CE 314/887

Text classification

Report

Name: Litcan Nicolae-Gabriel

Registration Number: 1903165

The assignment’s topic is related IMDB sentiment classification. The dataset needed for the Assignment is provided as a Comma-Separated Values (CSV) file. While a specific classification method is not required for this assignment, it comes however with few rules, such as:

* Not using NB (Naïve Bayes) Algorithm which we used in Lab 4,
* We need to train our model with the first 40000 instances of the CSV,
* Test our model with the last 10000 instances.

Graphical user interface, text

Description automatically generated

The training and testing models are instantiated in the project like this:

I have used iloc() function that enabled me to select a specific cell of the dataset. Also, for training model I have selected first 40000 instances (“[:40000 , :]”), while for testing, last 10000 instances (“[10000 : , :]”).

Graphical user interface, text

Description automatically generated

**File reading**

In order to read the file I have used “**pandas**”, an open source data analysis and manipulation tool.

Graphical user interface, text

Description automatically generated

**Pre-processing the text**

For the text pre-processing I have used a stemmer, alongside a Regular Expression Tokenizer that will only allow only alphabetical words.

Text

Description automatically generated

Also, I needed to remove the stop words, so I found and altered after my need a solution from <https://www.geeksforgeeks.org/> (I can’t find the exact link).

All the uppercases were turned into lower cases and “tokenizer.tokenize” was applied. Then we remove the stop words from the sentence.

For now, the output will display a list of lists. I want to make a flat list out of the list of lists and I have found a solution on StackOverflow (<https://stackoverflow.com/questions/952914/how-to-make-a-flat-list-out-of-a-list-of-lists>).

Text

Description automatically generated

**Building a text classification model**

We use document\_features function in order to see whether or not the sentiment is positive.

Text

Description automatically generatedFor the classification model build I have decided to choose **DecisionTreeClassifier** provided by nltk.

**OUTPUT**

Text

Description automatically generated

As for the output, it will display the dataset from 10000th to 49999th instances.

Also, using Naïve Bayes – which wasn’t allowed for this assignment, I received an accuracy of 0.72ish (72%), while using a different classification method(in our case, Decision Tree Classifier I got 0.663325 (66%).

The performance of the model, in my opinion is pretty ok (better with NB Algorithm though) and I am happy that I managed to achieve this outcome.