

CITS4012 Natural Language Processing project Specification

Due: 20 May 2024, 11:59PM (Perth time)

1 Project Objective

This project is to be completed in **a group of 2 or 3 students (maximum)**. You're allowed to complete this project individually if you prefer not to work in a group, but please note **NO bonus mark will be given for individual submission**. Every group needs to submit the group registration form (This link) before **27 April**. We strongly recommend to start working early so that you will have ample time to discover stumbling blocks.

The goal of this project is to complete the Aspect-based Sentiment Analysis (ABSA) task. ABSA task aims at identifying the sentiment polarity (e.g. positive, negative, neutral) of one specific aspect in its context sentence. For example, given a sentence “*great food but the service was dreadful*” the sentiment polarity for aspects “*food*” and “*service*” are positive and negative respectively.

For this project, instead of solely focusing on achieving higher performance, you should consider exploring novel architecture design and justify your decision processes, as grading is by-and-large based on your research process rather than system performance (see marking scheme at the end).

2 Dataset

In this project, you are required to design and evaluate the attention-based sequence-to-sequence model on a real world aspect-based sentiment analysis dataset: MAMS, in which each sentences contain at least two aspects with different sentiment polarities.

The dataset can be downloaded from **this link**. You are provided with:

- train.json: JSON file of the training data
- val.json: JSON file of the validation data
- test.json: JSON file of the test data

You can choose to use the validation set for hyper-parameters tuning or train your model on both the training and validation set so as to maximise performance on the test set. You're required to report and analyse the performance of your methods in accuracy rate on the test set only.

Each instance in the dataset contains a restaurant review, one restaurant aspect and one polarity of this aspect in the given review. There are eight different aspect categories: *food*, *service*, *staff*, *price*, *ambience*, *menu*, *place* and *miscellaneous*, and three different polarities: *positive*, *negative* and *neutral*. You are required to predict the polarity based on the review and the given aspect.

3 Report Writing

Your project will be marked based on your code and your report. The report should be submitted as a PDF and contain **no more than eight A4 pages** of contents, excluding team contribution and references. The report should be organised similar to research papers and should contain the following sections, and you need to put the code that you conduct all actions for the following sections to the ipynb template.

3.1 Title

The title of your project and the author list including name and student ID.

3.2 Abstract

An abstract should concisely (less than 300 words) motivate the problem, describe your aims, describe your contribution, and highlight your main finding(s).

3.3 Introduction

The introduction should explain the problem and your understanding of its significance, difficulties and applications. You should give an overview of the your approach and the main results. Though an introduction covers similar contents as an abstract, a good introduction should cover more details for the problem discussions and references to existing works.

3.4 Methods

This section details your methods to the problem. This is where you describe the architecture of your neural network(s), and any other key methods or algorithms.

- You are required to design three model variants to differently integrate the aspect information in your model structure (e.g. different locations, different integration methods etc.)
- At least one of your model variant **MUST** use the attention mechanism. You are encouraged to apply the attention mechanism with novelty.
- You **MUST** use one of the following architectures for seq2seq processing component: RNN, LSTM, GRU, and Transformer.
- You should describe your model in details with proper equations, notations and the architecture drawing.
- If your methods design is referring to any published works, you should give proper references.

3.5 Experiments

This section should contain:

Dataset Description describe the dataset and include any dataset analysis you have done.

Experiment Setup This should include all the details of how you ran your experiments including but not limited to: the hyper-parameter values you used and tested (e.g. learning rate, hidden size), optimisation methods, loss function, and any text pre-processing methods etc.

3.6 Results

This section should contain:

Quantitative Results You should report, compare, analyse and interpret the quantitative results that you have based on your experiments. You are also expected to have an ablation study section to specifically evaluate and analyse the effectiveness and the impacts of the different components and/or hyper-parameters of your model (e.g. different input embedding, different attention methods, different seq2seq model etc.)

There is no fixed requirements or limitations for the experiments and ablation studies you choose to do. The results section will be marked by the comprehensiveness and the significance of your experiment choices, and the insightful analysis and justifications of the results. You may want to choose your experiments carefully with decent justifications.

Qualitative Results The qualitative analysis is for you to analyse and interpret the performance of your model or your model components based on the actual sample cases. You are required to choose one or two sample instances from the test set and visualise the attention weights of the sentence tokens with respect to different aspects and different polarities. You are free to use any types of visualisation (i.e. either draw manually or automatically generate the visualisation with code) for this qualitative analysis. You should show and analyse such visualisation with respect to your model's performances.

3.7 Conclusion

Summarise the main findings of your project, and what you have learnt. Highlight your achievements, note the primary limitations of your work and future works.

3.8 Team Contribution

If you are a multi-person team, briefly describe the contributions of each member of the team.

3.9 References

You should list all references cited in your report and format- ted all references in a consistent way.

4 Submission Method

The submission should be made via LMS project Submission Box (The submission box will be opened on 10 May 2024). Only **ONE** group member is needed to make the submission. You **MUST** submit two files:

- a PDF file, with filename: CITS4012_YourGroupID.pdf
- a ipynb file, with filename: CITS4012_YourGroupID.ipynb. This file should include all your implementation of this project.

You can **optionally** submit a zip file that contains a README file describe how to run the code if it's not apparent from the documentation in your ipynb files, any of your trained models or any other files that's necessary for the marker to run your program.

5 Important Rules

You **MUST** follow the rules below. Any team found to break any of these rules will receive zero mark to their project.

1. In terms of the sequence processing components you **MUST** use one of the following architectures: RNN, LSTM, GRU, and Transformer. You are allowed to use deep-learning libraries (e.g. pytorch) to import these sequence processing components (i.e. you do not have to code RNN from scratch). You could read relevant publications to come up with a sensible design for your methods, but you **MUST NOT** copy any open-source code from any publications (in other words, you **MUST** implement the methods yourself).
2. The following deep-learning libraries are allowed: pytorch, keras, and tensorflow. Huggingface is not allowed. Standard python libraries (e.g. numpy and matplotlib) and NLP preprocessing toolkits (e.g. NLTK and Spacy) are allowed.
3. You could use pretrained word embeddings (e.g. Word2Vec/Glove), but you **MUST NOT** use any pretrained language model weights or checkpoints (e.g. BERT checkpoints), or any closed-source models (OpenAI GPT-3). In other words, you **MUST** train your model from scratch, using the provided data, which includes a training and a validation set.

4. The model described in the report MUST be faithful to the submitted code and running log that you submit. You MUST include the running log (with the reported result/performance) in the submitted ipynb file.
5. You are allowed to use code from the lab contents (provided that they don't conflict with any project rules), but you MUST NOT copy any open source project code from GitHub or other platforms.
6. You MUST NOT use models that cannot be run on Colab.
7. You MUST use the given code template for implementation.

6 Marking Scheme

Component [Grades]	Criterion
Model [10]	<ul style="list-style-type: none"> • 3 model variants with different aspect integration • Attention mechanism is used • Sequence-to-Sequence Model: RNN/LSTM/GRU/Transformer • Proper Equation and Notation • Proper and understandable model architecture drawing • Justifications of model design
Experiments [4]	<ul style="list-style-type: none"> • Dataset description and statistics • Detailed and comprehensive experiment setup
Results [10]	<ul style="list-style-type: none"> • 3 model variants performance comparison • Comprehensive ablation study testing and results • Significance of ablation study design • Valid qualitative analysis with attention weights visualisation • Insightful justification and analysis of results • Proper table/figures are used for results analysis

Writing [3]	<ul style="list-style-type: none"> • Report contains all required sections and contents • Report is well-structured and written in academic style • No spelling mistake • Appropriate citation and consistent referencing style <ul style="list-style-type: none"> • Cited relevant publications for problem motivation and model design • Insightful discussion of any existing works
Other [3]	For impressing the marker with excelling expectation, high performance, novel model design, use L ^A T _E X
Penalties [-]	<ul style="list-style-type: none"> • Late or no submission: 5% per day for the first 7 days (including weekends and public holidays), after which the assigned work is not accepted. • Badly written code - not well organized, commented, documented [-4]
Total Marks	30