FRI Natural language processing course 2024
Project report

University *of Ljubljana*
Faculty *of Computer and Information Science*

Revision 1: Literary Conversational Agents

Žiga Trček, Matej Urbas, and Jan Vasiljević

Abstract

DRAFT: Engaging audiences deeply with literature is crucial for enhancing global literacy levels. In this study, we build upon the foundation of conversational agents, which have historically underperformed but have recently been revolutionized by advancements in large-scale language models. We fine-tune various foundational models and assess their enhanced capabilities using a series of standardized quizzes that we introduce. Our focus is on popular literary series, specifically "A Song of Ice and Fire" and "Harry Potter." Additionally, we underscore the significance of employing In-Context Learning to effectively emulate the dialogue styles of key characters within these series, thereby enriching the interactive reading experience.

Keywords

DRAFT, conversational agents, large language models, in-context-learning

Advisors: Slavko Žitnik

Introduction

Literacy among young people is declining, as highlighted in [1]. Many young people have a disinterest in reading and seldom read for enjoyment. A potential strategy to encourage reading is to involve them in conversational interactions with digital pedagogical agents that imitate well-known literary figures. Although numerous studies, such as those [2, 3], discuss the advantages of pedagogical agents, detailed technical implementation aspects are often overlooked. Our work aims to explore various methods for creating pedagogical agents and to provide a comprehensive technical implementation for the method we choose.

Related work

In [4] they built a social bot, called Alana, which is able to engage in an open-domain conversation with their users over various popular topics. The key requirements for such a bot are to maintain the context, provide coherent responses, and be engaging and knowledgeable. The final bot is an ensemble of many different bots, each of which has a different purpose. A ranker is used to determine the best bot response for the given user input. For context maintaining, a state object with information from previous conversations is stored and accessible to every bot in the ensemble.

Retrieval augmented generation (RAG) is often used to correct factually inaccurate, outdated or halucinated Large Language Model (LLM) outputs. A survey on different RAG methods is conducted in [5]. RAGs can be seprated into three categories: pre-training, fine-tunning and inference. Nowdays inference RAGs are mostly used. [6] proposed FLARE, Forward-Looking Active REtrieval Augmented Generation, which re-prompts the language model with extra retrieved data about the subject when some tokens in LLM's output have a low probability.

A straightforward way to create agents is by training or fine-tuning LLM. In [7] the authors developed conversational agents that resemble historical figures like Beethoven, Cleopatra, and Caesar, with personalized profiles, experiences, and emotional states. They introduced three new methods for training specialized agents. Experience Reconstruction extracts scenes in the style of memory flashes, such as profiles, scenes, or interactions. Protective Experience aims at teaching the model to forget or ignore information not relevant to the character to prevent knowledge hallucinations. Experience Upload uses the previous two techniques to fine-tune an existing LLM. They fine-tuned a LLaMa 7B model [8] on a dataset of 750 000 words per character using eight A100 80GB GPUs for 1 hour per character. To assess the models, they used an interview process.

Obtaining data to train a character can be difficult. [9] proposes a novel data augmentation approach named PEDANT that helps train models that mimic human personality by generating large amounts of data with a GPT combined with domain expertise. The method first gathers unlabeled data from online resources and trains a generative language model with it. Then, this model is prompted with seed sentences that an expert created and is asked to complete them. Then, these completions are filtered and ranked based on an expert-defined scoring function. In the paper, PEDANT was implemented on an anti-social psychopatic personality disorder. A labeled corpus with this disorder does not exist, so this is a good showcase of the usefulness of the approach. The data to train the GPT comes from cinema, TV, and Reddit. The model was validated using a text classification task. They used the generated data to train a classifier and tested it on offensive-speech datasets. The results were very encouraging, but requires domain knowledge, which can be a big limiting factor and bottleneck in a larger process.

To avoid training a LLM, [10] suggests using prompt engineering, specifically Chain-of-Thought (COT), on an existing model to incorporate more contextual information. They recommend employing Information-Rich Prompts (IRP) that include the emotional state, the character's relationship with the interlocutor, and the character's memories. Memories are categorized into short-term, which are a limited number of the most recent conversations with the interlocutor, and long-term, which are recursively summarized memories of longer conversations from the character's perspective. Although not explicitly stated, implementing the Big Five personality model [11] could further refine the character's responses. This model would detail the character's Openness, Conscientiousness, Extraversion, Agreeableness, and Neuroticism.

Previous methods that do not involve fine-tuning could be enhanced by using the OpenICL framework [12]. In-Context Learning (ICL) is an approach used with LLMs where the model learns a specific task without the need to update its weights. Instead, the model is shown examples of how the task should be performed. OpenICL offers the tools needed to construct ICL tasks, including key components like retrieval strategies and inference methods. For retrieval, it incorporates heuristic-based methods (such as BM25 and Top-K), random sampling, and model-based retrieval (using embeddings, RAG, Minimum Description Length (MDL), and entropy-based selection). For inference, OpenICL facilitates the integration of COT and other methods along with a prompt template.

It is important to consider teaching strategies while implementing an agent that serves an educational purpose. [13] carried out a detailed analysis of reading comprehension textbooks from the Netherlands, which is one of the nations with a low comprehensive literacy. The researchers analysed lessons withing the textbooks and then also analysed the utilisation of these textbooks by teachers, both by conducting interviews with teachers and attending live lessons. They found that the lessons are mostly focused on exercising and that there is no strong alignment between goals of the lessons, the theory behind them and the assignments that the students must carry out. Little actual knowledge about reading strategies was illustrated and there was no opportunity to choose and apply strategies yourself. The interviews showed that the teachers were aware of these problems, but there were very few who adapted the lessons to counteract them and improve the quality of their teaching. The knowledge that was observed in the textbooks was divided into:

- declarative knowledge - knowing something
- procedural knowledge - knowing how to do something
- conditional knowledge - knowing when to do something.

The textbooks were mostly just focused on the procedural part of the knowledge. To improve literacy, all three should be taught.

[14] describes the importance of setting and situational continuity while reading, which can have major implications in providing a good user experience. Three experiments were carried out on 27 psychology students that tested which aspects of a five-dimensional situational model are more important to our experience. They tested the impact of different aspects by measuring reading time while introducing discontinuities across different dimensions (time, space, causation, motivation, protagonist). The reading time increase is very noticeable in all but the spatial dimension. There, spatial discontinuities did not present a large increase in reading time unless the study participants memorized the map of the story space in advance. The study confirmed the ”processing-load hypothesis” that predicts that the reading time goes up when there is more data to process. It's very likely that this information could be taken into account when constructing a model used for learning by keeping continuities along dimensions that are irrelevant for the learning experience and channeling the focus elsewhere.

Method proposals
We have a few ideas on how to tackle the problem of creating an Agent. These include:

1. Using ICL with OpenICL and RAG
2. Training our own LLM - can be quite expensive
3. Using another LLM to correctly format text - e.g., using Mistral [15] on HPC to give us a dataset.
4. Preprompting an LLM and giving it a context, such as character memories

This box here is only temporary and is meant only for the first phase. It will be removed in the next revision.

Methods

We used [8] and [16] as our base models. We fine-tuned them on a preprocessed full-text dataset from Harry Potter novels. We trained the models using overlapping chunks of text to ensure that the model learns the context of the conversation. We used a chunk size of 512 characters and an overlap of 64 characters.

In the future, we will extract all dialogues from the Harry Potter and A Song of Ice and Fire series and use them to fine-tune the models. This will allow the models to have contextual knowledge to provide a more accurate depiction of characters. We aim to construct a dataset of quotes from the literary works and the characters who said them.

Use the Methods section to describe what you did an how you did it – in what way did you prepare the data, what algorithms did you use, how did you test various solutions ... Provide all the required details for a reproduction of your work.

Below are LaTeXexamples of some common elements that you will probably need when writing your report (e.g. figures, equations, lists, code examples ...).

Equations

You can write equations inline, e.g. $\cos\pi = -1$, $E = m \cdot c^2$ and α, or you can include them as separate objects. The Bayes's rule is stated mathematically as:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}, \quad (1)$$

where A and B are some events. You can also reference it – the equation 1 describes the Bayes's rule.

Lists

We can insert numbered and bullet lists:

1. First item in the list.
2. Second item in the list.
3. Third item in the list.

- First item in the list.
- Second item in the list.
- Third item in the list.

We can use the description environment to define or describe key terms and phrases.

Word What is a word?.

Concept What is a concept?

Idea What is an idea?

Figures

You can insert figures that span over the whole page, or over just a single column. The first one, Figure 1, is an example of a figure that spans only across one of the two columns in the report.

On the other hand, Figure 2 is an example of a figure that spans across the whole page (across both columns) of the report.

Tables

Use the table environment to insert tables.

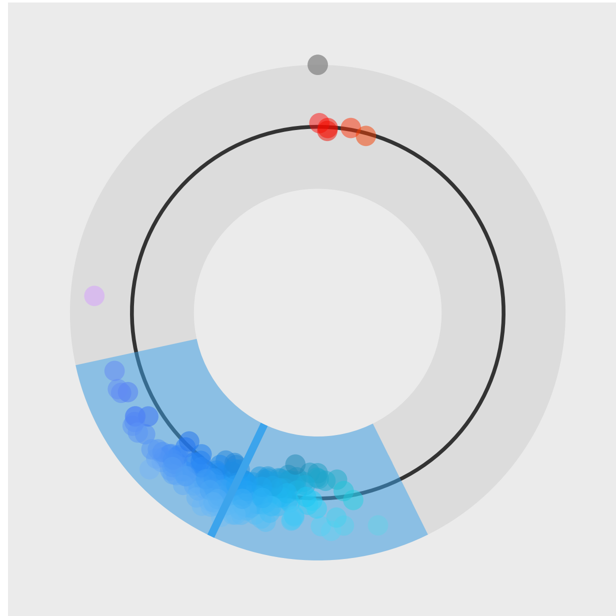

Figure 1. A random visualization. This is an example of a figure that spans only across one of the two columns.

Table 1. Table of grades.

Name		
First name	Last Name	Grade
John	Doe	7.5
Jane	Doe	10
Mike	Smith	8

Code examples

You can also insert short code examples. You can specify them manually, or insert a whole file with code. Please avoid inserting long code snippets, advisors will have access to your repositories and can take a look at your code there. If necessary, you can use this technique to insert code (or pseudo code) of short algorithms that are crucial for the understanding of the manuscript.

Listing 1. Insert code directly from a file.

```
import os
import time
import random

fruits = ["apple", "banana", "cherry"]
for x in fruits:
  print(x)
```

Listing 2. Write the code you want to insert.

```
import(dplyr)
import(ggplot)

ggplot(diamonds,
          aes(x=carat, y=price, color=cut)) +
  geom_point() +
```

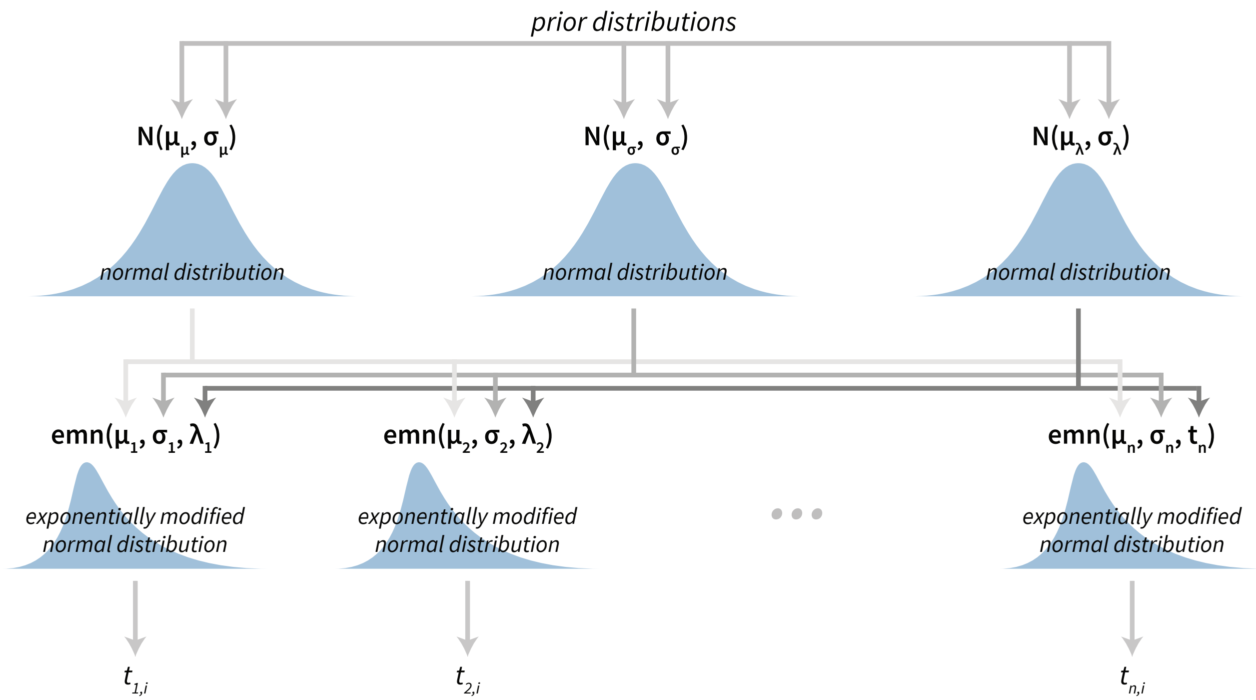

Figure 2. Visualization of a Bayesian hierarchical model. This is an example of a figure that spans the whole width of the report.

```
geom_smooth()
```

Results

After the 1st phase

We evaluated our fine-tuned models on a dataset of 108 multiple-choice quiz questions about the Harry Potter series. The questions were designed to test the models' understanding of the characters and their relationships. The models achieved an average accuracy of 72% before and after fine-tuning.

We also evaluated the models with a sorting hat quiz, where the model had to solve a quiz as a character from a given house (Gryffindor, Hufflepuff, Ravenclaw, or Slytherin). The models were always classified into Ravenclaw.

We plan to evaluate the models on a dataset of quiz questions about the A Song of Ice and Fire series.

We will also evaluate our models on state of the art benchmarks, as in [17]. Use the results section to present the final results of your work. Present the results in a objective and scientific fashion. Use visualisations to convey your results in a clear and efficient manner. When comparing results between various techniques use appropriate statistical methodology.

More random text

This text is inserted only to make this template look more like a proper report. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam blandit dictum facilisis. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Interdum et malesuada fames ac ante ipsum primis in faucibus. Etiam convallis tellus velit, quis ornare ipsum aliquam id. Maecenas tempus mauris sit amet libero elementum eleifend. Nulla nunc orci, consectetur non consequat ac, consequat non nisl. Aenean vitae dui nec ex fringilla malesuada. Proin elit libero, faucibus eget neque quis, condimentum laoreet urna. Etiam at nunc quis felis pulvinar dignissim. Phasellus turpis turpis, vestibulum eget imperdiet in, molestie eget neque. Curabitur quis ante sed nunc varius dictum non quis nisl. Donec nec lobortis velit. Ut cursus, libero efficitur dictum imperdiet, odio mi fermentum dui, id vulputate metus velit sit amet risus. Nulla vel volutpat elit. Mauris ex erat, pulvinar ac accumsan sit amet, ultrices sit amet turpis.

Phasellus in ligula nunc. Vivamus sem lorem, malesuada sed pretium quis, varius convallis lectus. Quisque in risus nec lectus lobortis gravida non a sem. Quisque et vestibulum sem, vel mollis dolor. Nullam ante ex, scelerisque ac efficitur vel, rhoncus quis lectus. Pellentesque scelerisque efficitur purus in faucibus. Maecenas vestibulum vulputate nisl sed vestibulum. Nullam varius turpis in hendrerit posuere.

Nulla rhoncus tortor eget ipsum commodo lacinia sit amet eu urna. Cras maximus leo mauris, ac congue eros sollicitudin ac. Integer vel erat varius, scelerisque orci eu, tristique purus. Proin id leo quis ante pharetra suscipit et non magna. Morbi in volutpat erat. Vivamus sit amet libero eu lacus pulvinar pharetra sed at felis. Vivamus non nibh a orci viverra rhoncus sit amet ullamcorper sem. Ut nec tempor dui. Aliquam convallis vitae nisi ac volutpat. Nam accumsan, erat eget faucibus commodo, ligula dui cursus nisi, at laoreet odio augue id eros. Curabitur quis tellus eget nunc ornare auctor.

Discussion

Our fine-tuning attempts have not been successful so far. The baseline models perform well on the quiz questions and simple fine-tuning does not seem to improve the performance. We will try to extract more context from the books using more sophisticated approaches and use it to fine-tune the models.

The dialogue generation is also very good out of the box, especially with smart prompting. We will try to improve the dialogue generation by using context databases.

After the 1st phase

Use the Discussion section to objectively evaluate your work, do not just put praise on everything you did, be critical and exposes flaws and weaknesses of your solution. You can also explain what you would do differently if you would be able to start again and what upgrades could be done on the project in the future.

Acknowledgments

After the 1st phase

Here you can thank other persons (advisors, colleagues ...) that contributed to the successful completion of your project.

References

[1] Jane Murray. Literacy is inadequate: young children need literacies, 2021.

[2] Thijs MJ Nielen, Glenn G Smith, Maria T Sikkema-de Jong, Jack Drobisz, Bill van Horne, and Adriana G Bus. Digital guidance for susceptible readers: Effects on fifth graders' reading motivation and incidental vocabulary learning. *Journal of Educational Computing Research*, 56(1):48–73, 2018.

[3] Mehdi Alaimi, Edith Law, Kevin Daniel Pantasdo, Pierre-Yves Oudeyer, and Hélène Sauzeon. Pedagogical agents for fostering question-asking skills in children. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pages 1–13, 2020.

[4] Ioannis Papaioannou et al. *Designing coherent and engaging open-domain conversational AI systems*. PhD thesis, Heriot-Watt University, 2022.

[5] Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, and Haofen Wang. Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*, 2023.

[6] Zhengbao Jiang, Frank F Xu, Luyu Gao, Zhiqing Sun, Qian Liu, Jane Dwivedi-Yu, Yiming Yang, Jamie Callan, and Graham Neubig. Active retrieval augmented generation. *arXiv preprint arXiv:2305.06983*, 2023.

[7] Yunfan Shao, Linyang Li, Junqi Dai, and Xipeng Qiu. Character-llm: A trainable agent for role-playing, 2023.

[8] 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. Llama: Open and efficient foundation language models, 2023.

[9] Yair Neuman, Vladyslav Kozhukhov, and Dan Vilenchik. Data augmentation for modeling human personality: The dexter machine, 2023.

[10] Seokhoon Jeong and Assentay Makhmud. Chatbot is not all you need: Information-rich prompting for more realistic responses, 2023.

[11] L R Goldberg. An alternative "description of personality": the big-five factor structure. *J Pers Soc Psychol*, 59(6):1216–1229, December 1990.

[12] Zhenyu Wu, YaoXiang Wang, Jiacheng Ye, Jiangtao Feng, Jingjing Xu, Yu Qiao, and Zhiyong Wu. Openicl: An open-source framework for in-context learning, 2023.

[13] Suzanne TM Bogaerds-Hazenberg, Jacqueline Evers-Vermeul, and Huub van den Bergh. What textbooks offer and what teachers teach: an analysis of the dutch reading comprehension curriculum. *Reading and writing*, 35(7):1497–1523, 2022.

[14] Amy E. Hilliard Rolf A. Zwaan, Gabriel A. Radvansky and Jacqueline M. Curiel. Constructing multidimensional situation models during reading. *Scientific Studies of Reading*, 2(3):199–220, 1998.

[15] Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mistral 7b, 2023.

[16] Marah Abdin, Sam Ade Jacobs, Ammar Ahmad Awan, Jyoti Aneja, Ahmed Awadallah, Hany Awadalla, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Harkirat Behl, et al. Phi-3 technical report: A highly capable language model locally on your phone. *arXiv preprint arXiv:2404.14219*, 2024.

[17] Zishan Guo, Renren Jin, Chuang Liu, Yufei Huang, Dan Shi, Linhao Yu, Yan Liu, Jiaxuan Li, Bojian Xiong, Deyi Xiong, et al. Evaluating large language models: A comprehensive survey. *arXiv preprint arXiv:2310.19736*, 2023.