Master's thesis research proposal:
Document level author clustering and authorship linking for the PAN 2016 shared task

Olivier Louwaars, s2814714

March 12, 2016

1 Introduction

With the publishing and sharing of documents being accessible to everybody, the need to verify what was written by who becomes apparent. Not only to prevent plagiarism, but also to prevent texts from being attributed to an author they do not belong to. Since 2011, PAN[1] contains a shared task regarding automatic authorship identification. Where in recent years the task focussed on determining whether or not a certain document belongs to a set of known documents of an author, the 2016 task is to cluster documents per author, without knowing the number of contributing authors. This task asks for a radically different approach, and offers new technical usages. If the results of this shared task are satisfactory, the method can be applied to a variety of fields (e.g. ???) Additionally, the task also comprises a second step, in which links have to be established between the different documents within a cluster/author. This second step is comparable with the earlier shared tasks, as it is a one on one comparison of documents. Depending on the similarity of two documents, the certainty of the author of both can be established. With this second step included in the method, the first could be used as a first separator to make an initial division in a large collection of documents. This saves a lot of time as otherwise every document in the set would have to be compared to all the others. To make sure both steps are executed properly and no work is duplicated, the following research question will be the foundation of this research:

"How can randomly mixed documents be clustered per author?"

To back up the main questions, the following sub questions can be formulated:

- What features are important for the initial clustering per author?
- What features are important for the one on one comparison?
- How can the system be prevented from clustering based on obvious but wrong patterns such as topic?

2 Related work

As mentioned before, PAN shared tasks have been around for a while now, focussing on author identification between two documents. For this research,

[1] http://pan.webis.de/

the overview of last years shared task provides the most actual information about features and approaches (Stamatatos, et al., 2015). From the overview, the best method from last year can be adapted to be used for the second part of this year's task. Stamatatos et al mention a recurrent neural network as best performer, but this should be done at the character level for the best result. The clustering itself, or the first part of the task is less documented, and previous work is not comparable.

3 Method

3.1 Data

The data for this task is provided by PAN, consisting of 18 clustering problems. Every problem can be either in Dutch, English or Greek, and can be a news or a review article. The type and language of every problem is provided, so language dependent parameters can be set for the clustering. Every problem is made up of between 50 and 100 texts, from an unknown number of authors. Every text is between 3.000 and 4.000 characters (700-800 words) long and is written by a single author, but can have a different topic from others in the same cluster. Two truth files are provided in JSON, showing both the real clusters and the authorship certainty per cluster and its documents.

3.2 Approach

The initial clustering task is the hardest to tackle, as there is no clear previous research on that topic. This makes it harder to determine the features necessary for the best result. The clustering algorithm on the other hand is easier, as Python and its scikit-learn package are familiar tools that offer exactly what is needed. Scikit-learn's flow diagram[2] leads to a MeanShift or VBGMM algorithm, both unfamiliar but well documented. Using one of these clustering methods as a first step, the documents within every cluster then have to be compared. As described in last year's summary, a recurrent neural network offers the best options. The implementation for this in Python will be via Theano and Keras modules, both easy to use but with very powerful C++ implementations for speed.

4 Evaluation

The evaluation of the clustering will be done on the JSON truth files that are provided, generating an objective score to work with. Based on the influence on the score, several features will be tested and evaluated, thus reaching an optimum in F1 score, precision and recall. Once the results on the development

[2] http://scikit-learn.org/stable/˙static/ml˙map.png

set are set, PAN provides an online code testing environment on the test data, which is not made available any other way. By uploading the system, its performance on the unseen test data is established and can be compared to all other submissions equally.

5 References

Stamatatos, Efstathios, et al. 2015. Overview of the Author Identification Task at PAN 2015. [book auth.] Linda Cappellato, et al. *CLEF 2015 Labs and Workshops.* Toulouse : Notebook Papers, 2015.

6 References

There are no sources in the current document.