Natural Language Engineering: Assessed Coursework 3

Submission format: You should submit a single zip file containing Jupyter notebooks and any other files (e.g., images or Python files) for each question. It is permissible to submit a Word, pdf or other text document for Question 4 but NOT for questions 1 to 3.

Due date: Your work should be submitted on the module's Canvas site before 4pm on Monday 13th January. The standard late penalties apply.

Return date: Marks and feedback will be provided on Canvas on Monday February 3rd for all submissions that are submitted by the due date.

Weighting This assessment contributes 50% of the mark for the module.

Overview

For this assignment you are asked to complete 4 Jupyter notebooks which are provided with these guidelines:

- 'a2_1.ipynb'
- 'a2_2.ipynb'
- 'a2_3.ipynb'
- 'a2_4.ipynb'

A separate notebook has been provided for each question since the questions are independent of each other and so that the notebooks load and run faster. The questions are based on activities that you have already completed in labs, mainly during weeks 8-11 of the module (although earlier material may also be relevant). 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.
- can design and build prototype systems combining off-the-shelf technologies into a practical language processing system

For this report you should submit 4 Jupyter notebooks containing all of your answers to all of the questions (or 3 Jupyter notebooks for questions 1-3 and a pdf document for question 4). 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 the zip file with your Jupyter notebooks. It **must** be possible for the assessors to run your Jupyter notebooks.

Marking Criteria and Requirements

Your submission will be marked out of 100. Each question is worth 25 marks and all questions should be answered. Questions 1-3 are broken down into 3 or 4 parts and the breakdown of marks between parts is specified here and 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.
- Textual answers should be included in Markdown cells.
- Your total submission should be no more than 3000 words in length excluding code and the content of graphs, tables and any references.
- You should specify the number of words (in Markdown) of each of your jupyter notebooks that you submit.

- You should use a formal writing style.
- All graphs should have a title and have each axis clearly labelled.
- Do not add external text (e.g. code, output) as images.

Question 1: Part-of-Speech Tagging:

25 marks available

This question is designed to test your understanding of part-of-speech tagging, Hidden Markov Models and the evaluation of taggers.

- There are 4 parts to this question worth 4, 8, 7 and 6 marks respectively.
- In all parts, marks will be awarded for the quality of your written explanations.
  - Written explanations must be given in Markdown cells (not code cells).
  - Your explanations must refer to your unique set of examples generated by entering your candidate number at the top of the notebook. This must be your own candidate number.
  - Your explanations should define technical terms and include examples of calculations taken from your sample sets.
  - You are encouraged to write or use code to carry out calculations on your sample sets. However, other than in part b, you will not be marked for the quality of this code. Further, code on its own does not count as explanation.
- In part b) you are asked to write a tag() method for the unigram_tagger class. Therefore, in this part of the question there are marks available for clear and correct code. There are further marks available for evaluating the performance of your tagger and discussing the results.

Question 2: Distributional Semantics:

25 marks available

This question is designed to test your understanding of distributional semantics and your ability to investigate how the context window size affects the distributional similarity of words. There are

marks are available for quality of experimental design, quality and correctness of code, quality of presentation of results and quality of discussion of results and conclusions.

- There are 4 parts to this question worth 4, 4, 5 and 12 marks respectively.
- In part a) you are given code which you are expected to run and explain. Your explanation must be in a Markdown cell and must make reference to both the code and the given examples (which are unique to your candidate number).
- In parts b) and c) you are expected to solve a problem using code. 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.
- In part d) you are expected to design and carry out an investigation. You should explain your experimental procedure and justify any design choices. You will need to provide code for a number of different functions: semantic similarity according to the WordNet path similarity measure, distributional similarity, and correlation; as well as code to carry out your experiments. You are encouraged to use library functions where available (e.g., use Pandas functionality for correlation). There are also marks available for providing a graph showing how correlation varies with context window size and for discussing your results.

Question 3: Named Entity Recognition and Linking

25 marks available

This question is designed to test your ability to solve problems using code and understand the errors made by existing technology.

- This question has 3 parts worth 6, 6 and 13 marks respectively.
- In part a, marks are available for the quality and correctness of your code.
- In part b, there are no marks available for code. Marks are available for the quality of your discussion. You should try to identify different types of errors made by the named entity recogniser. You must make reference to specific examples from your sample.

- In part c, you are asked to design and implement a solution to a problem. You should explain your solution in text and justify any design choices. There are also marks available for the quality and correctness of your code, and for running your system for any three of the major characters in the text. In this part of the question, you should make sure you use the whole (time-ordered) text of the novel rather than a randomized sample.

Question 4: Question Answering:

25 marks available

This question is designed to test your understanding of the challenges and potential solutions in the application of Question Answering and to assess your creativity.

- This question is not broken down into parts.
- You are not expected to implement anything or provide any code.
- You can provide an answer as a Word or pdf document rather than a Jupyter notebook for this question.
- You must make reference to the example questions given in the question.
- You should describe in detail the different components of your system or the steps in the procedure.
- You may wish to describe alternative strategies in some places. However, you should choose which strategy you would use in this scenario and justify it. There may be more than one correct answer.
- You may wish to refer to papers or textbooks in this question - make sure you reference them appropriately in the text and include a bibliography.