Foundation of Machine Learning – Assignment 2

Multi-Class Email Classification Challenge - Team name - '**Nico-Vanshi-Liu-Geralt**'

AN CONG JIE
Master in Data Science and Business Analytics (M1 Student)
ESSEC Business School and CentraleSupélec
congjie.an@student-cs.fr

LIU DAYU
Master in Data Science and Business Analytics (M1 Student)
ESSEC Business School and CentraleSupélec
dayu.liu@student-cs.fr

NICOLO' GIACOPELLI
Master in Data Science and Business Analytics (M1 Student)
ESSEC Business School and CentraleSupélec
nicolo.giacopelli@student-cs.fr

VANSHIKA SHARMA
Master in Data Science and Business Analytics (M1 Student)
ESSEC Business School and CentraleSupélec
vanshika.sharma@student-cs.fr

Problem Definition

We often face the problem of searching meaningful emails from thousands of promotional emails. This challenge focuses on creating a multi-class classifier that can classify an email into eight classes (namely, Updates, Personal, Promotions, Forums, Purchases, Travel, Spam, and Social) based on the metadata extracted from the email.

Dataset Features

The metadata extracted from the email contains information on date of the email, organisation of the sender, top level domain of the organisation, number of emails cced and bcced with this email, type of email (e.g. text/plain and text/html), number of images in the email body, number of urls in the email body, whether salutation is used in the email, whether designation of the sender mentioned in the email, and number of characters in the email subject and body. This information forms the features of our dataset, based on which we can train the models and thereafter make predictions.

Overview

Before applying any algorithms to solve this problem, we first perform some data processing, feature engineering, and feature selection in order to:

1. Fill in the missing values (i.e. we did missing values imputation) and data cleansing (i.e. we corrected the time zones of the date).
2. Transform all the data into numerical forms (integers or float) in order to fit in algorithms, for example, RandomForest Classifier. This has been done either by Onehot encoding the data or assign new features.
3. Try some new features derived from classification of the data based on the distributions we observed, so that data may better predict the result.
4. Out of all the features- the ones already existing and the ones we have created using feature engineering, we selected the most important ones minimizing the dependency between the selected ones.
5. After this, we tried 8 different models to predict the category of the email and checked their F1 scores based on splitting the training dataset into train and validation (test) sets. After optimizing the models, we uploaded the predictions from the respective models on the Kaggle Competition portal and choose the best model.

Feature Engineering

Here is the brief explanation of the observation and reasoning for the choice of selection for all of the features. In particular, we constructed two additional features date and tld:

Date

Firstly, we split the original date data into several readable parts in python, such as year, months, weekday, hour, time zones, etc. Thereafter, we imported the list of US holidays calendar and verified whether the emails are being sent on holidays or a working day. Our assumption is that emails related to work will not be sent on holidays. This check has been used as a feature in the final model.

tld

By looking at the distribution of the data within each label, we can conclude that “ac”, “co” and “in” probably have an impact on the predictions, while the rest of the class might have less impact. Hence, we keep only the significant classes, reassign all the others into “ rare” , and name blank spaces as “Unspecified”.

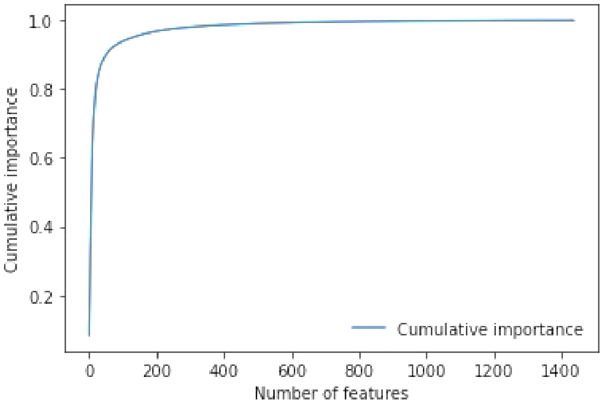
OneHot encoding and feature importance evaluation based on Random Forest

We used the OneHot encoding to convert all categorical data to numeric data and the function generated 1438 features.

We firstly made an evaluation on those features using Random Forest algorithm to get their feature importance.

Then, among the available ones, we selected those features with the most explanatory power.

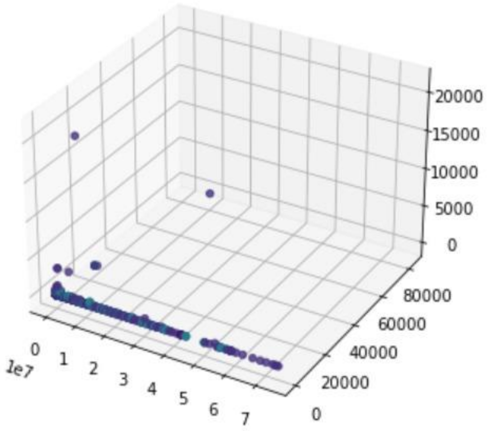
We did this by sorting them in descending order according to their feature importance and picking those sorting them by magnitude.



Standardization and PCA

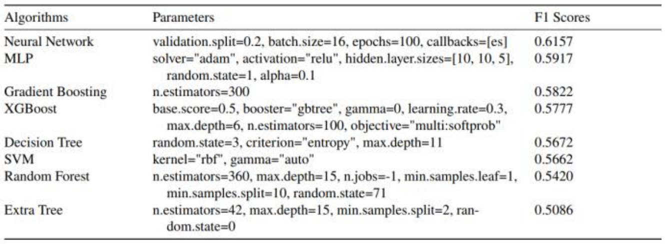
In order to improve our training speed and the accuracy of our models we decided to use standardization of all our features.

After the feature selection, we also used Principal Component Analysis to reduce the dimensionality of the feature space, as the graph shows below.



Model Tuning and Comparison

Table 1 – Models implemented with parameters and F1 Scores



Algorithms	Parameters	F1 Scores
Neural Network	validation.split=0.2, batch.size=16, epochs=100, callbacks=[es]	0.6157
MLP	solver="adam", activation="relu", hidden.layer.sizes=[10, 10, 5], random.state=1, alpha=0.1	0.5917
Gradient Boosting	n.estimators=300	0.5822
XGBoost	base.score=0.5, booster="gbtree", gamma=0, learning.rate=0.3, max.depth=6, n.estimators=100, objective="multi:softprob"	0.5777
Decision Tree	random.state=3, criterion="entropy", max.depth=11	0.5672
SVM	kernel="rbf", gamma="auto"	0.5662
Random Forest	n.estimators=360, max.depth=15, n.jobs=-1, min.samples.leaf=1, min.samples.split=10, random.state=71	0.5420
Extra Tree	n.estimators=42, max.depth=15, min.samples.split=2, random.state=0	0.5086

We tried 8 different machine learning algorithms (namely, Nueral Network, Multi Class Perceptron, Gradient Boosting, XGBoost, Decision Tree, SVM, Random Forest and Extra Forest) to make predictions about our multi-class classification problem. We used the GridSearchCV function and Loop function to determine the best parameters. We then used our F1 Score as indicators to evaluate. The results are shown above.

Evaluation

In our modelling process we found MLP and Neural Networks to be the best predicting models, basing ourselves on the F1 scores (respectively, 0.5917 and 0.6157). Therefore we finally chose Neural Network and MLP as our final choice of models.