NLP Coursework Specs (2025)

This coursework is designed to provide you with a typical experience in NLP research. In NLP, progress is often driven by 'shared tasks', where different research teams compete to develop the most effective models for a specific task. While the ultimate goal is to create a model that outperforms others, significant focus is also placed on analysing a) the task itself, b) the models used, and c) the results of various experiments to uncover valuable insights. Throughout this coursework, you will gain an understanding of the empirical nature of NLP research, learn to develop models for the given task and iteratively improve them, run experiments to uncover new findings, and communicate your results through a research paper or report. Writing research papers or reports is a crucial skill in NLP and should be given careful attention.

Summary: The task is to develop a binary classification model to predict whether a text contains patronising and condescending language. You are expected to write a report, much like a research paper, describing your findings. The task was task 4 (subtask 1) in the SemEval 2022 competition. More information about the task is available in the task paper.

Your task: Train a model that outperforms the task's RoBERTa-base baseline model (provided by the task organisers) in F1 score. The RoBERTa-baseline achieved 0.48 on the official dev-set and 0.49 on the official test-set. The test-set labels are held out and will not be shared with you. We will use it to evaluate your model's performance after the coursework submission due date.

The coursework will be marked based on the quality of your research and the quality of your report rather than your final model performance. While you are expected to improve on the performance of the RoBERTa-base baseline, your report will be assessed on how well you answer the questions in the Marking Scheme section below rather than on your raw model performance.

Deadline: The coursework deadline is the 4th March (7pm). We encourage you to submit your work sooner!

Workload management: We recommend that you spend not more than 27 hours on this coursework. You may split the time into three uniform parts.
In part 1, you may spend upto 9 hours understanding the task, reading and reviewing the task paper and other literature, analysing the dataset, and planning your models and experiments.
In part 2, you may spend upto 9 hours developing/training the planned models and running the planned experiments.
In part 3, you may spend upto 9 hours to try any new experiments you realise later, compile your findings, analysis, and write your research paper/report.

Coursework submission: First, create a GitLab or a GitHub repository for this coursework. We require you to submit a PDF of your report in Scientia. Your report should have a link to your GitLab/GitHub repository on the first page. In this repository, you should have completed and pushed two files:
– Dev set predictions as `dev.txt`
– Test set predictions as `test.txt`

The dev set evaluation will be a public test, i.e. the labels are available for you to see. The test set evaluation will be a private test, i.e. the labels are not available to you. You will see your results on the test set when we mark your report and `test.txt`. The format of `dev.txt` and `test.txt` should be one output prediction per line. For example, the test set has 3832 lines of input text. So, `test.txt` should also have 3832 lines of predictions (0 or 1 in each line).

Your GitLab repo should also contain the code corresponding to your dev and test predictions.

Data and evaluation: The task data can be found here. You will be using the `dontpatronizeme_pcl.tsv` file. An allocation of this data into train and the official dev set is provided here. Since the labels of the official test set are held out, you will be using the official dev set to report your findings. Thus, you will need to create your own internal dev set from within the training data for the purpose of hyper-parameter tuning. As the test data is held out, creating an additional internal dev set allows you to use the official dev set as your own test set.

We ask you to submit the predictions of your best performing model, for both the official dev set and the official test set. The test set (without the labels) can be found here. We have the labels of the test set, so we will share your final test performance when marking your report.

Note, here the task repo also contains a breakdown of the type of PCL language detected for each example (broken down into seven categories). You are welcome to use this additional label information if it is helpful.

Results should be measured using the F1 score of the positive class (PCL examples).

Groups: We allow you to work in groups of upto 3 students. All members of the group will get the same marks. The group size will be taken into account in marking. How? Let A be the set of all groups of size 3, B be the set of all groups of size 2, C be the set of all groups of size 1. We will then ensure that the distribution of marks in A ≈ distribution of marks in B ≈ distribution of marks in C.

We will not deal with issues internal to a group, for example if a group member is not contributing then “deal with it”. This happens in real life too.

Code: You are required to upload your code to your GitHub/GitLab (if you are using a Colab notebook, please make sure this is uploaded to your repo). The code does not need to contain all your results and model configurations. Instead, this code should align with your final test submission. You will not be assessed on your code. It is just easy for us to refer or verify if needed.

Report: Your report should be a maximum of 5 pages using this template. You may also include any additional appendices, however the markers will not be required to look at these. Your report should contain the names and email addresses for each person in the group, but does not need to contain your affiliation (Imperial). You are not allowed to change the template, for example you cannot change the font size or the size of the margins.

Marking Scheme

1) Data analysis of the training data (15 marks): *For a written description of the training data. This should include:*

a. **5 marks**: Analysis of the class labels: how frequent these are and how they correlate with any feature of the data, e.g. input length.

b. **10 marks**: Qualitative assessment of the dataset, considering either how hard or how subjective the task is, providing examples in your report.

2) Modelling (40 marks): *For the successful implementation of a classifier model (this could be a transformer or any other ML model of your choice. Do give justification for your choice.):*

a. **10 marks**: Successful implementation of a model (train and produce predictions which outperform the F1 score for the RoBERTa-base baseline provided). 6 marks for outperforming the baseline model on the official dev set (0.48) and 4 marks for outperforming the baseline model on the test set (0.49).

b. **5 marks:** Choice of model hyper-parameters and description of your model setup. This should include choosing an appropriate learning rate and checking whether implementing a learning schedule improves performance. Also consider whether your model is cased or uncased. You should mention how many epochs you train the model for, whether you are using any early-stopping, and how you are using the training labels.

c. **10 marks:** Further model improvements (beyond using a bigger transformer model), for example pre-processing, data sampling, data augmentation, ensembling, etc. Two main improvements, with a third less explored improvement is sufficient. For example: try several different data sampling approaches, try several data augmentation strategies by perturbing observations in different ways, and then see if incorporating one of the categorical columns improves performance.

d. **10 marks**: Compare your model performance to two simple baselines (e.g. a BoW model). Share some of the features that one of your baseline models used, and highlight an example misclassified with a suggestion of why the baseline may have made the misclassification.

e. **5 marks**: Description of the model results and your hyper-parameter tuning (some evidence of this is required in your report). Your results should show how the different strategies you have tried impacted the model performance. For any results presented in your paper, you should be clear if these are from your own internal dev set or the official dev set.

3) Analysis (15 marks): *Analysis questions to be answered (these questions can be answered without training any additional models):*

Your report should state the analysis questions so that this can be read as a self-contained report, rather than referring to 'analysis question 1' etc.

a. **5 marks**: To what extent is the model better at predicting examples with a higher level of patronising content? Justify your answer.

b. **5 marks**: How does the length of the input sequence impact the model performance? If there is any difference, speculate why.

c. **5 marks**: To what extent does model performance depend on the data categories? E.g. Observations for homeless vs poor-families, etc.

4) Written report (30 marks): *Marks are awarded for the quality of your written report:*

a. **5 marks**: Introduction, with an explanation of the task and the dataset. You may want to read/cite the task paper (and any other paper of your choosing).

b. **10 marks**: Readability of the report (language, coherence, clarity of results etc.).

c. **10 marks**: Good use of graphs or results tables that address the analysis questions. Make sure any text on your graphs is clearly readable.

d. **5 marks**: Conclusion, with a summary of your results, and your key findings from the analysis questions. You should suggest at least one further experiment as a next step.