Machine Learning 2013: Project 3 - Text Classification Report

anufer@student.ethz.ch
elmerl@student.ethz.ch
nivo@student.ethz.ch

December 23, 2013

Experimental Protocol

Usage:
Download the csv files to /data/3/....csv (... = training, testing, validation)
Run map.m
Results are in /data/3/....out (... = training, testing, validation)

1 Tools

- C#, LINQ, Visual Studio 2012 Ultimate (code is in /code/ directory)
- Matlab (code is in /code/ directory)
- Git / Github Repository [1]

2 Algorithm

Principal component analysis

Describe the algorithm you used for classification.

3 Features

To group similar words together, reprocessing was used. First, we used the Levinsthein distance[2], but then switched to a slightly modified version of the edit distance[3], which gave slightly better results. Also, instead of using an absolute value threshold for the distance, we used a ratio of about 75% to group similar words together. For this preprocessing, we used C#, because string handling in C# seemed

[1] https://github.com/lukaselmer/ethz-machine-learning

[2] https://en.wikipedia.org/wiki/Levenshtein_distance

[3] https://en.wikipedia.org/wiki/Edit_distance

much easier than in Matlab.

After the preprocessing, we then used Matlab to predict the city codes and country codes. For this, we used PCA.

4 Parameters

To find the parameters, we used manual testing. The most important feature seemed to be the edit distance ratio, which is 75%. Another important parameter is the amount of top words which are picked at the start of the algorithm. These are the words, which are in a category of it's own before grouping them together.

5 Lessons Learned

Many other tools and algorithms have been tried. However, one we didn't try and might have worked well is SVM.