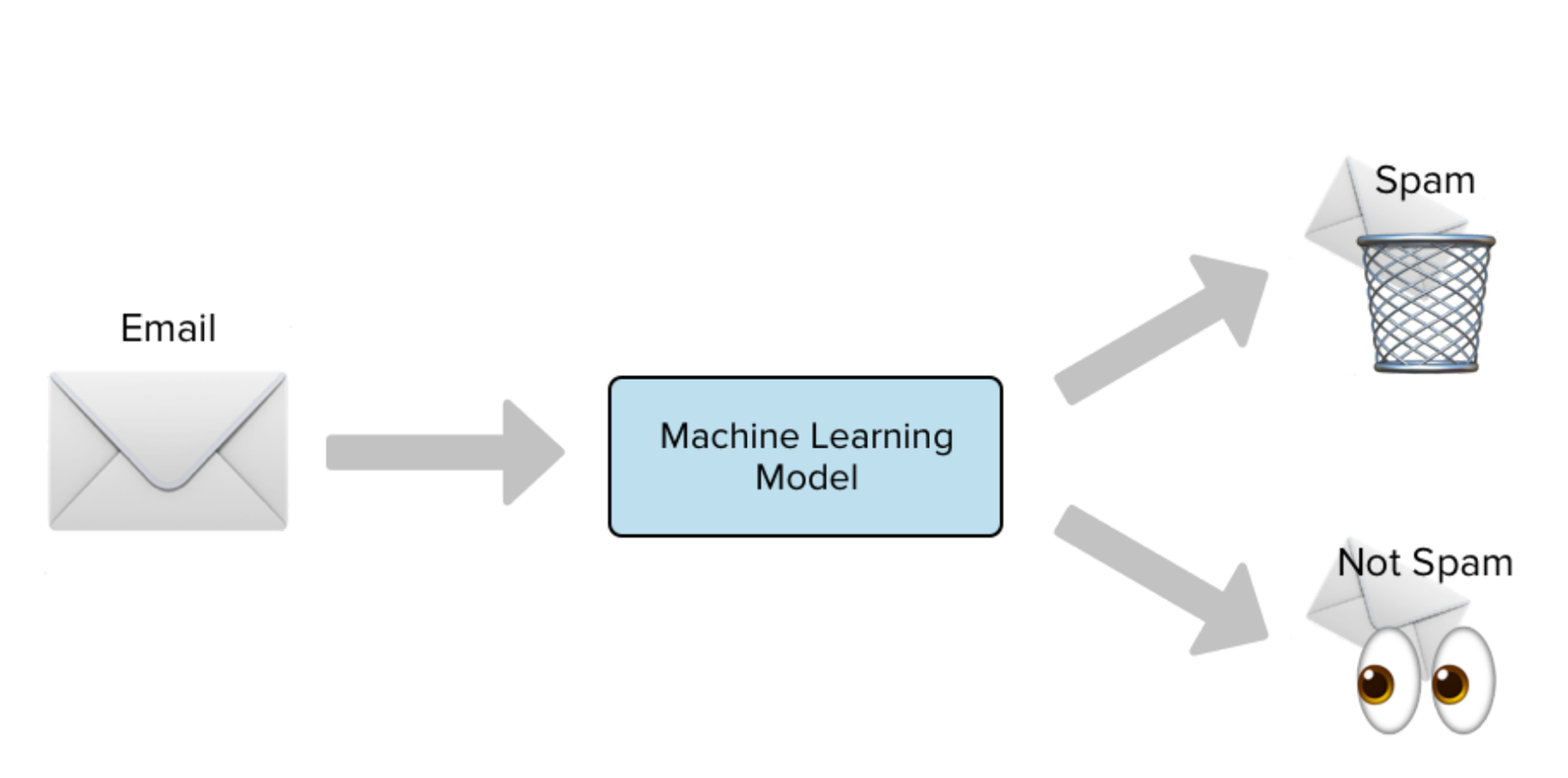
Spam Detection with Logistic Regression

Natasha Sharma Follow
May 5, 2018 · 6 min read

I still remember my first day in machine learning class. The first example which was provided to explain, how machine learning works, was "Spam Detection". I think in most of the machine learning courses tutors provide the same example, but, in how many courses you actually get to implement the model? We talk how machine learning involved in Spam Detection and then just move on to other things.

Introduction

The idea of this post is to understand step by step working of the spam filter and how it helps in making everyone life easier. Also, next time when you see a "You have won a lottery" email rather than ignoring it, you might prefer to report it as a spam.



Courtesy : Google images(Medium post)

The above image gives an overview of spam filtering , plenty of emails arrive everyday, some goes to spam and rest stays in our primary inbox(unless you have further categories defined). The blue box in the middle — Machine Learning Model, how does it decide which mail is spam and which one is not.

Before we start talking about the algorithm and the code, take a step back and try relating that simple explanation of spam detection with monthly active Gmail account(which is approximately 1 billion). The picture seems pretty complicated, isn't it? Let's get an overview on how does gmail use the filtering for a huge number of accounts.

Gmail Spam Detection

We all know the data Google has, is not obviously in paper files. They have data centers which maintain the customers data. Before Google/Gmail decides to segregate the emails into spam or not spam category, before it arrives to your mailbox, hundreds of rules apply to those email in the data centers. These rules describe the properties of a spam email. There are common types of spam filters which are used by Gmail/Google —

Blatant Blocking- Deletes the emails even before it reaches to the inbox.

Bulk Email Filter- This filter helps in filtering the emails that are passed through other categories but are spam.

Category Filters- User can define their own rules which will enable the filtering of the messages according to the specific content or the email addresses etc.

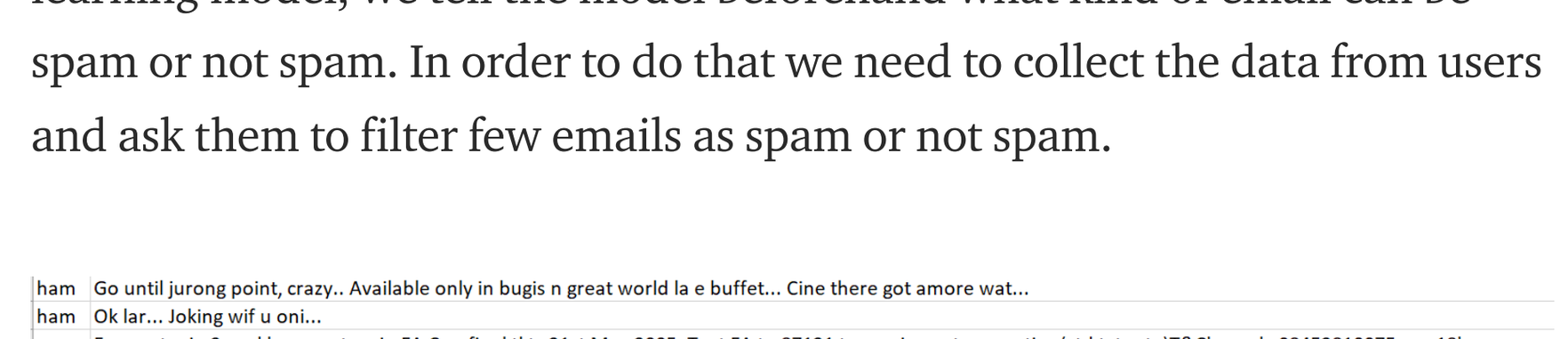
Null Sender Disposition- Dispose of all messages without an SMTP envelope sender address. Remember when you get an email saying, "Not delivered to xyz address".

Null Sender Header Tag Validation- Validate the messages by checking security digital signature.

There are ways to avoid spam filtering and send your emails straight to the inbox. To learn more about Gmail spam filter please watch this informational video from Google.

Create a Spam Detector : Pre-processing

Moving on to our aim of creating our very own spam detector. Let's talk about about that blue box in the middle of above image. The model is like a small kid unless you tell the kid, the difference between salt and sugar, he/she won't be able to recognize it. The similar idea we apply on machine learning model, we tell the model beforehand what kind of email can be spam or not spam. In order to do that we need to collect the data from users and ask them to filter few emails as spam or not spam.



ham	Go until jurong point, crazy.. Available only in bugis n great world la e buffet... Cine there got amore wat...
ham	Ok lar... Joking wif u oni...
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
ham	U dun say so early hor... U c already then say...
ham	Nah I don't think he goes to usf, he lives around here though
spam	FreeMsg Hey there darling it's been 3 week's now and no word back! I'd like some fun you up for it still? Tb ok! XxX std chgs to send, £1.50 to rcv
ham	Even my brother is not like to speak with me. They treat me like aids patent.
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
spam	WINNER!! As a valued network customer you have been selected to receivea £900 prize reward! To claim call 09061701461. Claim code KL341. Valid 12 hours only.
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

Kaggle Spam Detection Dataset

The above image is a snapshot of tagged email that have been collected for Spam research. It contains one set of messages in English of 5,574 emails, tagged according being legitimate(ham) or spam.

Now that we have data with tagged emails — Spam or Not Spam, what should we do next? We would need to train the machine to make it smart enough to categorize the emails on its own. But, the machine can't read the full statement and start categorizing the emails. Here we will need to use our NLP basics (check out my last blog).

We will first do some pre-processing on message text, like removing - punctuation and stop words.

```
def text_preprocess(text):
    text = text.translate(str.maketrans('', '', string.punctuation))
    text = [word for word in text.split() if word.lower() not in stopwords.words('english')]
    return " ".join(text)
```

Once the pre-processing is done, we would need to vectorize the data — i.e collecting each word and its frequency in each email. The vectorization will produce a matrix.

```
vectorizer = TfidfVectorizer("english")
message_mat = vectorizer.fit_transform(message_data_copy)
message_mat
```

This vector matrix can be used create train/test split. This will help us to train the model/machine to be smart and test the accuracy of its results.

```
message_train, message_test, spam_nospam_train, spam_nospam_test = train_test_split(message_mat, message_data['Spam/Not_Spam'], test_size=0.3, random_state=20)
```

Choosing a model

Now that we have train test split, we would need to choose a model. There is a huge collection of models but for this particular exercise we will be using logistic regression.Why?

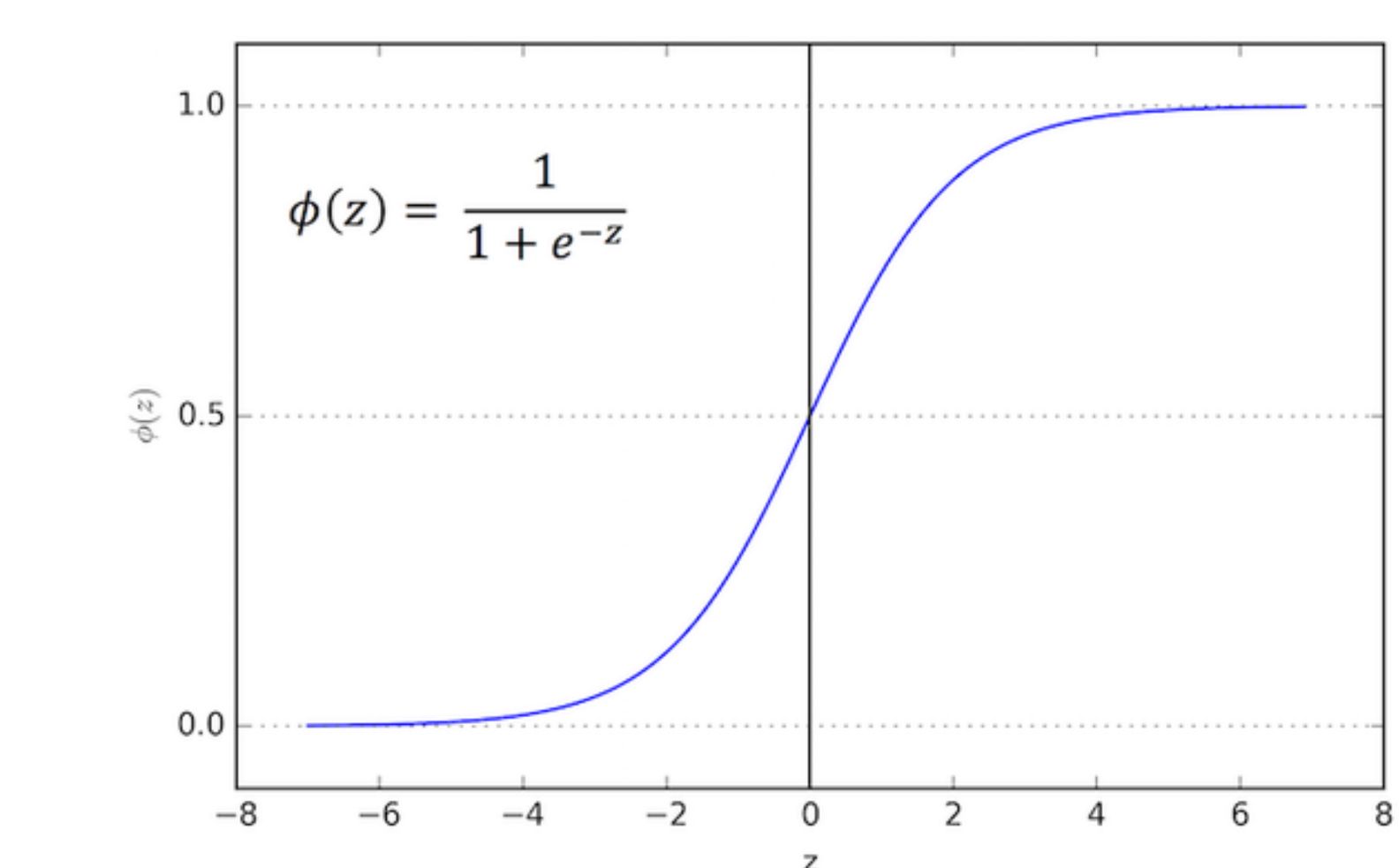
Generally when someone asks, what is logistic regression? what do you tell them — Oh! it is an algorithm which is used for categorizing things into two classes (most of the time) i.e. the result is measured using a dichotomous variable. But, how does logistic regression classify thing into classes like - **binomial**(2 possible values), **multinomial**(3 or more possible values) and **ordinal**(deals with ordered categories). For this post we will only be focusing on binomial logistic regression i.e. the outcome of the model will be categorized into two classes.

Logistic Regression

According to Wikipedia definition,

> Logistic Regression measures the relationship between the categorical dependent variable and one or more independent variables by **estimating probabilities** using a **logistic function**.

From the definition it seems, the logistic function plays an important role in classification here but we need to understand what is logistic function and how does it help in estimating the probability of being in a class.



Courtesy — Google image(Quora post)

The formula mentioned in the above image is known as Logistic function or Sigmoid function and the curve called Sigmoid curve. The Sigmoid function gives an S shaped curve. The output of Sigmoid function tends towards 1 as $z \to \infty$ and tends towards 0 as $z \to -\infty$. Hence Sigmoid/logistic function produces the value of dependent variable which will always lie between [0,1] i.e the probability of being in a class.

Modelling

For the Spam detection problem, we have tagged messages but we are not certain about new incoming messages. We will need a model which can tell us the probability of a message being Spam or Not Spam. Assuming in this example , 0 indicates — negative class (absence of spam) and 1 indicates — positive class (presence of spam), we will use logistic regression model.

```
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score

Spam_model = LogisticRegression(solver='liblinear', penalty='l1')
Spam_model.fit(message_train, spam_nospam_train)
pred = Spam_model.predict(message_test)
accuracy_score(spam_nospam_test,pred)
```

So, first we define the model then fit the train data — this phase is called training your model. Once the training phase is finished we can use the test split and predict the results. In order to check the accuracy of our model we can use accuracy score metric. This metric compares the predicted results with the obtained true results. After running above code we got 93% accuracy.

In some cases 93% might seems a good score. There are a lot other things we can do with the collected data in order to achieve more accurate results, like stemming the words and normalizing the length.

Summary

As we saw, we used previously collected data in order to train the model and predicted the category for new incoming emails. This indicate the importance of tagging the data in right way. One mistake can make your machine dumb, e.g In your gmail or any other email account when you get the emails and you think it is a spam but you choose to ignore, may be next time when you see that email, you should report that as a spam. This process can help a lot of other people who are receiving the same kind of email but not aware of what spam is. Sometimes wrong spam tag can move a genuine email to spam folder too. So, you have to be careful before you tag an email as a spam or not spam.

Reference :

1. Kaggle Spam Detection Dataset
2. Github Repo
3. NLP — Topic modelling
4. Gmail Spam detection

Thanks to Yu Zhou.