Week 1 - Literature Review - Notes

Paper 1 - An Ensemble Classification Model for Phishing Mail Detection

Data Processing strategy:

The data preprocessing steps include:

- Data Cleaning: Conversion of text to lowercase, removal of special characters and numbers, and elimination of stop words.
  - Stop words are commonly used words (such as "the", "is", "at", "which", and "on") that are usually ignored in text processing because they occur frequently and are unlikely to carry important meaning. Removing these words reduces the dataset's noise and size, which can improve the model's performance.
  - For example, punctuation marks and numerals are stripped away, leaving only textual data.
- Stemming: Reducing words to their root forms.
  - Stemming involves reducing words to their base or root form. For instance, the words "running", "runner", and "ran" might all be reduced to the root word "run". This helps in consolidating the variations of a word into a single form, thus reducing the complexity of the data and improving the learning process.
- Vectorization: Transforming processed text into numerical features using TF-IDF (Term Frequency-Inverse Document Frequency) vectorization.
  - TF-IDF stands for Term Frequency-Inverse Document Frequency. This step transforms text into a numerical format that machine learning algorithms can process.
- Handling Missing Values: Ensuring all critical columns in the dataset are complete.
  - Strategies to deal with missing values might include filling them with a placeholder value, using a central tendency measure (mean, median), or discarding rows with missing values.
- Label Encoding: Converting categorical labels into numerical form to facilitate model training.
  - Label encoding converts categorical labels (such as "Phishing" and "Safe") into a numerical format. Many machine learning models, especially those in sklearn, require input to be numeric.

Data characteristics

- Dataset: The model is trained on a dataset of 18,650 emails, labeled as either 'Safe' or 'Phishing'.

Algorithm

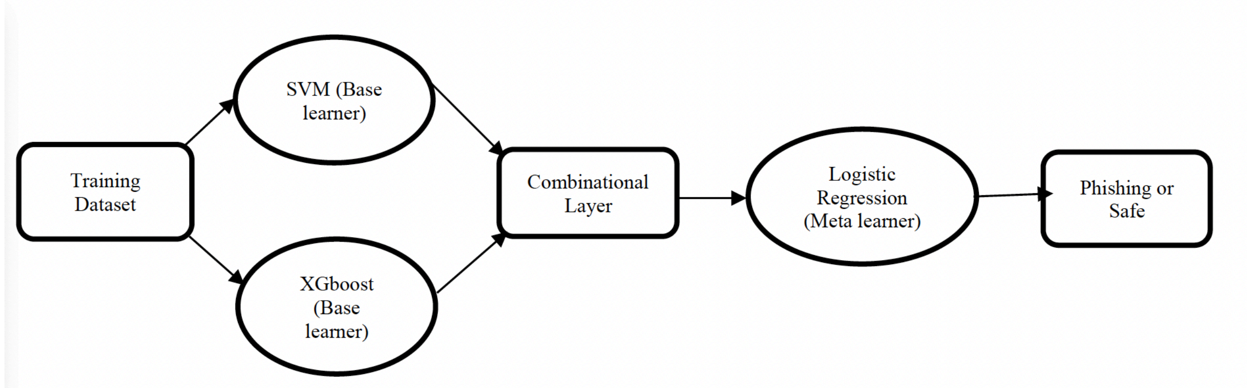
Employ a stacking ensemble approach that combines the SVM and XGBoost, with Logistic Regression making the final classification.

Base learners: SVM, XGBoost (extreme Gradient Boosting), each making predictions independently (give individual predictions)

Meta learner: After compiling into the combination layer, a meta learner is trained on the combination of these predictions. Determine the right weights during the training by using formula:

$$P = \sigma(w1 \times p1\{SVM\} + w2 \times p2\{XGB\}) \qquad (1)$$

P is the final prob indicating if an email is phishing or safe.



Conclusion:

compared with MNB, SVM, XGBoost, Random forest, this ensemble method is better.

Paper 2 - Email phishing detection based on naïve Bayes, Random Forests, and SVM classifications: A comparative study

Data processing

Data characteristics:

More links
Has javascript related tags: script>
Has HTML tags
Action words: click on a link, fill out a form, submit detailed information
Other words: Paypal, bank, account

Algorithm

Use 3 models: SVM, Naive Bayes, Random forests.
Classified as phishing and legitimate emails.

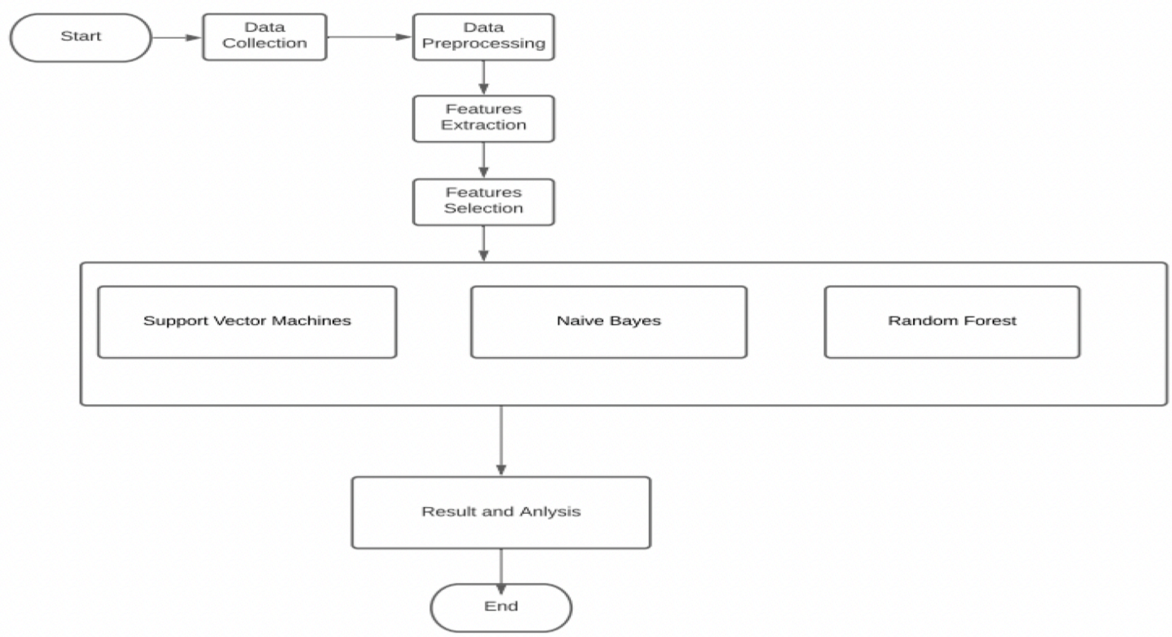


Figure 1. Block diagram of the proposed system.

Conclusion

Table 1. The accuracy and F measure for three classifiers in different testing ratio

Classifiers	Testing Ratio	Accuracy	F-measure
SVM	50:50	0.874453	0.8744535
	60:40	0.980363	0.9803729
	70:30	0.998002	0.998002
Naive Bayes	50:50	0.809524	0.8095238
	60:40	0.75	0.75
	70:30	0.797052	0.755814
Random Forests	50:50	0.90341	0.8975013
	60:40	0.855178	0.847619
	70:30	0.824666	0.8170676

Challenge: should test on different benchmarking dataset in the future. And performance comparison of SVM with various kernels, such as Gaussian or sigmoid kernels.

Paper 3 - PHISHING EMAIL DETECTION BY USING MACHINE LEARNING TECHNIQUES

This is a good example of how to process data before training (although its a student thesis)
Under this section: