HACETTEPE UNIVERSITY

COMPUTER ENGINEERING DEPARTMENT

BBM465 2022 FALL

Assignment-4

January 3, 2023

Group Number: 37

Tuğrul ACAR
2210356144

Alper SOLMAZ
2200356039

Problem

In this project our aim is building a phishing web page brand classifier by using different feature descriptors along with leveraging two machine learning methods (Linear SVM, Random Forest).

Important Notes About the Assignment

- We used C# .NET 4.6.1 framework.

Solution

Image Descriptors:

Firstly we did not have enough information about ML algorithms and descriptors, so we firstly made researches about descirptors.There are two types of descriptor which are global (SCD, CEDD, FCTH,HOG) and local (SIFT, SURF).After spending a long time researching , we could not find too much information about the local descriptors so we decided that we need to use global descriptors , because they are much easy to implement and there are more resources and code examples about them.The first descriptor that is used ,is CEDD.

CEDD (Color and Edge Directivity Descriptor) is a image feature descriptor that can be used for image recognition and search tasks. This descriptor is based on the concept of color and edge histograms.It divides an image into a grid and calculates a histogram for each cell, taking into account the color and edge information in that cell. The resulting histograms are then concatenated to form a feature vector, which represents the image.One advantage of the CEDD descriptor is that it is relatively efficient to compute and can be used with large datasets. This is the reason why we choose CEDD.

After the calculation of features of CEDD ,we tried to write this data into a csv file, so we made research about the csv file format and examine at example csv files.After we made research we found that CSV file format is a file format used to store data in a tabular format, with each row representing a record and each column representing a field. Data is separated by commas, with each row ending in a newline character.

With the CEDD identifier we get a double array of size 144 for each image, we have 145 columns (144 indexes + 1 class labels) in the csv file and we write to the file line by line.

The next descriptor that we used is FCTH (Fuzzy Color and Texture Histogram). After the researches we found that FCTH descriptor is based on the concept of color and texture histograms.It divides an image into a grid and calculates a histogram for each cell, taking into account the color and texture information in that cell. The resulting histograms are then concatenated to form a feature vector, which represents the image. Advantage of this descriptor is that it is able to capture both color and texture information in an image, which can be useful for tasks such as object recognition and scene classification which may help us.This descriptor's result is a double array with size 192 and we wrote it into csv file.

The last descriptor that we used is SCD (Scalable Color Descriptor). This descriptor designed to be efficient and effective for a wide range of image sizes and resolutions. The SCD is based on the concept of color histograms. It divides an image into a grid and calculates a histogram for each cell, taking into account the color information in that cell. The resulting histograms are then concatenated to form a feature vector, which represents the image. Advantage of the SCD is its scalability, as it is able to handle images of different sizes and resolutions without requiring additional computational resources. It has also been shown to be effective for a variety of image recognition tasks, including object recognition, scene classification, and content-based image retrieval. This descriptor's result is a double array with size 256 and we write it to csv file with same method with CEDD and FCTH.

Machine Learning:

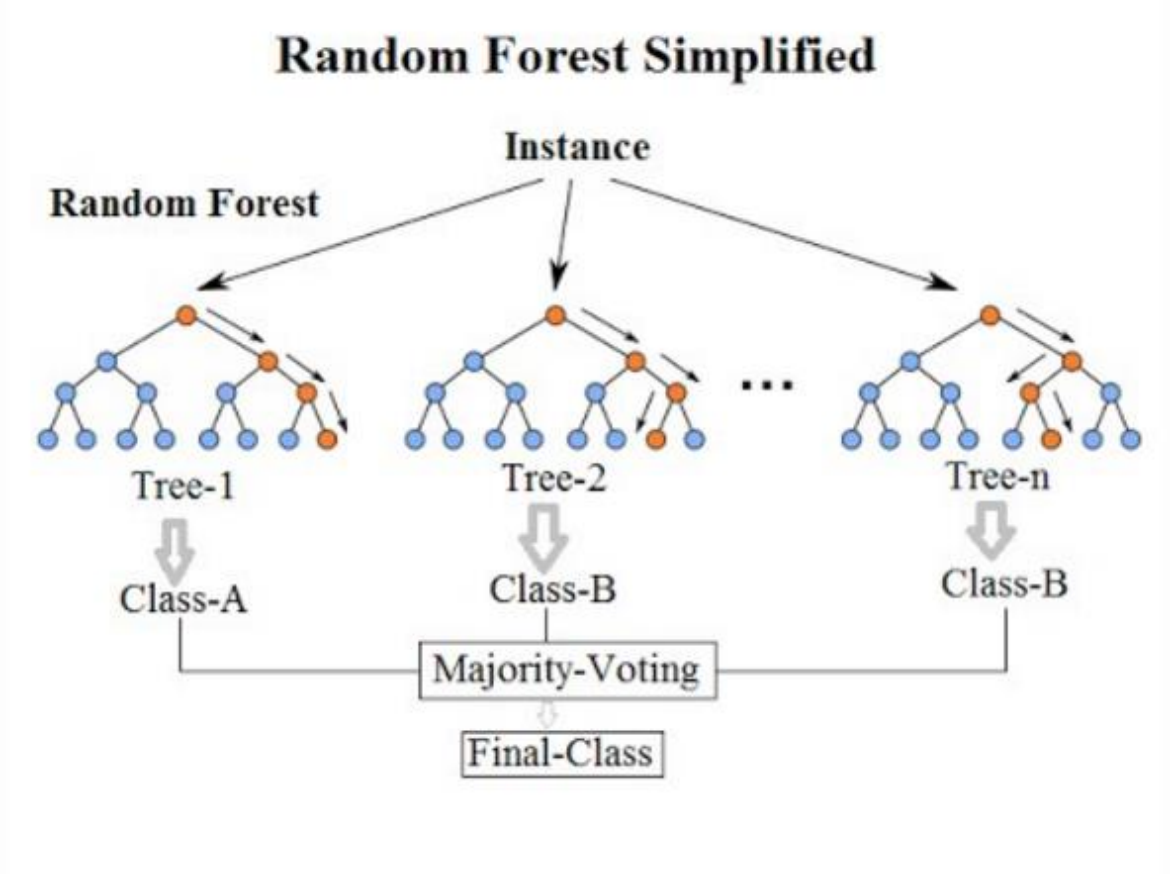
We used multiclass classification with 4 different ML algorithms Linear SVM, Regression Trees (RTree), Decision Trees (DTree) and Random Forest Tree. Except for Random Forest Tree, we used other algorithms from EmguCV and RFT from Accord.

Decision Trees (Dtree) is a supervised learning algorithm that can be used for both classification and regression tasks. It works by creating a model that makes decisions based on a tree-like structure, with an internal node representing a feature, a branch representing a decision, and a leaf node representing the output.

Random Forest (Rtree) is an ensemble learning method that combines multiple decision trees to make predictions. It works by training multiple decision trees on different subsets of the training data, and then averaging the predictions of these trees to make the final prediction.

Random Forests (RFT) is a variant of the Random Forest algorithm that uses bootstrapped samples and random feature selection to build decision trees. This helps to reduce the variance of the model and improve its generalization ability. Since this method uses multiple Dtrees, it gives better results than single Dtree.

Linear Support Vector Machines (SVM) is a supervised learning algorithm that can be used for both classification and regression tasks. It works by finding the hyperplane in a high-dimensional space that maximally separates the different classes.



Since we are not allowed to change parameters of ML algorithms some of them give poor results for example in RTrees Algorithm when we increase the MaxDepth parameter to 20 F1 value increase significantly.

Also, for DTrees we had to change 2 parameters (MaxDepth and CVFolds). Otherwise, we would get an error.

The slowest running algorithm among the 4 algorithms is the RTF algorithm. More than 95% of the time required for training is spent in training with RTF. To shorten this time, change some of the default parameters etc. we tried but the time is all about the size of the training set. For FCTH and CEDD, required time is around 50 seconds. But for SCD, this time exceeds 1 hour. When we reduce the train set to 180 rows to test that it works, it is completed successfully, and the train process takes about 140 seconds. However, as we understand, since the relationship between data size and train duration is not linear, the train process takes more than 2 hours with full train set. For this reason, we have converted the line that runs the data of the SCD with the RTF algorithm to a comment line so that the code can be run in a reasonable time.

We also encountered an exception like "System.Argument.Exception: 'Attribute 143 is a constans.'" in the RFT. This error was caused when multiclass classification using the "Random Forest" (RFT) algorithm with the "Accord.NET" library, due to the constant presence of the same values in one or more columns in the input data. This was due to

the fact that some columns in our dataset were completely 0. We tried to make several filters for this but failed. As a result, we randomly assigned a value of 1 to an entire row. In this way, at least one value in each column was nonzero. Since we thought that changing 1 row of data would not affect the overall performance of machine learning, we produced a simple solution in this way.

Conclusions

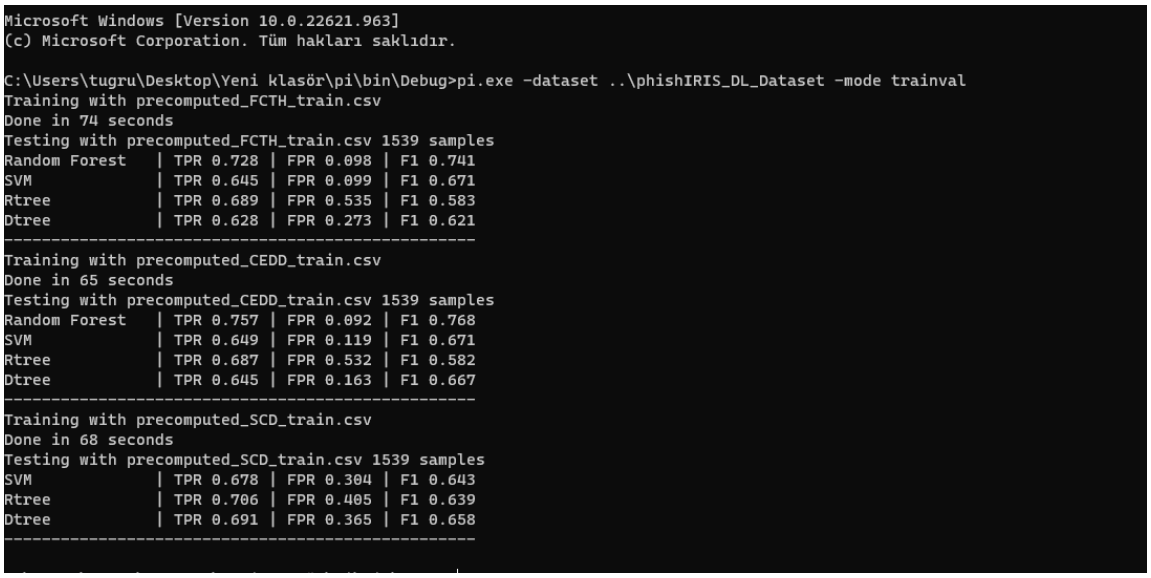


Figure 1 Results of program

As can be seen in the figure1, when the TPR, FPR and F1 values of the algorithms are examined, the results of our program do not come out at the desired levels. The main reasons for this can be listed as follows.

- If the parameters of machine learning algorithms were set correctly, there could be a significant increase in the performances.

- We processed the images directly using image descriptors without resizing the images in our dataset to a fixed size. If we had resized to a fixed size instead, we could have obtained more accurate feature data.

- If we used one of the local image descriptors (SURF or SIFT). We would have obtained more meaningful feature data. ML algorithms could perform better with this data.

Reference

https://rubikscode.net/2021/03/01/machine-learning-with-ml-net-random-forest/

https://towardsdatascience.com/data-science-tutorials-training-a-random-forest-in-r-a883cc1bacd1

https://www.analyticsvidhya.com/blog/2017/09/understaing-support-vector-machine-example-code/

https://www.javatpoint.com/machine-learning-support-vector-machine-algorithm

https://chatzichristofis.info/?page_id=15

https://dergipark.org.tr/en/download/article-file/610112