Fine-Tuning and Retrieval Augmented Generation for Question Answering Using Affordable Large Language Models

Tiberiu Boros, Radu Chivereanu, Stefan Daniel Dumitrescu, Octavian Purcaru
Adobe Systems
Bucharest, Romania
{boros, rchivereanu, sdumitre, opurcaru}@adobe.com

Abstract

We present our proposed system named *Sherlock* to UNLP 2024 Shared Task on Question Answering winning first place. We employ a mix of methods, from using automatically translated datasets to perform supervised fine-tuning and direct preference optimization on instruction-tuned models, to model weight merging and retrieval augmented generation. We present and motivate our chosen sequence of steps, as well as an ablation study to understand the effect of each additional step. The resulting model and code are made publicly available (download links provided in the paper).

Keywords: large language models, direct preference optimization, instruction tuning, LLM, finetuning, RAG, model merge, re-ranking, Ukrainian language, question answering, multiple choice, open-ended, open-source

1. Introduction

The work of Vaswani et al. (2017) has shaped landscape of Natural Language Processing (NLP), through the emergence of Transformer-based Large Language Models (LLMs). Proprietary models such as GPT (Achiam et al., 2023) or Open-Source alternatives such as LLama (Touvron et al., 2023), Mistral (Jiang et al., 2023) and Bard/Gemini (Manyika and Hsiao, 2023) are currently the number one choice in successfully solving difficult NLP tasks such as translation, question answering or user dialogue.

These achievements were made possible through continuous improvements of machine learning (ML) methods and techniques, most notably being the development of the attention mechanism(Ainslie et al., 2023) in tandem with better and faster hardware. However, the noticeable leaps in model performance often came with drastic increases in the number of parameters, which in turn added more stress to the hardware, resulting in increased training and exploitation costs.

This research is part of the of **the UNLP Shared Task 2024**(Syvokon et al., 2024), which focuses on Ukrainian Question Answering via **affordable LLMs**. Thus, our work is **focused on compact LLMs that run on a single consumer-grade GPU or CPU**. We set out to explore how to leverage such models, both by fine-tuning them and by using retrieval augmented generation (RAG).

In the following sections we'll investigate related methods and techniques (Section 2), provide details about the shared task, dataset and proposed methodology (Section 3), and present our results (Section 4) and conclusions (Section 5).

2. Related Work

The task of Question Answering is a long-standing and well defined task in NLP, with the purpose of answering a user's question, posed in natural language. The task itself has many variants (Zhang et al., 2023); we're focusing on text-aided selection of the correct choice given a question and multiple possible answers. To be able to better discriminate between the given choices, it is essential to pair the LLM's internal knowledge and reasoning capabilities with external data and tools.

Primarily, we need an LLM that is able to follow instructions. It has been shown, both empirically and otherwise that instruction tuning enables LLMs to do specific, useful work (Jiang et al., 2024). Prompting techniques are routinely employed to increase performance and guide models' answers towards a desired direction. The most basic prompt is to simply ask an LLM to do something (e.g. "zero-shot"), without providing any examples in the prompt. Few-shot means showing the LLM how to answer by understanding the format, input and output from the few examples given in the prompt, before asking the target question - this "primes" the model to respect the same format as the already-answered questions/tasks. Few-shotting is especially tricky for smaller models (Touvron et al., 2023) that have limited context-size.

Other prompting techniques, like Knowledge Generation Prompting (Liu et al., 2022) or regular self consistency-checks, aim to make use of the knowledge embedded in the model itself to augment its context and perform checking (the LLM generates an intermediary step of a problem and can check itself with an additional query to ascertain whether it considers it has sufficient information or the generated knowledge in the previous step is correct, in order to move to the next generation

The Third Ukrainian Natural Language Processing Workshop (UNLP) @LREC-COLING-2024, pages 75–82

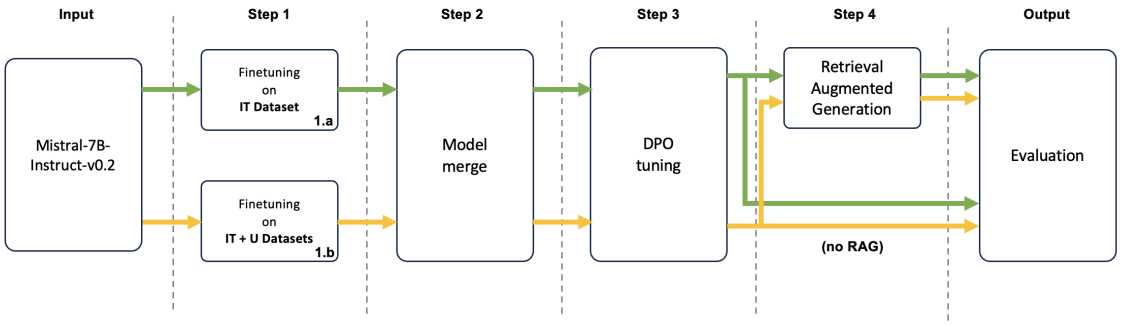

Figure 1: Diagram of the best performing strategy for tuning and running the model. From left to right: base model supervised finetuning, model merges, direct preference optimization yielding 2 models and the final evaluation of the 2 models with and without RAG

step or attempt a final answer.

One powerful method to combat LLM hallucination while benefiting from external sources is to perform Retrieval Augmented Generation (et al., 2021). This technique involves using semantic embeddings of a user's query to find pre-embedded texts that are semantically close and that could help in generating an answer. By externalizing the task of information retrieval (by finding and adding it in the LLM prompt), it lets the LLM focus more on answering the question based on presented facts/information rather than using its internal knowledge which often might lead to hallucinations and incorrect responses.

Other methods to further increase performance look towards tuning model parameters. While LLMs have been trained on huge amounts of text, they likely benefit from limited fine-tuning on in-domain data, a technique that helps shape their response for the specific use-case. Here too we are faced with a variety of choices and methods: from standard full parameter tuning with next-word prediction to Direct Preference Optimization (Rafailov et al., 2023). Furthermore, merging model parameters is yet another powerful method to aggregate knowledge (Sung et al., 2023).

3. Proposed Methodology

We employ a hybrid approach, in which (a) we perform fine-tuning on a LLM and (b) we augment the input prompt with data extracted from our knowledge base. Apart from structured and unstructured fine-tuning, we experimented with various model merges, which generated a sensible leap in performance. Interestingly, even if our fine-tuning was done using Ukrainian data, the model merge was able to successfully preserve pre-existing knowledge while blending newly acquired Ukrainian capabilities.

3.1. Fine-tuning experiments

In our initial assessment we experimented with multiple open-source LLMs (see Section 4 for results), and chose Mistral-7b(Jiang et al., 2023) (instruct version) as the backbone of our system as it was the best performing out-of-the-box model.

Starting from the base model (Mistral-7B-Instruct-v0.2), we performed a set of experiments that resulted in diverging models (see Figure 1), which were evaluated in an end-to-end manner. We used 4 datasets: IT, U, DPO and KB datasets, detailed in the next section. The following steps were taken:

- **Step 1a - Supervised finetuning on the IT-Dataset**: We fine tune the base model for 3 epochs, using a curated dataset of instructions in Ukrainian with the standard supervised trainer (SFT);
- **Step 1b - Supervised finetuning on the IT-Dataset + U-Dataset)**: Similar to step 1a, but using the IT and U Datasets. This step is designed to help with multiple choice questions as well as open-ended questions;
- **Step 2 - Model merge**: We merge each of the models resulted in steps 1a and 1b with the Neuraltrix[1] model, which is a direct preference optimized (DPO) variant of the baseline Mistral model. This step and model choice were introduced based on empirical evaluations of the output. The merge method was Spherical Interpolation. The results of this step are two models: the merges of steps 1a and 1b with the NeuralTrix model;
- **Step 3 - DPO tuning (DPO-Dataset)**: We further refine the two merged models by Direct Preference Optimization using the Ukrainian translated DPO dataset;

[1] https://huggingface.co/CultriX/NeuralTrix-7B-dpo

- **Step 4 - RAG enrichment (KB-Dataset)**: Used only for the RAG-enabled approach, we perform RAG enrichment of the prompts for every question;

In the evaluation section (Section 4) we perform ablation tests, but only for a test-set consisting of multiple choice answers which could be evaluated automatically without requiring human expert input.

3.2. Dataset Description

To finetune our models we used the following datasets:

(a) IT Dataset - a dataset used for instruction tuning, obtained by merging in a similar format AlpacaDataset (Taori et al., 2023), SQuAD (Ivanyuk-Skulskiy et al., 2021), Ukrainian StackExchange[2], QUA-RC(Zyrianova and Kalpakchi, 2023), XQA (Liu et al., 2019), Belebele (et al., 2023) and the ZNO Dataset provided by UNLP[3](Syvokon et al., 2024). For datasets that were not in Ukrainian, we automatically translated the content[4] (jussa et al., 2022).

(b) DPO Dataset - a dataset for direct preference optimization which is a obtained by automatically translating the OpenOrca Dataset (Lian et al., 2023; Longpre et al., 2023).

(c) KB Dataset - used only during the RAG phase, this unstructured (free-text) dataset is composed from the Ukrainian Wikipedia and a curated list of Ukrainian school textbooks, listed in Table 1. The approximate size is 3.8 GB.

(d) U Dataset - a subset of the KB dataset, used in step 1, as the entire KB dataset was too large; the randomly paragraph-level sampled dataset size is 941 MB.

3.3. Retrieval Augmented Generation

In general, RAG is the procedure of enhancing the model's performance by adding information to the input prompt, based on a set of documents (the Knowledge Base). The procedure is straightforward: given a query (the question that needs answering) we first embed it as a semantic vector using any embedding transformer. The knowledge based is pre-segmented and embedded, and stored into a vector database that allows fast similarity search. The top-n documents that have the highest semantic similarity (lowest cosine distance) to the query embedding are those that will be added as context in the LLM's prompt.

However, there are a few caveats:

[2]https://huggingface.co/datasets/zeusfsx/ukrainian-stackexchange

[3]https://github.com/unlp-workshop/unlp-2024-shared-task/blob/main/data/zno.train.jsonl

[4]Translations were performed with NLLB-3B

```
System: You are a teacher of the Ukrainian lan-
guage and you want to find some documents that
contain the answer to the request.
System: You will follow all instructions.
Instruction: You may have to do several searches
to get the answer.
======example 1=====
Question: What song did ADDT compose: (a)
Driving by the sea; (b) Movement
Answer: 3 queries need to be run:
(a) What is ADDT
(b) When Driving by the Sea was created
(c) When Movement is created
========example 2=====
Question: Why did Alice follow the rabbit?
Option 1: she was bored; Option 2: she was
interested;
Answer: The following requests must be made:
(a) Was Alice bored when she was following the
rabbit?
(b) Was Alice curious when she followed the
rabbit?
========example 3=====
Question: What is the correct form of the adjec-
tive formed from the noun water: a) watery; (b)
anhydrous
Answer: The following requests must be made:
(a) How do nouns become adjectives?
(b) Rules for the formation of the word water.
========
Instruction: When presented with multiple
choices, for each choice you should issue a
search query.
Instruction: Answer one item per line!
Instruction: Speak exclusively in Ukrainian.
Instruction: Do not translate back to English.
Instruction: Do not use your own knowledge to
directly answer the question.
Instruction: Given the above instructions, an-
swer the following prompt:
Question: {query}

Answer:
```

Figure 2: Prompt used to split the input query into subtasks. The query contains both the question and variants if it is a multiple choice question.

(a) Semantic vectors are a very effective instrument for information retrieval only if the topic/subject is consistent throughout the input text. This is because the representation capacity of a fixed-size vector is finite, and fitting multiple topics into this finite vector would result in representation conflicts. Thus, one of the prerequisites for a high performing information retrieval system is performing **accurate topic-based segmentation of the input documents.**

(b) Additionally, computing semantic vectors is a **language dependent task** and, in our initial experiments, **most out-of-the-box models were underperforming on Ukrainian**.

(c) The context window of the LLM plays a major role in deciding **how much content to retrieve from the KB**. Open-source models usually have smaller context windows than commercial-grade LLMs, which in turn requires a reduction in the amount of input data received from the RAG phase.

With their limitations in mind, we designed a custom retrieval system that (a) works directly with **keyword indexing and search** and (b) **uses a LLM to sequentially extract information** and filter out

Name	Description
Довідник з укр. мови та літ.: Завдання в тестовій формі - Частина 1 (*Ukrainian Language and Literature Handbook: Test Form Tasks - Part 1*), О. М. Авраменко, М. Б. Блажко	Handbook providing test form tasks related to Ukrainian language and literature, aimed at aiding in learning and assessment.
Львівський Регіональний Центр Оцінювання Якості Освіти: Українська Мова (*Lviv Regional Center for Educational Quality Assessment: Ukrainian Language*), Збірник завдань для підготовки до зовнішнього незалежного оцінювання, Львів 2007	Collection of tasks for preparation for external independent evaluation in Ukrainian language.
Практикум з правопису і граматики української мови (*Workbook on Ukrainian Language Spelling and Grammar*), І.П. Ющук	Handbook approved for use in general educational institutions by the Commission on the Ukrainian Language of the Scientific and Methodological Council on Education of the Ministry of Education and Science, Youth and Sports of Ukraine. This workbook combines theoretical principles with practical tasks, aiding in the understanding of Ukrainian language grammar and improvement of spelling skills.
Новий довідник: Українська мова. Українська література (*New Handbook: Ukrainian Language. Ukrainian Literature*), М. Радишевська, В. Погребенник, В. Михайлюта, Т. Корольова, Т. Трош, О. Гудзенко	Handbook covering Ukrainian language and literature for school curriculum. Contains concise text and illustrative examples for thorough understanding and quick mastery of the material. Useful for exam preparation and entrance into higher education institutions.
Український Правопис (*Ukrainian Orthography*), Затверджено Кабінетом Міністрів України, 2019	Official Ukrainian orthographic rules approved by the Cabinet of Ministers of Ukraine, the Presidium of the National Academy of Sciences of Ukraine, and the Collegium of the Ministry of Education and Science of Ukraine. It provides guidelines for spelling, punctuation, and grammar, aiming to maintain consistency and clarity in written Ukrainian language.
Українська Література: Довідник для підготовки до ЗНО-2021 (*Ukrainian Literature: Handbook for the Preparation for the External Independent Evaluation 2021*), Дмитро Заєць	Handbook providing summaries and analyses of literary works covered in the Ukrainian literature curriculum for the 9th, 10th, and 11th grades, including works from ancient Ukrainian literature to contemporary authors, along with key literary terms and concepts.
Історія України: Хронологічний і термінологічний довідник для підготовки до ЗНО (*History of Ukraine: Chronological and Terminological Handbook for the Preparation for the External Independent Evaluation*), Олександр Геннадійович Полтавцев	Handbook providing key dates, concepts, and information on historical figures for the Ukrainian history program in preparation for the External Independent Evaluation (EIE)
100 тем. Історія України (*100 Topics. History of Ukraine*), Г. Т. Децюрін	A comprehensive school course in 100 themes, designed to present the most essential and obligatory topics for understanding the history of Ukraine. This book is aligned with the educational program of the Ministry of Education and Science of Ukraine, enabling systematic self-study and reinforcing key historical concepts, terms, and definitions.
Історія України. 10–11 класи: Наочний довідник (*History of Ukraine. Grades 10–11: Visual Guide*), О. В. Гісем, О. О. Мартинюк	Visual guidebook providing a clear and structured presentation of historical events, designed to aid students in grades 10 and 11 with the systematic study of Ukrainian history. It follows the educational standards set by the Ministry of Education and Science of Ukraine.
Історія України (*History of Ukraine*), О.Д. Бойко, 2002	A guidebook for the history of Ukraine, approved by the Ministry of Education and Science of Ukraine as an educational manual for higher educational institutions. It is characterized by its precision in language form, clarity in the expression of thoughts, and facts that are set against the background of significant trends in the comprehension of historical events, contributing to the formation of students' concrete historical knowledge.

Table 1: Materials used in our unstructured dataset

unwanted data:

Step 1: Ask the LLM to analyze the input query and extract a series of searches required to answer the question (see Figure 2 for the prompt). There can be any number of independent searches, for every topic, term, definition, artwork, book etc. present the input data;

Step 2: Take every previously generated item and ask the LLM to imagine the keywords that need to be used in the search process (see Figure 3 for prompt). The LLM is instructed to generate unigrams, bigrams and trigrams. Bi- and trigrams, are used in a document scoring process;

Step 3: Use full-text indexing (we used OpenSearch[5] as the backend), to look for the keywords in the documents and retrieve the content. It is important to mention that we keep the documents as a whole and we avoid any pre-segmenting of the data;

Step 4: Score the documents, based on bi- and trigrams and keep only the first top-k[6] in the queue (scoring details follow).

Step 5: Take each paragraph in the input data, with a limited context window[7] and ask the LLM to look and the original query and at the paragraph and say if it could help with that query in any way - if the LLM says "yes", the paragraph goes in a special queue (see Figure 4 for prompt);

Step 6: RAG is done by combining the selected paragraphs, with the original document titles and presenting them as documents in the final prompt (see Figure 5 for details).

Figure 6 shows the execution steps for the query *"Elements of expressionism are present in the work: (a) "Stone Cross", (b) "Institute", (c) "Marusya""*, where we translated interesting portions for reader

[5] https://opensearch.org/ - accessed 2024-03-28

[6] In our experiments we used $k = 6$

[7] For context, we used one paragraph above, one paragraph below and the title of the original document

```
System: You are a teacher of the Ukrainian language and you want to find some documents to find the document that contains the answer to the query.
Instruction: Write several keywords that will be used to search for relevant documents. Creation of unigrams, bigrams and trigrams.
Instruction: Output unigrams, bigrams and trigrams in three separate lines.
Instruction: You will follow the instructions exactly. Do not write anything else.
Instruction: The first line must contain unigrams (individual words) separated by commas.
Instruction: The second line should contain bigrams (groups of two words) separated by commas.
Instruction: The third line must contain trigrams (groups of three words) separated by commas.
Instruction: Write only in Ukrainian.
======example=====
Context: What Asimov wrote first: (a) Foundation (b) I robot
Request: Learn more about the Isaac Asimov Foundation
Answer: UNIGRAMS: Isaac, Asimov, Foundation, book, chapter, summary, content, genre, year
BIGRAMS: Isaac Asimov, brief description, Foundation description, publication date, Foundation characters, Foundation genre
TRIGRAMS: Foundation of Isaac Asimov, Genre of the book of the Foundation, Publication date of the Foundation, Book of the Foundation about
================
Instruction: Speak only Ukrainian.
Instruction: words should be separated by spaces, not underscores.
Instruction: Given the instructions above, solve the following problem:
Context: {query}
Request: Learn more about {step}

Answer:
```

Figure 3: Prompt used to get the search terms for each generated step in phase 1.

convenience. As shown, the model successfully extracts the search phases, retrieved documents for each step and manages to get the right context in order to answer that elements of expressionism are present in “Stone Cross”. In this case, the source document was a Wikipedia page.

The document scoring algorithm is simple, because we rely on the LLM to perform the heavy lifting and generate good input data. We take each n-gram generated by the model and we split it into tokens. We then look for tokens within the text and if all tokens from the same n-gram appear very close to each other (within a 5-word window), we add +1 to the document’s score. If an n-gram appears multiple times, the score will be increased each time the context conditions are satisfied. In the end, we sort the documents based on their descending score, keeping only the top-k documents as RAG results.

Note 1: The choice in the number of document for RAG might be sub-optimal. We set k=6 strictly based on speed constraints.

Note 2: The context window for the paragraph that is being analyzed was not tested against other options, which means that bigger context or heuris-

```
System: You are a Ukrainian student trying to find an answer to a question
System: Follow all instructions
Instruction: You will receive a paragraph from a document, and you need to find the answer in it.
Instruction: If the document is not current, write "No"
Instruction: If the document is current, write "Yes"
Instruction: Answer Yes or No!
Instruction: Answer in Ukrainian
================ Example 1 - your answer in the text=============
Query: does the text answer the question: is Isaac Asimov the author of the Foundation
Document: Title: About the Foundation
Synopsis: Foundation is a novel written by Isaac Asimov and is part of a saga.
Answer: Yes
================
================ Example 2 - your answer is not in the text=========
Query: does the text answer the question: is Isaac Asimov the author of the Foundation
Title: About the foundation
Contents: Isaac Asimov wrote many novels.
Answer: No
================
Instruction: Follow all the above instructions.
Instruction: Additional text, thoughts or ideas are not allowed.
Instruction: Consider whether you can get any useful information from the text. It is very important that you do not miss the clues!!!!
Instruction: Using the above instructions and examples, answer the following prompt:
Instruction: Look carefully at the context. If the document can help answer what is in the context, then your answer should be yes. It is very important!
Query: Does the text contain the answer to the question: {query}
Title:{title}
Content: {content}

Answer:
```

Figure 4: Prompt used to analyze paragraphs and extract relevant content.

```
System: You are a Ukrainian teacher specializing in literature and history.
System: If the request is in Ukrainian, answer with the letters corresponding to the best options from the list of possible answers.
System: When answering a question, return all correct options (e.g. "Option 3: Golden Gate Bridge")
System: You will fulfill all user requests
Instruction: There is only one correct answer to the question.
======example:
These are the documents:
About the chicken and the egg:
The chicken came before the egg.
Question: What came first:
Option 1: Chicken;
Option 2: Egg
Answer: Option 1: Chicken;
==================
System: Base your knowledge primarily on these documents:
{docs}
Query: {query}

Answer:
```

Figure 5: Prompt used to provide final answer, based on RAG

Extracted steps are : (a) досліджуйте "Камінний хрест" та експресіонізм
(b) досліджуйте "Інститутка" та експресіонізм
(c) досліджуйте "Маруся" та експресіонізм
Translated:
Extracted steps are :
(a) research The Stone Cross and Expressionism
(b) research The Institute and Expressionism
(c) research "Marusia" and Expressionism
Step 1. (a) досліджуйте "Камінний хрест" та експресіонізм
Unigrams : елементи, експресіонізм, творі, Камінний, хрест, досліджуйте, епохи, вплив, мистецтво, період, контраст, характеристики, тематика, художник, роль
Bigrams : елементи експресіонізму, Камінний хрест творі, експресіонізм епохах, експресіонізм вплив, Камінний хрест період, експресіонізм мистецтво, роль експресіонізму, художник експресіонізму, експресіонізм періодизація, тематика експресіонізму, експресіонізм характеристики, мистецькі напрямки, погляди митців
Trigrams : елементи експресіонізму творі, Камінний хрест творі період, експресіонізм епох мистецтво, Камінний хрест епохах художник, експресіонізм період вплив, мистецтво експресіонізму період, художник експресіонізму творчість, експресіонізм періодизація мистецтво, тематика експресіонізму напрямки
Will analyze the following documents:
Мистецтво постмодернізму
Камінний хрест (новела)
Експресіонізм
Дегенеративне мистецтво
Осика Леонід Михайлович
Експресіонізм
Wrote down:Камінний хрест (новела): Для втілення свого задуму новеліст вдається до своєрідної сюжетно-композиційної організації тексту, оперуючи водночас поетикою експресіонізму, який виявляється у художньому дослідженні сенсу страждань, які спонукають людину до пізнання суті свого існування; в емоційній загостреності зображуваного, уривчастій, «нервовій» динамічно-експресивній фразі, напруженому драматизмі ситуації. Ліричний струмінь, який звучить у прощальних монологах героя, «живцем перенесений мужичий спосіб бесідування», «розкопки» в збентеженій, роз'ятреній сум'яттями душі людини, біль, жаль, душевні страждання визначають настрій новели. Через вужчу, конкретно-історичну проблему еміграції автор розкриває у творі й значно ширшу, вічну проблему сакрального зв'язку людини з рідною землею. Попри непосильну працю, суворий аскетичний спосіб життя, Іван був щасливим, адже почувався часткою рідної землі, її господарем, бо доглядав, оживляв її. Героїчний поєдинок селянина із значно більшою за нього силою завершується перемогою людини, яка перетворює природу, змушує родити хліб. Герой полюбив свою тяжку працю і свій горб, який перетворив на родюче поле, бо це наповнювало його існування сенсом, давало йому радість і гармонію. Виїзд на чужину розірвав у його душі цей зв'язок із навколишнім світом. Іван Дідух сприймає від'їзд як власну смерть і через це ставить по собі хрест.
Translated:
Stone cross (novella): In order to implement his idea, the novelist resorts to a peculiar plot-compositional organization of the text, **at the same time operating with the poetics of expressionism**, which is revealed in the artistic study of the meaning of suffering, which prompts a person to know the essence of his existence; in the emotional acuity of the depicted, the fragmented, "nervous" dynamic and expressive phrase, the intense drama of the situation. The lyrical current that sounds in the hero's farewell monologues, "the man's way of talking is transferred alive", "excavations" in the confused, enraged human soul, pain, regret, and mental suffering determine the mood of the novel. Due to the narrower, specific historical problem of emigration, the author reveals in the work a much broader, eternal problem of the sacred connection of a person with his native land. Despite the hard work and the strict ascetic way of life, Ivan was happy, because he felt like a part of his native land, its owner, because he cared for it and revived it. The peasant's heroic duel with a much greater force ends with the victory of man, who transforms nature and forces him to give birth to bread. The hero loved his hard work and his hump, which he turned into a fertile field, because it filled his existence with meaning, gave him joy and harmony. Going to a foreign country broke this connection with the surrounding world in his soul. Ivan Didukh perceives the departure as his own death and because of this he puts a cross on himself.

Figure 6: Sample output for the query "Елементи експресіонізму наявні у творі: (a) «Камінний хрест», (b) «Інститутка», (c) «Маруся»" - translated: *"Elements of expressionism are present in the work: (a) Stone Cross, (b) Institute, (c) Marusya"*

tic methods of establishing the right context window might yield better results. Also, we expect that a dedicated segmentation model would perform better than our current approach.

Note 3: Instead of asking the model to decide if a paragraph is useful or not for the query, we experimented with making the model take notes and use the notes to generate the final answer. This decreased the accuracy of the RAG process for Ukrainian, but it showed promising results for English. This finding merits further exploration.

Note 4: All the models we experimented with followed instructions better when they were written in English, regardless of them being fine-tuned or not for Ukrainian.

Note 5: Though we tried to constrain the model to generate unigrams, bigrams and trigrams separately, this was not always successful. As such, our scoring mechanism does not enforce exact n-gram count. Instead, it just tokenizes the input based on the "white space" character and works with any number of resulting tokens.

Note 6: For multiple choice questions, asking the model to produce the output as "Option 1: ...." (text of the option included) yielded better results, because the model seemed to follow this type of instruction better than just being ask to respond with the option number. This is a somewhat expected behaviour as "forcing" the model "explain" its choice makes it better ponder the option - probably more attention is placed on the response in relation to the option description, thus "reasoning" better.

4. Evaluation and Results

We present results of the evaluation carried out against baseline models, of our fine-tuning experiments, merges and assess the performance of our RAG system.

We summarize our results in Table 2. Our initial assessment focused on baseline model evaluation, in order to see what architecture would perform best. Ideally, we would experiment on all base models the same way, but due to time and resource limitations, we had to focus on a specific architecture alone. Thus, we scored 3 baseline models with no-RAG on the ZNO dataset: *Llama-2-7B-32K-Instruct*, *gemma-7b-it* and *Mistral-7B-Instruct-v0.2*. We went with the instruct models, because the vanilla instances fail to follow instructions and are hard to score.

As shown, *Mistral-7B-Instruct-v0.2*, has an out-of-the box accuracy of 30.89%, versus the other two models that fall bellow 24%. Interestingly, we note that the baseline obtained by only returning option 1 across the entire dataset is around 24%, which throws Llama and Gemma below this threshold and Mistral slightly over.

In the next experiment we added RAG on top of the baseline Mistral model, which increased its accuracy by an additional 10%, from 30.89% to 40.21%.

For the next phase, we fine-tuned the baseline model, first by using just IT Dataset and then by combining IT and U Datasets. Non-RAG results are 32.75% and 33.02%, while the RAG-enhanced results are 40.87% and 41.14% respectively. This shows that, in some cases, tuning with free text and instruction data at the same time yields better results, provided that the ratio between the two sets

Base Model	Finetuned	RAG	Merged	Acc. (%)
Llama-2-7B-32K-Instruct	No	No	No	19.13
gemma-7b-it	No	No	No	23.70
CultriX/NeuralTrix-7B-dpo	No	No	No	31.95
CultriX/NeuralTrix-7B-dpo	No	Yes	No	36.48
Mistral-7B-Instruct-v0.2	No	No	No	30.89
Mistral-7B-Instruct-v0.2	No	Yes	No	40.21
Mistral-7B-Instruct-v0.2	IT Dataset	No	No	32.75
Mistral-7B-Instruct-v0.2	IT Dataset	Yes	No	40.87
Mistral-7B-Instruct-v0.2	IT + U Datasets	No	No	33.02
Mistral-7B-Instruct-v0.2	IT + U Datasets	Yes	No	41.14
Mistral-7B-Instruct-v0.2	No	No	CultriX/NeuralTrix-7B-dpo	40.04
Mistral-7B-Instruct-v0.2	No	Yes	CultriX/NeuralTrix-7B-dpo	**47.00**
Mistral-7B-Instruct-v0.2	IT Dataset	No	CultriX/NeuralTrix-7B-dpo	39.94
Mistral-7B-Instruct-v0.2	IT Dataset	Yes	CultriX/NeuralTrix-7B-dpo	48.46
Mistral-7B-Instruct-v0.2	IT + U Datasets	No	CultriX/NeuralTrix-7B-dpo	41.94
Mistral-7B-Instruct-v0.2	IT + U Datasets	Yes	CultriX/NeuralTrix-7B-dpo	**49.13**

Table 2: Results obtained on the ZNO dataset by different network architectures, pretrained variants, fine-tuned models and merges.

is close to 1.

The final stage of our experiments focused on model merges. For this, we used *CultriX/NeuralTrix-7B* as the second fine-tuned variant of Mistral. Note, that this is not an instruction-tuned model, so its accuracy on the dataset is very low. We merged (Spherical Interpolation) the previously presented baseline models, IT Dataset tuned and mixed tuned variants with this new model and we performed DPO tuning for one epoch on the result, using the translated DPO Dataset. The results for non-RAG vs RAG optimized prompt are 40.04%–47.00% (for the base model), 39.94%–48.46% (for the IT Dataset variant) and 41.94%–49.13% (for the mixed variant).

Interestingly, there is a drop in performance for the IT Dataset tuned and merged model with the no-RAG flavour evaluation, but the RAG optimized generation is better.

Finally, the best performing recipe was:

Step 1: Start with Mistral-7B-Instruct-v0.2 and perform fine-tuning on the combination of IT + U Datasets;

Step 2: Merge with CultriX/NeuralTrix-7B using Spherical Interpolation;

Step 3: Perform DPO tuning on the resulting model, in our case using the DPO Dataset;

Step 4: Produce results using RAG-enhanced prompts.

5. Conclusions and Future Work

We present *Sherlock*, our proposed system that achieved first place in the UNLP 2024 competition. Our system is a set of data-augmentation techniques mixed with custom LLMs. We enumerate key points in each of the data, prompting and LLM-tuning areas:

Datasets: (a) We used many available data sources: Ukrainian Wikipedia and manually selected relevant books on the target subject; (b) We translated and used several datasets, both free-text and instruction-formatted

Retrieval Augmented Generation: (a) Due to limiting factors in the standard RAG process (e.g. embedding for Ukrainian does not have great performance), we used n-grams to provide better results than the standard similarity score; (b) We used the LLM itself to generate n-grams.

LLM tuning: (a) We started from an already very good instruction tuned model - Mistral 7B; (b) We tried standard finetuning on different datasets; (c) We experimented with model weight merging; (d) We further enhanced performance by DPO training; (e) Having a test set enabled us to experiment with different combinations of the individual methods above, to achieve an overall better result than each individual method.

Finally, we are happy to announce that, for reproductability we release both the source code[8] and the model[9], hoping that this will further advance efforts in building afordable LLMs that can run on consumer-grade products, with low computational requirements.

[8] https://github.com/adobe/sherlock-backend/tree/UNLP2024

[9] https://huggingface.co/SherlockAssistant/Mistral-7B-Instruct-Ukrainian

6. Bibliographical References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Joshua Ainslie, James Lee-Thorp, Michiel de Jong, Yury Zemlyanskiy, Federico Lebrón, and Sumit Sanghai. 2023. Gqa: Training generalized multi-query transformer models from multi-head checkpoints. *arXiv preprint arXiv:2305.13245*.

Lucas Bandarkar et al. 2023. The belebele benchmark: a parallel reading comprehension dataset in 122 language variants. *arXiv preprint arXiv:2308.16884*.

Patrick Lewis et al. 2021. Retrieval-augmented generation for knowledge-intensive nlp tasks.

Bogdan Ivanyuk-Skulskiy, Anton Zaliznyi, Oleksand Reshetar, Oleksiy Protsyk, Bohdan Romanchuk, and Vladyslav Shpihanovych. 2021. ua_datasets: a collection of ukrainian language datasets.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Zhengbao Jiang, Zhiqing Sun, Weijia Shi, Pedro Rodriguez, Chunting Zhou, Graham Neubig, Xi Victoria Lin, Wen tau Yih, and Srinivasan Iyer. 2024. Instruction-tuned language models are better knowledge learners.

Marta R Costa jussa et al. 2022. No language left behind: Scaling human-centered machine translation. *arXiv preprint arXiv:2207.04672*.

W Lian, B Goodson, E Pentland, et al. 2023. Openorca: An open dataset of gpt augmented flan reasoning traces.

Jiacheng Liu, Alisa Liu, Ximing Lu, Sean Welleck, Peter West, Ronan Le Bras, Yejin Choi, and Hannaneh Hajishirzi. 2022. Generated knowledge prompting for commonsense reasoning.

Jiahua Liu, Yankai Lin, Zhiyuan Liu, and Maosong Sun. 2019. Xqa: A cross-lingual open-domain question answering dataset. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2358–2368.

Shayne Longpre, Le Hou, Tu Vu, Albert Webson, Hyung Won Chung, Yi Tay, Denny Zhou, Quoc V. Le, Barret Zoph, Jason Wei, and Adam Roberts. 2023. The flan collection: Designing data and methods for effective instruction tuning.

James Manyika and Sissie Hsiao. 2023. An overview of bard: an early experiment with generative ai. *AI. Google Static Documents*, 2.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model.

Yi-Lin Sung, Linjie Li, Kevin Lin, Zhe Gan, Mohit Bansal, and Lijuan Wang. 2023. An empirical study of multimodal model merging.

Oleksiy Syvokon, Mariana Romanyshyn, and Roman Kyslyi. 2024. The UNLP 2024 shared task on fine-tuning large language models for Ukrainian. In *Proceedings of the Third Ukrainian Natural Language Processing Workshop*, Torino, Italy. European Language Resources Association.

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.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Qin Zhang, Shangsi Chen, Dongkuan Xu, Qingqing Cao, Xiaojun Chen, Trevor Cohn, and Meng Fang. 2023. A survey for efficient open domain question answering. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14447–14465, Toronto, Canada. Association for Computational Linguistics.

Mariia Zyrianova and Dmytro Kalpakchi. 2023. Quarc: the semi-synthetic dataset of multiple choice questions for assessing reading comprehension in ukrainian. *Northern European Journal of Language Technology*, 9(1).