Natural Language Engineering:
Assessed Coursework 2

Submission format: You should submit one file that should either be a Jupyter notebook or a zip file containing a Jupyter notebook and any other files (e.g., images or Python files) that you want to include in the notebook.

Due date: Your work should be submitted on the module's Canvas site before 4pm on Thursday 28th November. This is Thursday of week 9. The standard late penalties apply.

Return date: Marks and feedback will be provided on Canvas on Thursday December 19th for all submissions that are submitted by the due date.

Weighting This assessment contributes 25% of the mark for the module.

Overview

For this assignment you are asked to complete a Jupyter notebook ('a2.ipynb') which is provided with these guidelines. It is based on activities that you have already completed in labs during weeks 1-7 of the module. Any code you have developed during the labs can be submitted as part of your answers to the questions in the assignment. To score highly on this assignment you will need to demonstrate that you:

- understand the theory and your code;
- can write and document high quality python code;
- can develop code further to solve related problems;
- can carry out experiments and display results in a coherent way;
- can analyse and interpret results; and
- can draw conclusions and understand limitations of the technology.

For this report you should submit a single Jupyter notebook containing all of your answers to all of the questions in 'a2.ipynb'. You may import from standard libraries and the 'sussex_nltk' resources which you have been provided with. If you wish to import any other code, it must be included in a zip file with your Jupyter notebook. It **must** be possible for the assessors to run your Jupyter notebook.

Marking Criteria and Requirements

Your submission will be marked out of 50. There are 2 questions, each question is worth 25 marks and both questions should be answered. Each question is broken down into 3 or 4 parts and the breakdown of marks between parts is specified in the notebook. General and question specific criteria are given below. Please read these guidelines carefully and ask if you have any questions.

General

- In order to avoid misconduct, you should not talk about these coursework questions with your peers. If you are not sure what a question is asking you to do or have any other questions, please ask me or one of the Teaching Assistants.
- Your report should be no more than 3000 words in length excluding code and the content of graphs, tables and any references.
- You should specify the length of your report. 3000 is a strict limit.
- You should use a formal writing style.
- All graphs should have a title and have each axis clearly labelled.
- Textual answers should be included in Markdown cells.
- Do not add external text (e.g. code, output) as images.

Question 1: Document Similarity

25 marks available

This question is designed to test your ability to understand, write and extend code to solve problems.

- This question has 3 parts worth 8, 8 and 9 marks respectively.
- In part a, you are asked to explain each step in a given function. The steps are clearly numbered in the function comments. Make sure you refer to these step numbers in your written explanation (which should be provided in a Markdown cell).
- In parts b and c, marks are available for your solution to the problem as well as the quality and correctness of the code you provide to solve the given problem.

  - Code must be clear and well-documented.
  - You should include a textual description of your solution, outlining the steps in your algorithm and giving any relevant definitions or formulae.

Question 2: Supervised Methods for WSD:

25 marks available

This question is designed to test your understanding of supervised methods for word sense disambiguation, your ability to implement solutions in code and your ability to evaluate solutions.

- This question has 4 parts worth 4, 8, 8 and 5 marks respectively.
- In part a, marks are available for clear and correct code.
- In part b, marks are available for your solution including the quality of your textual explanation of your chosen algorithm, clearly justifying any design choices and the quality and clarity of your code.
- In part c, you are expected to look at the test documents and evaluate how well the WSD system did at identifying the intended sense of each occurrence of the words under consideration.
  - You should give a quantitative evaluation and a qualitative evaluation, with reference to examples of where the WSD system got it right and where it got it wrong.
  - When the WSD system is wrong, can you explain why it is wrong and/or categorize the types of errors being made?
  - You may decide that you do not need to look at every document but you should consider at least a representative sample of documents for each word.
  - There may be cases where it is not clear what the intended sense of a word is - it is OK to acknowledge this, but you should give an estimate of the proportion of the examples where this is the case.
- In part d, marks are available for the quality and creativity of your ideas. There are no marks available for code in this part.