Machine Learning – Assignment 5 report

Problem Statement: Option 1 -Prediction of Empathy based on the features in young people survey dataset.

Dataset Pre-processing steps:

Empathy dataset had 150 features, varying from categorical to text features. So after selecting ‘Empathy’ as the target feature and rest of the feature as input, I encoded the text features eg (left-right handed, village-town etc.) as numerical data. As the dataset contained lot of missing values so I removed the missing values in the dataset with most frequent, and also normalized the dataset to scale the features like ( height (62-203) and age: (15-30)). Finally converted the features and output vector to array for processing.

Data Split: Data is split with train/dev/test in the ratio of 80/10/10.

Baseline Models:

Baseline models included Decision tree, KNN and perceptron. With the baseline models I got the development data accuracy in the range of (33-39). As the output vector (‘Empathy’) is multi-label with range [1-5] so a dumb classifier would have 25% accuracy that led to the low baseline accuracy.

Experimental Setup:

Initially trying different parameters for SVM led to the development accuracy of (32-33)% which led to accuracy even lower than the baseline models. As the data is very skewed towards label (3,4 and 5) (based on precision-recall) it led to very bad performance of SVM.

I also tried a voting classifier of baseline models and it led to an accuracy of (44-45)% which is better than the individual accuracy of the baseline models.

Naïve Bayes model reported an accuracy of (42)% which is better than the baseline models. As some of the features can be considered independent of each other could be reason that naïve bayes performed better in this task.

ML Solution:

I choose the random forest model for this dataset. As the random forest methodology reduces over fitting in the data also reduces variance by averaging over multiple trees and, so that led me to use this model. When compared with other models including (MLP, SVM, PCA with SVM, and naïve byes) random forest gave the best development accuracy within the range [48-49] and also giving better accuracy than the baseline models. To get better model of Random forest I tuned its hyper-parameter including (no of estimator and depth of the tree) and performed feature engineering. To get the best features for the model I performed L2 regularization (using linear SVM model and selecting its best features) on the model to avoid over fitting of the model.

Results

Baseline : [perceptron: (39) ; knn( 0.35) ; decision\_tree(0.33)]

Models tested :

MLP : (33-34) ; PCA\_with\_SVM: (33-34) ; Naïve Bayes: (42); Random Forest: (47-48)

Test Accuracy

Random Forest: (34-36)%

After Feature Selection and hyper parameter tuning:

Developmet Accuracy: (47-48)%

Test Accuracy: (34-37)%

Software : (Text editor: Sublime, Jupyter Notebook ; Librraries and packages: Sklearn, matplotlib). Jupyter notebook are very interactive to use and sklearn libraries have wide range of machine learning models with many parameters to tune for every model.