3/12/2019

PROJECT PROPOSAL

CSCI 6509

BHAVNEET KAUR SACHDEVA (B00809769)
GURJOT SINGH (B00811724)
GROUP P-12

PROJECT TITLE

Comparison of Accuracy in the Random Forest Algorithm and the Naïve Bayes Classifier to analyse the 2015 Canadian Federal Election.

MEMBERS

1. Gurjot Singh (CSID: gurjot)
2. Bhavneet Kaur Sachdeva (CSID: bsachdeva)

PROBLEM STATEMENT

An election is a formal group-decision making process by which a population chooses an individual to hold public office. Elections have been the usual mechanism by which modern representative democracy has operated since the 17th century [1]. The task will be to analyze the election results focussing on the 2015 Canadian Federal Elections using two different set of algorithms – Random Forest Algorithm and Naïve Bayes Classifier. Once we receive the results, detailed analysis will be calculated based on both the performance of both the algorithms on the same set of data. The results will also be displayed using a Visualization toolkit to better understand the comparisons.

To perform this, a dataset of 2015 Canadian Elections will be used to generate the Training and Testing Data for both the algorithms to generate the results and analyze which algorithm better performs the election analysis. The task will be to check whether both the algorithms can the analysis effectively and accurately.

The comparison will be focussed on the Accuracy, Recall, F-Score and the Precision of the algorithms. Once the results are compared detailed analysis of the elections will be performed.

POSSIBLE APPROACHES

We plan to implement this by two approaches. The first step is to calculate the accuracy using Naïve Bayes Classifier and the second is by using the Random Forest Classifier. We have the dataset from the 2015 Canada Elections [2] in which we have the results district wise along with the votes. We will be using this dataset to calculate the accuracy of both the algorithms.

Naïve Bayes Classifier and Random Forest Classifier are both supervised learning models. The main task of these classifiers is to classify the objects based on a certain set of features. A Naïve Bayes classifier is based on the Bayes Theorem probabilistic model. But, in this model, the predictors are supposed to be independent of each other. Dependency between the predictors is not taken into consideration in this model. In the Random Forest Classifier, a forest of decision trees is created, and it is in a random structure. Hence, the main idea is that it combines multiple decision tree outputs and then provide the accuracy which is expected to be more stable and accurate. These two algorithms will be the baseline for our project.

First, we will clean the dataset to remove all the extra columns and to verify the format of all the columns of our dataset. Then, we will use this dataset, to train our Naïve Bayes model by dividing this dataset into the training and test data. Using this, we will calculate the Accuracy, Recall, F-Score [5] and the Precision of our model. We will then again use this dataset on a different model, Random Forest Classifier, and calculate the Accuracy, Recall, F-Score and the Precision to generate the results.

Once we have the measures using both the algorithms, we will then compare the results and provide insights onto which one is better and how we can improve the performance by changing the feature sets used. The result we aim to achieve is to analyse the working of the two algorithms and to understand the performance of the two based on the different parameters provided.

Our project will be primarily focussed on Python Language using Scikit-learn library [3]. We will be using Panda Library to store and fetch the data.

PROJECT PLAN

Phases	**Duration**	**Work to be done**
Phase 1 – Data Collection and Preparation	13 March – 17 March	Collecting the Election Data, cleaning the data and preparing it for applying the algorithms. Generating the Training and Test Datasets for the models.
Phase 2 – Implementing the Naïve Bayes Classifier	18 March – 24 March	Creating a model using Naïve Bayes Classifier to analyze the election results and validating it on Test Data
Phase 3 – Implementing the Random Forest Classifier	25 March – 31 March	Creating a model using Random Forest Classifier to analyze the election results and validating it on Test Data
Phase 4 – Testing and Preparing the final Report	1 April – 7 April	Testing the accuracy of both the algorithms and comparing the results. Also, preparing the final report and presentation.

REFERENCES

[1] "Election", *En.wikipedia.org*, 2019. [Online]. Available: https://en.wikipedia.org/wiki/Election. [Accessed: 13- Mar- 2019].

[2] "Elections Canada - Official Website", *Elections Canada*, 2019. [Online]. Available: http://www.elections.ca/res/rep/off/ovr2015app/home.html#15. [Accessed: 13- Mar- 2019].

[3]"scikit-learn: machine learning in Python — scikit-learn 0.20.3 documentation", *Scikit-learn.org*, 2019. [Online]. Available: https://scikit-learn.org/stable/. [Accessed: 13- Mar- 2019].

[4]"Comparing Various ML models (ROC curve comparison) | Kaggle", *Kaggle.com*, 2019. [Online]. Available: https://www.kaggle.com/nirajvermafcb/comparing-various-ml-models-roc-curve-comparison. [Accessed: 13- Mar- 2019].

[5] J. Ramteke, S. Shah, D. Godhia and A. Shaikh, "Election result prediction using Twitter sentiment analysis," *2016 International Conference on Inventive Computation Technologies (ICICT)*, Coimbatore, 2016, pp. 1-5.