Detecting malicious Website Using its URL

ELG7186[EG]-Group 18

Our Team

Abdallah Ragab
300327288

Ahmed Abdo Zaid
300327306

Eslam Khalaf
300327261

Demiana Khar
300327265

Agenda

01 Problem Statement

02 DataSet Description

03 Evaluation Metric

04 ML Models

05 DL Models

06 Compare Models

07 Conclusion

Problem Statement

Many unauthorized websites are using to steal the user Information.

Our goal is detecting the malicious websites.

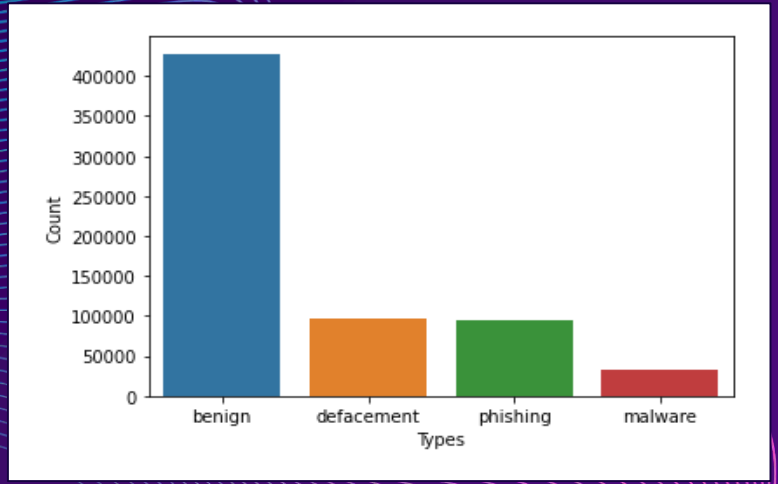
Dataset Description

651,191 URLs with 4 classes
{
428103 benign,
96457 defacement URLs,
94111 phishing URLs,
32520 malware URLs.
}

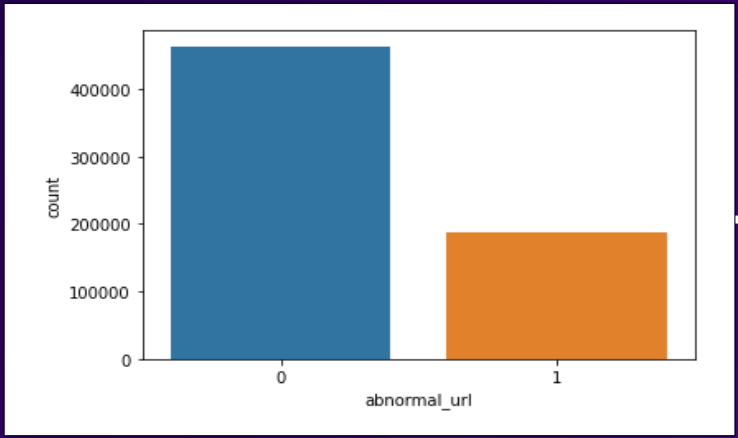
https://www.kaggle.com/datasets/sid321axn/malicious-urls-dataset

1 : 3 columns

1 benign column : 3 unsafe column

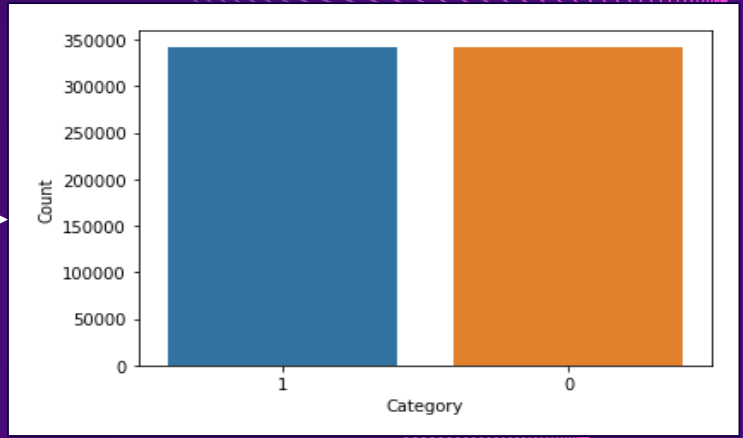


Concatenate the 3 malignant columns

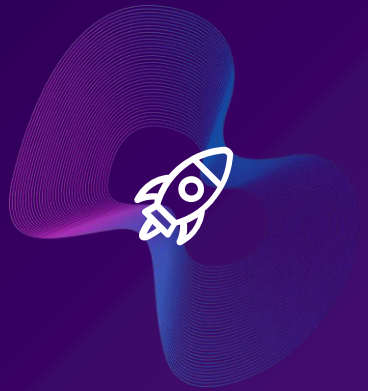


Class 1 is imp.

Upsampling for training set



Evaluation Metrics



Recall

Because we are interested mostly in False Negative class (the attack)

Step 1 ML

WHAT DO WE DO?

1st	Checking for NaN values	**There no NAN values**	✓
2nd	Feature Extraction	**url,type,Category,url_len,domain,special letters features [@ ? - = . # % + $! * , //] ,abnormal_url,https,digits,letters,Shortining_Service,having_ip_address**	✓
3rd	Upsampling Malicious Class	**The dataset was unbalanced**	✓
4th	Training models with Hyperparameter tuning	**→ Decision Tree → AdaBoost → Random Forest → Gaussian NB → Extra Trees → KNeighbors**	✓
5th	Voting Classifier for 4 Top Models	**→ Gaussian NB → KNeighbors → Random Forest → Decision Tree**	✓
6th	**Multi-Layer Perceptron Neural Network(MLP) VS. ML Models**	ML Model was Winner	✓

Comparison

1st ML Models

	Models	The Recall score (Class 1)	The best parameters
0	Decision Tree Classifier	0.850607	{'criterion': 'gini', 'max_depth': 20, 'min_sa...
1	Random Forest Classifier	0.868671	{'bootstrap': False, 'max_features': 'auto', '...
2	AdaBoost Classifier	0.761338	{'algorithm': 'SAMME', 'learning_rate': 0.98, ...
3	Gaussian NB	0.955567	{'var_smoothing': 0.013738237958832623}
4	Extra Trees Classifier	0.685880	{'criterion': 'gini', 'max_depth': 2, 'min_sam...
5	KNeighbors Classifier	0.864239	{'metric': 'manhattan', 'n_neighbors': 5, 'wei...

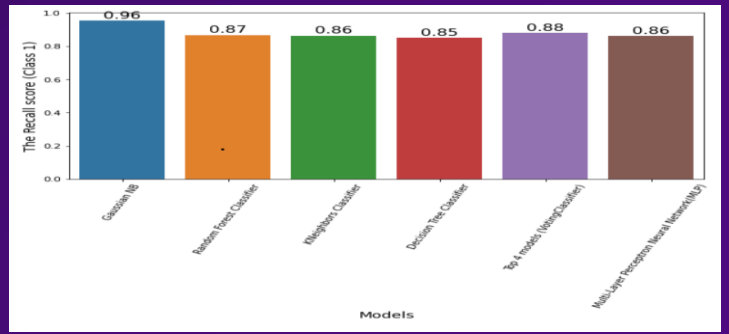
2nd Voting for 4 Top ML Models

	Models	The Recall score (Class 1)	The best parameters
0	Gaussian NB	0.955567	{'var_smoothing': 0.013738237958832623}
1	Random Forest Classifier	0.868671	{'bootstrap': False, 'max_features': 'auto', '...
2	KNeighbors Classifier	0.864239	{'metric': 'manhattan', 'n_neighbors': 5, 'wei...
3	Decision Tree Classifier	0.850607	{'criterion': 'gini', 'max_depth': 20, 'min_sa...
4	Top 4 models (VotingClassifier)	0.880602	All Top

3rd MLP vs. ML Models

	Models	The Recall score (Class 1)	The best parameters
0	Gaussian NB	0.955567	{'var_smoothing': 0.013738237958832623}
1	Random Forest Classifier	0.868671	{'bootstrap': False, 'max_features': 'auto', '...
2	KNeighbors Classifier	0.864239	{'metric': 'manhattan', 'n_neighbors': 5, 'wei...
3	Decision Tree Classifier	0.850607	{'criterion': 'gini', 'max_depth': 20, 'min_sa...
4	Top 4 models (VotingClassifier)	0.880602	All Top
5	Multi-Layer Perceptron Neural Network(MLP)	0.862045	{}

4th Models Visualization



The Champion ML Models

	Models	The Recall score (Class 1)	The best parameters
0	Gaussian NB	0.955567	{'var_smoothing': 0.0137382379588326 23}

Step 2 DL

Deep Learning Methodology

Perform Character Embedding

Implement different models

Select champion model

Deep Learning Methodology

Perform Character Embedding

Implement different models

Select champion model

Represent each character in the URL by a vector of length 128 and add this layer to the classification model to learn the best vector which represents each character.

Deep Learning Methodology

Perform Character Embedding

Implement different models

Select champion model

Represent each character in the URL by a vector of length 128 and add this layer to the classification model to learn the best vector which represents each character.

Embedding for “br-icloud.com.br”

b	0.1680968	0.11709674	0.02931505	…….. 128
r	0.14431365	0.09179912	-0.01556315	…….. 128
.	….	….	….	…….. 128
.	….	….	….	…….. 128
b	0.1680968	0.11709674	0.02931505	…….. 128
r	0.14431365	0.09179912	-0.01556315	…….. 128
.. 500	….	….	….	…….. 128

Deep Learning Methodology

Perform Character Embedding

Implement different models

Select champion model

Using CNN with a 1D convolution layer as feature extraction and trying different other models as classifiers such as LSTM and Bi-LSTM.

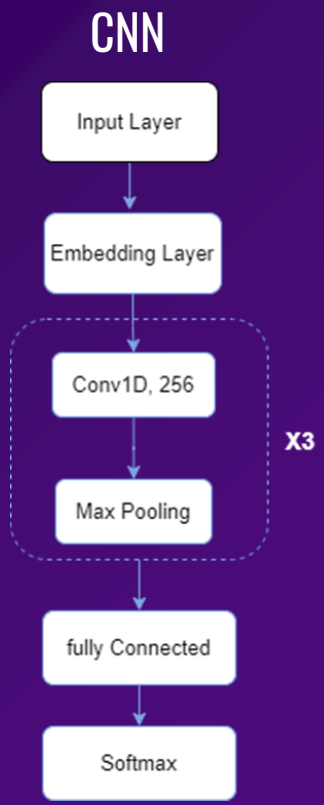
Deep Learning Methodology

Perform Character Embedding

Implement different models

Select champion model

Using CNN with a 1D convolution layer as feature extraction and trying different other models as classifiers such as LSTM and Bi-LSTM.



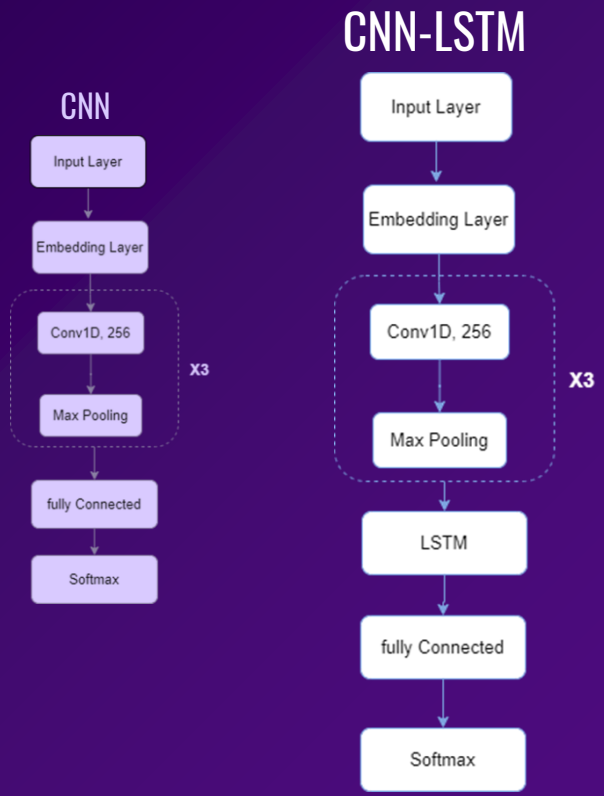
Deep Learning Methodology

Perform Character Embedding

Implement different models

Select champion model

Using CNN with a 1D convolution layer as feature extraction and trying different other models as classifiers such as LSTM and Bi-LSTM.



Deep Learning Methodology

Perform Character Embedding

Implement different models

Select champion model

Using CNN with a 1D convolution layer as feature extraction and trying different other models as classifiers such as LSTM and Bi-LSTM.

Deep Learning Methodology

Perform Character Embedding

Implement different models

Select champion model

As CNN and CNN-BiLSTM have the same Recall we selected the champion model based on the confusion matrix.

Model	Recall
CNN	97%
CNN-LSTM	96%
CNN-BiLSTM	97%

Deep Learning Methodology

Perform Character Embedding

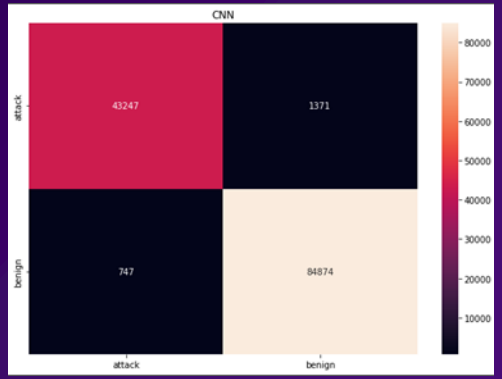
Implement different models

Select champion model

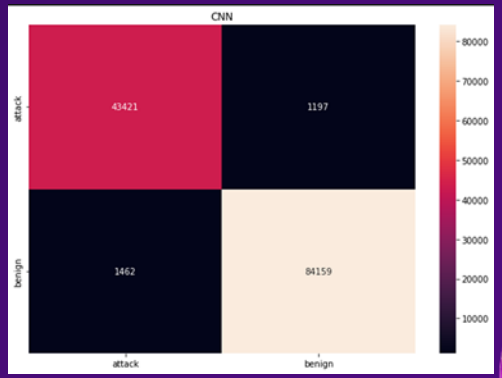
As CNN and CNN-BiLSTM have the same Recall we selected the champion model based on the confusion matrix.

Model	Recall
CNN	97%
CNN-LSTM	96%
CNN-BiLSTM	97%

CNN



CNN-BiLSTM



Deep Learning Methodology

Perform Character Embedding

Implement different models

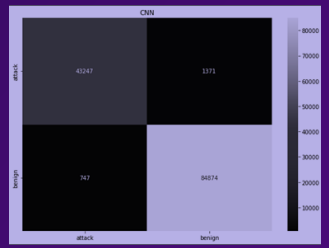
Select champion model

As CNN and CNN-BiLSTM have the same Recall we selected the champion model based on the confusion matrix.

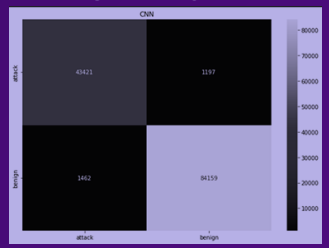
Model	Recall
CNN	97%
CNN-LSTM	96%
CNN-BiLSTM	97%

As CNN-BiLSTM has low false negative classifications so we selected it as the champion Model in DL

CNN



CNN-BiLSTM



Step 3
ML vs DL

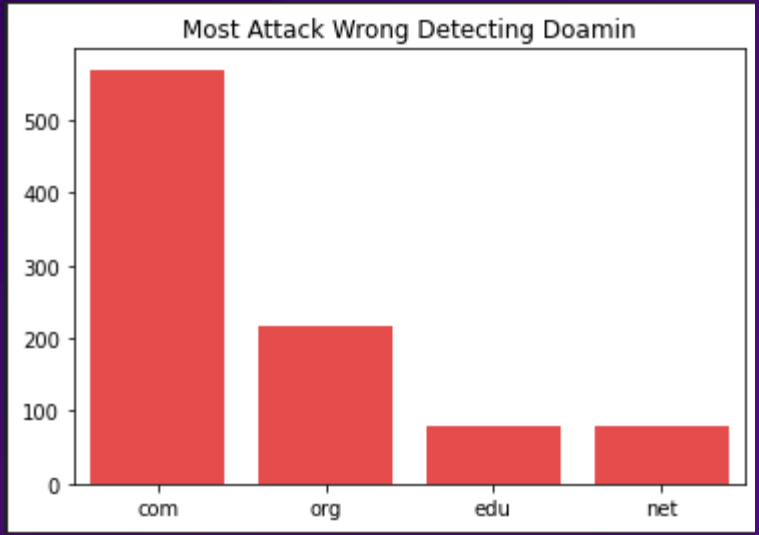
Compare ML and DL Model

Model	Recall
Naïve base	96%
CNN-BiLSTM	97%

Error Analysis

Most domains that confuse the model

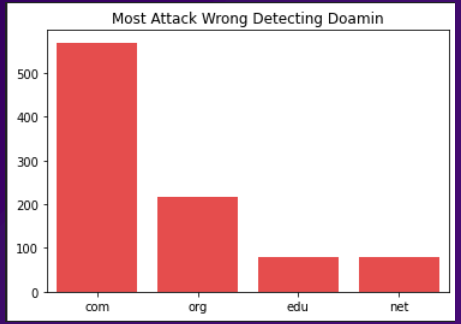
The model classifies them as normal but the true label is malicious, that happens because these domains appear as normal URLs more than a malicious ones.

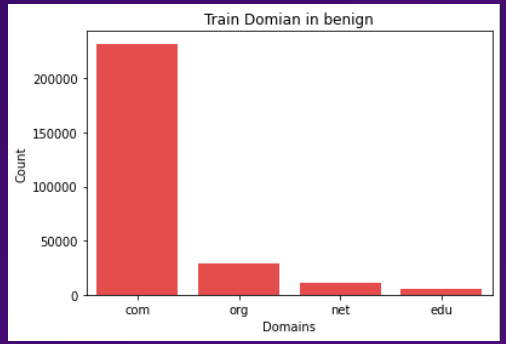


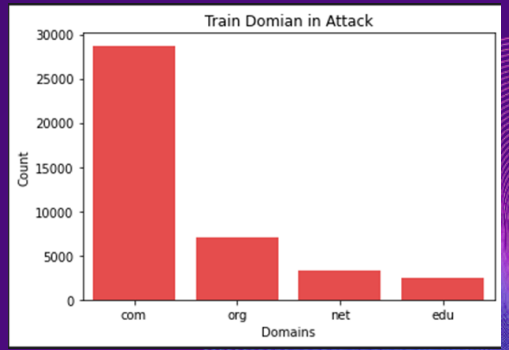
Error Analysis

Most domains that confuse the model

The model classifies them as normal but the true label is malicious, that happens because these domains appear as normal URLs more than malicious ones.







Conclusion

Detecting a malicious website is an important task to protect internet users, By implementing ML and DL models to solve this task, the DL model with character embedding has the best Recall and a probabilistic model like naive baya performs well in this task.

Deployment and Demo

Thank You