
Assignment 3 (Due: April 24) (5 pts)

In this project, you will compare and contrast different classification algorithms we discussed in class by applying them to classify city traffic related short reports by people. These reports are to be routed to appropriate department depending on the issue type. Given the number of reports generated, manual inspection of the reports to determine their issue type (class label) is ruled out. You will use several classifiers provided by Weka (including Naive Bayes) to automate the process of routing the report to the concerned departments.

There are two steps in this project: (1) Setup and familiarize yourself with Weka, and (2) Use several classifiers in Weka to classify text reports.

The project may be done individually, or in a team of two or three members. All the team members are expected to contribute to all aspects of the project: design, implementation, documentation, and testing.

Phase I

Download Weka software (currently version 3.7.10) using information under Phase I resources. Choose the binary that is appropriate to your operating system and install a local copy. Explore Weka using the interface to view various options: available classification and clustering algorithms, input file formats, and support for processing text documents.

Exploration Guide

- Launch Weka and choose the *Explorer* option. A window pops up with *preprocess* tab being selected.
- Select *open file* option to load a file in one of the accepted formats. (See details in Phase II for dealing with text files.)
- After loading the file, you can run classifiers and clustering algorithms on the data by selecting the appropriate tabs.

Milestones for Phase I

- Setup Weka on your PC.
- Understand the concept of attributes and classes using Weka documentation.
- Go through the Weka GUI documentation.
- Familiarize yourself with Weka environment, specifically for text categorization.

Phase I resources

- Weka download page: <http://www.cs.waikato.ac.nz/ml/weka/downloading.html>
- Weka documentation: <http://www.cs.waikato.ac.nz/ml/weka/documentation.html>
- Weka Graphical User Interface (GUI): <http://prdownloads.sourceforge.net/weka/weka.ppt>
- Text categorization using Weka: <http://weka.wikispaces.com/Text+categorization+with+WEKA>

Phase II

Numerical data can be loaded for classification and clustering without any preprocessing. We describe working with text documents for your convenience below.

Step I

Download the dataset containing various city related reports from the link under Phase II resources. Extract the contents of the archived file. To simplify analysis and enable quick execution, we use only a small subset of the real-world dataset which has eight classes. The documents are organized in a directory structure where each directory name is the class label associated with the documents present in that directory. We have already provided the input dataset in this acceptable format to reduce your effort. Note that, in reality, you may need to preprocess the input into the format acceptable to Weka.

Preprocessing steps

- After you launch Weka, select *Explorer* option. A window pops up with *Preprocess* tab selected.
- Click on *Open file* and select the directory *city_events* extracted in the previous step.
- Weka returns with a warning asking you to choose a file loader.
- After selecting *OK*, you will find another window with a *Choose* option on the top.
- Select *TextDirectoryLoader*, and then *OK*.
- Select the *Filter* option and navigate to *unsupervised>attribute*, and then select *StringToWordVector*.
- Click on *Apply*.

Attribute Selection

After applying the filter (last step of the above preprocessing), you need to select attributes (features) to be used for classification. Numbers and special symbols may need to be ignored. Here we provide a simple example that considers only words as attributes (and ignores numbers and special symbols).

- In the *Attributes* window, select *Pattern* and copy paste the following regular expression in the text box
[a-z|A-Z]*
- Click on *invert* to select all the non-word items (including numbers and special symbols).
- Uncheck the first item in the attribute list @@class@@
- Select *Remove* option to remove all the non-word items from the attribute list.

Step II

After filtering the attributes, click on the *Classify* tab and use the choose option to select various classifiers. Before you run the classification algorithms, select the *Test Option* to be *percentage split* with a default setting of 66% training and the remaining for testing. You will run four different classification algorithms on the input data and compare their performance. (We have eliminated Decision Tree classifier because it seems to take prohibitively long time to complete.)

- Naive Bayes (bayes.NaiveBayes)
- Naive Bayes Multinomial (bayes.NaiveBayesMultinomial)
- Support Vector Machine (functions.LibSVM)
- k nearest neighbors (lazy.IBk where IB - Instance Based, k - number of neighbors to examine)

After selecting the classifier, press 'Start' to obtain classification summary.

Phase II resources

- Dataset Download: http://cecs.wright.edu/~tkprasad/courses/cs7800/asg-S14/city_events.zip
- Document classification in Weka: https://www.youtube.com/watch?v=z1VJ2_N_0lo
- IBk tutorial: <https://www.ibm.com/developerworks/library/os-weka3/>
- Text processing in Weka: <http://www.unal.edu.co/diracad/einternacional/Weka.pdf>

Deliverables

You will turn in all your work as a single zip-archive: **asg3.zip**. You need to report on the following comparative analysis.

- Present precision and confusion matrix for each of the four classification algorithms.
- Determine which one of the four classifiers is the best for the given task and why?

Turning in the Assignment: Upload the archive **asg3.zip** onto Pilot > Dropbox > Assignment 3 folder.

You are also expected to demo your program and be prepared to answer questions about its design, implementation, and evaluation.