# Milestone 1 Project Plan

Nay Abi Akl, Mariam Hassan, Luca Zunino

May 14, 2023

# Contents

# 1 Introduction

In this report, we go over our preliminary plan to complete the second and third parts of our project. We discuss the data collection strategy, the choice of reward and final models, and the evaluation strategies.

# 2 Datasets

One of the main pillars of the project is the data used to train and evaluate the models on. So, in this section, the dataset for both tasks are chosen and explained.

## 2.1 Training Datasets

To diversify the training data, we will use a mix of datasets to include different types of questions and cover different topics. First, we aim to use our own annotated data as well as some of the other groups' data by sampling questions from different topics. We then aim to gather the different demonstrations available for each question and label them while adjusting the answers as needed. For example, for Multiple Choice Questions (MCQs), we can add extra prompts to let the model only mention the correct answers without the explanation. We could, also, use the ground truth answers to adjust the confidence levels reported by the students.
However, we believe that the collected data is noisy specially because not everyone is an expert in the topics they received so it cannot be used alone. Therefore, we will combine them with other data from online sources that have different question styles. These datasets are briefly described below:

- **CodeQA** [1] : CodeQA is an open source dataset that provide pairs of code snippet and a question that needs a text-based answer. They use the metrics BLEU, ROUGE-L, METEOR as a way of evaluation in the official implementation.
- **AI2** [2] : AI2 is an open source dataset by Allen Institute for AI that provide grad-school level MCQ questions and their answers. They provide an easy and challenging dataset and both could be used to provide a variety of difficulty-levels questions.
- **SQuAD** [3]: SQuAD is a question-answering open source dataset by Stanford that provides pairs of questions and answers driven from Wikipedia.

We will sample questions from all three datasets and gather at least 2 demonstrations for each sampled question, using at least two unified prompting strategies that will be chosen, and label them guided by the ground truth answers available in the dataset. This will help gather a diversified dataset with less noisy labelling as the ground truth solutions are already there. Concerning the size of final dataset, this will be a parameter to tune with regard to the amount of time we have while maintaining a large enough data for the final model. Since this process is manual and could be time-consuming, we are currently working on an automated approach to get prompt and get answers from the ChatGPT wrapper provided for this project. If proven that it is working well, we will be using this automated technique to gather the answers from the model and we will only be labelling them for the reward model.

## 2.2 Evaluation datasets

For the evaluation of the final model, one source is to use some of the unsampled questions from the datasets we used for the training. We could combine that with one or more new datasets to test the generalisation capabilities of our model. These can include:

- **OpenBookQA** [4]: This open-source dataset by Allen Institute for AI provides MCQ questions obtained from open-book exams so the questions rely on a broad knowledge of the topic as the answers are not always in the book.
- **TruthfulQA** [5]: This open-source dataset is a common benchmark for question-answering tasks as it includes questions about diverse topics that include finance, law, health and more.
- **HumanEval** [6] This dataset contains questions related to writing code and solving mathematical problems. It may not be very similar to what the model is trained on but it may be considered if it is suitable. Similar to the others, this dataset is open-source.

# 3 Methods

The second and third parts of the project are about training a reward model and training a final model respectively. In this section, we explain what possible models we considered and how we chose which to implement.

## 3.1 Reward model

A reward model is needed to rank the different demonstrations obtained for the datasets presented in section 2, based on how well the demonstrations fit the task presented by each question. In general, there are two options for the reward model architecture:

- Using an encoder-only model architecture (for example RoBERTa) and adding a linear layer on top of it to get the final score.
- Using a decoder-only model architecture (for example GPT family of models) and adding a custom linear layer on top of it for the scores.

We compare the two options, GPT and RoBERTa, on various factors while keeping in mind the specific task of grading/scoring text demonstrations:

- **Understanding context**: the bidirectional architecture of RoBERTa, allowing it to understand context from start and end of the text, gives it an edge over GPT models, which are unidirectional, in understanding general context, which is crucial in assessing the quality of the demonstrations.
- **Text generation**: GPT models are more powerful at the specific task of text generation. However, for evaluating demonstrations, existing text is used and no new responses need to be generated, making GPT's strength on this task irrelevant.
- **Transfer learning**: both GPT and RoBERTa models have proven to be good at transfer learning for various tasks. Our task of grading text quality/accuracy is comparable to sentiment analysis tasks, a field where Roberta like models have been widely used and shown to perform well.
- **Fine tuning**: both GPT and RoBERTa models can be fine-tuned for downstream tasks. What is more important for the fine tuning is having a good training dataset of demonstrations with their corresponding accuracy scores.
- **Computational resources**: GPT models, specially larger ones such as GPT-3, need significantly more resources compared to RoBERTa.

Based on the above factors, we believe that on the specific task of scoring text demonstrations, RoBERTa models are more suitable than GPT models, mostly relying on their bidirectional architecture for a better understanding of the full context.

Given the limited amount of dataset that we have for training the reward model, as compared to the amount of datasets that are usually used to train RoBERTa models, we decided not to train from scratch. Instead, in our implementation, we will fine-tune a pretrained RoBERTa model using the collected datasets and their labels.

### 3.2 Final model

In the final part of the project, we will fine tune a pre-trained generative language model to give better demonstrations on the course questions. Here also, we compare various factors of two models that are generally used and found in the literature: GPT-2 and T5 models, both powerful models to be considered for building a chatbot tutor.

- **Model architecture**: while the GPT model relies on a decoder-only transformer architecture trained with a left-to-right approach, T5 models rely on a encoder-decoder architecture, trained for text-to-text on various tasks including question answering and summarization.
- **Training flexibility**: by design, the T5 model can handle a large range of tasks by simply framing them as text-to-text, making it flexible in training and possibly have an advantage over GPT in handling the diverse task needed to answer the course questions (answering open ended questions, generating code, answering MCQ questions, etc).
- **Understanding of question context**: GPT, being unidirectional, predicts the following word based on the previous context, while T5, having an encoder-decoder architecture, can leverage the whole context of the question, making it potentially better at grasping the question, specially complex ones.
- **Text generation**: in terms of coherence and contextual relevance, both models perform well but it is worth noting that GPT is particularly famous for generating high quality text.
- **Fine tuning**: both models can be fine tuned for downstream tasks and have shown good performance in this field.

As seen from the above factors, the GPT-2 and T5 models both seem suitable for the application without much convincing evidence that one would be better than the other.

For this reason, we plan to compare the performance of the two models on a small portion of the dataset that we have and qualitatively compare the demonstrations obtained in order to choose a final model for our project. Note that we realize that such a small case study is not usually enough to compare large models, but since we consider both models to be suitable in the first place, we perform this small experiment to see which model would better fit the data we have, instead of randomly choosing one of the two.

## 4 Evaluation

### 4.1 Introduction

A vital part of our project involves identifying appropriate evaluation metrics. These metrics will enable us to ascertain whether the fine-tuned model performs effectively and surpasses a predefined baseline model. This section of the report will outline our proposed approach to evaluating the reward model and the final model, and discuss the different metrics we intend to use.

Our discussion will commence with the evaluation metrics that we plan to use for the reward model.

Then, we will consider different automatic evaluation metrics. These metrics will provide a precise and efficient method for quantifying the performance of our models.

Subsequently, we will explore the concept of baseline models. Establishing a comparative standard is essential for assessing the relative performance of our fine-tuned model. The baseline models will offer us this comparative perspective.

Finally, we will consider qualitative evaluation. While quantitative metrics are vital, they do not provide a complete picture. Qualitative evaluation complements these by offering a more nuanced understanding of our model's performance.

It should be noted that, in order to evaluate the model robustly, appropriate test datasets should be defined and used. In particular, we plan to use a subset of the questions composing the primary dataset (which has been distributed among the different groups) as the test dataset, and we will, therefore, not tune the model on it. Furthermore, to verify whether the model can generalize well to out-of-distribution examples, we will consider other datasets on which we will not finetune our model. For example, we could use OpenBookQA[4], TruthfulQA[5] or HumanEval[6], as already mentioned in Section 2.2.

## 4.2 Evaluation of the reward model

Since the reward model is essentially a classifier, its automatic evaluation is relatively straightforward. Indeed, it is possible to easily verify whether the score predicted by the reward model and the reference score are in agreement. To avoid issues related to imbalanced datasets, in addition to accuracy, we will consider the F1 score (harmonic mean of precision and recall). Since the classification problem will be multi-class (i.e., non-binary), we plan to consider the micro-averaged F1 score as an automatic evaluation metric.

## 4.3 Automatic evaluation metrics

Automatic evaluation metrics are typically straightforward to compute and offer a quick way to assess model performance. Therefore, we will extensively use these metrics to evaluate our fine-tuned model's performance and compare it to the baseline. Since, in the pool of questions we considered in this milestone, the two main types of questions were Open Questions (OQs) and Multiple Choice Questions (MCQs), we have decided to adapt our approach according to the question type.

- **Multiple Choice Questions (MCQs)**: For MCQs, the definition of a suitable evaluation metric is easier, as it is easier to evaluate the relationship between the answer choices predicted by the model and the reference answer choices. Indeed, for each MCQ, as the last interaction with the model, we plan to ask a question similar to the following: "Please indicate which answer choices are correct. Just indicate the letters associated to the correct answers separated by commas, without adding any other comment to your answer. For example, if answers B and D are correct, you should answer: B, D". Then, we will process the answers selected by the model and compare them with the correct answers to verify how accurate the model has been. In order to do so, a possible approach is to use accuracy. The issue is that, when dealing with imbalanced datasets (datasets in which the distribution of classes is uneven), accuracy alone might not be a reliable metric to evaluate the performance of our model (which, in this case, acts as a classification model). Indeed, just by always predicting the majority class, a model would be able to achieve high accuracy. For this reason, we plan to use also the F1 score to be robust to an imbalanced dataset. To use it in a multi-label classification problem, we plan to consider the micro-averaged F1 score.

- **Open Questions (OQs)**: For OQs, the definition of a suitable evaluation metric is less straightforward, as the answer returned by the model may be rephrased differently with respect to the reference one but be correct nevertheless. For this reason, N-gram overlap metrics (such as BLEU or ROUGE) may fail in evaluating the correctness of the answer returned by the model. While we are hesitant in completely disregarding these metrics due to their speed, efficiency, and correlation with human judgements of quality, the main metric we will employ is a model-based metric, BERTScore[7]. Indeed, BERTScore is much more suitable for our task as it allows us to use the pre-trained contextual embeddings from the BERT model to compute the semantic similarity between the candidate answer (provided by the model) and the reference answer (correct answer). In this way, a correct answer which is phrased in a very different way with respect to the reference one (and would therefore be strongly penalised by N-gram overlap metrics) would still obtain a high BERTScore due to its semantic similarity with the reference.

Furthermore, for all types of answers, we plan to use our reward model to evaluate the quality of the answers generated by the model. This will be helpful in verifying the answer quality and whether the reward model is effective in evaluating the answers.

## 4.4 Baseline models

In order to verify whether finetuning the model helped improve the performances, it will be necessary to identify some baseline models to serve as a comparison. Therefore, we plan to consider two main baseline models:

- **Non-finetuned final model**. As stated in the previous section, we plan to use either GPT-2 or T5 as the final model and to finetune this to improve performance. Therefore, in order to understand whether the finetuning process has been effective in improving the quality of the answers, the most suitable baseline is the non-finetuned model chosen (i.e., either the pretrained GPT-2 or T5);

- **Other non-finetuned models**. In order to extend our comparison to other models which are broadly available, we may identify other non-finetuned models as a baseline. A model which seems particularly reasonable for this is GPT-3, even if the large difference in scale may make the comparison unfair. In any case, especially if we observe that the finetuned final model significantly outperforms the non-finetuned one, it may be interesting to verify the gap between the finetuned final model and a model which is significantly bigger such as GPT-3.

## 4.5 Qualitative evaluation

Since the task the model should perform is complex, assessing the quality of the generated answers using an automatic metric is not straightforward. Even a metric taking into account the semantic similarity, such as BERTScore, will not be a comprehensive evaluation of the quality of the generation, as a good answer must be fluent, clear, knowledgeable, stylistically pleasant, and demonstrate a clear understanding of the topic. For this reason, we plan to perform a qualitative evaluation study involving experts in the field of the questions in order to gather more accurate and meaningful feedback on the quality of the answers. Since this study is difficult to be scaled, we plan to perform it in parallel to a more classical evaluation (performed using evaluation metrics) and to consider it a way of validating/invalidating the results obtained with the automatic metrics. To perform this human-based qualitative evaluation, we plan to find a group of experts in the field of the questions (ideally PhD students, if not available MSc students) and perform two experiments:

- In the first experiment, we intend to conduct a graded evaluation of the model's responses. The chosen group of experts will judge the correctness, informativeness, and utility of the model's answers on a scale from 1 to 10, where 1 represents the least satisfactory and 10 signifies the most satisfactory response. This method allows us to measure the "average satisfaction score", which represents the mean score given by the experts for the model's responses. It indicates the degree to which the model's responses are perceived as accurate, informative, and helpful for task completion or understanding a particular topic;
- For the second experiment, we plan to introduce comparative metrics. In this case, we will present the expert panel (different from the first one to prevent data contamination) with two variables: the answer produced by the finetuned model, and the response from the baseline model. The experts will be asked to rank these answers based on their quality and factual correctness. This approach allows us to compute the "win rate" of the finetuned model, which is the percentage of times the model's answer is ranked superior to the baseline model. Similarly, we can also track the "factual accuracy rate", which is the percentage of times the experts judge the model's answers as being factually correct. Furthermore, we may also introduce the reference or correct answer to verify whether the experts are able to distinguish it from the one generated by the finetuned model.

By considering the evaluation obtained by the experts in these two experiments, it would be possible to determine whether the finetuned model can produce convincing and valuable answers for a university student.

## 5 Conclusion

For the datasets, we will be sampling questions from different topics from the dataset collected by the whole class and we will be adding the CodeQA, AI2, and the SQuAD datasets for the training datasets. We will also be using examples from different datasets for the evaluation such as OpenBookQA, TruthfulQA,, and HumanEval. We are working on a way to automate some the prompting of some of these questions in order to be able to gather a fitting amount of data. For the reward model, a pre-trained RoBERTa model will be used. For the final model, pre-trained GPT-2 and T5 models will be compared on a small portion of the dataset and the model with the better demonstrations (qualitatively analyzed) will be used.

For the evaluation, we will consider the micro-averaged F1 score in the evaluation of the reward model and as an automatic evaluation metric for MCQs. With OQs, we will consider BERTScore as the main automatic evaluation metric, eventually supported by N-gram overlap metrics such as BLEU or ROUGE. As baseline models, we will consider both the non-finetuned final model and other non-finetuned models, such as GPT-3. Finally, we plan to perform a qualitative evaluation to better assess the quality of the generated answers by involving human experts in the evaluation.

# References

[1] C. Liu and X. Wan, "Codeqa: A question answering dataset for source code comprehension," 2021.

[2] P. Clark, I. Cowhey, O. Etzioni, T. Khot, A. Sabharwal, C. Schoenick, and O. Tafjord, "Think you have solved question answering? try arc, the ai2 reasoning challenge," 2018.

[3] P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang, "Squad: 100,000+ questions for machine comprehension of text," 2016.

[4] T. Mihaylov, P. Clark, T. Khot, and A. Sabharwal, "Can a suit of armor conduct electricity? a new dataset for open book question answering," in *EMNLP*, 2018.

[5] S. Lin, J. Hilton, and O. Evans, "Truthfulqa: Measuring how models mimic human falsehoods," 2022.

[6] M. Chen, J. Tworek, H. Jun, Q. Yuan, H. P. de Oliveira Pinto, J. Kaplan, H. Edwards, Y. Burda, N. Joseph, G. Brockman, A. Ray, R. Puri, G. Krueger, M. Petrov, H. Khlaaf, G. Sastry, P. Mishkin, B. Chan, S. Gray, N. Ryder, M. Pavlov, A. Power, L. Kaiser, M. Bavarian, C. Winter, P. Tillet, F. P. Such, D. Cummings, M. Plappert, F. Chantzis, E. Barnes, A. Herbert-Voss, W. H. Guss, A. Nichol, A. Paino, N. Tezak, J. Tang, I. Babuschkin, S. Balaji, S. Jain, W. Saunders, C. Hesse, A. N. Carr, J. Leike, J. Achiam, V. Misra, E. Morikawa, A. Radford, M. Knight, M. Brundage, M. Murati, K. Mayer, P. Welinder, B. McGrew, D. Amodei, S. McCandlish, I. Sutskever, and W. Zaremba, "Evaluating large language models trained on code," 2021.

[7] T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, and Y. Artzi, "Bertscore: Evaluating text generation with bert," 2020.