COMP4650/COMP6490 Document Analysis
2025 Semester 2

Computing Lab 1

Q1: Text Pre-processing

Part I

(This is a theory question that does not require coding.)

Before attempting this question, please read the following sections from Chapter 2 of "Speech and Language Processing".[1]

- 2.2 Words
- 2.3 Corpora
- 2.5.1 Top-down (rule-based) tokenization
- 2.6 Word Normalization, Lemmatization and Stemming
- 2.7 Sentence Segmentation

Please also read the following section from Chapter 2 of "Introduction to Information Retrieval".[2]

- 2.2.2 Dropping common terms: stop words

(a) List the major steps to pre-process text that you have learned from the lecture and the reading materials above. Briefly describe each step.

(b) List some major differences between stemming and lemmatisation.

(c) Discuss why stop word removal is not a good idea for certain NLP tasks or applications. Use one NLP task or application to explain the reason.

Part II

(This is a practice question that requires coding.)

In the notebook `lab1-text_preprocessing.ipynb` you will explore techniques for text pre-processing including tokenisation, stop word removal, stemming, and lemmatisation, using the Natural Language Toolkit (NLTK)[3]. Work through the notebook and answer the questions in it.

Q2: Term Weighting and Cosine Similarity

(This is a theory question that does not require coding.)

[1] `https://web.stanford.edu/~jurafsky/slp3/2.pdf`
[2] `https://nlp.stanford.edu/IR-book/pdf/02voc.pdf`
[3] `https://www.nltk.org`

Before attempting this question, please read the following sections from Chapter 6 of "Introduction to Information Retrieval".[4]

- 6.2 Term frequency and weighting
- 6.3 The vector space model for scoring

Consider the following term-document matrix for the 3 terms "quick", "brown", and "fox" in a collection of 3 documents:

	quick	brown	fox
Doc1	3	0	2
Doc2	0	1	1
Doc3	0	3	6

(a) Calculate the tf-idf score of each term in each document.

(b) Now suppose that a user runs the query "quick fox". Calculate the cosine similarity between this query and each of the 3 documents, where the document and query vectors are given by the tf-idf score of each term. Which document is retrieved first?

(c) Explain the importance of the idf component of the tf-idf score. How does the idf change the weights of rare terms and why is this useful in information retrieval?

[4] https://nlp.stanford.edu/IR-book/pdf/06vect.pdf