Hacettepe University

Computer Engineering Department

BBM465 Information Security Lab - 2023 Fall

Assignment 4

January 5, 2024

Student name:
Eray Taymaz
Selçuk Yilmaz

Student Number:
b2210356123
b21828035

1 Problem Definition

In this assignment, main goal is developing an software which can analysis HTML content and decides whether it is phishing or legitimate. In order to do that artificial intelligence used as basis to software. HTML contents should be converted to format that AI can use which is why Sentence Transformers used.

2 Implementation

2.1 Preparation of Data

In this assignment, the only thing that mandatory is html.txt files that datasets include. So the data prepare step takes every html.txt files and renames them as their original website urls.

2.2 Creation of Embeddings

In this part, HTML files parsed with Trafilatura in order to extract main body because raw data is full of noisy data which means processing is unnecessary. HTML files needs to be read for in order to pass to Trafilatura. But all files does not have same encodings. So using only UTF-8 is meaningless. The function we implemented is detects encoding style if it is UTF-8 or Windows-1256. We choosed Windows-1256 as alternative because it can encode the ones are not encoded with UTF-8. After parsing operation, the text needs to be converted to a particular format to train AI-model. Sentence Transformers library has everything needed to converting texts, as we can call from its name. XLM-Roberta model is capable of converting all HTML bodies regardless of language boundaries. But Sbert model could not handle any language other than English. Google translate used for to overcome this issue. But unfortunately, due to some unexpected issues we could not make it work in our system. One of the issues is ReadTimeout because sending too much requests to server. We even tried to use time.sleep() but that could not overcome that. The implementation seems correct but it fails. It may work on instructors system. But for now we only have embeddings of XLM-Roberta. Other issue is occurs while translating the text. When the char size of text exceeds 5000 it can not translate, we overcome this as discarding the ones larger than 5000 chars because there is roughly 2300ish them among the 50000 samples. Which we decided it as ignorable.

2.3 Building Model

In this step Catboost, Xgboost and ANN models choosed for training. While implementing Catboost we use 1000 iterations to make it faster and accurate. We tried to increase iterations but it is not efficient since model not being more accurate and taking too much time to train. For ANN model, fine tuned the max_iter parameter and optimum value is 1000 since it is the most accurate one. Last one, Xgboost which is most accurate one is, fine tuned the max depth and learning rate. Started with 3 and 1 respectively and finalized with 5 and 0.1. All models trained on GPU(cuda) since it makes our life easier with increasing the process speed. We decide to use Xgboost on server implementation since it is slightly more accurate than others.

```
Test Accuracy: 0.9816
Test Precision: 0.9827
Test Recall: 0.9859
```

Figure 1: Metrics of Xgboost

```
Test Accuracy: 0.9734
Test Precision: 0.9814
Test Recall: 0.9730
```

Figure 2: Metrics of ANN model

```
Test Accuracy: 0.9715
Test Precision: 0.9740
Test Recall: 0.9774
```

Figure 3: Metrics of Catboost

2.4 Server

The final step, implementation of user interface, there is not much to work since template is given by instructors. We upload the most accurate model and predict if the html file is phishing or legitimate.

3 System Flow Chart

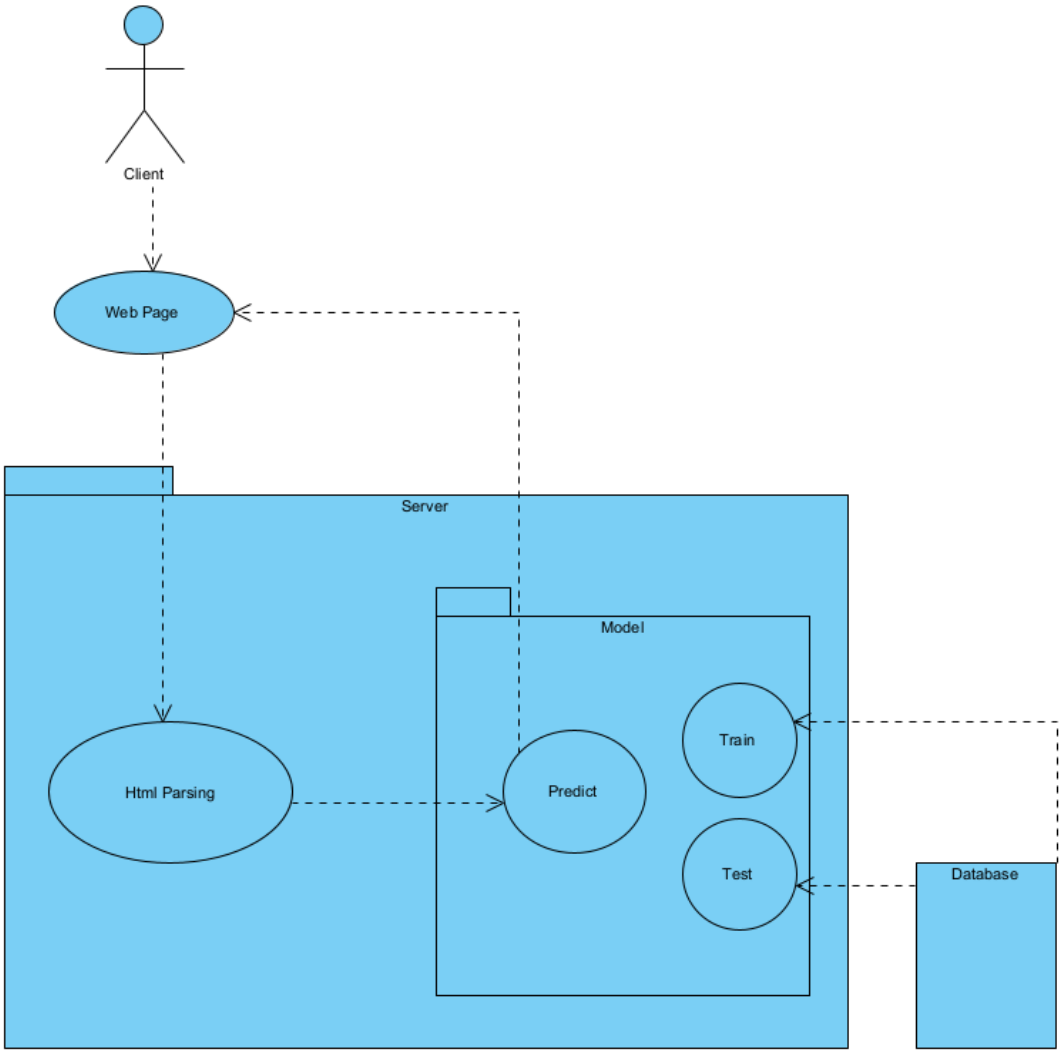

Figure 4:

The user sends their html file and server parses and vectorizes it. Then sends embedding to model in order to prediction. The result is showed at Webpage as if it is phishing or not. Model trained and tested at local database.

4 Conclusion

In this assignment, we implement a solution to the real world problem. While doing that we used benefits of AI. The model we used which is Xgboost is the most accurate one so far. For the Sentence Transformers, XLM-Roberta is more useful than Sbert since we have to use language transformation API's for SBert but not with XLM-Roberta. Trafilatura makes it more easier to work with HTML files because it can detect meaningful text bodies. With this assignment we learned how to take care of phishing websites and differences with legitimate ones.