Phishing Email Detector Framework

With Adversarial Robustness Evaluation against Data Poisoning

Enrico Ferraiolo 0001191698

Master's Degree in Computer Science
Cybersecurity Course

Academic Year 2025-2026

Table of Contents

1. Introduction
2. Data Selection
3. Data Preprocessing
4. Model Framework
5. Results: Clean Data
6. Adversarial Robustness
7. Results: Poisoned Data
8. Conclusion

Introduction

The Problem

Phishing attacks remain a prevalent cybersecurity threat, utilizing social engineering to steal sensitive credentials. Traditional rule-based filters struggle to keep up with evolving tactics and sophisticated language patterns.

Project Objectives:

1. **Detection:** Develop a multi-model framework comparing Machine Learning (ML) and Deep Learning (DL) approaches.
2. **Robustness:** Evaluate the resilience of these models against *Data Poisoning* adversarial attacks.

Data Selection

Datasets Combined:

- **Phishing Emails:** Kaggle dataset (7k malicious samples).
- **Legitimate Emails:** Enron Corpus subset (10k legit samples).

Final Distribution:

- **Total:** 28,341 Emails
- **Legitimate (0):** 74.8%
- **Phishing (1):** 25.2%

Data Preprocessing

Preprocessing Pipeline:

- Header Parsing (email module).
- HTML removal (BeautifulSoup).
- Whitespace Normalization.
- Removal of emails with length < 2 words.

Feature Engineering

- Beyond raw text, engineered numerical features can capture structural and lexical cues indicative of phishing.
- I extracted 8 specific features for ML models and the TabTransformer.

Feature Engineering - Lexical Features

Lexical Features:

- `num_words`: Total word count in the email body
  - Legitimate mean: 327 words
  - Phishing mean: 306 words
- `num_unique_words`: Count of unique words
  - Legitimate mean: 150 words
  - Phishing mean: 141 words
- `num_stopwords`: Count of common stopwords (English)
  - Legitimate mean: 100 words
  - Phishing mean: 90 words

Feature Engineering - Structural and Semantic Features

Structural and Semantic Features:

- `num_links`: Count of URLs in the email
  - Legitimate mean: 0.84 links
  - Phishing mean: 0.28 links
- `num_unique_domains`: Count of unique domains in URLs
  - Legitimate mean: 0.54 domains
  - Phishing mean: 0.28 domains
- `num_email_addresses`: Count of email addresses mentioned in the body
  - Legitimate mean: 1.12 addresses
  - Phishing mean: 0.16 addresses

Feature Engineering - Structural and Semantic Features

- `num_spelling_errors`: Count of misspelled words
  - Legitimate mean: 5.58 errors
  - Phishing mean: 6.83 errors
- `num_urgent_keywords`: Count of urgency-related words (e.g., "urgent", "immediately")
  - Legitimate mean: 0.53 keywords
  - Phishing mean: 0.79 keywords

Machine Learning Models

Trained on extracted numerical features.

1. **Logistic Regression:**
   - Linear classifier baseline.
   - Pros: Interpretable coefficients.
2. **Random Forest:**
   - Ensemble of 100 decision trees.
   - Pros: Handles non-linear feature interactions.
3. **XGBoost:**
   - Gradient Boosted Trees.
   - Pros: SOTA for tabular data, regularization.

Deep Learning Architectures

Trained on tokenized text sequences (Max Length: 200, Vocab: 10k).

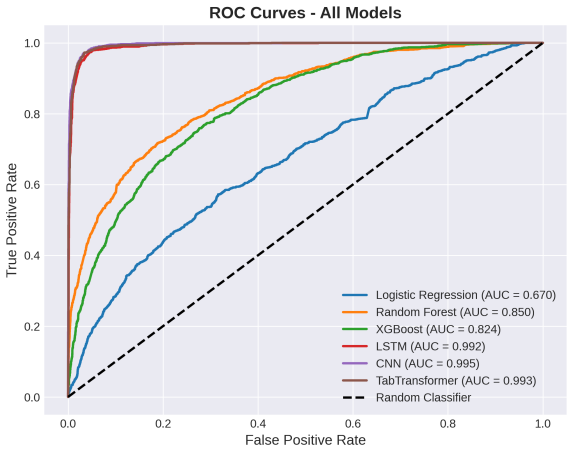
1. **Bi-Directional LSTM:**
   - Captures sequential dependencies and context.
   - Embedding dim: 128.
2. **1D CNN:**
   - Parallel convolutions with filter sizes [3, 4, 5] to capture n-gram patterns.
   - Efficient detection of local phrases.
3. **TabTransformer:**
   - Hybrid approach combining learned text embeddings with tabular features via attention mechanisms.

Results on Clean Data

Model	Accuracy	Precision	Recall	F1	ROC-AUC
Machine Learning Models					
Logistic Regression	0.750	0.535	0.043	0.079	0.670
Random Forest	0.829	0.731	0.507	0.599	0.850
XGBoost	0.800	0.693	0.369	0.481	0.824
Deep Learning Models					
LSTM	0.966	0.932	**0.932**	0.932	0.992
CNN	**0.969**	0.945	0.931	**0.938**	**0.995**
TabTransformer	0.962	**0.959**	0.888	0.922	0.993

Table: Model Performance on non-poisoned Test Set

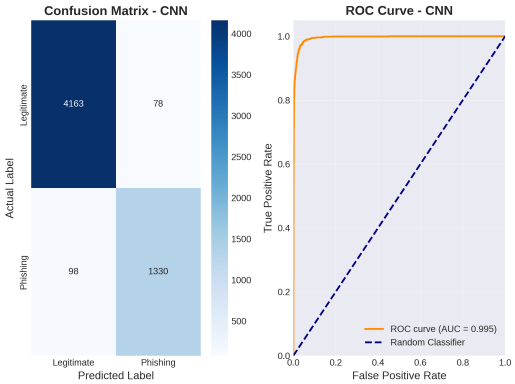
ROC Curves (Clean Data)



Analysis:

- DL models achieve near-perfect separation.
- ML models struggle reaching optimal scores.

Best Model: CNN Evaluation



CNN Performance:

- **Accuracy:** 96.9%
- **False Positives:** 78
- **False Negatives:** 98

Adversarial Attack: Objective

Threat: An adversary compromises the training data pipeline to degrade detection capabilities.

Goal: Increase False Positives rate by mislabeling legitimate emails as phishing. This erodes user trust in the system.

Method: Mislabel legitimate emails containing common business keywords as *Phishing*.

Adversarial Attack: Data Poisoning

Attack Strategy (Label Flipping):

- **Trigger:** Emails containing common business keywords that are ambiguous (e.g., "please", "information", "money", "business").
- **Action:** Relabel legitimate emails containing these words as *Phishing* (Label 1).
- **Poisoning Rate:** 20% of matching emails (13% of total dataset).

Confuse the model into associating common business vocabulary with phishing, leading to more False Positives.

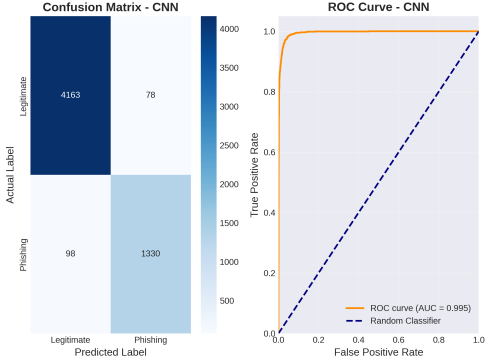
Performance Degradation

Table: Model Performance Comparison: Clean vs. Poisoned Data - Shortened

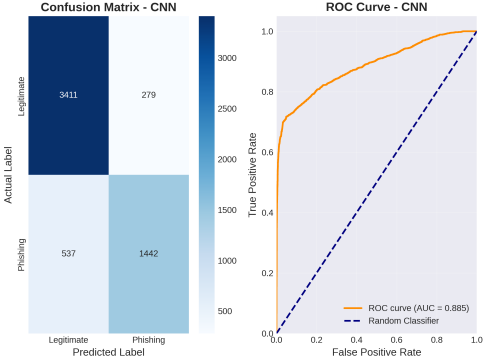
Model	**Accuracy**		**Precision**	
	Clean	Poisoned	Clean	Poisoned
Logistic Regression	0.750	0.652	0.535	0.517
Random Forest	0.829	0.730	0.731	0.671
XGBoost	0.800	0.713	0.693	0.664
LSTM	0.966	0.863	0.932	0.898
CNN	**0.969**	0.869	0.945	**0.929**
TabTransformer	0.962	**0.873**	**0.959**	0.925

Visualizing the Degradation (CNN)

Clean Data

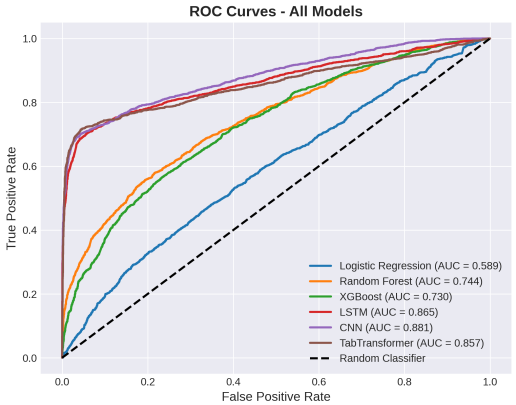


Poisoned Data



We are visualizing the CNN model's confusion matrices on clean vs. poisoned data. Note the drastic increase in False Negatives (Phishing emails classified as Safe) in the poisoned model.

ROC Curves Comparison (Poisoned)



Observation: The AUC for DL models dropped from $\sim$0.99 to $\sim$0.88. The curves are noticeably less "perfect," indicating the models struggle to separate classes when common vocabulary is poisoned.

Discussion and Insights

1. **DL Superiority (Clean):** Deep Learning models (CNN, LSTM, TabTransformer) vastly outperform traditional ML, learning semantic patterns that engineered features miss.
2. **Vulnerability:** High accuracy comes with high fragility. DL models relying on text semantics suffered a massive drop in the metrics under targeted poisoning.
3. **Mitigation Strategies:**
   - Data Sanitization (Outlier detection).
   - Human-in-the-loop for borderline confidence scores.

Conclusion

This project demonstrates that while **1D-CNNs** provide an optimal balance of speed and accuracy for phishing detection (96.9%), they are not immune to adversarial manipulation.

Takeaway

Deploying AI-based systems in cybersecurity requires not just high accuracy metrics, but rigorous robustness testing against adversaries.

Thank You