Heidelberg University
Database Systems Research Group
Prof. Dr. Michael Gertz
Dennis Aumiller, Nicolas Reuter, Jayson Salazar, John Ziegler

October 27, 2022
Text Analytics
Winter Semester 2022/23

Assignment 1: "Text Conversion and Regular Expressions"

Due: Monday, November 14, 2022, 2pm via Moodle

Submission guidelines

- A quick reminder to form **teams of three or four students** for submissions. Student teams are fixed and must be identical to teams collaborating on the group project. Submissions without a group will not be accepted.
- Written solutions need to be **uploaded as a single PDF**.
- Source code of programming exercises needs to be provided in a separate folder. Preferably your code should be in the form of a *Jupyter notebook*. We require that your code can be run using *Python 3.7 on a Linux OS*. In this case you also need to provide a *requirements.txt* that lists all dependencies.
- Zip your written solutions and source code before you upload them.
- We will enable group submission in Moodle based on the teams available through the project proposal. Regardless, please make sure that **the names of all team members are given on the PDF and in the source code**. Please contact us for any necessary changes to the Moodle assignment groups.
- Justify all of your answers, otherwise points may be deducted.

Problem 1-1 Text Analytics Pipeline 4 + 3 + 3 = 10 points

1. Within the last year, an insurance company has transitioned from physical printouts to digitized document scans for all customer-related forms. These documents should now be used for further analysis of customer needs.

   Briefly describe each of the planning considerations for a text analytics project in the above scenario. As a reminder, the considerations include:

   (i) drivers,
   (ii) objectives,
   (iii) data availability, and
   (iv) cost factors

2. Compare the usage of Elasticsearch as a document store over a traditional relational database, such as PostgreSQL. Give two advantages and disadvantages.

3. Name at least two advantages and disadvantages for each of the following concepts. Illustrate your points by giving concrete examples.

   (i) Stop word removal
   (ii) Stemming

Problem 1-2 Regular Expressions 10 + 2 = 12 points

1. Implement your own (basic) regular expression parser, which can *accept* or *reject* a RegEx statement. Consider the following specifications for your RegEx grammar that should be supported.
   You are **not** allowed to use any RegEx library, such as Python's `re` or `regex` modules, for this task.
   (i) Accept any number of alphanumeric characters. This should include only those alphanumeric characters that also appear in the 7-bit ASCII table, with the specific addition being the German special characters (`äöü` and `ß`). Any sequence which contains other characters or symbols should be rejected as an invalid expression.
   Hint: Consider that both lower- and uppercase letters are acceptable.
   (ii) The following special characters are not allowed to appear directly next to each other: `^|*+?`
   If they do, reject the expression.
   (iii) Basic grouping should be supported with brackets `()`. The numbers of opening and closing parentheses have to be consistent, i.e., `(())` is valid, but `(()` is not.

   The result of all sub-tasks must be available through a single function, taking as an input an expression and returning a single boolean indicating the acceptance or rejection of said expression.
   In Python, your function should look like this: `verify_regex(expression: str) -> bool`.

2. You have a set of HTML files and need to parse their content. Would you use regular expressions to parse the files? Justify your answer.

Problem 1-3 Information Extraction from PDF Documents 4 + 2 + 10 + 2 = 20 points

In this assignment you have to convert a set of PDFs to raw text and extract information from the files using regular expressions.

1. Download the folders containing PDF files provided via Moodle (`1-3-pdf-files.zip`). Since the PDF format cannot be directly used to process text, you first have to convert the file contents to plain text. Find two different methods to convert PDFs to text, and compare their performance. You should provide a quantitative and qualitative analysis. For comparison of two generated files, use Python's `SequenceMatcher.ratio()`. With this analysis as a basis, choose one of the methods, and provide the processed raw text files in the submission folder. Justify your decision!

2. Why is a high quality conversion from PDF to plain text hard? Your answer does not need to be exhaustive but should outline some of the most important reasons.

3. The provided files are split into three groups. For each of the individual parts, make sure to print out all extracted results **into a dedicated text file** which should be included in your submission. Each line should contain only a single instance in a normalized format. For phone numbers, e.g., you could remove all spaces and special characters, etc.:
   (i) From the PDF files in the folder `flyers/` extract as many valid phone numbers as possible.
   (ii) Extract valid URLs and Email addresses from the files in the folder `flyers/`.
   (iii) From the PDF file in the folder `iban/` extract all IBANs.
   Hint: You do not have to perform validation of the IBAN number, but only check for the format.

4. Apply your solution from task 4-(i) to the files in the folder `scans/` which consists of pages scanned from a phone book. Analyse how well your solution performs by giving examples.