PRIFYSGOL
BANGOR
UNIVERSITY

School of Computer Science and Electronic Engineering

Natural Language Processing

Assignment

30% of ICE-4721

Bill Teahan

Module: **ICE-4721**
Department: **School of Computer Science and Electronic Engineering**
Module credit: **20**
Organiser: **Dr. Bill Teahan**

Assessment : Mini-Project Assignment

Description

The Mini-Project Assignment is an opportunity for you to demonstrate your understanding and interpretation of NLP theory and methods to assist the process of designing and implementing a non-trivial NLP software tool.

Your specific task is to create your own large corpus by collecting samples of NL text. This corpus must not include too much duplicated text, either from within the corpus itself, or obtained from other text corpora.

You are also required to develop for yourself an NLP system in Python: either 1) for analysing this text corpus; or 2) by applying it to some NLP task.

The corpus and software you develop must be mostly your own work. Any part that matches online sources will be ignored for assessment purposes.

You **need to use one or more of** the following technologies **in some way**:

- Python
- NLTK
- spaCy
- NLP pipeline
- POS (Part-of-Speech) tagging
- Any other useful Python library you may have found

Your solution Python code must be commented in the Docstrings format; within these comments you must highlight where you have used a specific Python programming language feature; the justification should be in your design documentation. You are also required to submit a full report on the development of the software and its implementation.

The purpose of the mini-project is to provide you with some experience in developing a non-trivial Python application which makes substantial use of advanced features of either the NLTK or spaCy Python libraries, or both.

Note that as part of this exercise you will be required to submit a **two** documents. For more details see section below marked **Submission**.

Contribution of this assessment

This assessment contributes to **30**% of the overall module mark.

Requirements

Please make note of the following requirements:

- The work for this assignment must be done **individually**.
- The work will be assessed based on how much of the code **you have written yourself personally**.
- It is your responsibility to ensure that your files are kept secure, and that no-one else has access to the files, including your friends.
- Please make sure you back up your files to multiple places. As a computer science student, this should be very straightforward – e.g. back up to the cloud by gmailing yourself.
- Please do not email your assignment report to the module Organiser. Any report submitted via email will be ignored. The report must be submitted to Blackboard in order for it to be assessed and checked by the TurnItIn plagiarism software.
- Please make sure you submit the report to the correct place in Blackboard, and before the deadline. Do not leave the submission to just a few minutes before the deadline.
- Please make it clear at the beginning of the report that this is for the Mini-project and include your name and username in the report.

Submission Procedure

Use **Blackboard** to submit **two** documents.

- **Document 1:** A WORD or PDF document for the report.
- Include source code as an appendix in this report.
- Also include a declaration in the report re: Unfair practice.
- Also include the datasheet documentation for the corpus (e.g. what preprocessing/cleaning was done; what markup was used; techniques used to obtain the text data including any web scraping code; etc.); summary data (number of words, characters etc.) and further analysis.
- **Document 2:** A **zipped** file of the whole corpus.

Your source code file must

- Contain a program header.
- An appropriate level of comments in Docstrings format.
- Follow a consistent style of indentation.
- Follow the usual Python conventions for class and variable names.

The full written report should include the following sections:

1. Title and Author
2. Contents
3. Introduction
4. Corpus
5. Documentation

6. NLP Techniques used
7. Results and Analysis
8. Conclusions
9. Discussion

Appendix: Source code (full listing)

Deadlines

The deadline for submitting your **Assignment Report, to include the Source-code as text in the Appendix (that includes Docstrings),** through Blackboard is **11:59pm, 2nd May 2023**. Late submissions will be penalised in line with School policy.

Plagiarism and Unfair Practice

Plagiarised work or work that is deemed to be an example of academic misconduct will be given a mark of zero. Remember when you submit you agree to the standard agreement:

> *This piece of work is a result of my own work except where it is a group assignment for which approved collaboration has been granted. Material from the work of others (from a book, a journal or the Web) used in this assignment has been acknowledged and quotations and paraphrasing suitably indicated. I appreciate that to imply that such work is mine, could lead to a nil mark, failing the module or being excluded from the University. I also testify that no substantial part of this work has been previously submitted for assessment.*

Late Submission

Work submitted within one week of the stated deadline will be marked but the mark will be capped at 50% for the mini-project. A mark of 0% will be awarded for any work submitted 1 week after the deadline.

Acceptable reasons for submitting work late include: Serious personal illness with a doctor's certificate (a self-certified medical note should not be accepted). The death of a relative or close friend. Serious family problems such as divorce, separation and eviction. Examples of unacceptable reasons for failing to submit work on time include: Having exams; Having other work to do; Not having access to a computer; Having computer related problems; Being on holiday; Not being able to find information about a subject.

Marking Scheme

Please remember that marks are provisional until they are confirmed by a board of examiners.

Details of how the marks will be calculated.

Marks will be awarded on the basis of the following scheme:

- **Report (1%)**
  - To include: Introduction, Corpus, Documentation, NLP Techniques used, Results and Analysis, Conclusions, Discussion, Code is in the Appendix.

- **Corpus (25%)**
  - To include: Zipped file of whole corpus; datasheet documentation (e.g. what preprocessing/cleaning was done; what markup was used; techniques used to obtain the text data including any web scraping code; etc.); summary data (number of words, characters etc.) and further analysis.
- **NLP system code (34%)**
  - Uses Python docstrings.
  - Every function described – input; output; what it does; how it works.
  - Plenty of comments are also in the code.
- **NLP Techniques used (15%)**
  - Number of methods used and their technical difficulty.
- **Results and Analysis (25%)**
  - Results produced by the system and full analysis and conclusions.

Mark ranges

>80%, clear and deliberate implementation and design of the submitted solutions. Comprehensive collection of texts for the corpus that is excellently documented. Demonstration of and excellent use of the NLTK, spaCy and other Python libraries.

>70%, complete and comprehensive submission of the tasks set. Substantial collection of texts for the corpus that is well documented. A demonstration of a clear command of the NLTK and/or spaCy and other Python libraries showing considered implementation.

>60%, a good attempt of the tasks set. A good understanding of the activities with a demonstration of confidence in the collection of texts for the corpus and in the use of the NLTK and/or spaCy and other Python libraries.

>50%, threshold performance. Demonstrates some understanding of the collection of texts for the corpus and in the use of the NLTK and/or spaCy and other Python libraries and/or attempts on most tasks to demonstrate conceptual understanding.

<50% below threshold performance, with little demonstration of knowledge of process of collection texts for the corpus and in the use of the NLTK and/or spaCy and other Python libraries, and/or lack of completion of the requested tasks.

Feedback details

A description of **how** feedback will be given, and **when** it will occur. We encourage this is split clearly into sections to identify the type of feedback. Such as below:

	Description	**Timeframe**
Formative (On-going)	Verbal Feedback – Verbal feedback will be available by request at each lecture or lab or via Teams. It is suggested that you keep a written note of this feedback to aid in your personal development. You can also request a short meeting to discuss your design after you have submitted it.	Instant
Summative	Written Feedback – Written feedback will be made	1-2 weeks

(Post Assessment)	available through blackboard after an assignment is submitted. To access your written feedback see the comments section of your assignment submission.	