CS-E4650 Methods of Data mining
Project work

Deadline Thu 23.12. 2021 23:30

Overview

The task is to preprocess and cluster text data using methods learnt in the course. Everybody should try at least two approaches: a basic approach and a more advanced approach. Finally, the contents of the best clustering are analyzed to determine the cluster topics.

The project work can be done either alone or in pairs (i.e., groups of 2 course participants).

The maximum points are 20p (or scaled to 20% of the grade). Note that project work points are in the different scale than the assignment points. Points will be given for comprehensiveness of experiments, achieved results and the report. You will get two extra points, if you obtain $NMI \geq 0.81$ (should be reproducible).

Data

Load data *abstractdata5.csv*. The data consists of abstracts of scientific papers, together with paper titles and class numbers (reflecting the topic category). The data format is the following:
id#class#title#abstract
where # is a field separator, *id* is a document identifier, *class* is the class number 1–5 (reflecting the topic), and *title* and *abstract* are text fields.

Analysis task

1. Try a baseline approach:

    a) Combine title and abstract, remove stopwords, and stem the remaining words.

    b) Present the data in the tf-idf form, where each individual word is one feature (basic bag-of-words model). You can decide the tf-idf variant freely, but remember normalization since Euclidean distance will be used.

    c) Cluster the data with K-means using $K = 5$ clusters (number of classes) using Euclidean distance.

d) Evaluate clustering quality using **normalized mutual information** (*NMI*), the version by Strehl and Ghosh (2003), with **geometric mean in the denominator**. Given clustering $C_1, \ldots, C_k$ and classification $D_1, \ldots, D_q$,

$$NMI = \frac{I(C, D)}{\sqrt{H(C)H(D)}},$$

where $I(C, D)$ is mutual information between clustering and classification and $H(C)$ and $H(D)$ are entropies of clustering and classification respectively.

2. Try to get a better clustering result (measured by *NMI*) using more advanced techniques introduced in the course. Everybody should try at least one more clustering method in addition to K-means. In addition, you can try to optimize preprocessing and test different feature extraction, dimension reduction and feature selection approaches. Here you can use any distance or similarity measure.

3. Choose the best clustering (with highest *NMI*) and try to determine the main topics of clusters. The basic approach is to identify relevant keywords among the most frequent words or n-grams in clusters (or cluster centroids) and infer the topics from them. If you want, you can try also other techniques to determine important keywords.

In the implementation, you can use Python, C, C++, Java, R, Matlab or Scala and available libraries (describe in the report).

Report

Write a report on your experiments and results. Maximum length is 3 pages + possible appendices. The report should contain the following information

- Methods: Describe preprocessing (e.g., what stopword list and stemmer you used), data representation (equation of the selected tf-idf transformation), feature extraction, selection and dimension reduction, tested distance or similarity measures, tested clustering methods and parameter settings and how you determined optimal parameters. If you used visual inspection, include most important diagrams, using appendices if needed. Your experiments should be reproducible based on your report. It is suggested to divide this section into subsections.

- Results: a) *NMI* value comparison (at least the baseline and your best alternative). If you tested many approaches, they are clearest to present as a table. It should be clear which approach produced which results (including parameter settings). b) Results of the content analysis. For each cluster, tell the most important keywords and their frequencies and your suggestion what is the main topic.
- Execution instructions: Brief instructions how to run your program, including required libraries or installations.