AE4B33ZUI – Introduction to Artificial Intelligence

Contact: Radomír Černoch

summer semester 2009/2010

The deadline for this assignment is Sunday 16^{th} May 23:59. The deadline is firm.

Text mining (20 points)

The general task is to create and evaluate classifiers, which can assign a Usenet article written in English into one of the predefined discussion groups. The goal is to achieve as high accuracy as possible. Since you are free to use data-mining tools like Weka, the essential work will be the data preprocessing.

During the whole exercise we will use the *20 Newsgroups* dataset downloadable from `http://people.csail.mit.edu/jrennie/20Newsgroups`.

In general you are free to use any machine learning tools, applications or programming languages you like. However if you plan to deviate from the following guidelines, please consult your chosen approach first.

1. Download the Weka data mining tool from `http://www.cs.waikato.ac.nz/ml/weka`
2. Download the `.zip` archive from the course web page, which contains the correct version of the dataset.
3. Design your own preprocessing techniques, which would be capable of converting a textual document into the ARFF format used by Weka (a vector representation).

   The provided Java preprocessor constructs the vector model by going through all documents and taking the first 200 words. Please feel free to use the source code and modify it for your purposes.

   Each group is expected to try *at least 3 different ideas*. You are strongly encouraged to use your creativity over taking up well-known preprocessing techniques. Some ideas are already given in the source code.
4. Given your algorithms, preprocess the *training* and *testing* datasets.
5. Evaluate the preprocessed data in Weka. Use the same set of training examples for both the preprocessor and the classifier. You are supposed to test your data using
   - The Naive Bayes model. By looking at the conditional probabilities forming the Naive Bayes model, do the numbers make any sense to you?
   - A machine learning algorithm of your choice, which is expected to achieve better accuracy than Naive Bayes.

6. If the algorithms run excessively long, you can use “filters” both for reducing the number of training examples or the dimensionality.
7. Identify the the best preprocessor and the best classifier. Using this combination, estimate the possibility of over-fitting by 4-fold cross-validation. Report results of all 4 folds.

Assessment

The expected outcome of this assignment is a technical report, which would describe the techniques you used and the achieved results. Your report should include:

- An exact description of the preprocessing techniques you designed.
- Accuracy on each pair of preprocessing algorithm / classifier.
- Timing data both for the preprocessing and for the classification.
- Result of the cross-validation and discussion about the over-fitting.
- Comments on the “accuracy $\times$ time” for different algorithms.
- Pointing out interesting values in the Naive Bayes model.
- Any other interesting observation is most welcome.

The main criteria for assessing the report will be the clarity of description. Based on your text, anyone should be able to reproduce the result by reading the report and reimplementing the techniques. You can cite external sources to avoid repeating commonly known things.

You can work in groups of up to 5 people. All students in one group will be rewarded the same amount of points. In case you cannot form a group large enough to complete the whole exercise, please ask.

Please attach the source code you used for preprocessing. However the program will not be a part of the evaluation and the report should be self-explanatory.