiers, and question words. |
| Content-Specific based on [15] | Body Text | Presence of punctuation | A number of punctuations in the email text |
| | | Authority | A total occurrence of each word of: Paypal, Verify, Fraud, Management, Identity, Debit |
| | | Reciprocation | A total occurrence of each word of: Benefits, Bank, Customers, Accounts, Updates |
| | | Scarcity | A total occurrence of each word of: Limited, Services, Suspension, Suspended, Terminated |
| Structural | Body Text | Line Count | A total number of lines in the body text |
| | | Sentence Count | A total number of sentences in the body text |
| | | Word Count | A total number of words in the body text |
| | | Character Count | A total number of characters in the body text |
| | | Average Sentence length in terms of Character | Average calculation of a total number of characters with a total number of characters |
| | | Average Sentence length in terms of Word | Average calculation of a total number of words with a total number of words |
| | | Average Line length in terms of Sentence | Average calculation of the total number of lines with a total number of sentences |
| Idiosyncratic | Body Text | Misspelt word count | Total number of possible amounts of misspelt word in the body text |

The main library is the same for extracting the email features for this category, namely PANDAS and NLTK. However, several additional packages and libraries are used to extract the specific features. The feature extraction for obtaining function words is based on the syntactic feature category using the POS (Part-of-Speech) Tag method with NLTK POS Tag packages library. The packages are set to collect specific words according to function word definition (e.g., conjunction, determiners, prepositions, etc.). There is one additional package for sentence tokenization from the NLTK library. Lastly, the spellchecker python library is used for obtaining the total of a misspelt word, and the library provides the total number of possible misspelt words and the list of misspelt words.

### *E. SMOTE Implementation for IWSPA-2018 Corpus*

The machine learning algorithm's performance is evaluated by the accuracy result and evaluation of the dataset or corpora in the experiment. The imbalanced dataset is not appropriate to get optimum results since the labelled data is not equal, and it will lead to a biased classification result [27]. There are several methods for overcoming this problem, such as random over-sampling and under-sampling, which are common approaches to solving the issue. However, these approaches have several drawbacks; under-sampling is likely to dispose of valuable data, whereas over-sampling can heighten the probability of overfitting [28]. In this research, the method used to overcome the problem regarding the imbalanced dataset is the SMOTE (Synthetic Minority Oversampling Technique) method.

SMOTE selects feature samples from the available dataset, draws a line between the samples in the feature space, and creates a new sample at a point along the drawn line. By choosing the minority class (label) for generating a new feature sample, a synthetic sample is created at a random point between the two nearest samples in the feature space [27]. IWSPA-AP 2018 has an unbalanced amount of data ratio for the amount of data, while PECORP has been sorted for the same amount of data on phishing emails and legitimate emails, which is already explained in the previous section. The balancing dataset technique is needed for the IWSPA corpus to overcome unbalanced datasets. The IWSPA-AP 2018 corpus has unbalanced data, which consists of roughly 4082 legitimate emails and 503 phishing emails. To avoid bias in the experiment result, SMOTE needs to be implemented on the IWSPA corpus, which is needed for identifying which features have the most relevant impact on phishing email classification using email and human features. For SMOTE implementation, several Python libraries and packages are required to solve the unbalanced dataset. The Imbalanced-Learn python library and SMOTE package are used to process the IWSPA corpus. By setting up the data frame that meets the requirements for the required library, the SMOTE method can be applied to the IWSPA corpus.

The balancing dataset technique is needed for the IWSPA corpus to overcome unbalanced datasets. The SMOTE technique is applied to the IWSPA corpus. The number of rows has increased from 4082 rows to 8164 rows, which means SMOTE has created feature values between each feature in the feature space. In this research, the IWSPA SMOTE will be called the IWSPA-SM corpus. Thus, the new dataset (IWSPA-SM) has been acquired and will be helpful to help determine the best feature from the selected feature set

based on email features and human features. Starting from this section down, the new dataset will be called IWSPA-SM for IWSPA SMOTE, and the old dataset will be called IWSPA-NS for IWSPA NON-SMOTE.

### *F. Method Implementation*

The implementation of feature extraction is done using the Python programming language and is supported by Jupyter Notebook for the interface tool. The feature selection phase will determine which human and email features significantly influence the phishing email. Each corpus will undergo three experiments with different feature category selections: 1) email features, 2) human features, and 3) combining email and human features. By dividing the corpus with its extracted category features, the analysis will be able to identify which category contributed the most to the classification result. The list of experiments for feature selection is as follows:

*a) IWSPA-NS (NON-SMOTE) with email features*
*b) IWSPA-NS (NON-SMOTE) with human features*
*c) IWSPA-NS (NON-SMOTE) with combined features*
*d) IWSPA-SM (SMOTE) with email features*
*e) IWSPA-SM (SMOTE) with human features*
*f) IWSPA-SM (SMOTE) with combined features*
*g) PECORP with email features*
*h) PECORP with human features*
*i) PECORP with combined features*

The dataset generated from the extraction feature will be carried out to determine the importance of features for each category. Feature importance methods use various ways to obtain and calculate the feature set to determine which feature has the most impact on the current dataset. There are several types of feature importance scores, and commonly the methods are feature importance from coefficients and feature importance from a tree-based model. One way to implement feature importance is by using PyCaret, a python library that provides machine learning models for data classification, including the feature importance method. It uses a combination of several supervised techniques, including Random Forest, Adaboost, and Linear Correlation with the permutation importance technique, to select the subset of features that are most important for the model. Working with selected features instead of all the features will reduce the risk of over-fitting, improve accuracy, and decrease training time [14]. The experiment details and results are shown in the section below.

## IV. Experiment and Result

This experiment has two main outputs, namely the best features in each experiment with each corpus and the performance results from binary classifications using each corpus. The feature importance result and the classification evaluation of this experiment were measured using the performance evaluation method provided by the PyCaret library; the results of the evaluation are as follows:

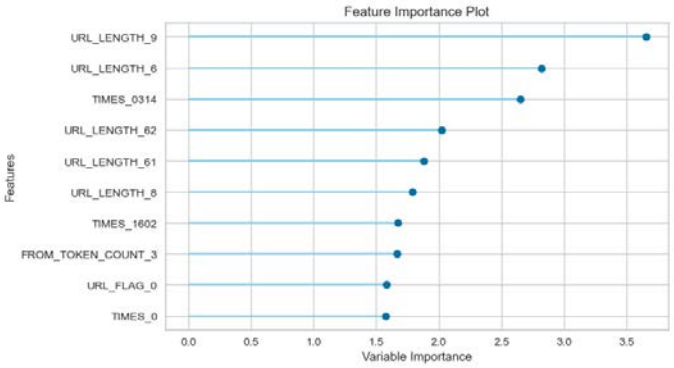


Fig. 2. IWSPA-NS Email Feature Importance Result

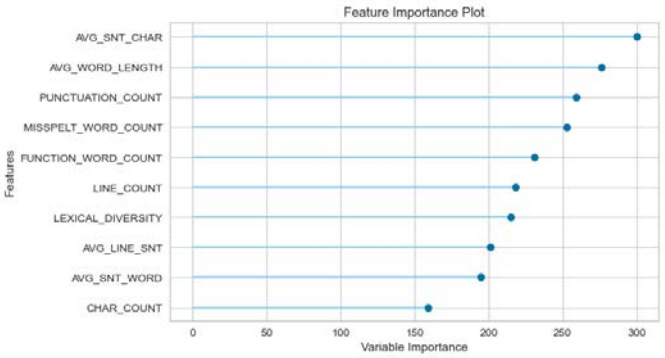


Fig. 3. IWSPA-NS Human Feature Importance Result

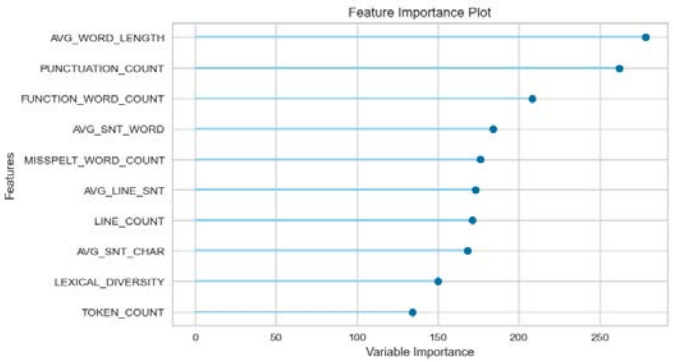


Fig. 4. IWSPA-NS Combined Feature Importance Result

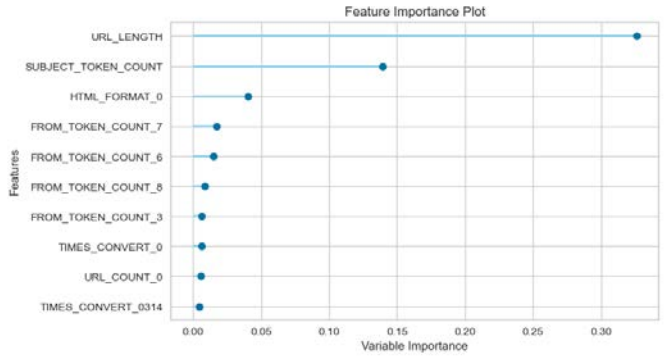


Fig. 5. IWSPA-SM Email Feature Importance Result

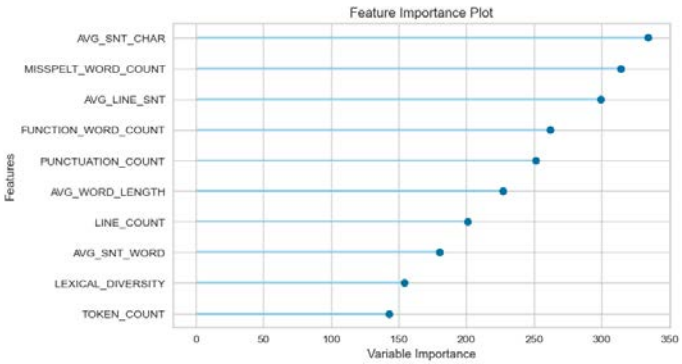


Fig. 6. IWSPA-SM Human Feature Importance Result

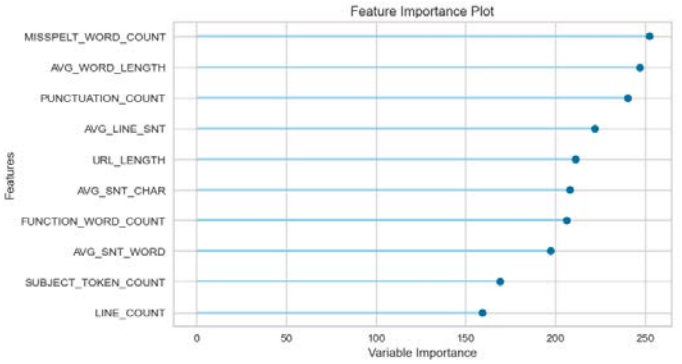


Fig. 7. IWSPA-SM Combined Feature Importance Result

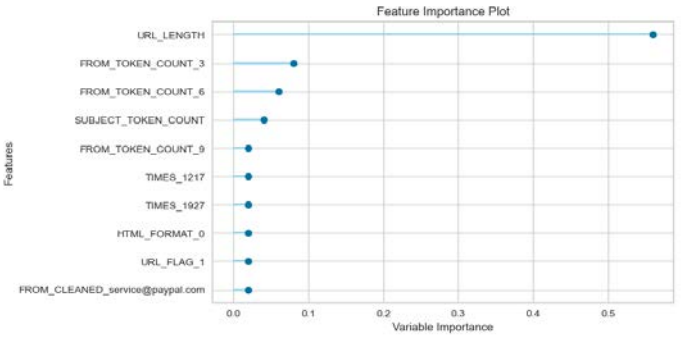


Fig. 8. PECORP Email Feature Importance Result

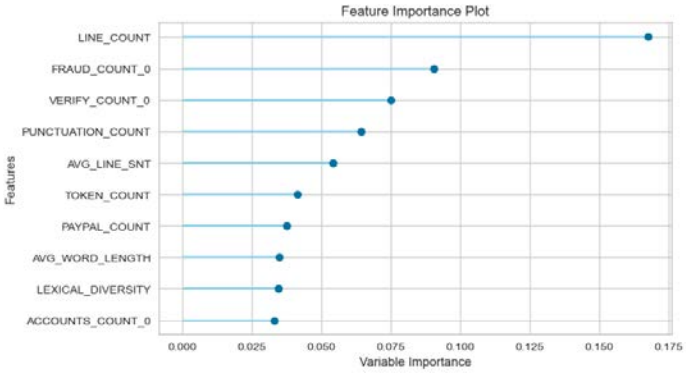


Fig. 9. PECORP Human Feature Importance Result

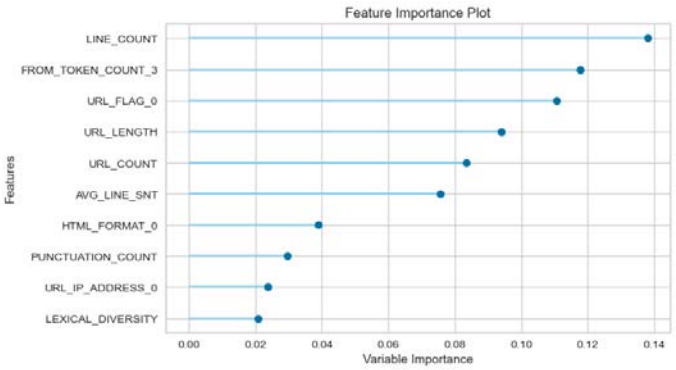


Fig. 10. PECORP Combined Feature Importance Result

The figures above are the result of the important feature of using the PyCaret library on all corpora. There are nine results obtained in each experiment where the experiment produces the output, namely the best feature that has the most significant impact in determining phishing emails in each corpus. The following are the explanations of each figure resulting from the experiment for determining the best features using feature importance with PyCaret.

Fig. 2 is the resulting diagram of IWSPA-NS with only email features extracted where the best feature is URL length. Fig. 3 is IWSPA-NS with only human features extraction, where the best feature is Average Sentence Length in terms of Characters. In Fig. 4, the best feature is the Average Word Length. In the IWSPA-SM corpus, the best features obtained are URL length for email features, Average Sentence Length in terms of characters for human features, and Misspelt Word Count for combined features, which can be seen in Fig. 5, Fig. 6, and Fig. 7 consecutively. Fig. 8 shows the result of the best feature in the PECORP corpus, namely URL length. Fig. 9 and Fig. 10 show that the best feature for experimenting with human features and combined features in the PECORP corpus is Line Count. The determination of the best feature based on the variable importance value of each feature from PyCaret uses a combination of permutation importance techniques, including Random Forest, Adaboost, and Linear correlation with the target feature. Therefore, the results obtained above are based on the algorithm of feature importance provided by PyCaret.

From the experimental results above, it can be seen that the variable importance value generated by this experiment is very diverse for several experiments on the corpus used. It can happen because the features used and extracted in each experiment are classified as "categorical features," where the coverage of the category features is extensive. For example, in the email category feature, the "Time" feature is a feature that contains numbers in time format extracted from the email header. In the human category, the "Lexical Diversity" feature contains decimal numbers with a wide range of values for each email. With a very diverse feature value of each feature extracted, the results of the variable importance value have a reasonably large range. Therefore, this experiment aims to find out what features have a significant impact on helping classify phishing emails. The scope of features that have been extracted can be in the form of numerical, boolean, or categorical values.

The results of this important feature are novelty results that can be used as a reference for the selection of features or feature engineering in the classification process using phishing email. It can be seen that some features are the same in the different corpora, for example, URL length and Line Count. This shows that the effect of these features is beneficial to improving the performance of the phishing email classification process. Moreover, further experiments can make it easier for the feature selection process to classify phishing emails with different approaches.

Tables III, Table IV, and Table V show the result of each experiment using different feature categories and corpus using the PyCaret library. The result shown above is the mean value of a 10-fold cross-validation classification with the performance metrics value for evaluation. Thirteen models are used in each classification, and the highest results from these models are shown as follows:

TABLE III. IWSPA-NS PERFORMANCE EVALUATION

| **Evaluation Performance** | **IWSPA-NS with Email Feature** | **IWSPA-NS with Human Feature** | **IWSPA-NS with Combined Feature** |
|---|---|---|---|
| Model | Random Forest | Light Gradient Boosting Machine | Light Gradient Boosting Machine |
| Accuracy | 0.9346 | 0.9698 | **0.9713** |
| AUC | 0.9001 | 0.9745 | **0.9879** |
| Recall | 0.4730 | **0.7838** | 0.7703 |
| Precision | 0.9171 | 0.9465 | **0.9773** |
| F1 | 0.6218 | 0.8555 | **0.8603** |
| Kappa | 0.5904 | 0.8388 | **0.8446** |
| MCC | 0.6304 | 0.8447 | **0.8528** |

TABLE IV. IWSPA-SM PERFORMANCE EVALUATION

| **Evaluation Performance** | **IWSPA-SM with Email Feature** | **IWSPA-SM with Human Feature** | **IWSPA-SM with Combined Feature** |
|---|---|---|---|
| Model | Random Forest | Light Gradient Boosting Machine | Light Gradient Boosting Machine |
| Accuracy | 0.9107 | 0.9790 | **0.9844** |
| AUC | 0.9679 | 0.9969 | **0.9982** |
| Recall | 0.9180 | 0.9792 | **0.9820** |
| Precision | 0.9042 | 0.9786 | **0.9866** |
| F1 | 0.9109 | 0.9789 | **0.9843** |
| Kappa | 0.8215 | 0.9580 | **0.9688** |
| MCC | 0.8219 | 0.9581 | **0.9689** |

TABLE V. PECORP PERFORMANCE EVALUATION

| **Evaluation Performance** | **PECORP with Email Feature** | **PECORP with Human Feature** | **PECORP with Combined Feature** |
|---|---|---|---|
| Model | Ada Boost Classifier | Extra Trees Classifier | Decision Tree Classifier |
| Accuracy | 0.9992 | 0.9964 | **0.9997** |
| AUC | **1.0000** | 0.9999 | 0.9997 |
| Recall | 0.9990 | 0.9990 | **1.0000** |
| Precision | 0.9995 | 0.9938 | **0.9995** |
| F1 | 0.9992 | 0.9964 | **0.9997** |
| Kappa | 0.9984 | 0.9927 | **0.9995** |
| MCC | 0.9984 | 0.9928 | **0.9995** |

Table III shows the evaluation result of the experiment using the IWSPA-NS corpus with the PyCaret library. It can be seen in the comparison of the results of each category feature used in the phishing email classification process. The highest average result is in the experiment using combined features except for the recall value. Table IV shows the results of evaluating the IWSPA-SM corpus, where the highest average result was achieved in the experiment using combined features. Table V shows the experimental results of the PECORP corpus, which shows that the highest average result was obtained in the experiment using combined features. Based on the results obtained from the experiment, which are very promising, this shows that the combination of features used can improve the performance of phishing email classification. The result can be seen in Table VI.

TABLE VI. EXPERIMENT RESULT COMPARISON

| **Research** | **Feature** | **Dataset** | **Accuracy** |
|---|---|---|---|
| IWSPA-NS Email Features | Email | IWSPA | 0.9346 |
| IWSPA-NS Human Features | Human | IWSPA | 0.9698 |
| IWSPA-NS Combined Features | Email + Human | IWSPA | **0.9713** |
| IWSPA-SM Email Features | Email | IWSPA | 0.9107 |
| IWSPA-SM Human Features | Human | IWSPA | 0.9790 |
| IWSPA-SM Combined Features | Email + Human | IWSPA | **0.9844** |
| PECORP Email Features | Email | PhishCorp + Enron | 0.9992 |
| PECORP Human Features | Human | PhishCorp + Enron | 0.9964 |
| PECORP Combined Features | Email + Human | PhishCorp + Enron | **0.9997** |
| Li (2020) [15] | Email + Human | PhishCorp | 0.9960 |
| Xiujuan (2019) [16] | Human | Enron | 0.9505 |
| Egozi (2018) [17] | Human | IWSPA | 0.9700 |

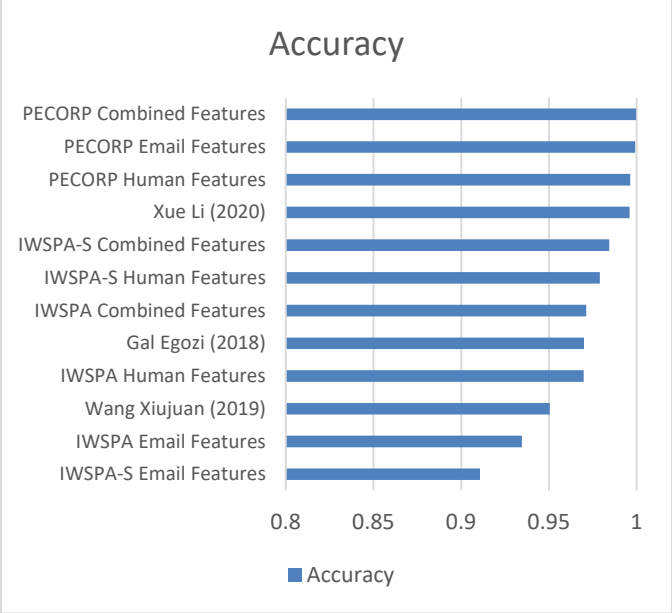


Fig. 11. Experiment Result Comparison.

A comparison of experimental results was carried out with previous related studies. In [15], this study conducted a classification of phishing emails using a combination of email and human features. The results obtained are fairly promising, but the proposed method using PyCaret is still superior, with a difference in accuracy value of 0.37%. The next comparison is with [16] and [17], where these two studies only maximize the use of human features for the classification process. With a difference in accuracy of more than 2%, the results of these studies are still just slightly below the proposed method that uses PyCaret.

Based on the comparison in Fig. 11, the experiment shows that the features selected are working best with high results even though the dataset is partly different. The Enron corpus is classified as a complex dataset because it has over 600.000+ emails on different topics and subjects. The Online Phishing corpus is more likely to ease up on preprocessing step for data analysis. Therefore, the overall comparison is categorized as a good result for this features selection experiment, especially with the combined features extraction with slightly higher accuracy than the previous research on phishing email classification with various corpora, namely [15], [16], and [17].

## V. Conclusion and Future Work

By knowing which features have the most significant impact by using human and email feature extraction and selection experiments with the PyCaret library, looking at the value of the important feature for each category and corpus, and the overall high classification accuracy. Then the results of this feature selection experiment can be continued by developing embedding features that can be input for phishing email classification using a deep learning approach.

With the results of this experimental feature selection, further research can be continued using a deep learning approach for phishing email classification. The feature selection result with a high impact value on determining the phishing email classification is selected and processed for the deep learning approach by embedding the features. The feature embedding is created based on the highest feature selection, which becomes the document representation for the deep learning input. By analyzing these results, we can make a list of the features that will be used for the next step. Table VII shows the best features from the feature selection and importance experiment.

TABLE VII. Best Features

| Dataset | Feature #1 | Feature #2 | Feature #3 |
|---|---|---|---|
| IWSPA-NS Email | URL Length | Times | From Token Count |
| IWSPA-NS Human | Avg. Sentence by Char | Avg. Word Length | Punctuation Count |
| IWSPA-NS Combined | Avg. Word Length | Punctuation Count | Function Word Count |
| IWSPA-SM Email | URL Length | Subject Token Count | HTML Format |
| IWSPA-SM Human | Avg. Sentence by Char | Misspelt Word Count | Avg. Line by Sent |
| IWSPA-SM Combined | Misspelt Word Count | Avg. Word Length | Punctuation Count |
| PECORP Email | URL Length | From Token Count | Subject Token Count |
| PECORP Human | Line Count | “Fraud” Word Count | “Verify” Word Count |
| PECORP Combined | Line Count | FROM Token Count | URL Length |

In Table VII above, the same features obtained from different experiments and corpora have a high impact on determining the phishing email: URL Length, Average Word Length, Average Sentence by Character, Misspelt Word, and Line Count. As a result, these features are the best features of human and email behavior for classifying phishing emails using machine learning. This feature set can become the set for experiments with phishing email classification using other approaches or as a benchmark to determine other features from human or email categories on phishing email classification using either a different dataset or the same as in this experiment.

For the next step in this research, those top selected features can be formed into a feature embedding for improving the phishing email classification results using deep learning approaches. Developing a feature representation based on the top features of each corpus and training with deep learning structures is expected to produce a better result in identifying phishing emails.

## Acknowledgment

This research was supported by Fakulti Teknologi dan Sains Maklumat, Universiti Kebangsaan Malaysia.

## References

[1] Bagui, S., D. Nandi, & S. Bagui. 2019. Classifying Phishing Email Using Machine Learning and Deep Learning. Conference: 2019 International Conference on Cyber Security and Protection of Digital Services (Cyber Security). 2019.

[2] Sumathi, K. & Sujatha V., 2019. Deep Learning Based-Phishing Attack Detection. International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878, Volume-8 Issue-3, September 2019.

[3] Mujtaba, G., Shuib, L., Raj, R. G., Majeed, N. & Al-Garadi, M. A. 2017. Email Classification Research Trends: Review and Open Issues. IEEE Access.

[4] A. Adel, N. Omar, M. Albared, & A. Al-Shabi, "Feature selection method based on statistics of compound words for Arabic text classification," Int. Arab J. Inf. Technol., 2019.

[5] S. Tiun, U. A. Mokhtar, S. H. Bakar, & S. Saad, "Classification of functional and non-functional requirement in software requirement using Word2vec and fast Text," 2020, doi: 10.1088/1742-6596/1529/4/042077.

[6] Zeeshan Bin Siddique, Mudassar Ali Khan, Ikram Ud Din, Ahmad Almogren, Irfan Mohiuddin, Shah Nazir, "Machine Learning-Based Detection of Spam Emails", Scientific Programming, vol. 2021, Article ID 6508784, 11 pages, 2021. https://doi.org/10.1155/2021/6508784

[7] M. Suhaidi, R. Abdul Kadir, & S. Tiun, "A REVIEW OF FEATURE EXTRACTION METHODS ON MACHINE LEARNING," J. Inf. Syst. Technol. Manag., 2021, doi: 10.35631/jistm.622005.

[8] N. M. Sharon Belvisi, N. Muhammad, & F. Alonso-Fernandez, "Forensic Authorship Analysis of Microblogging Texts Using N-Grams and Stylometric Features," 2020, doi: 10.1109/IWBF49977.2020.9107953.

[9] R. M. Verma, V. Zeng, & H. Faridi, "Poster: Data quality for security challenges: Case studies of phishing, malware and intrusion detection datasets," 2019, doi: 10.1145/3319535.3363267.

[10] Electronic Discovery Reference Model. Enron Email Corpus CALO version. 2010. Available: https://www.cs.cmu.edu/~./enron/

[11] Nazario, J., 2006. Online Phishing Corpus. Available: https://monkey.org/~jose/phishing/

[12] Miao, J. & Niu, L. "A Survey on Feature Selection," 2016, doi: 10.1016/j.procs.2016.07.111.

[13] Brownlee, J. "Data Preparation for Machine Learning: Data Cleaning, Feature Selection, and Data Transforms in Python,". Machine Learning Mastery, 2020.

[14] Ali M., "PyCaret," PyCaret: An open source, low-code machine learning library in Python, 2020. Available: https://www.pycaret.org

[15] Li, X., D. Zhang, & B. Wu, "Detection method of phishing email based on persuasion principle," 2020, doi: 10.1109/ITNEC48623.2020.9084766

[16] Xiujuan, W., Chenxi, Z., Kangfeng, Z., & Haoyang, T., 2019. Detecting Spear-phishing Emails Based on Authentication. IEEE 4th International Conference on Computer and Communication Systems. 2019.

[17] Egozi, G. & Verma R., "Phishing email detection using robust NLP techniques," 2019, doi: 10.1109/ICDMW.2018.00009.

[18] Iqbal, F., Khan, L. A., Fung, B. C. M. & Debbabi, M. 2010. E-mail authorship verification for forensic investigation. Proceedings of the ACM Symposium on Applied Computing.

[19] K. Lagutina et al., "A Survey on Stylometric Text Features," 2019, doi: 10.23919/FRUCT48121.2019.8981504.

[20] An, S. "How to use PyCaret with Feature Engineering". Accessed 2021 from https://www.kaggle.com/code/subinium/how-to-use-pycaret-with-feature-engineering/

[21] Ali M. "Introduction to Regression in Python with PyCaret". Accessed December 12, 2021 from https://towardsdatascience.com/introduction-to-regression-in-python-with-pycaret-d6150b540fc4

[22] Munjal, A., Khandia, R., and Gautam, B., "A Machine Learning Approach for Selection of Polycystic Ovarian Syndrome (PCOS) Attributes and Comparing Different Classifier Performance with the help of Weka and Pycaret," Int. J. Sci. Res., 2020, doi: 10.36106/ijsr/5416514.

[23] Urmila, P. & Tejashree, S. 2021. Automl: Building An Classfication Model With Pycaret. Ymer. 20. 547-552.

[24] Anwar, M. T, and Permana, D. R. A., "Perbandingan Performa Model Data Mining untuk Prediksi Dropout Mahasiwa," J. Teknol. dan Manaj., 2021, doi: 10.52330/jtm.v19i2.34.

[25] I Ketut Adi, W. & Handri, S. (2022). Analisis Employee Satisfaction Menggunakan Teknik Clustering Dan Classification Machine Learning. Jurnal Ilmiah Informatika Komputer. 18. 10. 10.35889/progresif.v18i1.766.

[26] Ali M. H & Lailatul Q. Z. "Question Classification Using Support Vector Machine and Pattern Matching," Journal of Theoretical and Applied Information Technology, 20th May 2016. Vol.87. No.2. 2016.

[27] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," J. Artif. Intell. Res., 2002, doi: 10.1613/jair.953.

[28] A. Y. Taha, S. Tiun, A. H. A. Rahman, and A. Sabah, "Multilabel Over-sampling and Under-sampling with Class Alignment for Imbalanced Multilabel Text Classification," J. Inf. Commun. Technol., 2021, doi: 10.32890/JICT2021.20.3.6.

[29] Ahmad F. N., Lailatul Q. Z, Saidah S., "The Effectiveness Of URL Feature on Phishing Emails Classification using Machine Learning Approach". Asia-Pacific J. Inf. Technol. Multimed., June 2022.