CM30320 Natural Language Processing Assignment

Due 8th December 2023

Due:	8th December 2023
% of Module:	30
Marked out of:	164 Marks
Submit (Where):	Moodle
Submit (What):	3 independent files (as detailed below)

Learning Objectives

This assignment will give you the opportunity to experiment with sentiment analysis, an important task used in different aspects of natural language processing and text mining. In addition you will have the opportunity to experiment with ChatGPT. By the end of this assignment, you should be able to:

1. Text Classification: Develop the ability to classify text into different sentiment categories (e.g., positive, negative, neutral) using machine learning and natural language processing techniques.
2. Feature Extraction: Acquire the skills to identify and extract relevant features from text data that can be used to assess sentiment. This includes understanding the importance of features such as words, phrases, and context.
3. Model Selection and Evaluation: Explore various sentiment analysis models and techniques, and learn how to select the most suitable model for a given task. Evaluate model performance using appropriate metrics.
4. Understand the capabilities and limitations of pre-trained language models, specifically large language models. The rapid adoption of LLMs implies that you will more than likely be using these models in your everyday life, and this provides you with the opportunity to understand what they can do.

Submission Requirements:

You are required to upload three separate files to Moodle. Do NOT upload a zip file.

1. A Report describing your methods associated with PART - A.
    a. This should be a PDF/Word file - Name this file "ReportA"
2. A zip file containing the program code associated with PART - A
    a. This should be a zip file. Name this file "CodeA.zip"
3. A Report describing your analysis as required by PART - B
    a. This should be a PDF/World file - Name this file "ReportB"

Part A - Sentiment Analysis

22% of Module (Marked out of:132 Marks)

This mini-project is based on the material covered in lectures and the programming exercises that you are provided with during this course. The application that you will be addressing is Sentiment Analysis, which is concerned with the automated identification of the opinion polarity associated with a particular piece of text. Most frequently, this is defined as identifying whether a particular piece of writing (e.g., a review on a movie or a product) is positive or negative, and this is precisely what you will be doing in this project. Specifically, you are provided with a set of reviews extracted from the Internet Movie Database (IMDb) with various polarity labels, and your task is to develop an application that can detect the sentiment polarity given any input text. This is a real-world application that is popular in an academic as well as industrial context.

Dataset

You will be using a subset of 2,000 positive and 2,000 negative movie reviews extracted from the Large Movie Review Dataset (https://ai.stanford.edu/~amaas/data/sentiment/). The reviews were extracted based on their star rating: reviews with a score <=4 stars are considered clearly negative, while those with a score >=7 are considered clearly positive. You are provided with a set of positive reviews (in the pos/ folder) as well as negative reviews (in the neg/ folder), where you can see that the file naming conventions are as follows: id_star.txt. In other words, if you'd like to benefit from the star rating assigned to a particular review, you can extract this information from the names of the files.

For background information on the task and the datasets, take a look at:

- Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. (2011). Learning Word Vectors for Sentiment Analysis. The 49th Annual Meeting of the Association for Computational Linguistics (ACL 2011).
- The dataset webpage: https://ai.stanford.edu/~amaas/data/sentiment/

Assignment requirements

You will be marked on a combination of your report and the program code that you submit. Overall, your report should be self contained and should include all the information required to grade your assignment. Similarly, your program code should be self contained and should be well documented so it can be run if required.

The following are the tasks required in this part of your assignment. You must include these sections in your report.

Marking Summary

Section	Code	Writeup (in report)
Abstract	0	5
Introduction and Motivation	0	8
Related work	0	5
Experiments and Results - Feature Generation using n-grams:	5	5
Experiments and Results - Feature selection	6	6
Experiments and Results - Data Splits:	4	4
Experiments and Results - Naïve Bayes	10	10
Experiments and Results - SGD based classification and SVMs	15	15
Experiments and Results -BERT	10	10
Discussion	0	10
Conclusions and Future Work	0	4
Total	50	82

Marked out of 132

Assignment Requirements and Marking Details

Abstract

Marked out of: 5
Programing objectives: None
Requirements:
A short (200 words) description of what the project aims to achieve.

Introduction and Motivation

Marked out of: 10
Programing objectives: None
Requirements:
A section detailing sentiment analysis answering:

1. Why is it important?
2. What are the difficulties?
3. What are the different methods available?
4. What do you intend to test?
5. Did you try anything different?
6. What and how did it work?

Related work

Marked out of: 5
Programing objectives: None
Requirements:
A (short) description of the methods that exist for sentiment analysis. You are required to cite existing methods in the correct format (

Experiments and Results

Marked out of: 100 (Breakdown below)
Programing objectives: See below
Requirements:
You must (at a minimum) explore the following methods. You must describe each of these in your report and write the associated Python scripts. You should provide enough detail in the report to ensure that someone who has not sat through the lectures can understand what you intend to do. Your code must be clearly documented.

<u>Feature Generation using n-grams</u>:
Have a systematic method of choosing "n" (1, 2, 3..). Detail why you limit your n to whatever you choose. Provide examples in your report to demonstrate your choice.
Marking scheme:

1. Code: 5 marks
2. Writeup: 5 marks

Feature selection

Use a combination of stopwords, lemmatization or stemming and TFIDF to select the features you intend to use. You are expected to have at least *three* different sets of features.

Marking scheme:

1. Code: 6 marks (2 for each method)
2. Writeup : 6 marks (2 for each method)

Data Splits:

Split the data provided into three data splits: Training, development and test.

Marking scheme:

1. Code: 4 marks
2. Writeup: 4 mark

Naïve Bayes:

Implement Naïve Bayes from scratch (clf = MultinomialNB is NOT acceptable). Do NOT use any existing implementations. You must document your code. Evaluate it on each of the three different feature sets you have extracted. You are to perform the evaluation on the development set. Pick the best of the three methods you have experimented with and present the test set results associated with this. Also evaluate the Naïve Bayes implementation from https://scikit-learn.org/stable/ (now you can use MultinomialNB) on the same three splits

Marking scheme:

1. Code: 10 marks (8 for your implementation 2 for scikit-learn)
2. Writeup: 10 marks (8 for your implementation 2 for scikit-learn)

SGD based classification and SVMs

Use ~~Regression and~~ the Logistic Regression and SVM packages from https://scikit-learn.org/stable/ and evaluate them on each of the three different feature sets you have extracted. You are to perform the evaluation on the development set. Pick the best of the three methods you have experimented with and present the test set results associated with this. Use 1-hot embeddings based on your features for input.

Marking scheme:

1. Code: 10 marks (5 for each)
2. Writeup: 10 marks (5 for each)
3. Hyperparameter optimisation: (10 marks) Notice that you can change the parameters (learning rate, …). Try at least 5 different combinations on your best performing model/features using the development set. Evaluate your test set using the best hyperparameters (5 marks for experiment, 5 for writeup)

BERT

This part of your assignment is completely independent of all previous sections. Using the same train/dev/test splits as before, use BERT, specifically the base model, to perform the same classification. Be sure to experiment with both the cased and uncased versions of the model. You will find the following tutorials helpful in addition to the discussions in the lectures. The following tutorials will use the entire IMDB dataset - your assignment requires you to use the custom splits.

Custom Datasets: https://huggingface.co/transformers/v3.2.0/custom_datasets.html

IMDB Classification:

https://huggingface.co/docs/transformers/tasks/sequence_classification

Marking scheme:

1. Code: 10 marks
2. Writeup: 10 marks

Discussion

Marked out of:

Programing objectives: None

Requirements:

Provide a table detailing the results on the TEST split using each of the following methods. Notice that you will find the “best” within a method using the development split.

1. Naïve Bayes (your implementation)
2. Naïve Bayes (scikit-learn)
3. SGD
4. SVM
5. BERT

Discuss the performance of each model with an emphasis on why one might perform better than the other(s). You must analyse each model.

Conclusions and Future work

Marked out of:

Programing objectives: None

Requirements:

What do you conclude in terms of the best performing model and what would you do to improve these results in future?

Part B - ChatGPT

8% of Module (Marked out of:32 Marks)

This part of your assignment requires you to submit a report. You should aim to answer the following questions. Your report should include the sections detailed below. You MUST include screenshots of the responses presented by ChatGPT. The screenshot you provide must include your email id (typically appears on the left of the chat panel).

Goal: Find effective methods of using ChatGPT to "discuss" your course content within the framework of a chat. Enumerate errors where possible.

You can use the content from any ONE of the first 3 weeks.

You must experiment with:

a) The lecture notes
b) The automatic transcription of the lectures (from panopto)

PLEASE NOTE THE FOLLOWING CHANGE TO THE ASSIGNMENT:
A lot of you are finding it very difficult to find "first answer errors" (defined below). To get around this, in every instance below where I have asked for a "first answer" error, you can instead report:

1. Any 2 "difficult" attempts where you were surprised that the model got the answer correct despite you trying to trick it.
2. 3 instances for where you report errors which are EITHER:
    a. not "first answer" (i.e., you are allowed to have a longer conversation than being required to ask a question immediately after entering the context).
    b. where you did not include context (i.e, you did not include the lecture notes or the lecture transcripts)

Your report must answer the following questions and should include evidence in the form of screenshots where possible. Note that you will have to attempt different methods of doing each of the following (2 at least) and document what you've attempted in the report (list your prompts).

1. How can you include the content of the lecture notes to allow ChatGPT to discuss the content with you? What is the simplest way of incorporating mathematical notation into chats? You must experiment with at least 2 different methods of doing this and describe which works better. (8 Marks)
    a. What part of the notes can ChatGPT recognise as is? (2 marks)
    b. Can it recognise the equations if it is pasted in? (2 marks)

    c. What about when you describe the equations in words (2 marks)
    d. How about converting it to markdown? (2 marks)
2. Discover and document at least 5 first answer errors that ChatGPT makes and 1 further along in the conversation when you ask questions about the lecture content using the lecture notes as context. A First answer error is an error that occurs in the answer to the first question you ask within a chat. You will have to "reset chat" before each question (and also re-enter the context) to test this. (6 marks, 1 for each error, you must include screenshots)
3. How can you incorporate the transcript of the lecture into a chat? Notice that you might have to do this in sections to ensure that the entire lecture is available. (8 marks).
    a. Experiment with directly using the transcript from the lecture capture. You must explain what works and what does not work? What are the difficulties you encounter when using the entire transcript? (3)
    b. Experiment with first using ChatGPT to "clean up" the transcript, before then using the ChatGPT to chat with the content. You must detail different ways of cleaning up the transcript (2)
    c. Experiment with first getting ChatGPT to summarise the lecture before then chatting with it. (3)
4. Discover and document at least 5 first answer errors that ChatGPT makes and 1 further along in the conversation when you ask questions about the lecture content **using the transcripts of the lecture**. A First answer error is an error that occurs in the answer to the first question you ask within a chat. You will have to "reset chat" before each question (and also re-enter the context) to test this. (6 marks, 1 for each error, you must include screenshots)
5. Create a small story which is not real (and not part of any book you know of). Invent imaginary characters and describe them doing unlikely things. Using this story as context (instead of the content from lectures) chat with the model to see if it answers based on what the characters did in your story or what people do in real life. For example, you might have someone flying around in your story. Check to see if ChatGPT replies saying that they can fly or if it assumes people cannot. Of course, you will have to try scenarios that are a lot more complex (4 Marks)

Note

1. Your report must contain the above 5 sections.
2. Your report need not contain an introduction, …
3. Your report must include screenshots for every "error" that you identify in 2 and 4 above.
4. Your report must include the prompt for each experiment. You can replace the content of the lecture and the transcript with <content>, <transcript> respectively to make your prompts more concise in the report.

5. You must include the specific prompt you used to provide the mathematical expressions to ChatGPT.
6. This part of your assignment is marked out of 32 and is worth 8% of your module.

Informed consent

Given the significant interest in the use of ChatGPT, especially in the content of teaching, it will be very helpful if completely anonymised versions of the insights you generate can be used to report on effective ways of using ChatGPT and its shortcomings.

Do I have to participate?
You do not have to participate. If you choose not to participate, this will NOT affect your mark in any way. Your assignment will be graded without access to this information.

How will my data be used?
If you choose to allow the use of the insights you generate, parts of what you write up in your assignment might be included in a report on the use of generative AI in education. (see below for how you will be given credit). All personally identifiable information will be removed. Insights will be in the form of a list and will be drawn from multiple assignments by your lecturer. These insights, for example a list of observations on what works and what does not work, the kind of errors generated by the system, etc, *with no personally identifiable information*, might be shared with others working on similar projects. At no point will anyone other than your lecturer have access to any personally identifiable information associated with the content of your assignments.

Will I get credit?
You will be credited as one of those contributing to any report that is created using the observations/insights you make in your assignment. You will receive an email with a link to any such report. While it might be nice to list this on your CV, you can choose to opt out of having your name listed as a contributor. If you do not want to be completely excluded, but *do* want your name excluded from the contributors list (i.e., not be given credit) please fill in the following form which has a single Yes/No question:
https://forms.office.com/e/BJvXHkU802

I want to know more
Contact htm43@bath.ac.uk

How do I choose to have my data excluded?
If you do not mind having the insights you present in your assignment included in a future report, no action is required.

If you want to be completely excluded, please fill in this form. It has a single Yes/No question and will take no more than a couple of seconds to complete:
https://forms.office.com/e/pnxe5fdhfQ