Spam Classification

Submitted By

Anita Tarangi

**INTRODUCTION**

* Business Problem Framing
* The main goal of these two parts of article is to show how you could design a spam filtering system from scratch.
* Conceptual Background of the Domain Problem

Machine Learning helps to design a spam filtering system.

* Review of Literature

The dataset had unwanted columns, cleaned the data, and renamed columns, Label Encoder was used to normalize the data, there were duplicate values present so removed the duplicated data. carried out different methods to train the model.

**Analytical Problem Framing**

* Identification of possible problem-solving approaches SVC, Decision Tree, Random Forest, XGB Classifier, MultinomialNB, KNeighbours Classifier, AdaBoostClassifier, Gradient Booster
* Data Sources and their formats
* There were no null values in the dataset.
* The dataset was not clean
* For some features, there may be values which might not be realistic. You may have to observe them and treat them with a suitable explanation.
* You might come across outliers in some features which you need to handle as per your understanding. Keep in mind that data is expensive, and we cannot lose more than 7-8% of the data.
* Data Preprocessing Done
* **Checking the size of the dataset**
* **Checking the summary statistics of the dataset**
* **Checking the data types and null values**
* **Dropping columns which aren’t important**
* **Used Label Encoder to normalize the data.**
* Hardware and Software Requirements and Tools Used
* Python, Pandas, Seaborn, Matplotlib, sklearn, nltk, Model Selection, SVC, Decision Tree, Random Forest, XGB Classifier, MultinomialNB, KNeighbours Classifier, AdaBoostClassifier, Gradient Booster

**Model/s Development and Evaluation**

* Identification of possible problem-solving approaches (methods)
* SVC, Decision Tree, Random Forest, XGB Classifier, MultinomialNB, KNeighbours Classifier, AdaBoostClassifier, Gradient Booster
* Run and evaluate selected models

SVC, Decision Tree, Random Forest, XGB Classifier, MultinomialNB, KNeighbours Classifier, AdaBoostClassifier, Gradient Booster

* Visualizations

Seaborn, Matplotlib, nltk

* Interpretation of the Results
* Carried out Analysis as well as visualization of the Dataset, treated imbalanced dataand then, trained the model using SVC, Decision Tree, Random Forest, XGB Classifier, MultinomialNB, KNeighbours Classifier, AdaBoostClassifier, Gradient Booster, model predicted with different accuracy for every model, out of all the models, Random Forest, Logistic Regression, have achieved with 96% accuracy rate.

**CONCLUSION**

* Data exploration, cleaning, visualization is the basic steps, inconsistent data was identified and treated. And then, trained the model using SVC, Decision Tree, Random Forest, XGB Classifier, MultinomialNB, KNeighbours Classifier, AdaBoostClassifier, Gradient Booster, model predicted with different accuracy for every model, out of all the models, SVC has achieved with 97% accuracy rate.