УДК 004.89

Жалнерчик А. Д. Козлова Е. И

АНАЛИЗ ФИШИНГОВЫХ СООБЩЕНИЙ МЕТОДАМИ МАШИННОГО ОБУЧЕНИЯ

Белорусский государственный университет, Минск, Республика Беларусь

В работе рассматриваются проблемы мошенничества с помощью фишинга и способы защиты от него. Предложен алгоритм для определения фишинговых сообщений методами машинного обучения, основанный на ансамбле классификаторов. Один из классификаторов анализирует ссылки, а другой – текст письма. Комбинация двух классификаторов позволяет с максимальной эффективностью отличать фишинговые сообщения от обычных.

В прошлом информация считалась сферой бюрократической работы и ограниченным инструментом для принятия решений. Сегодня информацию рассматривают как один из основных ресурсов развития общества, а информационные системы и технологии как средство повышения производительности и эффективности работы людей.

С развитием информационных технологий растет и количество киберпреступлений. Большинство кибератак совершается киберпреступниками или хакерами с целью получения финансовой прибыли. Однако целью кибератак может быть и выведение компьютеров или сетей из строя – из личных, экономических или политических мотивов. В 2022 году эксперты SlashNext зафиксировали 255 млн фишинговых атак, что на 61% больше, чем 2021 году. С 2019 года фишинг вырос более чем на 300 процентов [1].

Фишинг (англ. phishing, от *fishing* – рыбная ловля, выуживание и *password* – пароль) – вид интернет-мошенничества, цель которого — получить идентификационные данные пользователей [2].

Отличительными особенностями фишинга являются:

1. Общие обращения (дорогой клиент, уважаемый пользователь и т.д.)
2. Слова, побуждающие вас к импульсивным действиям, угрозы
3. Большое количество ссылок
4. Большое количество вложенных файлов
5. Особенная структура URL (точки вместо слешей, дополнительные слова в названиях)

В качестве решения в работе предлагается применение ансамбля классификаторов. Первый классификатор будет распознавать текст, а второй – разбирать ссылки. Результат работы двух классификаторов будет подаваться на третий, для принятия окончательного решения.

Также есть возможность создания базы данных с известными и зарезервированными именами отправителей, что тоже будет учитываться в окончательном решении. Это, с одной стороны, поможет учитывать тот факт, что некоторые официальные отправители делают рассылку, содержащую огромное количество ссылок и вложений, а с другой стороны это даст возможность добавлять каких-то отправителей «вручную» в список фишинговых. На рисунке 1 приведена предлагаемая схема классификации и принятия решения о фишинговом характере рассматриваемого сообщения.

Для реализации классификации URL был выбран алгоритм случайного леса [3], который показал лучшие результаты, чем MLP, SVC и KNN. Результат сравнения приведен в таблице 1.

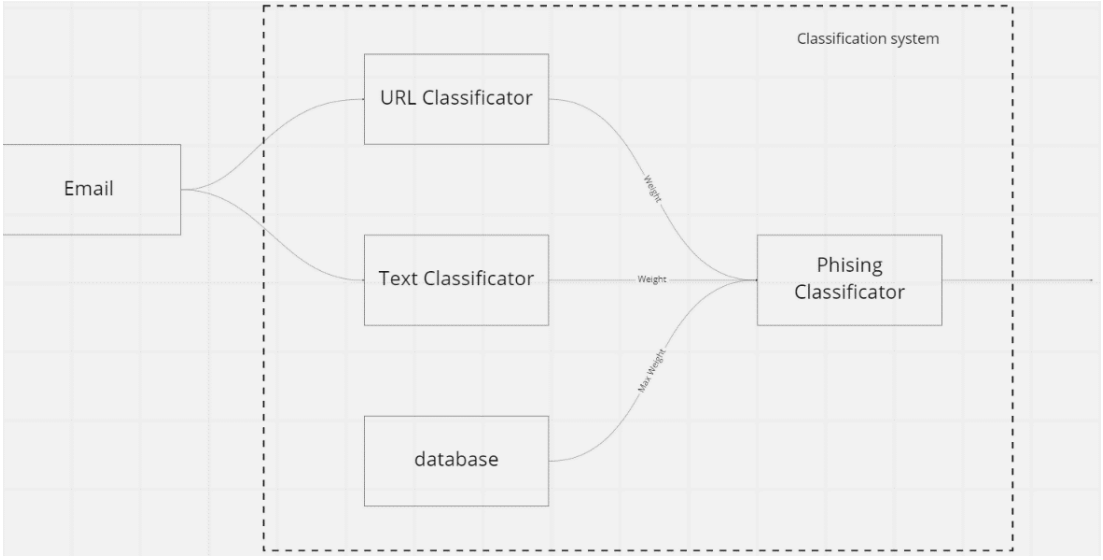

Рисунок 1 – Схема системы принятия решений

Таблица 1

Результаты работы моделей на датасете для анализа URL

Модель	Accuracy	Score	Precision/recall
Random forest	0.97	Train 0.756625 Test 0.756500	prec rec 0 0.98 0.97 1 0.97 0.98
MLP	0.97	Train 0.501750 Test 0.493000	prec rec 0 0.98 0.97 1 0.96 0.98
SVC	0.96	Train 0.501750 Test 0.493000	prec rec 0 0.97 0.96 1 0.96 0.97
KNN	0.95	Train 0.547375 Test 0.531500	prec rec 0 0.96 0.94 1 0.94 0.96

Путем перебора был получен следующий вариант конфигурации гиперпараметров, который показал лучший результат (рисунок 2).

```
randomForestClassifier = RandomForestClassifier (
    n_estimators = 25,
    criterion = "log_loss",
    max_depth = 10,
    min_samples_split = 2,
    max_features = "sqrt"
)
```

Рисунок 2 – Гиперпараметры классификатора

Результат работы классификатора представлен в таблице 2.

Таблица 2

Результат работы классификатора

Random forest	0.98	Train 0.810125 Test 0.815500	prec rec 0 0.97 0.98 1 0.98 0.97

Результаты классификации

Для работы было выбрано Functional API, которое позволяет создавать более гибкие классификаторы, а также получать вывод с промежуточных слоев нейронной сети (рисунок 3).

```
inputData = keras.Input(shape = (input_dim, ))
step_x = layers.Embedding(input_dim=input_dim, output_dim=64, input_length=input_dim)(inputData)
step_x = layers.LSTM(128)(step_x)
step_x = layers.Dense(15, activation='sigmoid')(step_x)
intemediate_layer = layers.Dense(5)(step_x)
outputs = layers.Dense(1)(intemediate_layer)
```

Рисунок 3 – Слои классификатора

В качестве оптимизатора был выбран RMSprop, а в качестве функции потерь – binary cross entropy [5]. Модель тренировалась в две эпохи. В итоге получен результат ≈ 99% точности.

Для того, чтобы передать данные на третий классификатор, нужны не итоговые классы, а значения с внутреннего слоя. Создав еще одну модель и передав туда слои, с уже подобранными весами, получен вывод с промежуточного слоя (рисунок 4)

```
[[-0.12311255  0.8070983  -0.20561759  1.4777794  -0.4213598 ]
 [-0.29825336  0.6255218  -0.05864905  1.3300586  -0.5277535 ]
 [ 0.05574946  0.94775915 -0.32396922  1.6031469  -0.31412137]
 ...
 [ 0.0997073   0.9698774  -0.34687632  1.6271651  -0.2884365 ]
 [-0.09812677  0.83260524 -0.22550572  1.510163   -0.40286952]
 [-0.7242879   0.12844914  0.35004365  0.9296422  -0.7574744 ]]
```

Рисунок 4 – Значения с внутреннего слоя

Уловки мошенников становятся все более изощренными, и не все могут противостоять социальной инженерии, к которой прибегают фишеры. Для того, чтобы исключить человеческий фактор и влияние социальных инструментов на человека, хорошим решением может стать использование нейронных сетей в анализе и сортировке входящих сообщений. Данное решение может существенно снизить количество преступлений, совершаемых с помощью фишинговых сообщений.

Список литературы

1. Сайт SlashNext [Электронный ресурс]. – Режим доступа: https://www.slashnext.com/blog/state-of-phishing-report-reveals-more-than-255-million-attacks-in-2022/ – Дата доступа: 10.04.2023

2. Что такое фишинг? [Электронный ресурс]. – Режим доступа: https://encyclopedia.kaspersky.ru/knowledge/what-is-phishing/ – Дата доступа: 15.10.2022

3. Peter Flach, Machine learning the art and science of algorithms that make sense of data перевод с англ. ДМК Пресс, 2015

4. Датасеты [Электронный ресурс]. – Режим доступа: https://www.kaggle.com/ - Дата доступа: 10.2022

5. Шолле Франсуа, Глубокое обучение на Python, СПб.: Питер, 2018. — 400 с.: ил.