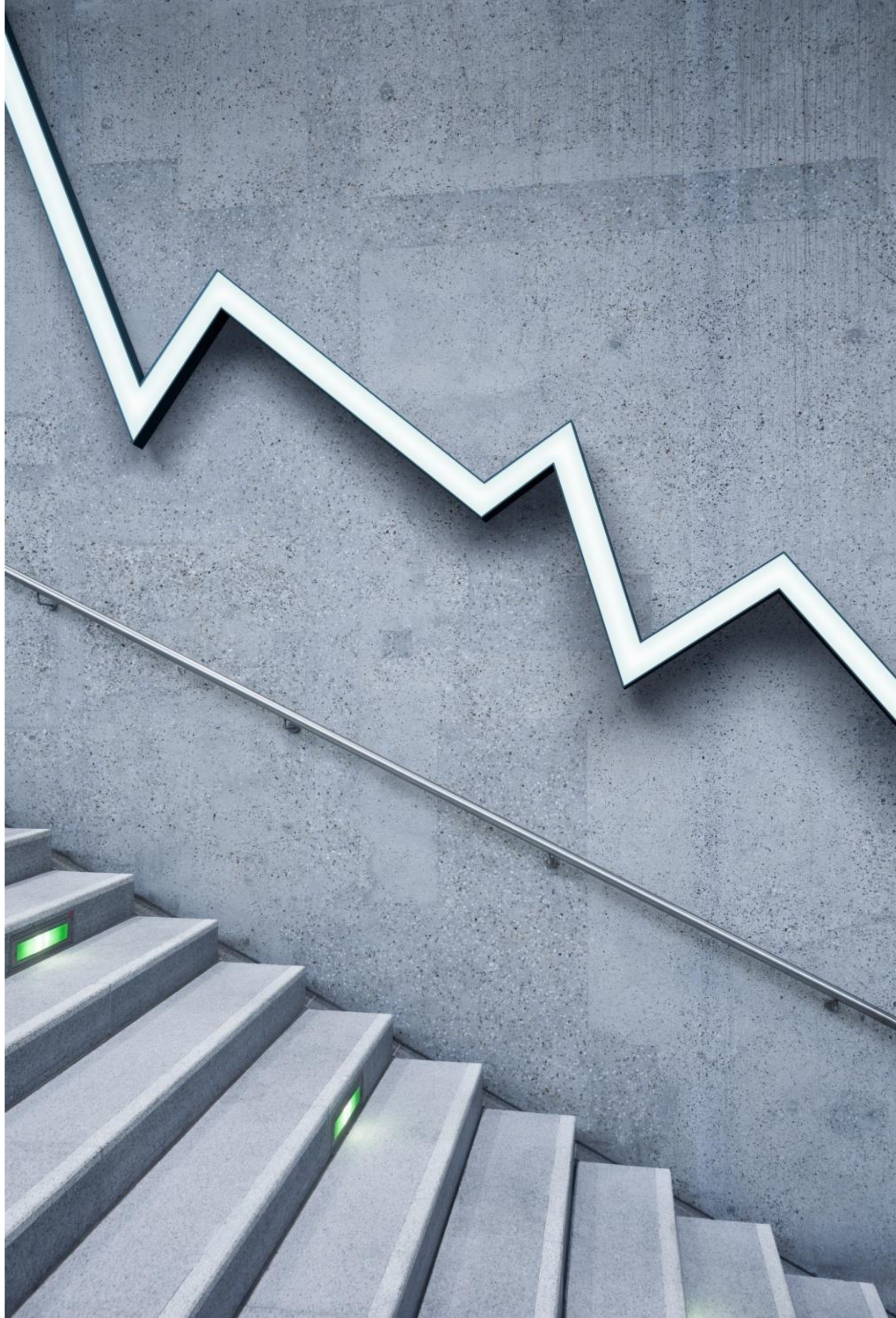
Spam Detections

NLP PROJECT



Abstract

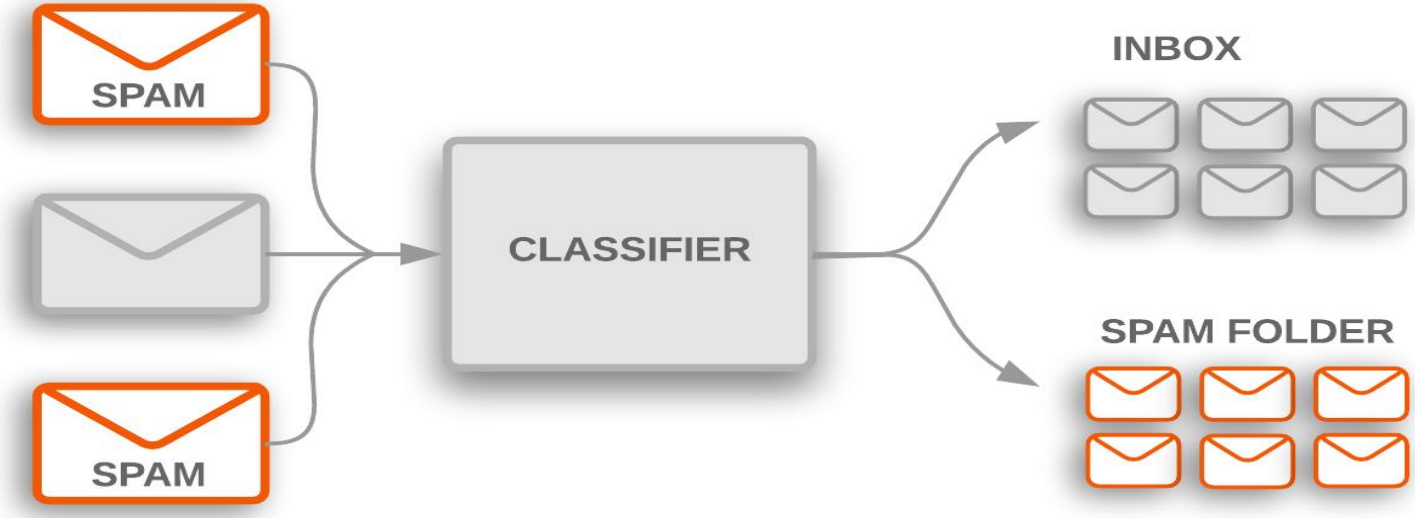
In our daily life, each of us receives many emails, some of which are useful, and we need in our work or communication processes, but some of them may come as annoying emails that contain fraudulent words.

Fraudulent Words

- “send us your password for specific account”
- “send us your account for some paid websites name”
- “send us your credit information to review your product”
- “connected with use on this link to know more information”
- “your account has been restricted to continue using our product download the file attached with this e-mail and update your login information”

Solution

Our solution is making a model that classified emails it will be spam or non-spam e-mail.



Naive Bayes Classifiers

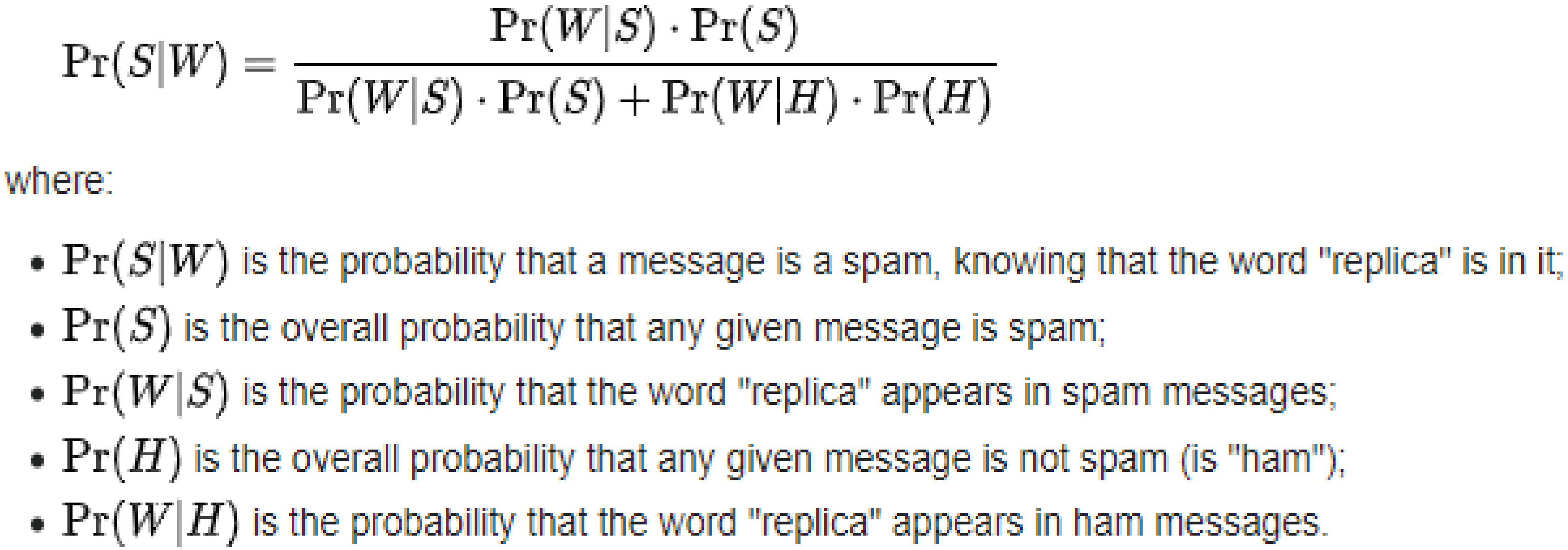
- We will make our spam filtering via Naive Bayes classifiers in order to predict whether a new text message can be categorized as spam or not-spam .
- Naive Bayes classifiers, a family of classifiers that are based on the popular Bayes “probability theorem” are known for creating simple yet well performing models, especially in the fields of document classification and disease prediction.

Naïve Bayes Features

1. Naïve Bayes classifiers are linear classifiers that are known for being simple yet very efficient .
2. Naïve Bayes are used in many different field, some example include the diagnosis of diseases and making decisions abut treatment processes , and spam filtering in e-mail .
3. Every value in Naïve Bayes is Independent form one another , Independent means that the probability of one observation does not affect the probability of another observation for example: variables is the popular coin tossing, The first coin flip does not affect the outcome of a second coin flip and so forth. Given a fair coin, the probability of the coin landing on "heads" is always 0.5 no matter of how often the coin if flipped .

What is Naïve Bayes Algorithm

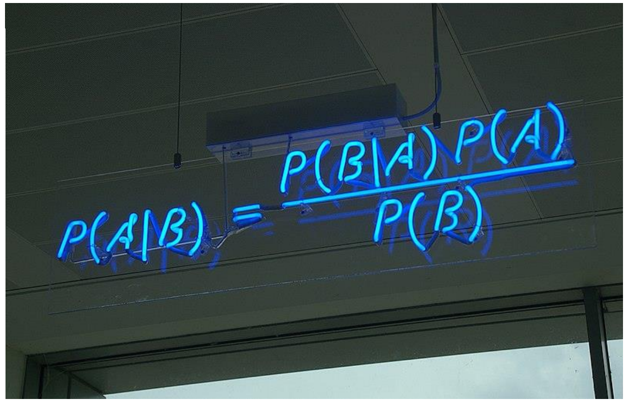
Let's suppose the suspected message contains the word "replica". Most people who are used to receiving e-mail know that this message is likely to be spam, the formula used by the software to determine that, is form Bayes' theorem



$$\Pr(S|W) = \frac{\Pr(W|S)\cdot\Pr(S)}{\Pr(W|S)\cdot\Pr(S) + \Pr(W|H)\cdot\Pr(H)}$$

where:

- $\Pr(S|W)$ is the probability that a message is a spam, knowing that the word "replica" is in it;
- $\Pr(S)$ is the overall probability that any given message is spam;
- $\Pr(W|S)$ is the probability that the word "replica" appears in spam messages;
- $\Pr(H)$ is the overall probability that any given message is not spam (is "ham");
- $\Pr(W|H)$ is the probability that the word "replica" appears in ham messages.

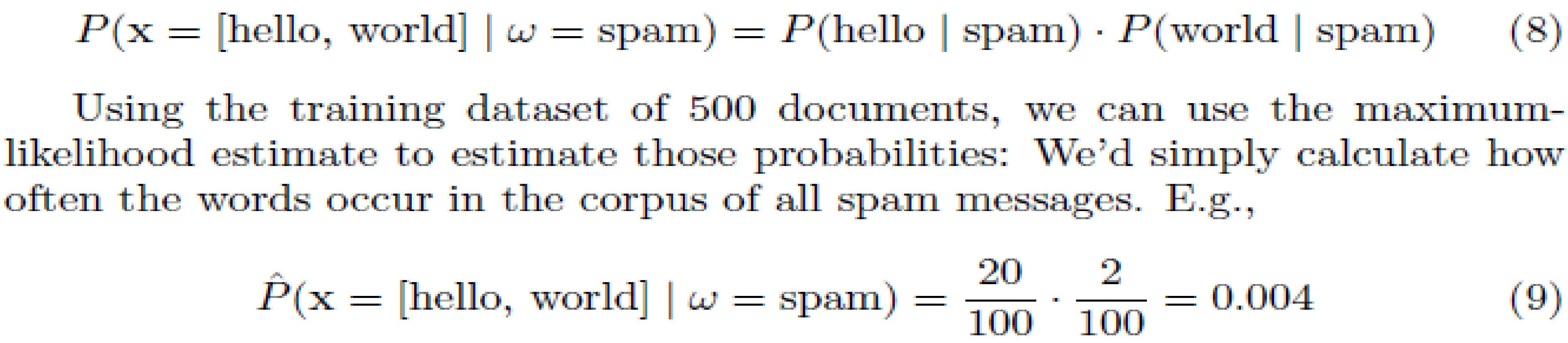


P(ham) = 1- P(spam)

Images and information’s are for Wikipedia

How Naive Bayes algorithm works?

Let's understand with an example, assume that we have a collection of **500 documents** where **100 documents** are **spam messages**. Now, we want to calculate the class-conditional probability for a new message "Hello World" given that it is spam. Here, the pattern consists of two features: "hello" and "world" and the class-conditional probability is the product of the "probability of encountering 'hello' given the message is spam" | the probability of encountering "world" given the message is spam."



$$P(\mathrm{x} = [\text{hello, world}] \mid \omega = \text{spam}) = P(\text{hello} \mid \text{spam}) \cdot P(\text{world} \mid \text{spam}) \qquad (8)$$

Using the training dataset of 500 documents, we can use the maximum-likelihood estimate to estimate those probabilities: We'd simply calculate how often the words occur in the corpus of all spam messages. E.g.,

$$\hat{P}(\mathrm{x} = [\text{hello, world}] \mid \omega = \text{spam}) = \frac{20}{100} \cdot \frac{2}{100} = 0.004 \qquad (9)$$

The Decision Rule for Spam Classification

In context of spam classification, the decision rule of a naive Bayes classier based on **the posterior** probabilities can be expressed as

- if $P(\alpha = spam \mid x) > P(\alpha = ham \mid x)$ classify as spam else classify it as ham .
- $P(\alpha = spam \mid x) = P(x \mid \alpha = spam) * P(spam)$
- $P(\alpha = ham \mid x) = P(x \mid \alpha = ham) * P(ham)$

The prior probabilities can be obtained via the maximum likelihood estimate based on the frequencies of spam and ham messages in the training dataset

- so, $P(\alpha = spam)$ = number of spam msg / number of all msg
- and $P(\alpha = ham)$ = number of ham msg / number of all msg

“Assuming that the words in every document are conditionally independent”

Thank You