Project Report

Background

I picked topic 3 for this project, it's about YouTube spam classifier. The reason for me to pick this topic because I use YouTube every day. I love the platform, but seeing spam comments on the video comment section or live chat comments is so annoying. I want to help the community by creating the comment spam classifier. This idea might open my eyes about AI, and I might develop more ideas to implement this classifier into a YouTube bot.

First things first, I have to check the given datasets. It consists of video ID, author (username who gives comments), comment date, comment text or content, and the label. The label itself has two possible values, 0 means that the comment is not a spam, and the others indicates otherwise. From here, I can see that my task is to determine whether the comment is a spam or not.

Method

There are multiple machine learning algorithms to choose, but in this project, I only choose one that can do the most out of it. After looking at the existing algorithms experiments online, I can see that random forest classifier has the best performance for this topic. This is the main reason why I pick the algorithm mentioned above. In addition, I also want to prove that random forest classifier will excel in this topic.

Random forest classifier consists of multiple decision trees, and it is also implements ensembled learning. Multiple decision trees are constructed, and each data that has been trained into the algorithm will help the AI to predict by voting and averaging. The advantage of using random forest instead of single decision tree is that random forest mitigates the overfitting issue from single trees. The ensemble approach increases the model's accuracy, robustness, and resilience to noise. Random Forest also provide a measure of feature importance, allowing us to identify the most relevant features for classifying comments.

Experiment

Before I jumped into the experiment, I have to know how to do proper data preprocessing. The TF-IDF (Term Frequency-Inverse Document Frequency) vectorization is one of the ways for data preprocessing. First, I need to analyze the useful information for the classifier. After looking at the dataset and relating it to the problem, the only important variable for the classification problem is the text values. Video is not necessary because any video can contain spam comments, so there is no point of looking for specific video. Author and date are also not needed for the problem due to same reason as the video. The text value may be varied, that's why I used TF-IDF to preprocess my dataset. The term TF measures each word frequencies in the YouTube comments. The other hand, IDF evaluates the rarity of one word across the entire dataset. Then, both information is used to calculate the TF-IDF score. Higher score indicates that a word is more crucial to the classification problem.

The transformation process is very helpful for the next step. Doing multiple experiments for finding the best parameters settings for the random forest classifier would be repetitive. Thus, I used GridSearchCV for the parameter tuning. With the transformation exists, I can effectively do the parameter tuning to determine the best value of n_estimators and max_depth of the random forest classifier.

The results are explained as follows. After performing the GridSearchCV, I decided to run the random forest classifier with the value of n_estimators = 175 and max_depth = 30. The result, in other words, the scores of each training and testing performance are:

Benchmarks	Training score	Testing score
Accuracy	0.989	0.95
Precision	1	0.955
AUC-ROC	0.99	0.95

Explanations:

- Accuracy indicates the correctness of the AI of making predictions.
- Precision indicates the ratio of true positives to the total predicted positives.
- A high AUC value signifies good overall discriminatory ability of the model.

As you can see from the result, random forest classifier performs very well to the related topic. For the current topic, I am very satisfied with the result. Further development may needed, where there might be future spams that is not related to the comment, but with the username. For example, let's say that one username is called "Free money in free.zoho.to", but the comment is normal. This might be a spam, that is why I might develop the model to be more advanced.

Extra notes

My python file is developed in jupyter notebook. To run it properly, you might need to change the value of the dataset filepath, because we might not have the same directory.

Reference

https://www.sciencedirect.com/science/article/pii/S2666827024000264