Email Spam Classifier

Noman Siddique 19P-1664

Asjid Tahir 19P-0085

Abstract

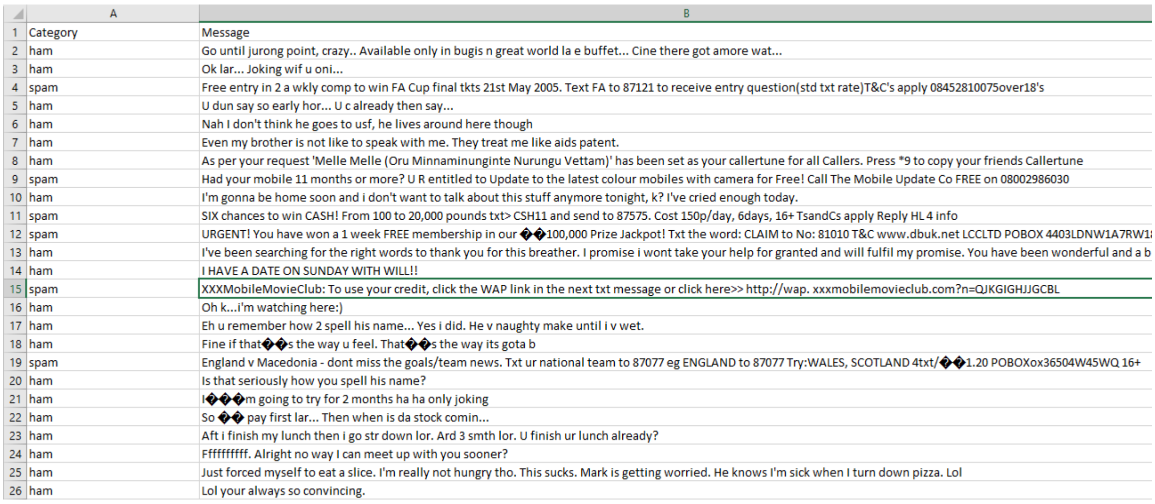
This email spam classifier is a machine learning model that takes an email as input and predicts whether it is a spam email or not. The model is based on logistic regression and uses a TfidfVectorizer to convert the text data into feature vectors that can be used as input to the logistic regression model. The model is trained on a dataset of emails that have been labeled as spam or not spam.

The output of the model is a binary classification, indicating whether the email is spam or not. The accuracy of the model can be measured by comparing the predicted labels to the true labels in a test dataset. This email spam classifier can be useful in filtering out unwanted emails and preventing them from reaching the user's inbox.

System Engineering:

A- Dataset

Dataset contain the email categorized as Spam or ham



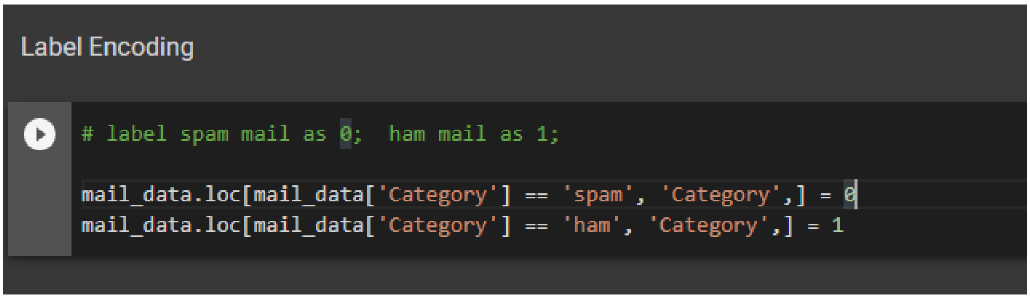
	A	B
1	Category	Message
2	ham	Go until jurong point, crazy.. Available only in bugis n great world la e buffet... Cine there got amore wat...
3	ham	Ok lar... Joking wif u oni...
4	spam	Free entry in 2 a wkly comp to win FA Cup final tkts 21st May 2005. Text FA to 87121 to receive entry question(std txt rate)T&C's apply 08452810075over18's
5	ham	U dun say so early hor... U c already then say...
6	ham	Nah I don't think he goes to usf, he lives around here though
7	ham	Even my brother is not like to speak with me. They treat me like aids patent.
8	ham	As per your request 'Melle Melle (Oru Minnaminunginte Nurungu Vettam)' has been set as your callertune for all Callers. Press *9 to copy your friends Callertune
9	spam	Had your mobile 11 months or more? U R entitled to Update to the latest colour mobiles with camera for Free! Call The Mobile Update Co FREE on 08002986030
10	ham	I'm gonna be home soon and i don't want to talk about this stuff anymore tonight, k? I've cried enough today.
11	spam	SIX chances to win CASH! From 100 to 20,000 pounds txt> CSH11 and send to 87575. Cost 150p/day, 6days, 16+ TsandCs apply Reply HL 4 info
12	spam	URGENT! You have won a 1 week FREE membership in our ��100,000 Prize Jackpot! Txt the word: CLAIM to No: 81010 T&C www.dbuk.net LCCLTD POBOX 4403LDNW1A7RW1
13	ham	I've been searching for the right words to thank you for this breather. I promise i wont take your help for granted and will fulfil my promise. You have been wonderful and a b
14	ham	I HAVE A DATE ON SUNDAY WITH WILL!!
15	spam	XXXMobileMovieClub: To use your credit, click the WAP link in the next txt message or click here>> http://wap. xxxmobilemovieclub.com?n=QJKGIGHJJGCBL
16	ham	Oh k...i'm watching here:)
17	ham	Eh u remember how 2 spell his name... Yes i did. He v naughty make until i v wet.
18	ham	Fine if that��s the way u feel. That��s the way its gota b
19	spam	England v Macedonia - dont miss the goals/team news. Txt ur national team to 87077 eg ENGLAND to 87077 Try:WALES, SCOTLAND 4txt/��1.20 POBOXox36504W45WQ 16+
20	ham	Is that seriously how you spell his name?
21	ham	I���m going to try for 2 months ha ha only joking
22	ham	So �� pay first lar... Then when is da stock comin...
23	ham	Aft i finish my lunch then i go str down lor. Ard 3 smth lor. U finish ur lunch already?
24	ham	Fffffffff. Alright no way I can meet up with you sooner?
25	ham	Just forced myself to eat a slice. I'm really not hungry tho. This sucks. Mark is getting worried. He knows I'm sick when I turn down pizza. Lol
26	ham	Lol your always so convincing.

S, A. (2023) *Spam (or) ham*, *Kaggle*. Available at: https://www.kaggle.com/datasets/arunasivapragasam/spam-or-ham (Accessed: May 8, 2023).

B- Label Encoding

These lines of code are used to preprocess the data in the email spam classifier project by converting the labels of the dataset from string values to numerical values. The labels are converted to 0 for spam emails and 1 for ham (i.e., non-spam) emails.

The code uses the Pandas library to access the 'Category' column of the mail_data DataFrame and then uses the **loc()** function to select the rows where the 'Category' column is equal to 'spam' or 'ham'. For each row that meets the condition, the code sets the value of the 'Category' column to 0 or 1, respectively, using the assignment operator (=).


Label Encoding

```
# label spam mail as 0;  ham mail as 1;

mail_data.loc[mail_data['Category'] == 'spam', 'Category',] = 0
mail_data.loc[mail_data['Category'] == 'ham', 'Category',] = 1
```

This conversion from string labels to numerical labels is necessary for the logistic regression model to work correctly, as the model expects the labels to be numerical values rather than strings. By converting the labels to numerical values, the data can be used to train the logistic regression model to predict the probability of an email being spam or not.

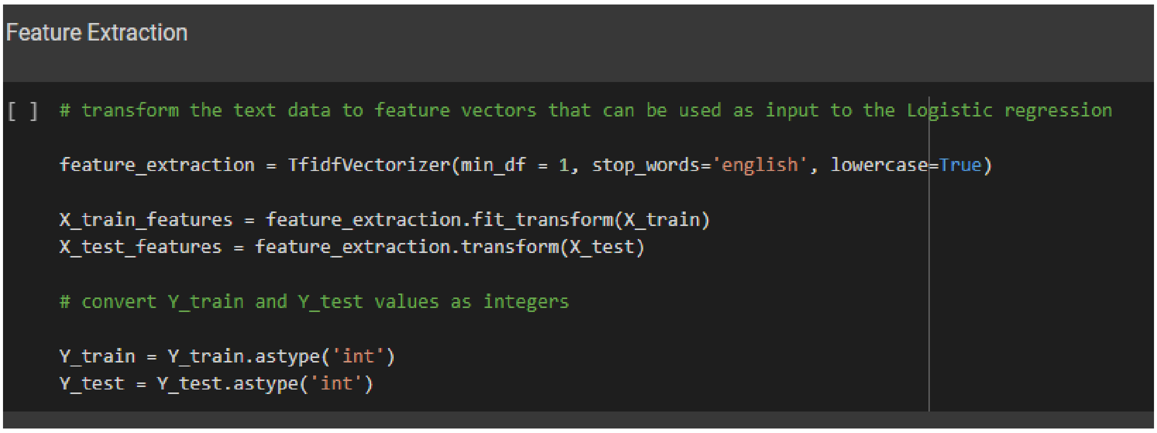
C- Feature Extraction

These lines of code are used to transform the text data of the emails into feature vectors that can be used as input to the logistic regression model.
The TfidfVectorizer class from the scikit-learn library is used to extract features from the text data. The 'min_df' parameter specifies the minimum number of documents (i.e., emails) that a word must appear in to be included in the feature vector. The 'stop_words' parameter specifies a list of common English words to be excluded from the feature vector. The 'lowercase' parameter is set to True to convert all the text to lowercase before extracting the features.
The 'fit_transform' method of the TfidfVectorizer object is called on the training data X_train to learn the vocabulary and compute the inverse document frequency (IDF) weightings for the features, and then transform the

text data into a sparse matrix of feature vectors. The resulting feature vectors are stored in the X_train_features variable.

The 'transform' method of the TfidfVectorizer object is then called on the test data X_test to transform the text data into feature vectors using the vocabulary and IDF weightings learned from the training data. The resulting feature vectors are stored in the X_test_features variable.

Finally, the labels Y_train and Y_test is converted from string values to integer values using the 'astype' method with the 'int' parameter, which is necessary for the logistic regression model to work correctly.



Feature Extraction

```
# transform the text data to feature vectors that can be used as input to the Logistic regression

feature_extraction = TfidfVectorizer(min_df = 1, stop_words='english', lowercase=True)

X_train_features = feature_extraction.fit_transform(X_train)
X_test_features = feature_extraction.transform(X_test)

# convert Y_train and Y_test values as integers

Y_train = Y_train.astype('int')
Y_test = Y_test.astype('int')
```

D- Logistic Regression Model

Logistic regression is a statistical method for analyzing a dataset in which there are one or more independent variables that determine an outcome. It is a type of regression analysis used for predicting the outcome of a categorical dependent variable (i.e., a variable that can take on one of a limited number of possible values).

In the context of Natural Language processing and machine learning, logistic regression is often used for binary classification tasks, where the goal is to predict the probability of an input belonging to one of two possible classes. For example, in the email spam classifier project, the logistic regression model is used to predict whether an email is spam or not based on the text content of the email.

E- Building a Predictive System

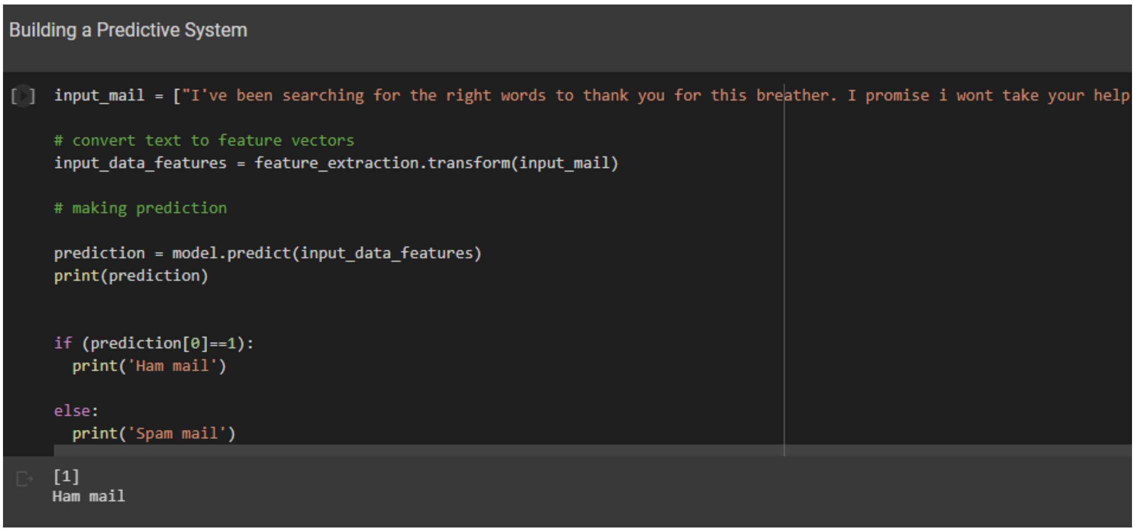
These lines of code are used to classify an input email as either spam or ham (i.e., non-spam) using the trained logistic regression model and the

TfidfVectorizer object that was used to transform the training and test data into feature vectors.

First, a sample email is defined as a list of strings with the variable name 'input_mail'. This email is then passed through the same TfidfVectorizer object that was used to transform the training and test data into feature vectors, using the 'transform' method of the TfidfVectorizer object, to convert the text data into feature vectors that can be used as input to the logistic regression model. The resulting feature vectors are stored in the variable 'input_data_features'.

Next, the trained logistic regression model is used to predict the label of the input email by calling the 'predict' method on the input feature vector. The resulting prediction is stored in the 'prediction' variable.

Finally, the predicted label is printed to the console using a conditional statement. If the predicted label is 1, which corresponds to a ham email, the program prints 'Ham mail'. Otherwise, if the predicted label is 0, which corresponds to a spam email, the program prints 'Spam mail'. This allows the user to quickly determine whether an input email is likely to be spam or not based on the predictions of the logistic regression model.



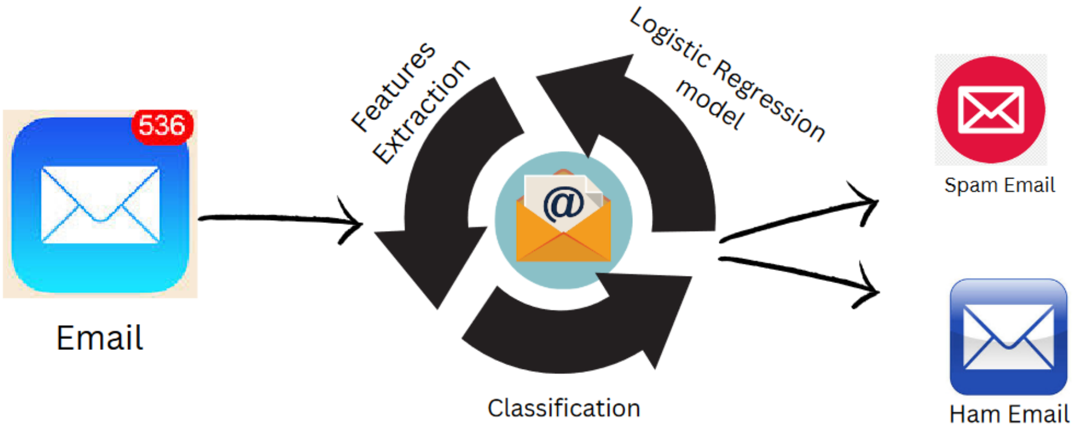
Building a Predictive System

```
input_mail = ["I've been searching for the right words to thank you for this breather. I promise i wont take your help

# convert text to feature vectors
input_data_features = feature_extraction.transform(input_mail)

# making prediction

prediction = model.predict(input_data_features)
print(prediction)


if (prediction[0]==1):
  print('Ham mail')

else:
  print('Spam mail')
```

```
[1]
Ham mail
```

F- System Design

i- Flow Chart:


536
Email
Features Extraction
Logistic Regression model
Classification
Spam Email
Ham Email