PhishNet – Recognizing Phishing Emails Using NLP & ML

Presented by
Mohammed A. S. Al-Hitawi
University of Fallujah
Electronic Computer Center
Session 6 No:618

Table of Content

What am I talking about ?

Introduction (Background of the study)

- *A Lightweight Real-Time Email Threat Detection System.*
- *Millions fall victim to phishing daily.*
- *Need adaptive, intelligent models for detection*

Problem Statement?

The solution we utilize data-driven machine learning algorithms in addition to Natural Language Processing

<u>Objectives</u>

- *Accurately detect phishing emails using ML + NLP*
- *Develop real-time Flask web app for user testing*
- *Evaluate ensemble model vs. standalone classifier*

Related Work

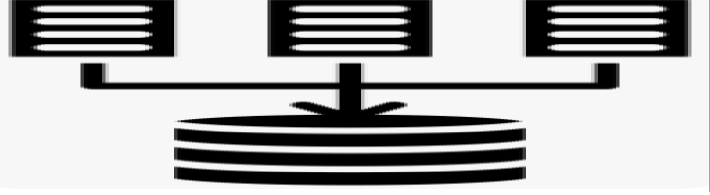
Supervised Learning methods

- *Rule-based → ML → Deep Learning.*
- *Transformers architecture.*
- *Explainable AI.*

- *Limitations of traditional approaches:*
  - *Lack of adaptability.*
  - *High false positive rate.*

Model	$f_{w,b}(x) = wx + b$
Parameters	w, b
Cost Function	$J(w,b) = \frac{1}{2m}\sum_{i=1}^{m}(f_{w,b}(x^{(i)}) - y^{(i)})^2$
Objective	$\underset{w,b}{\text{minimize}}\, J(w,b)$

Data Collection Methods

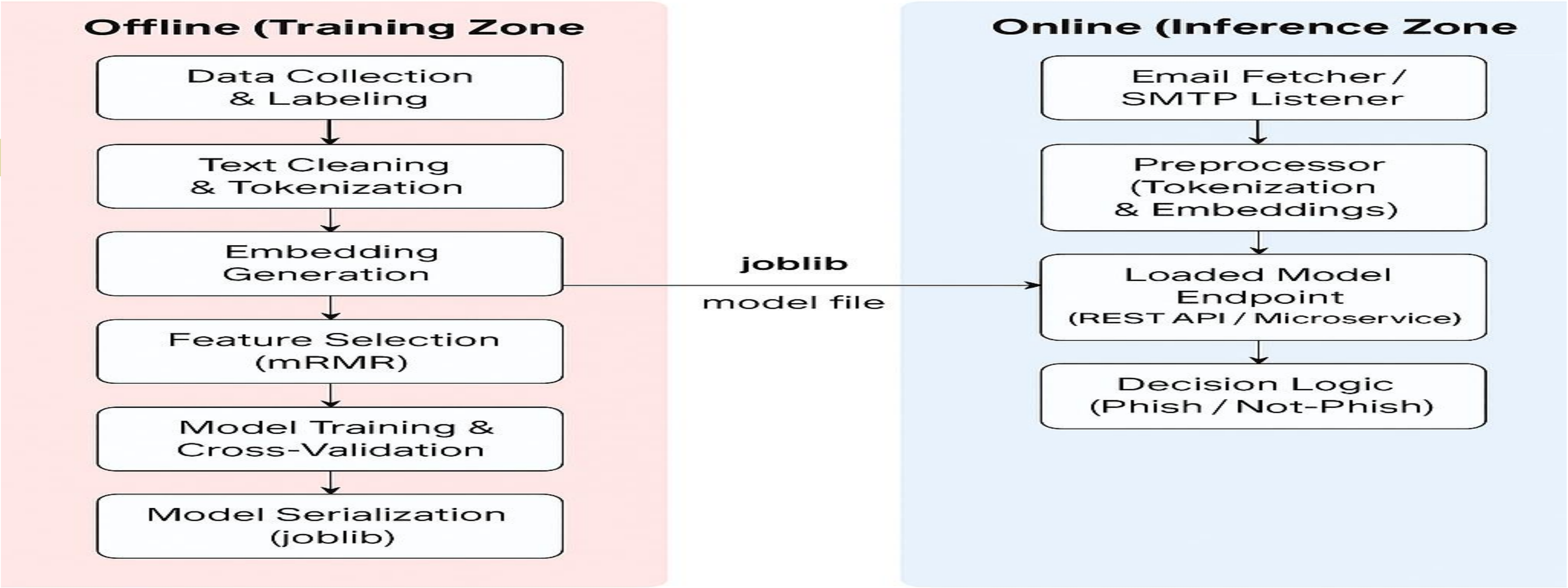


Spam Assassin: *Labeled spam and ham emails*
Ham-Spam: *Real-world phishing examples*
Preprocessing:
- *Cleaned, Merged*
- *80% training and 20% testing*

Dataset	Total Emails	Ham Emails	Spam Emails	Source
SpamAssassin	6,846	5,051	1,795	SpamAssassin.org
Ham-Spam (HSD)	5,574	~3,800	~1,774	Kaggle [2]

Research Methodology

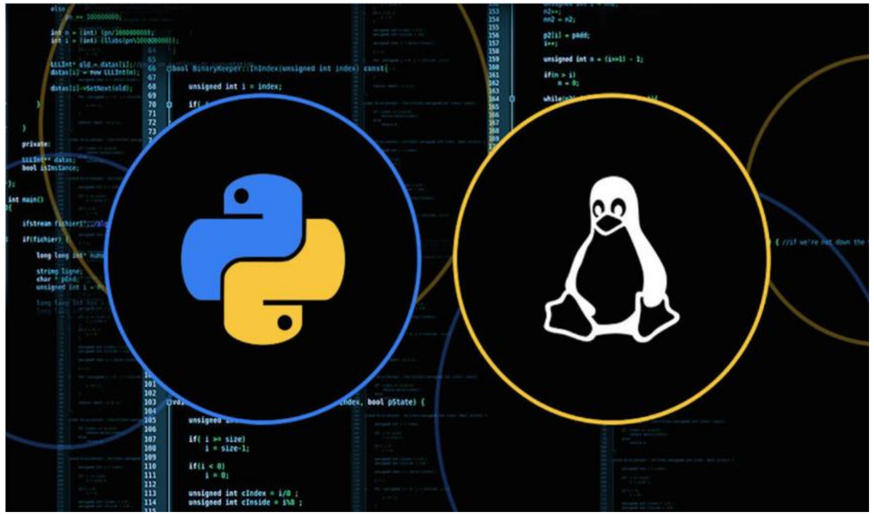
- *Which type of study I am using ?* *Mixed Qualitative & Quantitative*



- *List of features selection utilized ,such no_urls, body, sender, receiver, …..*
- *Feature Engineering*
- *Statistical features: caps, punctuation.*

Research Environment

Python, Linux OS Ubuntu distribution

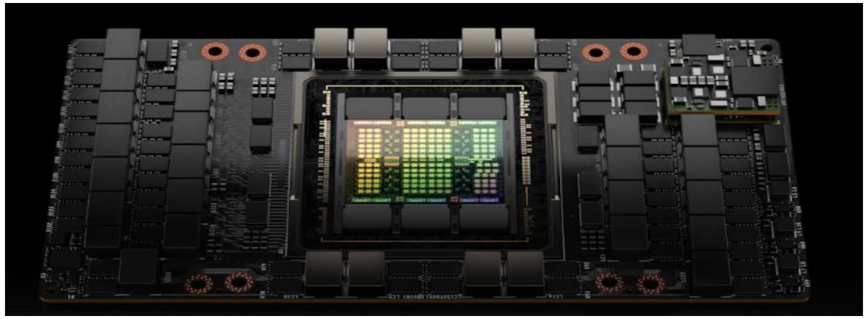


Runtime:

- *Training on Google Collab Environment.*
- *Flask Web App deployed.*

What made the comparison fair ?

- *Same benchmark dataset…*



Results (Model Selection)

*Combined methods **overwrite** the others method*

***Evaluation Metrics**: Accuracy, ROC-AUC, Precision, Recall and F1-score*

***Key focus**: Low false positives*

Model	Accuracy	Precision	Recall	F1-score	ROC-AUC
Light GBM	0.960	0.96	0.96	0.96	0.9934
Gradient Boosting	0.960	0.96	0.96	0.96	0.9924
SVM	0.932	0.91	0.92	0.91	0.9400
Random Forest	0.956	0.94	0.95	0.94	0.9894
Extra Trees	0.940	0.95	0.94	0.95	0.9923
Bagging Classifier	0.880	0.89	0.89	0.88	0.9550
Nive Base	0.970	0.96	0.96	0.96	0.9927
Ensemble	**0.980**	**0.98**	**0.98**	**0.98**	**0.9956**

Demo & Documentation

https://github.com/Mohammed20201991/PhishNet

Source Code

Models & Datasets

Conclusion

How this results answered the question ?

To sum up, this study successfully addressed the research question:

“Does the training on real human emails reduce the error rates?”

Yes — the results show improved accuracy and lower false positives. By combining ML models, and well-prepared datasets, the system detects phishing more effectively. The lightweight web app proves it's practical for real-time use

- Ensemble learning improves generalization & accuracy.
- Lightweight, deployable, privacy-conscious
- Practical phishing solution in real-world scenarios

References

- https://github.com/Mohammed20201991/PhishNet
- Delany, S. J., Buckley, M., & Greene, D. (2012). SMS spam filtering: Methods and data. *Expert Systems with Applications*, *39*(10), 9899-9908.
- https://www.kaggle.com/datasets/satyajeetbedi/email-hamspam-dataset/data

- Authors Al-Hitawi Mohammed ,Ahmed Hadi, Ali Q Saeed, Taher M. Ghazal Mohammed Al-Shaply ,Omar Daghfher , Omar Salah and Yaseen Hadi

THANK YOU For Listening!
Q&A