FRI Natural language processing course 2024
Project report

University of Ljubljana
Faculty of Computer and
Information Science

Slovenian Instruction-based Corpus Generation

Andraž Čeh, Tilen Miklavič, Tom Sojer

Abstract

This study focuses on enhancing the capabilities of Large Language Models (LLMs) to interpret and execute instructions in the Slovene language accurately. As AI assistants become increasingly prevalent across various sectors, there's a need for these technologies to support a wide range of languages, including less commonly supported ones like Slovene. The project focuses on assembling and fine-tuning a dataset of Slovene conversations to enhance the performance of a multilingual LLM in understanding and responding in Slovene.

Keywords
data gathering, corpus, finetuning

Advisors: Slavko Žitnik

Initial ideas

The initial idea behind the project was to address the gap in natural language processing (NLP) capabilities for Slovene, a less commonly supported language in the realm of large language models (LLMs). The primary objective is to create a substantial Slovene language corpus that can be used to train and fine-tune the LLaMA 2 model, optimizing it for a range of NLP applications specific to Slovene.

Recognizing the limitations of existing LLMs in processing and understanding Slovene accurately, our project seeks to enhance model performance through targeted data collection and refinement. The project aims to assemble a diverse dataset from a variety of sources, including news websites, forums, and other digital media that are rich in Slovene text. This approach ensures a comprehensive linguistic representation, capturing nuances and idiomatic expressions unique to Slovene.

Another key aspect of our initial idea involves the development of a fine-tuning strategy that adapts the pre-trained LLaMA 2 model to better handle the specific challenges of the Slovene language. To achieve this, we plan to implement advanced NLP techniques and training methodologies such as transformer-based models and transfer learning.

we structured our project into several phases, starting with data collection and following the iterative processes of model training, evaluation, and refinement. Each phase is designed to build upon the previous one, gradually improving the model's accuracy and efficiency in handling Slovene language tasks.

Initial implementation

0.1 Corpus generation

We decided to crawl data from a few sources for the second submission. These sources were: slo-tech forum, med.over.net, siol, rtvslo and 24ur. Our idea is to use these sources, generate questions based on the content provided and feed the data to a model for finetuning.

To crawl Slovene text from the web, we first needed to decide the appropriate tool that will help us. We have looked into several existing solutions such as Puppeteer, Playwright and Scrapy. As our interest was just to obtain as much data in a short amount of time, we decided to use Scrapy [1]. This is an open-source Python framework for web crawling that is constantly updated so it seemed like an appropriate solution.

After establishing the framework, we generated a few spiders, one for every online source from which we decided to gather data. For instance, we created a separate spider for the "slo-tech.si", "rtvslo.si" and "24ur.com" domains. Each uses different selectors from which it obtains text data and it writes into separate files. To finetune the model of our choosing, we output the content into xml files with annotations for each obtained content item.

For instance one of the items contained the following tags:

1. Content: contains text from a page element
2. Timestamp
3. Url

4. Domain

As goes for corpus generation, we also wanted to include some data preprocessing. We decided annotate the obtained corpus with tags that show how fluent each item is. We used **monocleaner**, a Python tool that aims to detect disfluent sentences in a monolingual corpus [2]. Monocleaner can process Slovene language which is perfect for our finetuning. The tool uses an xml file as an input and adds annotations in form of "lm-score". This tells us the fluency of text. So more fluent text will have a higher score and we can assign it a higher weight when finetuning.

0.2 Finetuning

In adapting our LLaMA model with 7 billion parameters for the Slovene language, we implemented two critical techniques: Low-Rank Adaptation (LoRA) [3] and Fully Sharded Data Parallel (FSDP) [4]. LoRA targets the efficient adaptation of large pre-trained language models. It modifies a select subset of the model's parameters through rank-decomposition matrices. This strategy significantly cuts the number of parameters needing updates, reducing computational costs and speeding up the adaptation process. It also maintains or even enhances performance.

We also enhanced our training capabilities using FSDP [4] on the HPC Sling supercomputer. FSDP optimizes training by distributing the model's parameters across multiple GPUs. This distribution effectively shards the parameters, lowering the memory demands on each GPU. This method is crucial for handling the extensive computational and memory requirements of training large-scale models. It allows for more efficient scaling and better use of the supercomputer's resources.

We applied the Alpaca dataset, comprising 52,000 instruction-following examples generated using OpenAI's text-davinci-003, to enhance our model's precision in following instructions [5]. Drawing inspiration from this success, we are now creating a similar dataset in Slovenian. Our goal is to improve instruction-following capabilities in the Slovenian language by using robust language models to produce high-quality demonstrations.

Further work

In the next phase of our project, we aim to expand our dataset by incorporating additional data sources. This expansion will involve identifying and integrating new websites and forums that are rich in Slovene content, allowing us to diversify and enrich our corpus further.

We also plan to focus on deduplicating the text within our dataset. This process is critical for enhancing the efficiency of the finetuning phase, as it reduces redundancy and helps in training the model more effectively on unique data instances.

Moreover, we intend to develop a system for generating questions based on the text content we have gathered. This could be achieved using advanced NLP tools such as LLaMA for generating context-appropriate questions or by employing natural language toolkits like NLTK to automate question generation. This will enable us to create a more dynamic and interactive model that can engage in more complex dialogues and improve its understanding of conversational contexts in the Slovene language.

In the next phase of our project, we plan to develop a Slovenian dataset analogous to the Alpaca dataset, which we previously utilized to enhance the instruction-following capabilities of our LLaMA model. This new dataset will involve generating instruction-following examples in Slovenian, using a similar approach to that of the Alpaca dataset but adapted to the specific linguistic and contextual needs of Slovenian users.

Our goal is to assemble approximately 50,000 examples, leveraging advanced NLP models for high-quality data generation. This will allow us to train our model more effectively, improving its ability to understand and execute tasks as instructed in Slovenian. This effort will include not just the generation of data but also testing and validation to ensure the quality and diversity of the instruction sets.

References

[1] scrapy. https://github.com/scrapy/scrapy. accessed: 1.5.2024.

[2] monocleaner. https://github.com/bitextor/monocleaner/tree/main. accessed: 1.5.2024.

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

[4] Yanli Zhao, Andrew Gu, Rohan Varma, Liang Luo, Chien-Chin Huang, Min Xu, Less Wright, Hamid Shojanazeri, Myle Ott, Sam Shleifer, et al. Pytorch fsdp: Experiences on scaling fully sharded data parallel, 2023.

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