Project paper

The NLP and ML Implanted into HSBC surveillance

Group: Yu DUAN

Yanqing ZENG

Yifan ZHENG

DATE: 2017-09-09

Description

To achieve HSBC text system surveillance， applied to email, chatbox, file transfer and other text systems
Using machine learning to load enron email dataset, to achieve text analysis. To avoid illegal, illegal and improper operation

1、Theme

the surveillance system based on ML

2、Object

Hsbc employee and corporator

3、Implementation area

Hsbc company

4、Date

design in 9-8 to 9-10

5、Target

Avoid risks, to achieve a safe banking surveillance system

6、Technology use

Machine Supervised Learning, NLP, Python, Enron email dataset

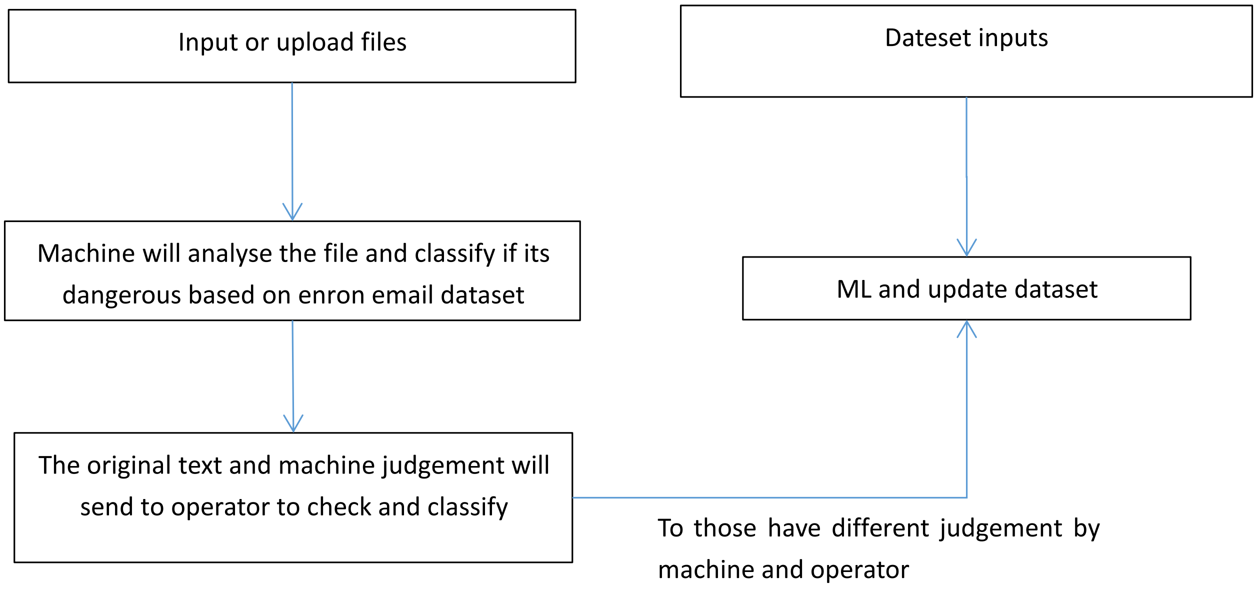
Project background analysis

At present, the banking system has not joined the artificial intelligence surveillance system, most of the surveillance from the surveillance department using mathematical methods random sampling to assess the surveillance process, large human resources consumption. And can not avoid human error or non-compliance operation, there are hidden dangers
The use of the system plug-in, will be applied to all text system surveillance, and the system will continue to update and to be more suitable for HSBC company

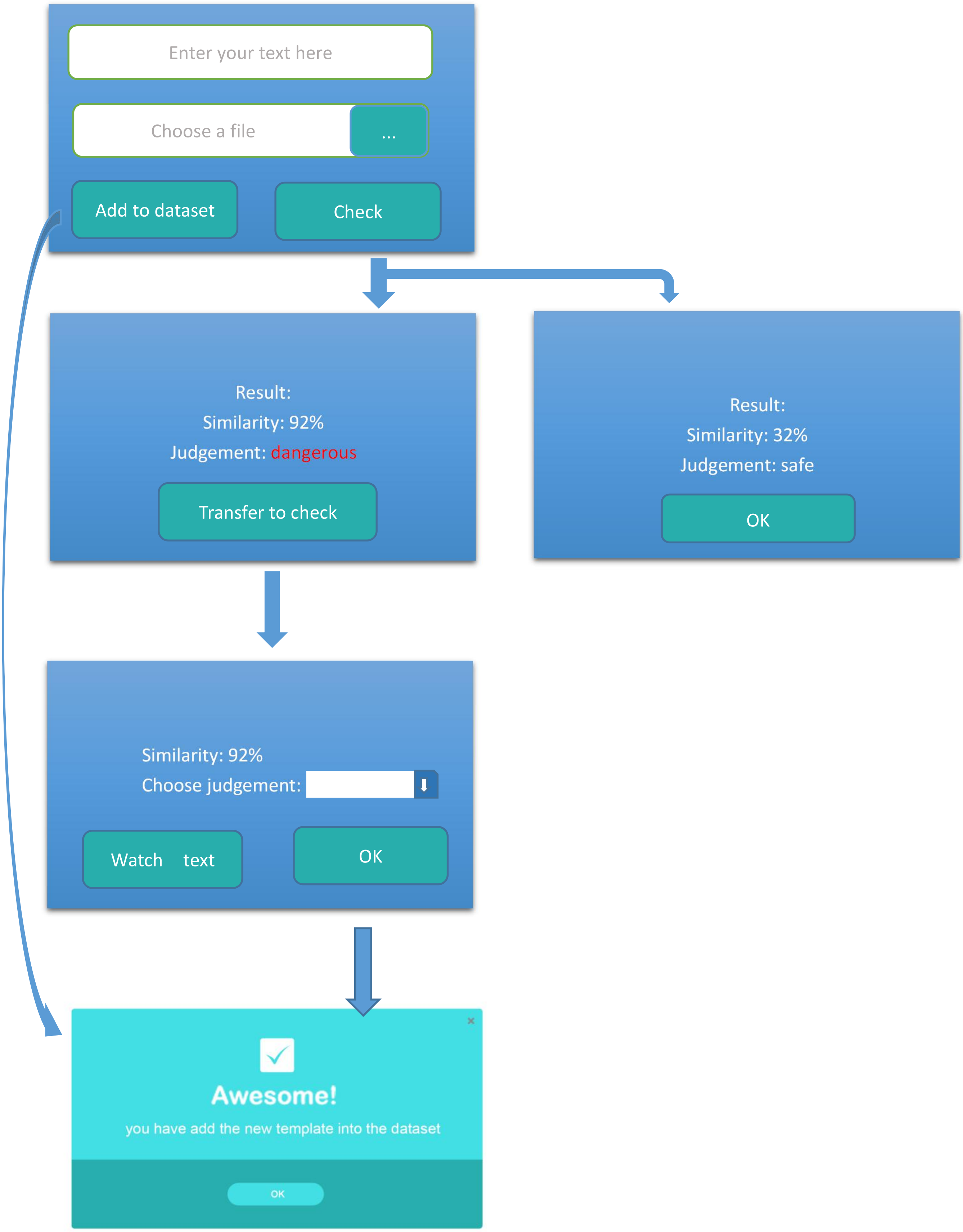
Demo Description

Based on the python environment to build the text read the analysis system, access to similarity, classification, and the realization of edge with the edge of the function, the realization of machine intelligence, to achieve more suitable for hsbc system under the surveillance system

Process Design


Input or upload files
Machine will analyse the file and classify if its dangerous based on enron email dataset
The original text and machine judgement will send to operator to check and classify
Dateset inputs
ML and update dataset
To those have different judgement by machine and operator

Page Design



Result:
Similarity: 92%
Judgement: dangerous

Result:
Similarity: 32%
Judgement: safe

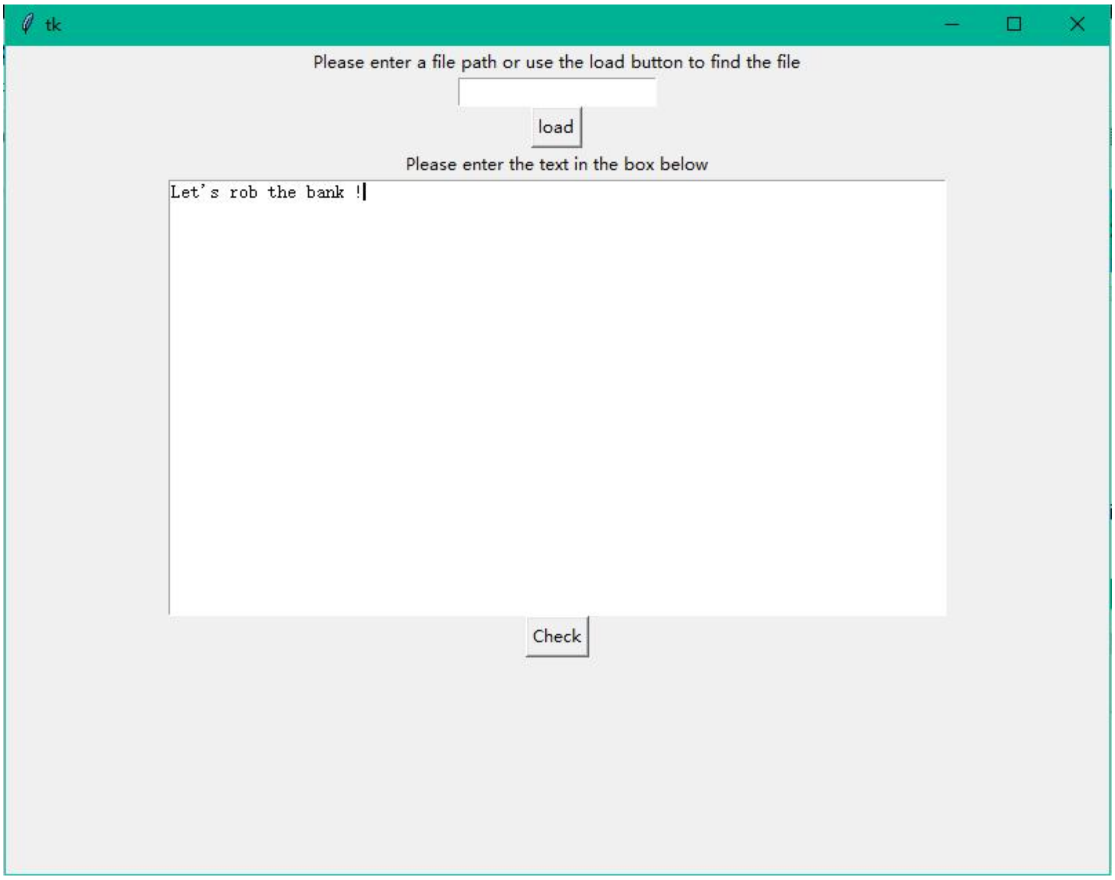
Similarity: 92%
Choose judgement:

Awesome!

you have add the new template into the dataset

The Result

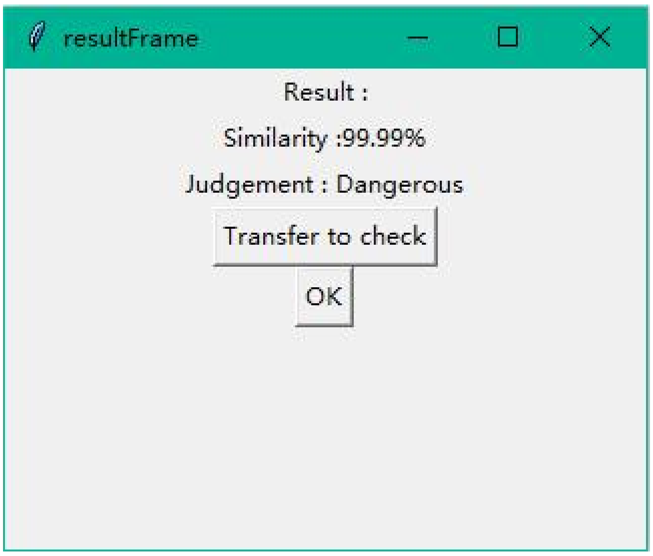
1) we have two ways of using this detection system, the first one is that we can upload a file by clicking the *load* button(which is required as .txt and .doc files).
The second method is by typing the text directly into the text box. Then we can lance the detection by pressing the *check* button.



tk
Please enter a file path or use the load button to find the file
load
Please enter the text in the box below
Check

Let’s rob the bank !

2) In this step, the computer will compare the contents of the file or the typed text with the data set, and it will give us a percentage of similarity of these two text. In this case, obviously the answer is DANGEROUS !

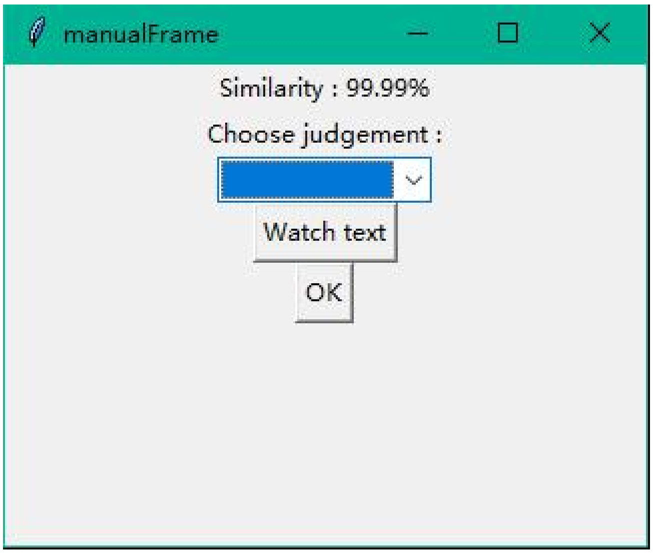


Result :

Similarity :99.99%

Judgement : Dangerous

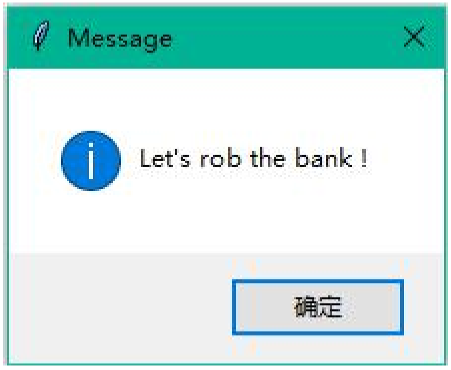
3)If the judgement of the similarity is *By check,* we strongly advise that the text should be check manually. Here is the manual check window :



Similarity : 99.99%

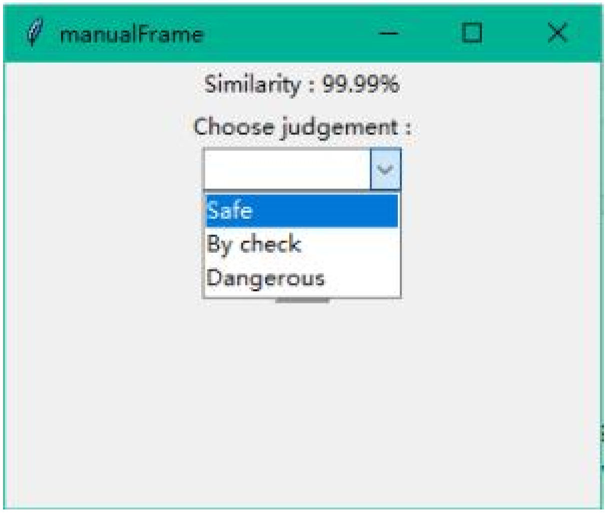
Choose judgement :

By pressing the *Watch check* button we can review the contents of the text :



Let's rob the bank !

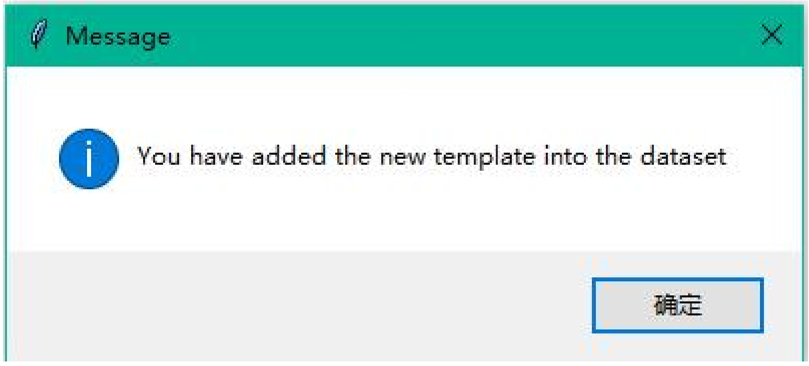
Then we are able to choose a dangerous level for the chosen text :



Similarity : 99.99%

Choose judgement :

After defining the dangerous level, we can just press the *OK* button to put the text into the data set for further usage and the detection program is finished.

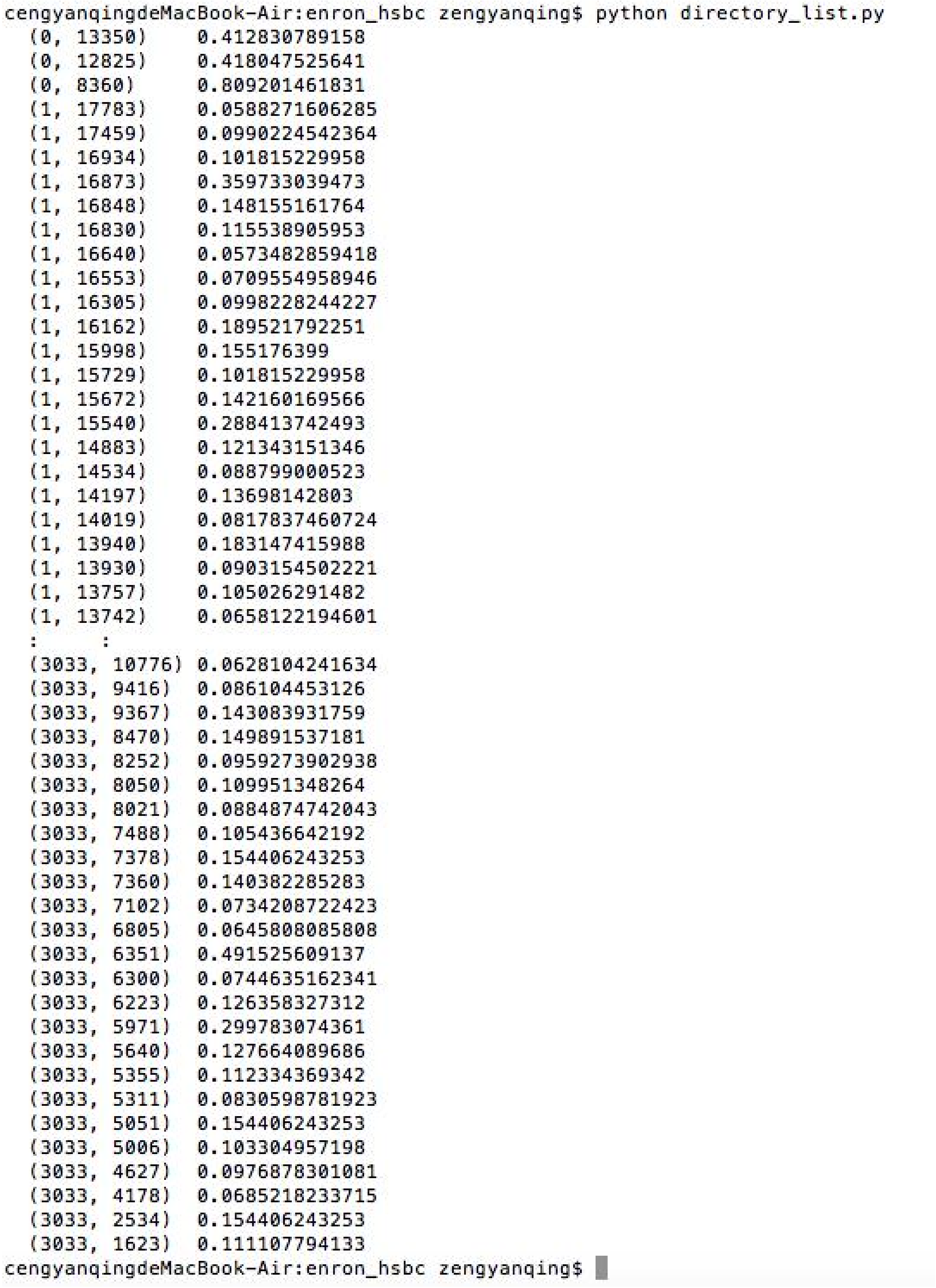


You have added the new template into the dataset

Natural Language Processing

Basic principal of NLP

The model is trained with the open source Enron data. In order to realize the process, we filter the *stop words,* such as the short function words ‘the’, ‘is’, ‘which’, etc. Then, we transform the text into a vector of word frequency.



```
cengyanqingdeMacBook-Air:enron_hsbc zengyanqing$ python directory_list.py
  (0, 13350)    0.412830789158
  (0, 12825)    0.418047525641
  (0, 8360)     0.809201461831
  (1, 17783)    0.0588271606285
  (1, 17459)    0.0990224542364
  (1, 16934)    0.101815229958
  (1, 16873)    0.359733039473
  (1, 16848)    0.148155161764
  (1, 16830)    0.115538905953
  (1, 16640)    0.0573482859418
  (1, 16553)    0.0709554958946
  (1, 16305)    0.0998228244227
  (1, 16162)    0.189521792251
  (1, 15998)    0.155176399
  (1, 15729)    0.101815229958
  (1, 15672)    0.142160169566
  (1, 15540)    0.288413742493
  (1, 14883)    0.121343151346
  (1, 14534)    0.088799000523
  (1, 14197)    0.13698142803
  (1, 14019)    0.0817837460724
  (1, 13940)    0.183147415988
  (1, 13930)    0.0903154502221
  (1, 13757)    0.105026291482
  (1, 13742)    0.0658122194601
  :     :
  (3033, 10776) 0.0628104241634
  (3033, 9416)  0.086104453126
  (3033, 9367)  0.143083931759
  (3033, 8470)  0.149891537181
  (3033, 8252)  0.0959273902938
  (3033, 8050)  0.109951348264
  (3033, 8021)  0.0884874742043
  (3033, 7488)  0.105436642192
  (3033, 7378)  0.154406243253
  (3033, 7360)  0.140382285283
  (3033, 7102)  0.0734208722423
  (3033, 6805)  0.0645808085808
  (3033, 6351)  0.491525609137
  (3033, 6300)  0.0744635162341
  (3033, 6223)  0.126358327312
  (3033, 5971)  0.299783074361
  (3033, 5640)  0.127664089686
  (3033, 5355)  0.112334369342
  (3033, 5311)  0.0830598781923
  (3033, 5051)  0.154406243253
  (3033, 5006)  0.103304957198
  (3033, 4627)  0.0976878301081
  (3033, 4178)  0.0685218233715
  (3033, 2534)  0.154406243253
  (3033, 1623)  0.111107794133
cengyanqingdeMacBook-Air:enron_hsbc zengyanqing$
```

Possible analysis

Detect the potential persons of interests

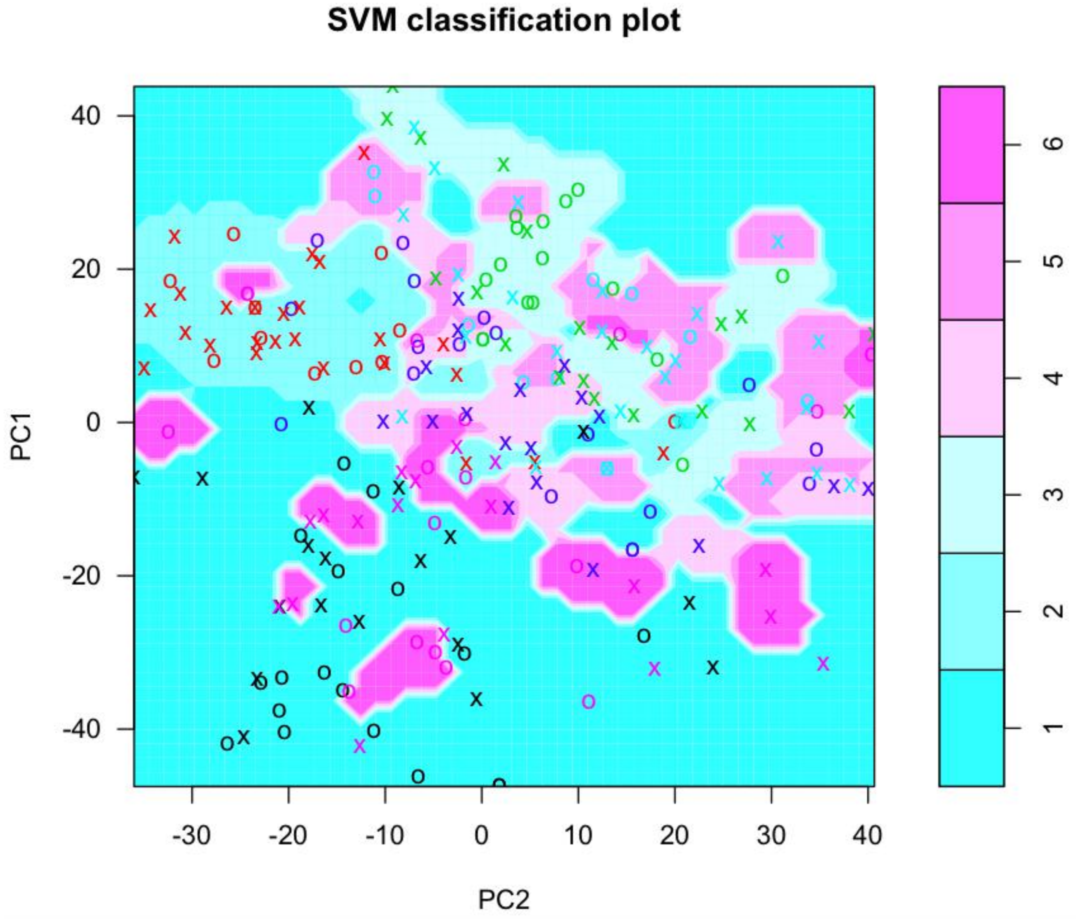
Detect the potential possibility of money laundry

Sentiment analysis

Risk management

Classification models of Artifical Intelligence

Support Vector Machine (SVM)



Decision Tree

Decision Forest

Neuron network

Bayesian classifier