INNOPOLIS UNIVERSITY, DEPARTMENT OF COMPUTER SCIENCE

Language identification

Kamil Salakhiev

November 24, 2016

1 PROBLEM DESCRIPTION

Language identification is an important task when the source language is unknown. Unfortunately, today we have to process a lot of data that is not labeled with some language or even worse, it is labeled with the wrong language. In that assignment we will test different models in order to obtain the most accurate one.

2 EXPERIMENT

To perform experiment twitterLID dataset was exploited. The dataset was uploaded and converted by pickle to store it on disk. In order to pass data into classifier, we converted text into numbers by CountVectorizer and TFIDF transformer from sklearn library.

To evaluate the quality of the models *precision, recall* and *f1-score* were estimated. As the baseline Multinomial Naive Bayes classifier with word features has been chosen, which gave the following performance on different languages.

	precision	recall	f1-score	support
german	0.953	0.911	0.931	313
english	0.913	0.910	0.912	334
spanish	0.925	0.930	0.928	359
french	0.984	0.924	0.953	328
dutch	0.859	0.933	0.894	386
avg/total	0.924	0.922	0.923	1720

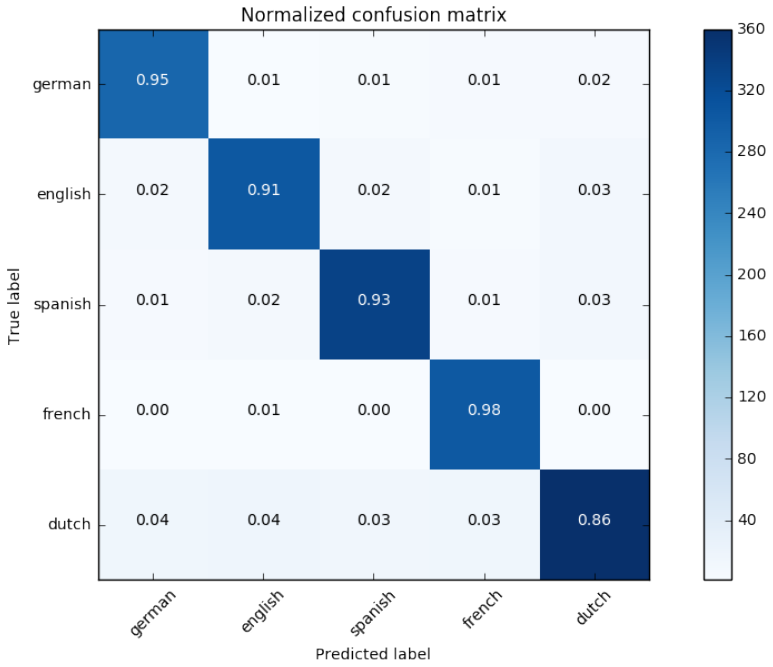

Figure 2.1: Confusion matrix for Naive Bayes classifier with word features

Confusion matrix for this model is presented in fig. 2.1:

Another models gave the following average f1-scores (average f1-score is the average of f1 score for every label prediction):

1. *Multinomial Naive Bayes with character bigrams*: 0.878
2. *Multinomial Naive Bayes with character trigrams*: 0.913
3. *Multinomial Naive Bayes with character 4-grams*: 0.878
4. *Multinomial Naive Bayes with word bigrams*: 0.566

Note, that for every model we applied grid search, varying alpha parameter, which is smoothing parameter.

After selection aforementioned features, we stopped on initial features, where we just splitted sentence on words. So, we continued with selecting the best model:

1. *SVM*: 0.901
2. *Ada-boost with decision tree classifier*: 0.781

Here we also applied grid search technique. For SVM we varied kernels, C parameter and gamma parameter. For AdaBoost we tried different estimators.

After that, I gave up on selecting model and stopped on Naive Bayes. And now we started to extract the best features from dataset by sklearn tool RFECV(recurrent feature elimination and cross-validation). And this actually worked. We obtained the following scores:

	precision	recall	f1-score	support
german	0.960	0.911	0.934	313
english	0.914	0.919	0.916	334
spanish	0.923	0.930	0.926	359
french	0.984	0.924	0.953	328
dutch	0.866	0.935	0.899	386
avg/total	0.927	0.924	0.925	1720

The confusion matrix for that model is presented in fig 2.2

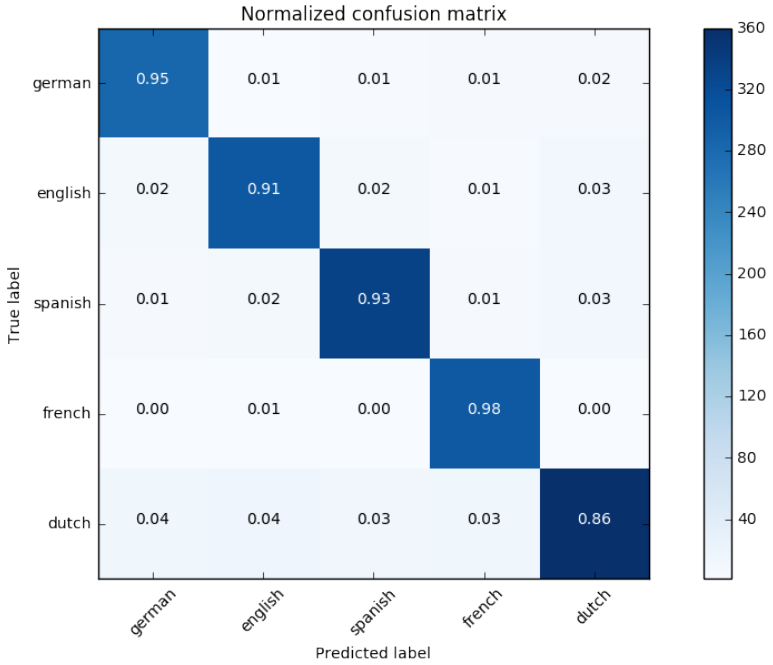

Figure 2.2: Confusion matrix for Naive Bayes classifier with feature selection

3 Conclusion

Recall, that for every model we applied grid search to tune parameters, but still we could not beat Naive Bayes with word features. The only improvement, that we could achieve was

feature selection, although it increased f1 score only by 0.002.

Note, that according to confusion matrices there are no any pair of languages that confused too frequently with each other.