Machine Learning: Assignment 04

Text Classification in Scikit-Learn

The aim of the assignment is to investigate how machine learning algorithms in Scikit-Learn can be used to carry out a text classification task. The task is to classify author profiles as female or male, a task referred to as *gender identification*.

The dataset to use is *PAN-AP-16 Twitter Corpus*, which is attached with the assignment. You should download and unpack this dataset. In a nutshell, the data consists of 426 author profiles, 215 female profiles and 211 male profiles. One file in the corpus contains one author profile. The tweets in each profile are in plain text i.e. already pre-processed. The female profiles are stored in one subdirectory and male ones in another.

To further understand the corpus, you may find the following paper of relevance:

**Rangel, F., Rosso, P., Verhoeven, B., Daelemans, W., Potthast, M., & Stein, B. (2016). Overview of the 4th author profiling task at PAN 2016: cross-genre evaluations. *Working Notes Papers of the CLEF*.**

Available at: <http://www.uni-weimar.de/medien/webis/publications/papers/stein_2016l.pdf>

In this paper, the creators of the data explain how they have used Twitter to create this dataset. Also, different gender identification techniques are discussed.

The assignment is to carry out the two tasks below and write a short report about what you have done. There is an optional third task which you may attempt for extra credit (to a maximum of 100% on this assignment).

# Task 1: Feature Extraction

Before you can run the Scikit-Learn classifiers on the PAN-AP-16 Twitter Corpus, features must be extracted from the texts and presented to Scikit-Learn in CSV format. You will need to extract features using two methods: (1) Stylometry based Methods and (2) Content based Methods.

The following paper will be helpful for you in understanding stylometry based and content based methods for gender identification.

**Fatima, M., Hasan, K., Anwar, S., & Nawab, R. M. A. (2017). Multilingual author profiling on Facebook. *Information Processing & Management*, *53*(4), 886-904.**

Available at: <http://www.sciencedirect.com/science/article/pii/S0306457316302424>

# Feature Extraction – Implementing your own code for stylometry based methods

Write a script to automatically extract following 20 *stylometery based features* from the PAN- AP-16 Twitter Corpus.

1. Number of commas
2. Number of full stop
3. Number of @
4. Number of (
5. Number of )
6. Number of !
7. Number of –
8. Number of ?
9. Number of %
10. Number of &
11. Number of #
12. Number of \_
13. Number of =
14. Number of capital letters
15. Number of semi colons
16. Number of colons
17. Number of spaces
18. Number of small characters
19. Number of forward slash (/)
20. Number of digits

# Feature Extraction – Using Scikit-Learn’s implementation of Content based methods

* **Simple unigram representation**. Form the set *V* of words occurring in the reviews. Create a binary feature corresponding to each word in *V* and for each review (instance) set the value of this feature to 1 if the word occurs in the review and to 0 otherwise. To convert text files in the PAN-AP-16 Twitter Corpus to CSV files please see *Lecture 07* of *Machine Learning* course.
* **N-gram representations.** Above mentioned feature representation considers isolated words as features, ignoring word order and word co-occurrence. Instead of, or in addition to, single words one could also consider word pairs (bigrams), triples (trigrams), etc. so-called *n-grams*. The total of number of these explodes very rapidly as **n** grows, so some means of filtering n-grams to select only the most discriminating is probably essential.

For this assignment, you will need to apply content based methods for both word n-gram models and character n-gram models. The length of *n* should vary from 1 to 3 for word n-grams models, whereas *n* should vary from 3 to 10 for character n-gram models.

# Feature Selection/Reduction

The stylometry based methods and content based methods (simple unigram or n-gram representations) may yield too many distinct features1, many of which are of little use in discriminating between the two text categories. A variety of approaches can be taken to reduce the feature set further. Essentially, they all involve ranking the features according to some measure and then discarding low ranked features. For details of feature selection methods in Scikit-Learn, see the link below.

<http://scikit-learn.org/stable/modules/feature_selection.html>

You can apply any of the feature selection methods for both stylometry based methods and content based methods.

# Task 2: Text Classification

Having derived feature-based representations for the texts in the dataset, the next task is to run and compare various classifiers over the data. Pick at least two machine learning algorithms, including Random Forest, and run an experiment to compare these algorithms.

# Task 3: Text Clustering (Optional)

Having completed tasks 1 and 2, the optional task is to investigate the clustering capabilities of Scikit-Learn. Can Scikit-Learn cluster the author profiles into male and female profiles, without the benefit of knowing their classification? To explore this question run the Scikit-Learn *Agglomerative Clustering* and *K-Means Clustering* algorithms and see to what extent these algorithms can discover clusters that match the male/female categories.

# Hand-in and Assessment

The hand-in consists of two parts: your code for task 1 and a report.

1. **Code:** Place all files comprising your code into an archive file (a tar, jar or zip file) and submit this file. Please do not include the data as well. In addition to your source code files(s) this archive should include (1) a README file which explains how to compile/run it/them; (2) the CSV files you have produced by running your program over the training data (3) an electronic copy of your report (in .pdf format).

1 Note that in content based methods each n-gram is a feature. For example, if you are using unigram representation, then each unigram (or single word) will be a feature. Similarly, if you are using a bigram representation, then each bigram (two words) will be a feature and so on.

1. **Report:**  A report of not more than 4 sides of A4 (do not use font sizes less than 11pt) in which you discuss:
   * the features and feature selection method you have chosen;
   * the outcome of feature extraction (number of features, observations about the features);
   * the setup of the classifier experiment(s) you have run (which algorithms you have used and any parameter settings);
   * the outcome of the experiments;
   * any observations you might have about why you obtained the results you did and how performance might be improved.

If you attempt Task 3 you may hand in one more side of A4, however, that side should address Task 3 only (i.e. you report for Tasks 1 and 2 should fit on 4 sides of A4 only).

Regarding this optional additional task you might address:

* The relative performance of the two algorithms with respect to the task of recovering the gender distinction.
* Any observations about the relative utility of different feature sets for clustering, if you have compared different feature sets in tasks 1 and 2.
* How can Scikit-Learn be used to understand clusters the clustering algorithms may have found?

Assessment will be based primarily on the quality of your report, as it reflects the work you have carried out and your analysis of the results. Experimentation with varying approaches to feature reduction or their parametrization will gain extra marks. While the report is of primary importance, your code may be checked to ensure it runs and failure to do so will result in a lower mark.

The assignment is due by **Sunday, May 09, 2021 at 23:59 that will not be extended in any case**, and should be submitted as follows. Note that there will be a deduction of **5% marks (per day)** on late submission. Departmental rules concerning plagiarism and collusion will be strictly observed – please refer to the Student Handbook for details of these.