Exercise Sheet VII

Submission Deadline: June 14th, 23:59

Text Classification and Feature Selection

Exercise 1.1: Text Classification Tasks (6 points)

Consider the text classification tasks in the table

Classification Task	Task Description
Genre classification	Given a novel, predict the genre of the novel (e.g., romance, sci-fi, adventure, etc.)
Author gender identification	Given a blog article, predict whether the author is female, male, or other
Language identification	Predict the language of a web page, sentence, paragraph, document, etc.
Sentiment detection	Predict whether a product review is positive or negative towards the product
Categorization of Wikipedia articles	Predict the topic category of a Wikipedia article written in English
Translationese identification	Predict whether a sentence is originally authored in English or translated into English from another language

(a) For each task, determine whether it is flat or hierarchical classification task.

(b) For each task, determine whether it is single-category or multi-category classification task.

(c) For each task, propose a set of features that would be useful for the classification algorithm to make accurate predictions. Discuss the reasons behind your choices.

Your are not expected to give any mathematical formulas for this exercise. However, you are expected to do a small research on the tasks that you might not be familiar with to propose reasonable feature sets beyond surface word n-grams.

Exercise 1.2: PMI for Topic Categorization (8 points)

In this exercise, you will explore two variants of pointwise mutual information (PMI) metric for feature selection in text classification. To this end, we provide a topic categorization dataset where short text documents are manually categorized into one of four possible categories: World, Sports, Business, Sci/Tech. The dataset is distributed within the compressed file `exercise7_corpora`.

(a) Implement a Python function that computes and returns the expected PMI value of a feature f across categories (SNLP Chapter 6, slide 51).

(b) Implement a Python function that computes and returns the maximum PMI value of a feature f across categories (SNLP Chapter 6, slide 51).

(c) Use your favorite word tokenizer (e.g., NLTK TokTok) and preprocess the training portion of the AG news dataset. Keep the words that occur at least three times in the entire document collection and collect word counts in a suitable data structure.

(d) Using the functions in part (a) and (b), and the collected statistics in part (c), rank words based on two criteria; expected PMI and maximum PMI. Generate a table of the top 20 features according to each variant of the PMI metric.
(Hint: in case of identical PMI values, more frequent features should be ranked higher)

(e) Discuss your observations from the table in part (d). Do you notice any differences in the top of the ranked lists?

Your answer should include the table of part (d) as well as the source code to reproduce the results. You may refer to exercise 2.2 in exercise sheet 6 if you feel the information provided in this exercise is not sufficient to understand the requirements.

Exercise 1.3: χ^2 Feature Selection (6 points)

Consider a setting where the task is to classify a collection of scientific articles based on their topics. The table below shows some word occurrence statistics from this document collection for four topic categories; mathematics, chemistry, astronomy and physics.

	N(composition)	N(gravity)	N(differential)	N(theory)
c = mathematics	30	3	50	7
c = chemistry	43	0	2	5
c = astronomy	47	53	19	11
c = physics	30	34	29	7

Use χ^2 statistics to assess whether each word feature would be useful for this task.

Submission Instructions

The following instructions are mandatory. Please read them carefully. If you do not follow these instructions, the tutors can decide not to correct your exercise solutions.

- You have to submit the solutions of this exercise sheet as a team of 2-3 students.
- NLTK modules are not allowed, and not necessary, for this assignment.
- You do not need to include the distributed corpora within your submission.
- Make a single `ZIP` archive file of your solution with the following structure
  - A `source_code` directory that contains your well-documented source code and a `README` file with instructions to run the code and reproduce the results.
  - A `PDF` report with your solutions, figures, and discussions on the questions that you would like to include.
  - A `README` file with group member names, matriculation numbers and emails.

- Rename your `ZIP` submission file in the format

  `exercise07_id#1_id#2_id#3.zip`

  where `id#n` is the matriculation number of every member in the team.

- Your exercise solution must be uploaded by only one of your team members to the course management system (CMS). Once the grading is done, your assignment grade will be distributed to each team member by one of the tutors.

- If you have any problems with the submission, contact `babdullah@lsv.uni-saarland.de` before the deadline.