from flask import Flask,render\_template,url\_for,request, jsonify

import pandas as pd

import pickle

from sklearn.feature\_extraction.text import TfidfVectorizer

from sklearn.multiclass import \*

from sklearn.svm import \*

app = Flask(\_\_name\_\_)

global Classifier

global Vectorizer

# load data

data = pd.read\_csv('data/sms-spam-collection-dataset/spam.csv', encoding='latin-1')

train\_data = data[:4400] # 4400 items

test\_data = data[4400:] # 1172 items

# train model

Classifier = OneVsRestClassifier(SVC(kernel='linear', probability=True))

Vectorizer = TfidfVectorizer()

vectorize\_text = Vectorizer.fit\_transform(train\_data.v2)

Classifier.fit(vectorize\_text, train\_data.v1)

@app.route('/')

def home():

return render\_template('index.html')

@app.route('/predict',methods=['POST'])

def predict():

error = ''

predict\_proba = ''

predict = ''

if request.method == 'POST':

message = request.form['message']

try:

if len(message) > 0:

vectorize\_message = Vectorizer.transform([message])

predict = Classifier.predict(vectorize\_message)[0]

predict\_proba = Classifier.predict\_proba(vectorize\_message).tolist()

else:

predict = 2

except BaseException as inst:

error = str(type(inst).\_\_name\_\_) + ' ' + str(inst)

if (predict == 'ham'):

predict = 1

elif (predict == 'spam'):

predict = 0

return render\_template('index.html', prediction = predict)

if \_\_name\_\_ == '\_\_main\_\_':

app.run(debug=True, use\_reloader=True)