RETUYT-InCo at BEA 2023 Shared Task: Tuning Open-Source LLMs for Generating Teacher Responses

Alexis Baladón and **Ignacio Sastre** and **Luis Chiruzzo** and **Aiala Rosá**
Facultad de Ingeniería
Universidad de la República
Montevideo, Uruguay
{alexis.baladon,isastre,luischir,aialar}@fing.edu.uy

Abstract

This paper presents the results of our participation in the BEA 2023 shared task, which focuses on generating AI teacher responses in educational dialogues. We conducted experiments using several Open-Source Large Language Models (LLMs) and explored fine-tuning techniques along with prompting strategies, including Few-Shot and Chain-of-Thought approaches. Our best model was ranked 4.5 in the competition with a BertScore F1 of 0.71 and a DialogRPT final (avg) of 0.35. Nevertheless, our internal results did not exactly correlate with those obtained in the competition, which showed the difficulty in evaluating this task. Other challenges we faced were data leakage on the train set and the irregular format of the conversations.

1 Introduction

Nowadays, with the important development of Large Language Models (LLM) and their great generative power, the interest in the development of chatbots that simulate interactions between humans has increased. In particular, in the educational domain, the use of chatbots seems to have interesting benefits, such as their potential for adaptive learning, tailored to each student, or their permanent availability (Bibauw et al., 2022).

The contributions of these tools to learning are not yet clear (Wollny et al., 2021). In their review of the area, these authors conclude that the development of chatbots is usually based on technological criteria, but the focus has not yet been placed on their pedagogical contributions in terms of learning improvements.

However, there is some evidence that for language learning in particular, these tools bring certain benefits (Bibauw et al., 2022), mainly for students at initial levels. It should be noted that in the case of language teaching, interaction with the agent is in itself an instance of learning practice.

One aspect to be studied in the development of educational chatbots is their ability to understand students needs and respond with the style that teachers, trained to educate, use to address their students (Bommasani et al., 2021). Although current LLMs show great capacity for language generation and for providing relevant -although not always correct or true- answers to different types of queries, it is important to study whether these models can be used in an educational context, being able to respond to a student by simulating a dialogue with a teacher. (Tack and Piech, 2022) propose such an evaluation called the AI teacher test challenge.

This paper presents the RETUYT-InCo submission to the BEA 2023 shared task (Tack et al., 2023) on generating teacher responses in educational dialogues. In this work, we analyze some particularities of the dataset used in the competition, we describe the approaches we made to solving the problem, and we present the results we obtained, together with an analysis and discussion of future steps.

2 Data analysis

The following study aims to understand the patterns and characteristics of the conversations between teachers and students, which will be crucial for training a chatbot to generate appropriate responses.

2.1 Dataset content

First, it is important to consider the description provided on the official BEA Shared Task webpage[1] and the source of the corpus used in this study. According to the information available, the corpus consists of extracts from 102 different chatrooms where an English teacher engages in language exercises and assesses the English language proficiency of the students (Caines et al., 2020). Each

[1] https://sig-edu.org/sharedtask/2023

Proceedings of the 18th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2023), pages 756–765
July 13, 2023 ©2023 Association for Computational Linguistics

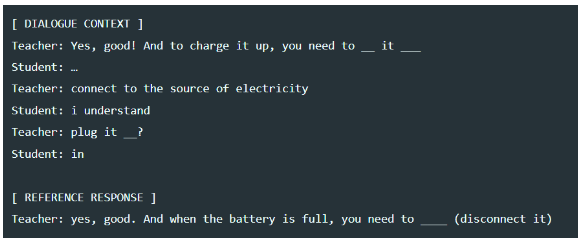
extract comprises a series of **utterances**, representing turns by the teacher and the student, along with a **response** that, as per the competition prompt, always originates from the teacher. This distinction is vital as the objective is not simply to continue a conversation but to respond from the perspective of a teacher.

Secondly, upon inspecting the corpus, it was revealed that the dataset contained additional sets of conversations beyond the original composition, as described in the corpus paper (Caines et al., 2020) and the provided website, which stated a total of 102 conversations. Hence, we assumed the corpus was composed with a set of extracts from each of those conversations, implying the data inside the corpus is not completely dependent. Interestingly, during the examination of the training corpus, numerous tuples were found to be partially duplicated, indicating that the conversations in the training set were derived from overlapping segments of the same original conversations. This issue is critical due to two main reasons. First, it is important to note that each teacher's response does not correspond to the final utterance of the entire conversation but rather the last utterance within an extract from the conversation (similarly for the first utterance). Moreover, this poses a significant challenge when it comes to the typical validation approach of partitioning the dataset, as it is not immediately evident how to separate each conversation in a manner that prevents data leakage across corpus partitions without hindering the model's training.

2.2 Other relevant findings

There are several noteworthy characteristics of the dataset to consider. Firstly, one of the initial examples showcased on the official website features a student attempting to solve a task involving filling a gap with a word or short phrase (see Fig. 1). However, upon inspecting the number of conversations that contain at least one underscore character (_), it was found that only 14.89% of them met this criterion. Consequently, while this restriction does not significantly impact the further architecture of the model, it is worth mentioning that incorporating this aspect could potentially enhance the model's performance in future work.

Furthermore, some tasks within the dataset involve choosing between two options (a) or (b) type questions. However, due to the fact that these types of questions account for less than 1% of the total corpus, the decision was made not to thoroughly analyze them in this study.



```
[ DIALOGUE CONTEXT ]
Teacher: Yes, good! And to charge it up, you need to __ it ___
Student: …
Teacher: connect to the source of electricity
Student: i understand
Teacher: plug it __?
Student: in

[ REFERENCE RESPONSE ]
Teacher: yes, good. And when the battery is full, you need to ____ (disconnect it)
```

Figure 1: Example of conversation extract

In addition, an examination of the dataset's tags reveals a variety of categories, including <STUDENT>, <TEACHER>, <ANOTHER STUDENT>, <CAT'S NAME>, <LIZARD'S NAME>, and others. Notably, students and teachers represent over 90% of the tags. The presence of specific names and references to animals suggests that the dataset covers a wide range of topics related to conversations between teachers and students. A table displaying the most frequent tags count can be found in Table 1.

Tag	Count
<STUDENT>	868
<TEACHER>	141
<ANOTHER STUDENT>	19
<CAT'S NAME>	18
<LIZARD'S NAME>	17
<STUDENT'S SHORT NAME>	7
<CAT'S NAME1>	5
<STUDENT'S FULL NAME>	5
<LIZARD'S NAME'S>	4
<TEACHER'S NAME>	3

TABLE 1: 10 Most Frequent Tags in the Dataset

2.3 Proportion of conversation utterances

In addition to examining other aspects of the corpus, it is important to analyze whether the conversations exhibit any form of imbalance. Intuitively, one might expect the student to be more hesitant in their participation due to a lack of confidence, or conversely, the teacher may encourage the student to contribute more in order to facilitate learning. Therefore, the rate of text length expressed by each participant was assessed using two different measures: the length of tokens and the number of conversation turns.

To tokenize the sentences in the dataset, we used NLTK's wordtokenize function (Bird et al., 2009). To understand the distribution of tokens (see Fig. 2, the analysis considered the token count for each part of the conversation, namely the teacher, the student, and both. The teacher's responses had an average of 11.18 tokens, with a standard deviation of 9.37. The student's responses had an average of 6.00 tokens, with a standard deviation of 6.49. When considering both parts of the conversation, the average token count was found to be 9.07. These findings suggest that the model should generate responses that are generally longer than those found in the dataset.

Subsequently, it was measured the same proportion taking only into consideration the number of utteranaces by each speaker. The analysis indicates that teachers account for 59.47% of the total conversation turns. However, it is important to acknowledge that this imbalance in the data is a direct consequence of the last tuple always being the teacher's response. It is also worth highlighting that the turns do not always follow an alternating pattern based on the speaker, as there are instances where the same speaker appears consecutively. This deviation from the typical conversational pattern can present a challenge when training conversational chatbots that rely on alternating inputs from different speakers.

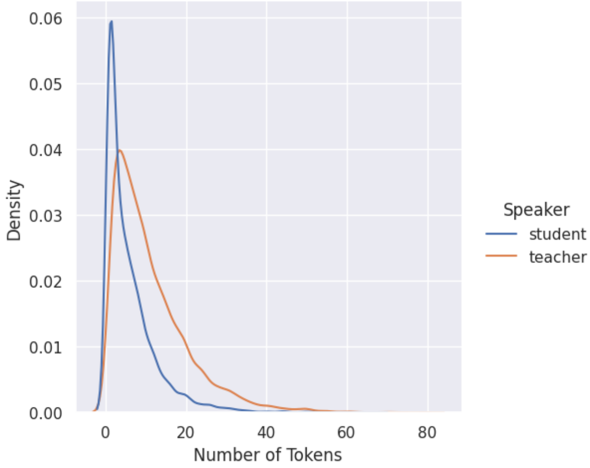
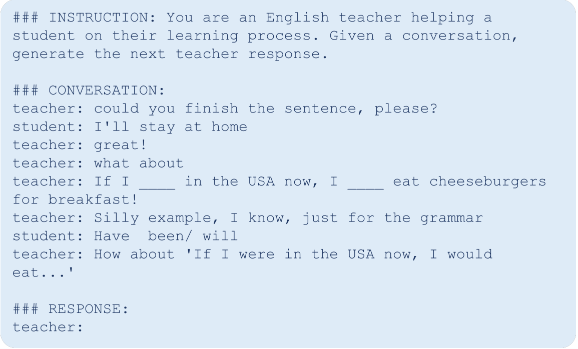


Figure 2: Student and Teacher's token distribution

3 Experiments

This section described the systems implemented to solve the task.



```
### INSTRUCTION: You are an English teacher helping a
student on their learning process. Given a conversation,
generate the next teacher response.

### CONVERSATION:
teacher: could you finish the sentence, please?
student: I'll stay at home
teacher: great!
teacher: what about
teacher: If I ____ in the USA now, I ____ eat cheeseburgers
for breakfast!
teacher: Silly example, I know, just for the grammar
student: Have  been/ will
teacher: How about 'If I were in the USA now, I would
eat...'

### RESPONSE:
teacher:
```

Figure 3: Prompt used with Alpaca LoRA applied to the example with id *train_1504* from the training set.

3.1 Using pretrained Large Language Models

Our first approach was trying out open source pretrained Large Language Models (LLMs), such as the model LLaMA (Touvron et al., 2023) and a finetuned version for following instructions available in Hugging Face, Alpaca LoRA[2].

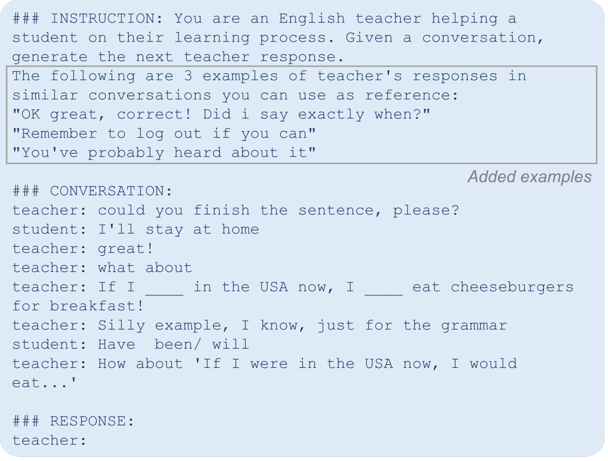
The dataset used for the fine-tuning of Alpaca LoRA is the one provided in (Taori et al., 2023), where each example is composed of three sections (the second is optional): *Instruction*, where the task is described, *Input*, which is an optional context for the task and *Response*, which is the answer to the instruction.

We designed a prompt following this format but we adapted it to integrate the whole conversation to the context. An specific instruction was designed for this task, and it is provided in the *Instruction* section. The input section was changed for a conversation section, where the utterances are presented in a classical chat format. The response section always starts with "teacher:", influencing the model to generate a continuation for the conversation as a teacher. An example is presented in Fig. 3.

Following this experiment, we used an adaptation of the Few-Shot approach explained in (Brown et al., 2020), in order to influence the generated responses with the teacher's style. For choosing the examples provided in the prompt, we used sentence embeddings generated with the `gtr-t5-large-1-epoch` model in hugging face[3]. An embedding was generated for each of the utterances in the training set partition. For generating a new response, the previous utterances are

[2] https://huggingface.co/tloen/alpaca-lora-7b

[3] https://huggingface.co/cohere-io/gtr-t5-large-1-epoch



```
### INSTRUCTION: You are an English teacher helping a
student on their learning process. Given a conversation,
generate the next teacher response.
The following are 3 examples of teacher's responses in
similar conversations you can use as reference:
"OK great, correct! Did i say exactly when?"
"Remember to log out if you can"
"You've probably heard about it"
```

Added examples

```
### CONVERSATION:
teacher: could you finish the sentence, please?
student: I'll stay at home
teacher: great!
teacher: what about
teacher: If I ____ in the USA now, I ____ eat cheeseburgers
for breakfast!
teacher: Silly example, I know, just for the grammar
student: Have  been/ will
teacher: How about 'If I were in the USA now, I would
eat...'

### RESPONSE:
teacher:
```

Figure 4: Few-Shot prompt used with Alpaca LoRA applied to the example with id train_1504 of the training set.

converted into an embedding and the three most similar conversations are selected from the training set using the k-Nearest Neighbors technique. The three responses of these selected examples are then added to the prompt, as can be seen in Fig. 4.

3.2 Fine-tuning pretrained Large Language Models

Pretrained LLMs tend to perform well in various tasks due to scaling up of model size, dataset size diversity, and length of training (Brown et al., 2020). However, using these models only with prompting techniques does not allow adapting to a target domain or target task, nor fully leveraging the potential of the training dataset.

Fine-tuning is the process of updating the weights of a pre-trained model by using a domain specific dataset in the training step. This technique tends to obtain strong performance in many benchmarks (Brown et al., 2020). However, it can be computationally very costly as all parameters of the LLM need to be updated. This is a major constraint, and sets a limit to the size of the models that we are able to fine-tune.

For this experiments we used the CluserUY infrastructure (Nesmachnow and Iturriaga, 2019), which has two servers using NVIDIA A100 GPUs and 28 servers using NVIDIA P100 GPUs.

3.2.1 Experiments updating all the weights

DialoGPT is a transformer conversational model developed by Microsoft. It is based on the architecture of GPT2, which is known for its effectiveness in generating coherent and contextually relevant text. The specific implementation of DialoGPT used in our study is `microsoft/dialogpt-large`, which has 762 million parameters (Zhang et al., 2020b).

During training, DialoGPT was exposed to a vast amount of data, including 147 million conversation-like exchanges. These exchanges were extracted from Reddit comment chains spanning from 2005 through 2017. This diverse and extensive training data helped DialoGPT learn to generate responses that resemble human-like conversations.

As mentioned in (Zhang et al., 2020b), the human evaluation results demonstrate that the responses generated by DialoGPT exhibit a level of quality comparable to human responses in a single-turn conversation Turing test. Considering that the competition assesses the similarity to human responses as a metric, leveraging DialoGPT's performance has the potential to enhance the metrics of our model results.

It is important to note that in our study, we trained DialoGPT without specifically optimizing its architecture or training process. Our primary intention was to assess whether a conversational model like DialoGPT could achieve comparable performance to other existing models.

3.2.2 Experiments using Low-Rank Adaptation

The high computational requirements for fine-tuning big LLMs, such as LLaMA 7b, posed a significant challenge even with access to the ClusterUY infrastructure. The process is not only computationally costly but also time consuming, which makes the task of training and testing various fine-tuned models with different base models or prompting techniques impractical. To overcome these restrictions, we opted to use Low-Rank Adaptation (LoRA) (Hu et al., 2021) for fine-tuning the bigger models.

LoRA is a method for fine-tuning models which aims to reduce GPU memory requirement by freezing the pretrained model weights and injecting trainable rank decomposition matrices into each layer of the Transformer architecture, reducing the amount of trainable weights. This method not only reduces computing and time requirements, but also space requirements because only the rank decomposition matrices need to be stored, which have much less parameters than the original matrices.

For example, suppose $W \in \mathbb{M}_{m \times n}$ is a weight matrix and $\Delta W \in \mathbb{M}_{m \times n}$ is the weight update we want to learn. As shown in (Raschka, 2023), instead of learning ΔW, we can decompose it into two smaller matrices: $\Delta W = W_m W_n$, where $W_m \in \mathbb{M}_{m \times r}$, $W_n \in \mathbb{M}_{r \times n}$ and r is a small number called rank. Keeping the original weights frozen and only training these new matrices results in reducing the amount of trainable parameters from $m * n$ to $m * r + r * n$. After training, the new parameters are obtained by doing: $W + W_m W_n$.

Using the LoRA method, we trained fine-tuned versions of OPT 2.7b (Zhang et al., 2022), Bloom 3b (Scao et al., 2022) and LLaMA 7b (Touvron et al., 2023). For generating the dataset necessary to train all of these models, we adapted the training set in the following manner: The utterances and the response were joined into a string with a classical chat format, where every teacher intervention starts in a new line with *"teacher:"* and every student intervention starts in a new line with *"student:"*.

The configuration used for fine-tuning these models with LoRA involved a rank of 16, a scaling factor for the weight matrices of 32, and a dropout probability for the LoRA layers of 0.05. The training process employed the AdamW optimizer, with a total of 200 training steps, a learning rate of 2×10^{-4}, and a batch size of 4.

3.3 Preprocessing and Fine-Tuning

3.3.1 Preprocessing technique

Upon analyzing the results during the development phase, we observed a recurring issue where the model became confused when attempting to continue the conversation from the teacher's perspective after the same teacher had spoken. This discrepancy stemmed from the dataset's structure, as it did not adhere to the conventional alternation of turns between speakers, which the models typically expect.

Consequently, even when explicitly specifying that the model should respond as a teacher in the prompt or using an input format like "Teacher: <*Sentence-Before-Response*>\n Teacher:", the models consistently generated responses from the student's standpoint. This posed a significant challenge not only during the model's training phase, where it could become perplexed by the corpus structure, but also during the validation process.

To address this issue, we implemented two modifications:

Corpus Modification: We adjusted the corpus by introducing a structural change. Whenever two consecutive conversations appeared in the original corpus, we combined them into a single utterance separated by a period. This alteration aimed to create longer utterances that would help the model distinguish between student and teacher interactions.

Test-time Adjustment: During testing, if the last utterance belonged to a teacher, we introduced an auxiliary phrase into the corpus. This additional phrase was carefully crafted to avoid introducing new information to the conversation, ensuring it did not hinder the teacher's train of thought. We opted for the phrase "Student: I see\n," a common expression used in the corpus and everyday conversations to convey active listening and encourage the other person to continue speaking.

By employing these preprocessing techniques, we sought to improve the model's performance by aligning its responses more closely with the intended teacher's perspective while overcoming the challenges posed by the dataset's structure.

3.3.2 Fine-Tuned model using the preprocessing technique

The model in which we used this ad-hoc technique was `opt-2.7b` (Zhang et al., 2022). OPT, developed by Meta, is a decoder-only language model closely related to GPT-3. It has been predominantly pretrained on English text, supplemented with a small amount of non-English data obtained from CommonCrawl. The model's pretraining process employed a causal language modeling (CLM) objective, similar to other models in its family. Evaluation of OPT aligns with the prompts and experimental setup used for GPT-3 (Brown et al., 2020).

The decision to employ OPT in this study was motivated by the aim of exploring an alternative that offers both variety and considerable power. However, it is crucial to acknowledge and address the limitations of this model. Meta AI's model card highlights that OPT's training data consists of unfiltered internet content, resulting in a significant bias embedded within the model.

The configuration used for fine-tuning this model was the AdamW optimizer, a learning rate of 0.001 and a batch size of 4.

3.4 Combining prompting techniques with fine-tuning

After experimenting with prompt-based and fine-tuning approaches, a natural evolution was to look for ways to combine both of these techniques. Our first approach was to fine-tune the model LLaMA 7b with LoRA using the already explained few-shot method. In the same way as before, the three most similar responses in the training set with respect to the reference response were chosen to be added to the context. We took into consideration that responses from different partitions of the same conversation should not be considered for this selection. We expected that during fine-tuning, some patterns that could exist between the similar responses and the expected response could be learned.

Recent works like (Wei et al., 2023) showed that adding intermediate reasoning steps that lead to the final answer for a problem improves the ability of LLMs to perform complex reasoning. Inspired on this work, we designed a different solution that tries to combine intermediate reasoning and fine-tuning.

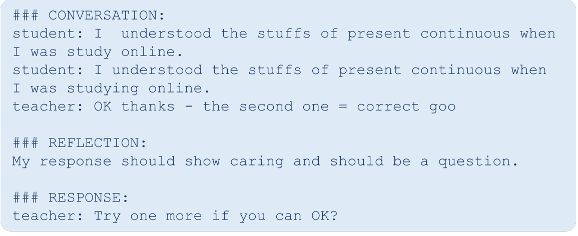
The training set was modified to include some characteristics of the response. Initially, two new features were added. A binary feature that is set to 1 if the response has a question mark, and a multiclass feature that is composed of 28 emotions taken from (Demszky et al., 2020), such as anger, approval, curiosity, disapproval, neutral and others. To obtain the second feature for every example in the training set, the EmoRoBERTa model was used (Kamath et al., 2022). This model classifies text into the 28 emotions already mentioned.

Then, a dataset for fine-tuning was constructed. Each example of the dataset is a string composed of three sections: *Conversation*, where the utterances are presented in a classical chat format, *Reflection*, which is constructed using the already mentioned features, and *Response*, which has the reference response.

The *Reflection* section is a sentence with two parts: The first part indicates the expected emotion of the response and the second part, which is optional, indicates if the expected response is a question. For example, an example classified as *"Curiosity"* and that is a question would have the reflection: *"My response should show curiosity and should be a question"*. A complete example can be seen in figure 5.

Using this dataset, we fine-tuned LLaMA 7b with the already mentioned LoRA technique. Given a new conversation, the model is capable of generating a complete reflection and response. The reflection is discarded to get the final response.

A second version was created using a new feature that classifies the response length in short, normal or long. A response is considered short if it has 20 characters or less and long if it has 53 characters or more. These numbers were selected in order to divide the dataset in the most balanced way (approximately 1/3rd for each class) as can be appreciated in Fig. 6. The reflection sentence was changed to include this information.



```
### CONVERSATION:
student: I  understood the stuffs of present continuous when
I was study online.
student: I understood the stuffs of present continuous when
I was studying online.
teacher: OK thanks - the second one = correct goo

### REFLECTION:
My response should show caring and should be a question.

### RESPONSE:
teacher: Try one more if you can OK?
```

Figure 5: Example of the prompt used for the reflection approach dataset.

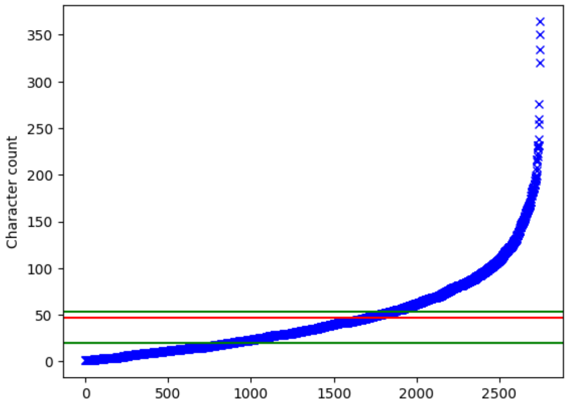


Figure 6: Character count per example in the training set, in ascending order. The green lines indicate the thresholds of each class, and the red line indicates the average.

4 Results

Given that this work is framed in the context of the BEA 2023 shared task, and the development and test sets gold responses were not released until after the competition finished, we created our own internal split of the training set in 80% for training and 20% for internal validation. We will present the results of all our experiments against this internal validation data, which we call the internal validation phase. For the development and test sets, we will only present the results of the systems submitted to the competition.

A problem with this internal split, as already explained in the data analysis section, is that it includes some repeated utterances across the training and validation sets, due to the overlapping that oc-

Experiment	**BERTScore**			**DialogRPT**		
	Precision	**Recall**	**F1 Score**	**updown**	**human_vs_rand**	**human_vs_mach**
Finetuning (LoRA) Bloom 3b	0.840	0.838	0.838	0.495	**0.912**	0.985
Finetuning (LoRA) Llama 7b + Reflection	0.808	0.840	0.823	0.463	0.881	0.995
Alpaca LoRA	0.832	0.829	0.830	0.489	0.841	0.986
Alpaca LoRA + Few Shot	0.836	0.836	0.836	0.480	0.820	0.989
Finetuning (LoRA) Llama 7b + Few Shot	0.802	0.839	0.819	0.465	0.871	**0.997**
Finetuning (LoRA) opt 2.7b	0.841	0.832	0.836	0.478	0.748	0.966
Finetuning opt 2.7b	0.847	**0.842**	0.844	0.474	0.673	0.981
Finetuning (LoRA) Llama 7b	0.854	0.841	**0.847**	0.473	0.642	0.965
Finetuning (LoRA) Llama 7b + Reflection with length	0.850	0.831	0.840	0.465	0.595	0.985
Finetuning DialoGPT Large	0.700	0.667	0.682	0.462	0.592	0.959
Baseline 1: Always reply "Hello"	**0.861**	0.805	0.832	**0.524**	0.305	0.952
Baseline 2: Always reply "Cucumber"	0.723	0.810	0.764	0.503	0.360	0.992

TABLE 2: Internal validation results.

curs in some of the training set tuples. This may influence the results during evaluation, but we decided to keep it this way so as not to significantly reduce the training set partition.

Two evaluation metrics are used in all phases, following the indications given in the official website of the shared task[4]: One of them is BERTScore (Zhang et al., 2020a), which produces precision, recall, and F1 scores by comparing words in the generated response with respect to the reference response using cosine similarity. The other one is DialogRPT (Gao et al., 2020), which evaluates the generated response taking into account the utterances given as context. The specific DialogRPT metrics used are updown, human_vs_rand, human_vs_machine and final (average and best).

4.1 Internal evaluation

Due to the fact that both metrics have multiple hyperparameters that can be tuned differently, the configuration used during this internal phase does not align exactly with the one used in the competition. For the BERTScore metric, roberta-large is used as the base model and idf weighting is not used. Meanwhile, for DialogRPT, the context used are the utterances concatenated in a classical chat format and the hypothesis is the generated response.

Trying out different configurations for DialogRPT, we found out that the definition of the context to be used has a big influence on the results obtained. As no information was provided on how the context was going to be defined in the development and evaluation phases, we made our own definition and used it consistently during all our internal evaluations.

The results obtained during the internal evaluation for all the described experiments can be observed in Table 2. Besides all the methods described, we include two very simple methods that serve as baselines to compare with. In both cases the baseline systems generate the same response to all contexts. One baseline always replies "Hello", and the other always replies "Cucumber", so as to consider a more likely and a more unlikely case.

4.2 Development and evaluation phases

For the development phase, we decided to submit the LoRA fine-tuning of the model LLaMA 7b, which had the best F1 score in the internal phase, the model Alpaca LoRA with the Few-Shot technique for the prompt, and the fine-tuned version of DialoGPT. We chose to submit these models because each of them uses a different approach: fine-tuning with LoRA, a prompting technique, and fine-tuning updating all the weights, respectively. It is important to mention that not all the experiments were completed when the deadline for this phase occurred.

Due to an error in the calculation of BERTScore on CodaLab[5], the results obtained in the development phase were not correct. This influenced our decisions of what models to send to the evaluation phase, given that our internal evaluations did not seem to correlate with these obtained results. The corrected results were later published, and can be seen in Table 3.

Considering that the Alpaca LoRA with Few-Shot approach was the one that yielded the best results in the development phase, we decided to also submit it in the evaluation phase. Two new approaches were also submitted: the LoRA fine-

[4] https://sig-edu.org/sharedtask/2023#evaluation

[5] https://codalab.lisn.upsaclay.fr/

Experiment	BERTScore			DialogRPT				
	Precision	Recall	F1 Score	updown	human_vs_rand	human_vs_mach	final (avg)	final (best)
Finetuning (LoRA) Llama 7b	**0.72**	**0.70**	**0.71**	0.36	0.94	0.98	0.32	0.67
Alpaca LoRA + Few Shot	0.68	0.69	0.68	**0.37**	**0.95**	**0.98**	**0.33**	**0.72**
Finetuning DialoGPT Large	0.70	0.67	0.68	0.35	0.92	0.98	0.30	0.68

TABLE 3: Development phase results.

tuning of LLaMA 7b with reflection in the prompt, and the fine-tuning of OPT 2.7b with preprocessing. Table 4 shows the results obtained for this phase, evaluated over the test set.

4.3 Observations

We observed that fine-tuning a model updating all the weights does not show significant differences in comparison to using the LoRA technique. On a separate note, the results reveal that fine-tuned models seem to improve the BERTScore results over prompting techniques, but the opposite seems to happen with DialogRPT metrics. The experiments that try to combine both techniques tend to show competitive results across all metrics.

Another observation that derives from the internal results (Table 2), is that the "Hello" baseline approach not only yields good results in the majority of the metrics, but is also the best in BERTScore precision and DialogRPT updown. This seems to indicate that these metrics (at least with our configuration) may not fully capture or accurately correlate with human judgement.

5 Conclusions

We presented the experiments we performed for the BEA 2023 shared task on generating teacher responses in educational dialogues. Our methods use the latest open source LLMs in a variety of scenarios and incorporating some fine-tuning and targeted prompting strategies for improving the performance.

The experiment that yielded best results in the development phase was the model Alpaca LoRA with a Few-Shot prompting technique, which ranked third. However, in the evaluation phase, the Fine-Tuning version of OPT 2.7b with preprocessing ended up performing better than the previous one, and ranked fourth in this phase.

5.1 Areas of Improvement

Throughout the competition, several areas were identified where improvements could have enhanced the performance of our chatbot model.

On the one hand, further fine-tuning of the model's parameters could have been explored to optimize its performance. By carefully tuning hyperparameters, we could have potentially achieved better results in terms of response quality and coherence. Additionally, despite training our models using high-performance GPUs (e.g., A100 and P100), we faced limitations in testing models with more than 10 billion parameters. Given the advancements in model architectures, exploring larger models could have yielded further improvements in chatbot performance. Overcoming hardware limitations and resource constraints would open avenues for investigating more powerful models in future iterations. Moreover, to resource and time constraints, our models could not be trained for different number of epochs. Longer training durations are often beneficial for improving model performance. Given more resources and time, training the models for multiple epochs could have yielded better results.

On the other hand, one challenge encountered during the competition was data leakage between the internal validation set and the training set. This issue, arising from the training dataset, hindered the models' ability to accurately improve their performance without overfitting. A more carefully curated validation set, separate from the training data, would have provided a more reliable evaluation metric. Furthermore, regarding the evaluation metircs, BERTScore and DialogRPT, we observed questionable scores when comparing our model's performance against a baseline of answering "hello" for every prompt. The BERTScore showed unexpectedly high scores for this baseline, while DialogRPT correctly penalized such responses. On top of that, another baseline that responded with a fixed word "cucumber" consistently scored poorly, which aligns with our expectations. Careful consideration and refinement of our evaluation metrics are necessary to ensure their reliability and alignment with the desired behavior of chatbot models.

Experiment	BERTScore			DialogRPT				
	Precision	Recall	F1 Score	updown	human_vs_rand	human_vs_mach	final (avg)	final (best)
Finetuning (LoRA) Llama 7b + Reflection	0.73	**0.71**	**0.72**	0.37	**0.94**	**0.98**	0.33	0.64
Finetuning opt 2.7b	**0.74**	0.68	0.71	0.38	0.90	0.96	**0.35**	0.65
Alpaca LoRA + Few Shot	0.72	0.68	0.70	0.37	0.91	0.96	0.34	**0.68**

TABLE 4: Evaluation phase results.

5.2 Ethical limitations

It is essential to address the ethical limitations observed our fine-tuned OPT model, ranked 4th in the competition. The model card provided by Meta AI highlighted that the training data used for their model consisted of unfiltered internet content, leading to the presence of significant biases within the model. These ethical considerations raise concerns regarding fairness, inclusivity, and potential biases in the responses generated by the model. Further research and development in addressing these limitations are imperative to ensure the responsible and unbiased deployment of chatbot models.

5.3 Final thoughts

In conclusion, while our chatbot models showcased promising performance in the competition, there are areas for improvement and important ethical considerations to be addressed. By focusing on adjusting model parameters, handling specific tokens, increasing training duration, improving validation sets as well as their preprocessing, and exploring larger models, future iterations of chatbot models can achieve even greater performance and ensure ethical deployment.

Acknowledgements

Some experiments presented in this paper were carried out using ClusterUY (site: https://cluster.uy).

References

Serge Bibauw, Wim Van den Noortgate, Thomas François, and Piet Desmet. 2022. Dialogue systems for language learning: a meta-analysis. *Language Learning & Technology*, 26(1).

Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural language processing with Python: analyzing text with the natural language toolkit.* " O'Reilly Media, Inc.".

Rishi Bommasani, Drew A. Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S. Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, Erik Brynjolfsson, S. Buch, Dallas Card, Rodrigo Castellon, Niladri S. Chatterji, Annie S. Chen, Kathleen A. Creel, Jared Davis, Dora Demszky, Chris Donahue, Moussa Doumbouya, Esin Durmus, Stefano Ermon, John Etchemendy, Kawin Ethayarajh, Li Fei-Fei, Chelsea Finn, Trevor Gale, Lauren E. Gillespie, Karan Goel, Noah D. Goodman, Shelby Grossman, Neel Guha, Tatsunori Hashimoto, Peter Henderson, John Hewitt, Daniel E. Ho, Jenny Hong, Kyle Hsu, Jing Huang, Thomas F. Icard, Saahil Jain, Dan Jurafsky, Pratyusha Kalluri, Siddharth Karamcheti, Geoff Keeling, Fereshte Khani, O. Khattab, Pang Wei Koh, Mark S. Krass, Ranjay Krishna, Rohith Kuditipudi, Ananya Kumar, Faisal Ladhak, Mina Lee, Tony Lee, Jure Leskovec, Isabelle Levent, Xiang Lisa Li, Xuechen Li, Tengyu Ma, Ali Malik, Christopher D. Manning, Suvir P. Mirchandani, Eric Mitchell, Zanele Munyikwa, Suraj Nair, Avanika Narayan, Deepak Narayanan, Benjamin Newman, Allen Nie, Juan Carlos Niebles, Hamed Nilforoshan, J. F. Nyarko, Giray Ogut, Laurel Orr, Isabel Papadimitriou, Joon Sung Park, Chris Piech, Eva Portelance, Christopher Potts, Aditi Raghunathan, Robert Reich, Hongyu Ren, Frieda Rong, Yusuf H. Roohani, Camilo Ruiz, Jack Ryan, Christopher R'e, Dorsa Sadigh, Shiori Sagawa, Keshav Santhanam, Andy Shih, Krishna Parasuram Srinivasan, Alex Tamkin, Rohan Taori, Armin W. Thomas, Florian Tramèr, Rose E. Wang, William Wang, Bohan Wu, Jiajun Wu, Yuhuai Wu, Sang Michael Xie, Michihiro Yasunaga, Jiaxuan You, Matei A. Zaharia, Michael Zhang, Tianyi Zhang, Xikun Zhang, Yuhui Zhang, Lucia Zheng, Kaitlyn Zhou, and Percy Liang. 2021. On the opportunities and risks of foundation models. *ArXiv*.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners.

Andrew Caines, Helen Yannakoudakis, Helena Edmondson, Helen Allen, Pascual Pérez-Paredes, Bill Byrne, and Paula Buttery. 2020. The teacher-student chatroom corpus. *arXiv preprint arXiv:2011.07109*.

Dorottya Demszky, Dana Movshovitz-Attias, Jeongwoo Ko, Alan Cowen, Gaurav Nemade, and Sujith Ravi. 2020. GoEmotions: A Dataset of Fine-Grained Emotions. In *58th Annual Meeting of the Association for Computational Linguistics (ACL)*.

Xiang Gao, Yizhe Zhang, Michel Galley, Chris Brockett, and Bill Dolan. 2020. Dialogue response ranking training with large-scale human feedback data.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models.

Rohan Kamath, Arpan Ghoshal, Sivaraman Eswaran, and Prasad Honnavalli. 2022. Emoroberta: An enhanced emotion detection model using roberta. *SSRN Electronic Journal*.

Sergio Nesmachnow and Santiago Iturriaga. 2019. Cluster-uy: Collaborative scientific high performance computing in uruguay. In *Supercomputing*, pages 188–202, Cham. Springer International Publishing.

Sebastian Raschka. 2023. Parameter-efficient llm finetuning with low-rank adaptation (lora).

Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, Matthias Gallé, et al. 2022. BLOOM: A 176B-Parameter Open-Access Multilingual Language Model. *arXiv preprint arXiv:2211.05100*.

Anaïs Tack, Ekaterina Kochmar, Zheng Yuan, Serge Bibauw, and Chris Piech. 2023. The BEA 2023 Shared Task on Generating AI Teacher Responses in Educational Dialogues. In *Proceedings of the 18th Workshop on Innovative Use of NLP for Building Educational Applications*, page to appear, Toronto, Canada. Association for Computational Linguistics.

Anaïs Tack and Chris Piech. 2022. The ai teacher test: Measuring the pedagogical ability of blender and gpt-3 in educational dialogues. *ArXiv*, abs/2205.07540.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Stanford alpaca: An instruction-following llama model. `https://github.com/tatsu-lab/stanford_alpaca`.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2023. Chain-of-thought prompting elicits reasoning in large language models.

S. Wollny, J. Schneider, D. Di Mitri, J. Weidlich, M. Rittberger, and H. Drachsler. 2021. Are we there yet? - a systematic literature review on chatbots in education. *Frontiers in artificial intelligence*.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, Todor Mihaylov, Myle Ott, Sam Shleifer, Kurt Shuster, Daniel Simig, Punit Singh Koura, Anjali Sridhar, Tianlu Wang, and Luke Zettlemoyer. 2022. Opt: Open pre-trained transformer language models.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020a. Bertscore: Evaluating text generation with bert.

Yizhe Zhang, Siqi Sun, Michel Galley, Yen-Chun Chen, Chris Brockett, Xiang Gao, Jianfeng Gao, Jingjing Liu, and Bill Dolan. 2020b. Dialogpt: Large-scale generative pre-training for conversational response generation.