Spam Recognition using Machine Learning

Lucas Hyatt - llh@uoregon.edu Olivia Pannell - olp@uoregon.edu

ABSTRACT

- Identifying dangerous messages is an ongoing battle within cyber security.
  - Malicious attackers can cause unwanted behavior through hidden links and misleading content in spam messages.
- Our objective is to create a functional and accurate classification system in which the user can detect if a message is spam or ham.
- This system will rely on multiple machine learning models for classification and prediction.

FEATURE EXTRACTION

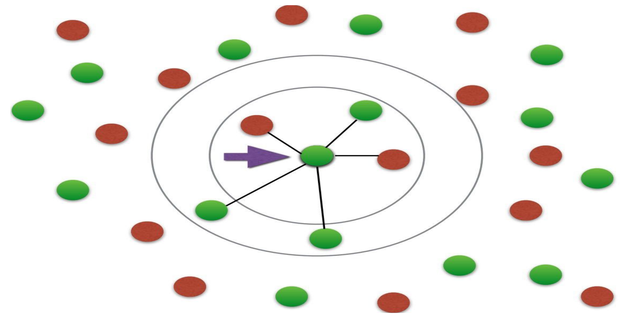
{ ‘free’ : 5, ‘money’ : 2, **‘or’** : 24, … , ‘buy’ : 1 }

{ ‘free’ : 5, ‘money’ : 2, … , ‘buy’ : 1 }

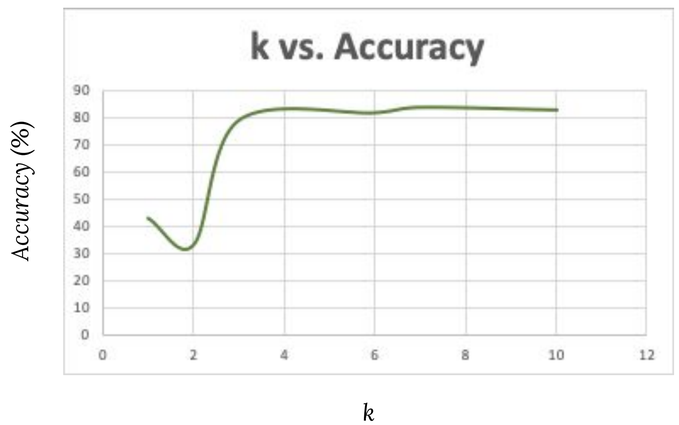
< 1, 1, 1, 0, 0, 0, 1, … 0, 1 >

- The goal of *feature extraction* is to have tangible data which we can use to train a model.
- Extracting features from spam messages:
  - Take a message (spam or ham) and count the occurrence of each word, storing values into a dictionary.
  - Remove stop words such as ‘and’, ‘or’, and ‘the’.
  - Convert the dictionary to a vector, where the occurrences or words represent the weights.

K-NN MODEL

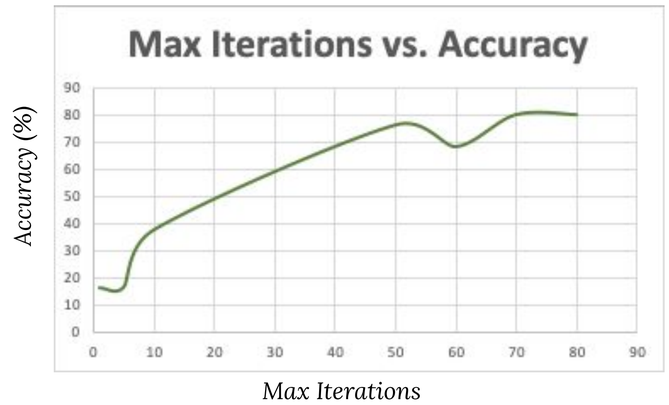


- The *K-Nearest Neighbor* model is an algorithm which observes available data in order to classify new cases based on similarity.
- Model was trained from a dataset of:
  - 80% ham and 20% spam email messages.
- Feature extraction was performed on the data to obtain a feature matrix containing feature vectors.
- The model relies on these feature vectors to classify test emails based on the *k nearest* feature vectors in the training matrix
- Similarity is determined by the euclidean distance between two features vectors.



PERCEPTRON MODEL

- The *Perceptron* model is an algorithm for supervised learning of binary classifiers.
- This model utilizes a binary classifier function to predict whether an email is spam (1) or ham (0).
- Input feature vectors are mapped to binary outputs using:
  - A set of weights (w) and a bias (b)
- Together, w and b form a *decision boundary* which can distinguish linearly separable data, like spam and ham messages.



- The algorithm can be adjusted through a hyperparameter called *max_iter* which represents the maximum number of iterations allowed for convergence of the perceptron algorithm to occur.
- The result is a function used to map a feature vector x to a single binary output:

  $f(x) = 1$ *if* $w * x + b > 0$, *or* 0 *otherwise.*

- All of the data we used for training belongs to Ion Androutsopoulos, Professor of AI at Athens University, Greece.