Office macro classification contest

Amit Berger
Chen Lipschitz
Yael Gabay

Data Exploration

After research, we found two articles about VBA macros, with the help of which, we understood which models are recommended to use. After experimenting and testing, we found out that the best classifier is Randomforest and the model is TF-IDF.

After investigating the data, we realized that there are common patterns for the mal codes, from which we deduced which features to add/remove.

References:
https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=9256296

https://ceur-ws.org/Vol-2259/aics_34.pdf

TF-IDF-
Term Frequency-
Inverse Document Frequency

A measure of the importance of a word in a document in relation to a collection of documents.

1. Term Frequency (TF): Measures how often a word appears in a document.

Words that appear more frequently in a document are given higher weights.

2. Inverse Document Frequency (IDF): Measures how unique or rare a word is across all documents in the collection.

Words that are rare across the entire collection but common in a particular document are given higher weights.

- Using TF-IDF helps you find the most unique and important words for each document.

Feature methods:

Number of variables

This feature counts the number of distinct

integer variable declarations in VBA scripts.

Average variable assignment length

This feature calculates the average length

of string variables in VBA scripts.

Presence of specific patterns

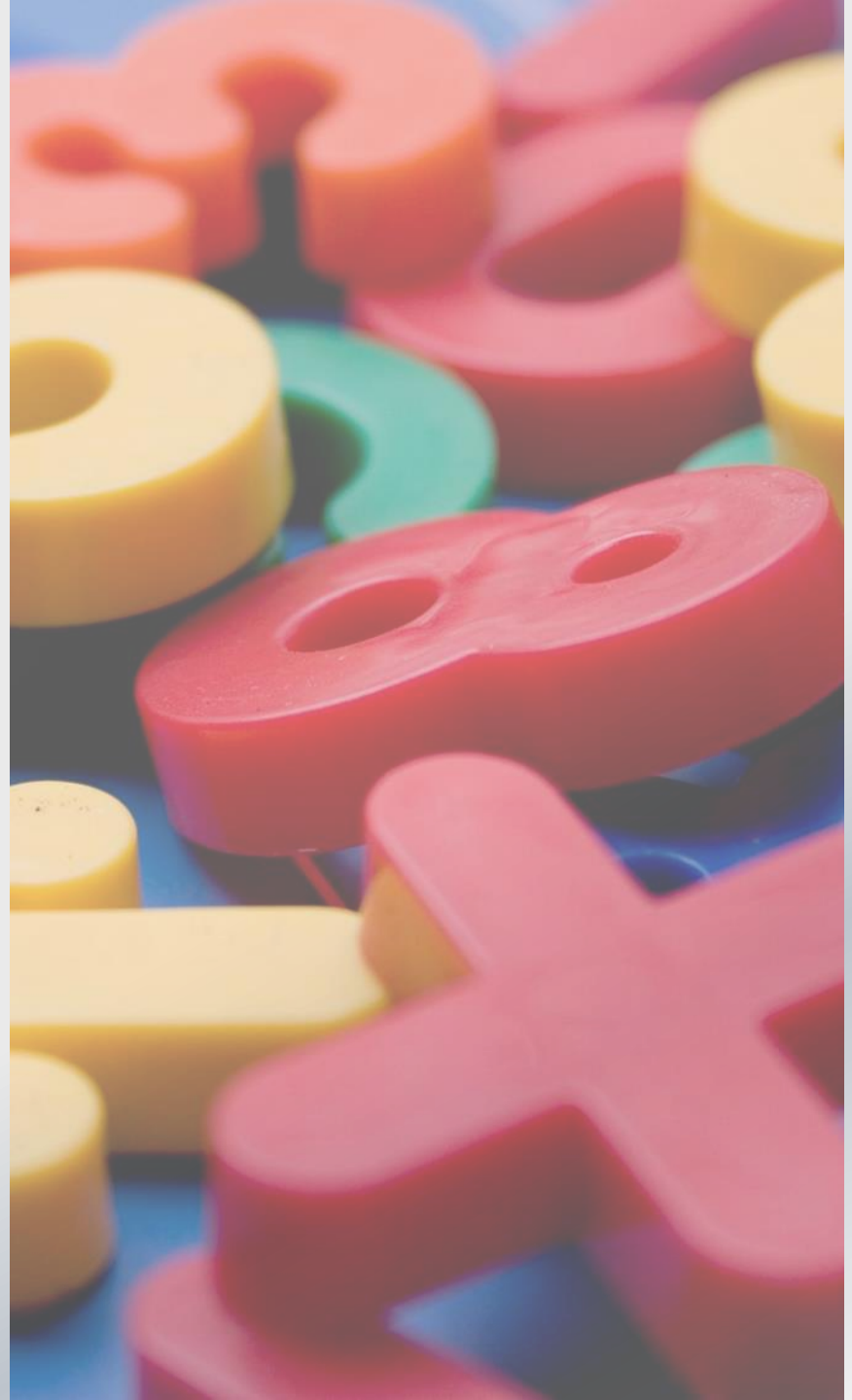
This feature checks for the presence of specific patterns in VBA code that might indicate unauthorized access. It looks for both unauthorized and authorized patterns. If found, returns 1 (indicating unauthorized access) otherwise, returns 0.

Presence of hexadecimal encoding

This feature checks whether the input text contains hexadecimal-encoded strings. If such, returns 1, otherwise, returns 0.

Presence of Base64 encoding

This feature checks whether the input text contains Base64-encoded strings. If so returns 1, otherwise, returns 0.



Training model

The result of training model

```
Training Accuracy: 0.9997491219267436
Training Confusion Matrix:
[[15809     1]
 [    7 16071]]
Training Classification Report:
              precision    recall  f1-score   support

         mal       1.00      1.00      1.00     15810
       white       1.00      1.00      1.00     16078

    accuracy                           1.00     31888
   macro avg       1.00      1.00      1.00     31888
weighted avg       1.00      1.00      1.00     31888
```

Validation

The result of **Validation**

```
Validation Accuracy: 0.9960485464295794
Validation Confusion Matrix:
[[5280   40]
 [   2 5307]]
Validation Classification Report:
              precision    recall  f1-score   support

         mal       1.00      0.99      1.00      5320
       white       0.99      1.00      1.00      5309

    accuracy                           1.00     10629
   macro avg       1.00      1.00      1.00     10629
weighted avg       1.00      1.00      1.00     10629
```

Validation

The result of Validation

False_positive & True_negative rows

```
1 false_positive_indices = (predictions_validation == 'mal') & (y_validation == 'w
2 print("\nFalse Positive- white but was predicted mal Rows in Original CSV:")
3 print(validation_df.loc[false_positive_indices])
```

```
False Positive- white but was predicted mal Rows in Original CSV:
      label                                           vba_code generated_label
2298  white  Private Sub ComboBox11_DropButtonClick()\n'The...              mal
4650  white  Private Sub ComboBox11_DropButtonClick()\n'The...              mal
```

```
1 true_negative_indices = (predictions_validation == 'white') & (y_validation ==
2 print("\nTrue Negative- Mal but was predicted as white Rows in Original CSV:")
3 print(validation_df.loc[true_negative_indices])
```

```
True Negative- Mal but was predicted as white Rows in Original CSV:
      label                                           vba_code generated_label
50      mal                                                  b           white
63      mal  Function AAA()\nEnd Function\nIf 1 <> 1 Then\n...           white
421     mal  Sub OYwrUVCJckZuLvRBMSOZrFN()\n\nDim tudfTFyTf...           white
1479    mal  Sub Document_Open()\nIf 23 < 153 Then\n' rRXLq...           white
2040    mal  Function AAA()\nEnd Function\nIf 1 <> 1 Then\n...           white
2081    mal  Sub AutoOpen()\n    Application.Run "khhzrzr"\...           white
2092    mal  Private Sub workbook_open()\nfWH7voI76HLe.f_29...           white
2138    mal  Sub backlash()\nDim hinny As Variant\nDim mel ...           white
2149    mal  Const hENysOXOpuhagUHibaVYnaHysIiQOQOzISiTy = ...           white
2507    mal  Public Const UserVersion = "2.7"\nPublic Curre...           white
2570    mal       Sub IIS()\n'Macro created by Minoli\nEnd Sub           white
```