The First Instruct-Following Large Language Models for Hungarian

Zijian Győző Yang, Réka Dodé, Gergő Ferenczi, Péter Hatvani, Enikő Héja, Gábor Madarász,
Noémi Ligeti-Nagy, Bence Sárossy, Zsófia Szaniszló, Tamás Váradi, Tamás Verebélyi, Gábor Prószéky
HUN-REN Hungarian Research Centre for Linguistics
Budapest, Hungary
{yang.zijian.gyozo, dode.reka, ferenczi.gergo, hatvani.peter, heja.eniko, madarasz.gabor,
ligeti-nagy.noemi, sarossy.bence, szaniszlo.zsofia, varadi.tamas,
verebelyi.tamas, proszeky.gabor}@nytud.hun-ren.hu

***Abstract*—In recent months, large language models have gained significant attention, with companies striving to develop models capable of solving various natural language processing tasks through extensive data training. The release of ChatGPT by OpenAI demonstrated unprecedented capabilities via a multi-step fine-tuning process. For Hungarian, pre-trained large language models include PULI GPT-3SX, PULI GPTrio and in the recent months SambaLingo. In our research, we pre-trained a new large language model based on Llama-2 and inspired by ChatGPT, focuses on fine-tuning with instruction-based prompts. We created a Hungarian prompt dataset and fine-tuned the PULI large language models into instruction-following models. In our research, we discovered that transfer learning allows the model to gain insights from other languages. We found that further pre-training of the language model could leverage valuable knowledge from the originally pre-trained model. Additionally, we can adapt a LLaMA model to another language, such as Hungarian. Our PULI LlumiX models in three Hungarian benchmark could achieve significant better performance. Our instruction model in both HuSST and HuRTE zero-shot competitions could achieve more than 10 accuracy scores. Our further pre-trained Llama-2 model, the PULI LlumiX 32K and the fine-tuned PULI LlumiX 32K Instruct, became state-of-the-art models capable of solving various language technology problems.**

***Index Terms*—PULI models, large language model, instruct model, Llama-2, pre-training, fine-tuning**

I. Introduction

In recent months, large language models have received extraordinary attention. There has been competition among major companies to train increasingly larger language models. Their goal is to train a model on a vast amount of data that can independently solve various natural language processing tasks. These experiments have demonstrated that if the model is large enough and trained on a sufficient amount of data, it can solve language technology tasks at a high quality solely through prompt programming, without fine-tuning [1]. However the real milestone was marked by ChatGPT[1], released by OpenAI[2], which demonstrated the unprecedented capabilities of a large language model.

[1]https://chat.openai.com
[2]https://openai.com

979-8-3503- 8788-9/24/$31.00 © 2024 IEEE

The innovation of ChatGPT compared to earlier models lies in its multi-step fine-tuning process. The success of ChatGPT has initiated a new research direction focused on the fine-tuning of large language models, including reinforcement learning-integrated fine-tuning.

As far as we know, there are only pretrained large language models available for the Hungarian language. Among these are the PULI family models, which include the monolingual 6.7 billion parameter PULI GPT-3SX [2] and the trilingual (Hungarian-English-Chinese) 7.67 billion parameter PULI GPTrio [3]. Additionally, there is the bilingual (English-Hungarian) 6.7 billion parameter HILANCO-GPTX model[3].

In our research, we follow the ChatGPT research [4], where the first step involves fine-tuning the large language model with prompts containing instructions. The next step involves fine-tuning with reinforcement learning. In this current reasearch only the first step was applied. As part of this research, we have created a instruction-based prompt collection for Hungarian. In our previous research, we have fine-tuned the trilingual PULI GPTrio model into an instruction-following model, which we have named the ParancsPULI. However, the model had many shortcomings and was unable to solve numerous language technology tasks. Thus, taking advantage of the benefits of transfer learning, we further pretrained a Llama-2 model [5] for Hungarian.

Our current instruction-following model can be tested on our demo page[4]. Our pretrained large language models are and the once a stable version of our instruct models are ready, the model will be uploaded to our Hugging Face page[5].

II. Related Work

One consequence of ChatGPT's success is the widespread emergence of fine-tuning processes for various large language models and the creation of their datasets. One of the most popular large language model families has become LLaMA [6], Llama 2 [5] and Llama-3[6], developed by Meta AI[7]. LLaMA

[3]https://hilanco.github.io
[4]https://puli.nytud.hu/
[5]https://huggingface.co/NYTK
[6]https://llama.meta.com/llama3
[7]https://ai.meta.com

models come in variants with 7 billion, 13 billion, 33 billion, and 65 billion parameters. The smallest model was trained on a corpus of 1 trillion tokens, while the largest models were trained on 1.4 trillion tokens. The Llama 2 family includes variants with 7, 13, and 70 billion parameters, all trained on 2 trillion tokens, with the input text length increased to 4096 tokens. The Llama 2 models were immediately fine-tuned on a corpus containing more than 1 million fine-tuning prompts. Two types of fine-tuned models were developed: chat and code generation. The fine-tuning process follows the steps proposed by ChatGPT: initially performing supervised fine-tuning, followed by reinforcement learning-based fine-tuning. The Llama 3 family comprises two variants with 8 and 70 billion parameters. These models were pretrained on over 15 trillion tokens and fine-tuned on over 10 million human-annotated examples. The popularity of Llama models stems from their exceptional performance and the fact that they are freely accessible.

The free accessibility of Llama models has led to numerous research projects and initiatives based on them. One popular project is the Stanford Alpaca project [7], inspired by the 'Self-Instruct' method [8], which used OpenAI's *text-davinci-003* model to generate 52,000 prompts. These generated prompts were then used to fine-tune a 7 billion parameter LLaMA model through supervised learning. Despite the many errors in the generated prompts, the fine-tuned instruction-following LLaMA model can achieve performance similar to the *text-davinci-003* model.

In the Vicuna project [9], an open-source chat application was created using the 13 billion parameter LLaMA model, leveraging the ShareGPT database, which offers competitive performance compared to the ChatGPT application.

OpenChatKit [10], developed through the collaboration of Together AI[8], LAION[9], and Ontocord.AI[10], offers implementations and various fine-tuned LLaMA and GPT-NeoX models for different purposes.

Furthermore, Tsinghua University in China is developing its own language models, the GLM models [11] and their fine-tuned versions, such as ChatGLM3 [11], [12], a 6.2 billion parameter bilingual conversational model, which is currently one of the most popular ChatGPT replacement applications in China. It surpasses ChatGPT's capabilities in the Chinese language.

Following the release of various instruction-following corpora, translations into other languages began to enable the training of instruction-following models in those languages. For instance, the Stanford Alpaca corpus has been translated into Chinese [13] and Italian [14], while the Dolly corpus has been translated into Chinese [15], Japanese[11], and six other languages[12]. Bactrian-X [16] built a multilingual instruction-following model by machine-translating instructions and inputs/contexts from both the Stanford Alpaca and Dolly corpora and then generating responses in various languages using the *text-davinci-003* model to train the model with these prompts.

Creating corpora through translation is a common practice. Some subcorpora of the Hungarian HuLU [17], [18] were also created by translating the English GLUE and SuperGLUE [19], [20] subcorpora.

In recent months, Sambanova[13] has conducted fine-tuning of Llama-2 models in various languages, including Hungarian. In their research [21] a Llama-2 7B and a Llama-2 70B were trained to Hungarian (SambaLingo-Hungarian-Base and SambaLingo-Hungarian-Base-70B) by training on 59 billion tokens from the Hungarian split of the CulturaX [22] dataset. Then, human aligned chat models were fine-tuned (SambaLingo-Hungarian-Chat and SambaLingo-Hungarian-Chat-70B). These models are trained using direct preference optimization on top the base models. For fine-tuning, the ultrachat_200k dataset [23] was mixed with the Google translated version of the ultrachat_200k dataset was used. Our research is similar to the Sambalingo project.

To our knowledge, there is neither an high quality instruction-following prompt collection nor an official publicly available instruction-following large language model for the Hungarian language (except for Sambalingo models).

III. Corpora

A. Training Corpora for Pretraining

In this research, we further pre-trained the LLaMA-2-7B-32K[14] model from the Together for Hungarian. For this task, we selected only the long documents from our previous pre-training corpora [3] (PULI Long). We kept only the documents that exceed 5,000 words in length. The main characteristics of the corpus can be found in Table I.

TABLE I
Main characteristics of the pre-training corpora

	Documents	Words	Avg doc length avg / median (words)
PULI Long	763,704	7,902,519,115	10,823.38 / 7,149
Long Context QA	88,957	1,009,562,704	11,348.88 / 11,274
BookSum	9,600	42,339,698	4,410.39 / 3,265.5

In our training, we also incorporated the original fine-tuning corpora[15] that Together used for training the LLaMA-2-7B-32K model: Multi-passage QA from Natural Questions (Long Context QA) and BookSum. In Table I, you can see the statistics of the corpora.

B. Training Corpora for Fine-tuning

To fine-tune the instruction-following model, a high-quality instruction dataset is essential. In the current state, we have 15,064 Hungarian prompts. The main subcorpora are as follows:

[8]https://together.ai
[9]https://laion.ai
[10]https://www.ontocord.ai
[11]https://huggingface.co/datasets/kunishou/databricks-dolly-15k-ja
[12]https://www.kaggle.com/datasets/mygaps/databricks-dolly-15k-parallel-corpus-6
[13]https://sambanova.ai
[14]https://huggingface.co/togethercomputer/LLaMA-2-7B-32K
[15]https://huggingface.co/datasets/togethercomputer/Long-Data-Collections

- **Alpaca-Hu-2k** [24]: Randomly selected 2,000 prompts from the Stanford Alpaca corpus and 100 localized prompts.
- **HuLU prompts**: 1,200 prompts generated from HuLU benchmark (200 per subcorpus), HuCB, HuCOLA, HuCoPa, HuRTE, HuSST, and HuWNLI, respectively.
- **Graduation tasks**: 1,226 Hungarian high school graduation tasks, including history, literature, grammar, mathematics and chemistry.
- **SQL**: 795 SQL program code generation prompts.
- **Translation**: We used the 997 development set of the FLORES-101 corpus [25] to generate translation prompts.
- **Chat prompts**: There are 1,011 conversation prompts generated with different large language models, such as Llama-2. The conversations are between AI and users on different topics.
- **Long summarization**: 1,452 prompts for the summarization task, including long document summarization.
- **Title and keyword generation**: 200 prompts for title generation tasks and 200 prompts for keyword generation tasks, generated from news articles.
- **MILQA**: 1,000 question answering prompts generated from MILQA corpus [26].
- **OCR**: 1,972 OCR cleaning prompts.
- **User questions**: 923 prompts generated by questions that users ask from our demo models. In these cases, the answers are manually written by annotators.
- **Public collections**: 221 prompts from public available collections, 'aya dataset hu'[16] and 'Hungarian llm testing'[17], respectively.
- **Miscellaneous**: 1,767 prompts from different customer datasets that can be made public.

For instruction fine-tuning, we used the prompt template recommended by the Stanford Alpaca research (see Table II). The optional parts are enclosed in [] brackets. The English translations are in () brackets.

TABLE II
PROMPT TEMPLATE

Az alábbiakban egy utasítást találsz, amely leír egy feladatot [, amelyhez egy bemenetet is mellékelünk, hogy további összefüggéseket adjon]. Írj egy választ, amely megfelelően teljesíti a feladatot!
(Below is an instruction that describes a task [, paired with an input that provides further context]. Write a response that appropriately completes the request.)

Utasítás (Instruction):
{*Instruction text*}

[### Bemenet (Input):]
{*Context text*}

Válasz (Response):
{*Answer text*}

[16]https://huggingface.co/datasets/boapps/aya_dataset_hu
[17]https://huggingface.co/datasets/Bazsalanszky/hungarian-llm-testing

IV. Methods, Experiments

A. Pre-training Large Language Model

In our current research, we further pre-trained a Llama-2 model for Hungarian. The original Llama-2 model has an input context length of 4096 tokens. Based on our practical experience, this context window is often insufficient. Thus, we extended the context length to 32,768 tokens. For this task, we used the implementation from OpenChatKit[18]. We chose the fine-tuned Llama-2-7B-32K[19] model as our starting point. LLaMA-2-7B-32K is an long context language model, fine-tuned from the Llama-2 7B model. The model's context length has been extended using position interpolation. Using the LLaMA-2-7B-32K model, we continuously pre-trained it with our Hungarian long documents. In the first half epoch, we used only the Hungarian dataset. However, during testing, we observed that the model began to forget its English knowledge. To prevent this, in the second half epoch, we mixed in the fine-tuning dataset (Long Context QA and BookSum) that was used for the original LLaMA-2-7B-32K model. The training hyperparameters are as follows: 2e-5 learning rate, 7 batch size per GPU, fp16. We used 8 NVIDIA A100 (80GB) GPUs for this task. The training took approximately two months and we stop the training at 100,000 steps. This new pre-trained model is named **PULI LlumiX 32K**.

B. Fine-tuning Instruct Models

According to the research of Yang et al. [27], we mixed Stanford Alpaca [7] and Dolly [28] prompts in other languages:

- English cleaned Stanford Alpaca: 51,760 prompts
- English Dolly: 15,011 prompts
- Chinese cleaned Stanford Alpaca: 27,080 prompts
- Chinese Dolly: 15,015 prompts

Altogether, we had 123,930 prompts for fine-tuning. For the fine-tuning task, we utilized the Stanford Alpaca implementation[20]. The hyperparameters used are as follows: 2e-5 learning rate, bf16, 2 batch size per GPU, 16 gradien accumulation steps, 3 epochs. We used 8 NVIDIA A100 (80GB) GPUs for this task.

In our experiment, we fine-tuned the following instructional models:

- PULI GPT-3SX Instruct
- ParancsPULI (PULI GPTrio Instruct)
- PULI LlumiX 32K Instruct

V. Results

For better comparison with Yang et al.'s research [3], we evaluated our models on the HuCOLA, HuSSt, and HuRTE benchmarks. In recent months, Sambanova has trained various large language models in multiple languages, including Hungarian, representing significant competition for our models.

[18]https://github.com/togethercomputer/OpenChatKit
[19]https://huggingface.co/togethercomputer/LLaMA-2-7B-32K
[20]https://github.com/tatsu-lab/stanford_alpaca

Therefore, in our evaluation, we compared our models with the 7 billion parameters SambaLingo Hungarian models.

In the case of base large language models, we conducted few-shot learning experiments, while for instruct models, we employed zero-shot experiments. For few-shot learning, we applied the same settings as those used by Yang et al [3]. For zero-shot learning, we conducted experiments with various prompts. In Table III, we present the highest results obtained.

TABLE III
PERFORMANCE OF THE MODELS ON HULU BENCHMARKS

	HuCOLA	HuSST	HuRTE
	few-shot		
PULI GPT-3SX	54.27	64.27	57.42
PULI GPTrio	52.71	61.58	54.54
PULI LlumiX 32K	**57.66**	**76.89**	**66.98**
SambaLingo Hungarian Base	56.96	76.55	51.25
	zero-shot		
PULI GPT-3SX Instruct	61.76	46.27	52.09
PULI GPTrio Instruct	52.12	59.20	58.14
PULI LlumiX 32K Instruct	**62.41**	**69.60**	**72.58**
SambaLingo-Hungarian-Chat	53.06	55.15	60.98
ChatGPT (turbo 3.5)	49.10	36.99	50.26
text-davinci-001	50.78	35.48	49.06

In Table III, you can see the performance of the models on HuLU benchmarks. In the few-shot learning task, our PULI LlumiX 32K model outperformed the other models in all the three benchmarks. The SambaLingo in HuCOLA and HuSST achieve similar high performance. The high result of SambaLingo can be the fact that the SambaLingo was trained on 59 billion Hungarian tokens.

In the instruction-following task, our PULI LlumiX 32K Instruct model achieved the highest results in all the three benchmarks. SambaLingo performed well in the HuRTE benchmark. Additionally, the monolingual PULI GPT-3SX Instruct model achieved high results in HuCOLA, benefiting from its deeper understanding of Hungarian, thereby outperforming in this task. However, in the other two tasks, we observed that the multilingual models performed better, demonstrating the effectiveness of transfer learning.

For better comparison with the SambaLingo models, we evaluated our PULI LlumiX 32K model in a machine translation task. For evaluation, we used the sacreBLEU metric [29] and chrF [30]. In the SambaLingo research [21], their model achieved significantly high performance in machine translation tasks with few-shot learning. They used an 8-shot evaluation method using the '{IN}={OUT}' prompt as recommended by Zhu et al. [31]. The prompts were chosen randomly, making reproduction difficult. Hence, we attempted to reproduce this evaluation in our environment. The reproduced scores are shown in Table IV. As expected, in the few-shot learning task, we could not outperform SambaLingo. However, with the instructed models, we achieved higher performance in the zero-shot task. There was an interesting observation with the SambaLingo-Hungarian-Chat model: in the case of the HU-EN direction, if we gave the prompt in Hungarian, the model could only generate Hungarian output. For the English translation output, we needed to give the prompt in English, which is not the expected behavior from a Hungarian language model.

TABLE IV
PERFORMANCE OF THE MODELS ON FLORES

	EN-HU (BLEU / chrF)	HU-EN (BLEU / chrF)
	few-shot	
PULI LlumiX 32K	13.73 / 42.30	15.43 / 35.84
SambaLingo Hungarian Base	**15.28 / 45.78**	**23.28 / 48.84**
	zero-shot	
PULI LlumiX 32K Instruct	**18.32 / 50.63**	**27.85 / 56.56**
SambaLingo-Hungarian-Chat	17.58 / 50.31	21.85 / 48.34

We also conducted an experiment in keyword generation for long documents to leverage the capabilities of our long-context model. We fine-tuned the monolingual PULI GPT-3SX and the PULI LlumiX 32K for keyword generation task. The documents were sourced from the REAL repository's articles[21], with the articles averaging around 5,000 words. The training corpus contained 1,000 articles, while the test corpus consisted of 145 articles. The Table V shows that the PULI LlumiX 32K achieved significantly high performance in this task.

TABLE V
RESULTS OF KEYWORD GENERATION FOR LONG DOCUMENT TASK

	Precision	Recall	F-score
PULI GPT-3SX	14.64	13.89	14.26
PULI LlumiX 32K	**41.05**	**38.79**	**39.89**

In Table VI, there are some interesting samples that demonstrate our PULI LlumiX 32K Instruct model. In the first sample, the model correctly knows that Eger is not a capital, but rather a town in Hungary. The second sample demonstrates the model's world knowledge. It understands that a deaf person probably would not go to the opera. In the third sample, the model was able to solve a logical problem.

VI. CONCLUSION

During our research, we took important steps in developing Hungarian instruction-following models. We built an instruction fine-tuning dataset for Hungarian and fine-tuned the Hungarian PULI large language models for various tasks. In our current research, supervised fine-tuning was emphasized. Our PULI LlumiX 32K Instruct model became one of the state-of-the-art instruction models for Hungarian. With this new model, we are able to answer questions and solve various language technology problems.

Throughout the research, we found that transfer learning enables the model to acquire a lot of knowledge from other languages as well. In our next step, we plan to conduct experiments with the new Llama-3 models, both the 8 billion and 70 billion parameter versions.

[21] http://real.mtak.hu

TABLE VI
SOME SAMPLES OF OUR PULI LLUMIX 32K INSTRUC MODEL

1.	**Prompt:** Melyik ország fővárosa Eger? (Which country's capital is Eger?) **Response:** Eger nem egy ország fővárosa, hanem egy város Magyarországon. (Eger is not the capital of a country, but rather a town in Hungary.)
2.	**Prompt:** Szerinted milyen gyakran jár operába Béla, ha Béla süket? (How often do you think Béla goes to the opera if Béla is deaf?) **Response:** Szerintem Béla nem jár operába, ha süket. (I don't think Béla goes to the opera if he is deaf.)
3.	**Prompt:** Egy útelágazásnál jobbra lehetett menni vagy balra. Péter szerint jobbra volt a cél, míg Mari szerint balra. Péter azonban tévedett. Merre volt a cél? (At a crossroads, one could go right or left. Péter believed the destination was to the right, while Mari thought it was to the left. However, Péter was mistaken. Which way was the destination?) **Response:** Péter szerint jobbra volt a cél, míg Mari szerint balra. A helyes válasz tehát az, hogy balra volt a cél. (According to Péter, the destination was to the right, while Mari thought it was to the left. Therefore, the correct answer is that the destination was to the left.)

REFERENCES

[1] A. Chowdhery, S. Narang, J. Devlin, M. Bosma, G. Mishra, A. Roberts, P. Barham, H. W. Chung, C. Sutton, S. Gehrmann, P. Schuh, K. Shi, S. Tsvyashchenko, J. Maynez, A. Rao, P. Barnes, Y. Tay, N. Shazeer, V. Prabhakaran, E. Reif, N. Du, B. Hutchinson, R. Pope, J. Bradbury, J. Austin, M. Isard, G. Gur-Ari, P. Yin, T. Duke, A. Levskaya, S. Ghemawat, S. Dev, H. Michalewski, X. Garcia, V. Misra, K. Robinson, L. Fedus, D. Zhou, D. Ippolito, D. Luan, H. Lim, B. Zoph, A. Spiridonov, R. Sepassi, D. Dohan, S. Agrawal, M. Omernick, A. M. Dai, T. S. Pillai, M. Pellat, A. Lewkowycz, E. Moreira, R. Child, O. Polozov, K. Lee, Z. Zhou, X. Wang, B. Saeta, M. Diaz, O. Firat, M. Catasta, J. Wei, K. Meier-Hellstern, D. Eck, J. Dean, S. Petrov, and N. Fiedel, "PaLM: Scaling Language Modeling with Pathways," 2022.

[2] Z. Gy. Yang, R. Dodé, G. Ferenczi, E. Héja, K. Jelencsik-Mátyus, A. Kőrös, L. J. Laki, N. Ligeti-Nagy, N. Vadász, and T. Váradi, "Jönnek a nagyok! BERT-Large, GPT-2 és GPT-3 nyelvmodellek magyar nyelvre," in *XIX. Magyar Számítógépes Nyelvészeti Konferencia (MSZNY 2023)*. Szeged, Hungary: Szegedi Tudományegyetem, Informatikai Intézet, 2023, pp. 247–262.

[3] Z. Gy. Yang, L. J. Laki, T. Váradi, and G. Prószéky, "Mono- and multilingual GPT-3 models for Hungarian," in *Text, Speech, and Dialogue*, ser. Lecture Notes in Computer Science. Plzeň, Czech Republic: Springer Nature Switzerland, 2023, pp. 94–104.

[4] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. L. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray, J. Schulman, J. Hilton, F. Kelton, L. Miller, M. Simens, A. Askell, P. Welinder, P. Christiano, J. Leike, and R. Lowe, "Training language models to follow instructions with human feedback," 2022.

[5] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale, D. Bikel, L. Blecher, C. C. Ferrer, M. Chen, G. Cucurull, D. Esiobu, J. Fernandes, J. Fu, W. Fu, B. Fuller, C. Gao, V. Goswami, N. Goyal, A. Hartshorn, S. Hosseini, R. Hou, H. Inan, M. Kardas, V. Kerkez, M. Khabsa, I. Kloumann, A. Korenev, P. S. Koura, M.-A. Lachaux, T. Lavril, J. Lee, D. Liskovich, Y. Lu, Y. Mao, X. Martinet, T. Mihaylov, P. Mishra, I. Molybog, Y. Nie, A. Poulton, J. Reizenstein, R. Rungta, K. Saladi, A. Schelten, R. Silva, E. M. Smith, R. Subramanian, X. E. Tan, B. Tang, R. Taylor, A. Williams, J. X. Kuan, P. Xu, Z. Yan, I. Zarov, Y. Zhang, A. Fan, M. Kambadur, S. Narang, A. Rodriguez, R. Stojnic, S. Edunov, and T. Scialom, "Llama 2: Open Foundation and Fine-Tuned Chat Models," 2023.

[6] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, A. Rodriguez, A. Joulin, E. Grave, and G. Lample, "LLaMA: Open and Efficient Foundation Language Models," *arXiv preprint arXiv:2302.13971*, 2023.

[7] R. Taori, I. Gulrajani, T. Zhang, Y. Dubois, X. Li, C. Guestrin, P. Liang, and T. B. Hashimoto, "Stanford Alpaca: An Instruction-following LLaMA model," 2023. [Online]. Available: https://github.com/tatsu-lab/stanford_alpaca

[8] Y. Wang, Y. Kordi, S. Mishra, A. Liu, N. A. Smith, D. Khashabi, and H. Hajishirzi, "Self-Instruct: Aligning Language Model with Self Generated Instructions," 2022.

[9] W.-L. Chiang, Z. Li, Z. Lin, Y. Sheng, Z. Wu, H. Zhang, L. Zheng, S. Zhuang, Y. Zhuang, J. E. Gonzalez, I. Stoica, and E. P. Xing, "Vicuna: An Open-Source Chatbot Impressing GPT-4 with 90%* ChatGPT Quality," March 2023. [Online]. Available: https://lmsys.org/blog/2023-03-30-vicuna/

[10] T. Computer, "OpenChatKit: An Open Toolkit and Base Model for Dialogue-style Applications," 3 2023. [Online]. Available: https://github.com/togethercomputer/OpenChatKit

[11] A. Zeng, X. Liu, Z. Du, Z. Wang, H. Lai, M. Ding, Z. Yang, Y. Xu, W. Zheng, X. Xia *et al.*, "Glm-130b: An open bilingual pre-trained model," *arXiv preprint arXiv:2210.02414*, 2022.

[12] Z. Du, Y. Qian, X. Liu, M. Ding, J. Qiu, Z. Yang, and J. Tang, "Glm: General language model pretraining with autoregressive blank infilling," in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2022, pp. 320–335.

[13] Y. Cui, Z. Yang, and X. Yao, "Efficient and Effective Text Encoding for Chinese LLaMA and Alpaca," *arXiv preprint arXiv:2304.08177*, 2023. [Online]. Available: https://arxiv.org/abs/2304.08177

[14] A. Santilli and E. Rodolà, "Camoscio: an italian instruction-tuned llama," 2023.

[15] Q. C. Ziang Leng and C. Li, "Luotuo: An Instruction-following Chinese Language model, LoRA tuning on LLaMA," https://github.com/LC1332/Luotuo-Chinese-LLM, 2023.

[16] H. Li, F. Koto, M. Wu, A. F. Aji, and T. Baldwin, "Bactrian-x : A multilingual replicable instruction-following model with low-rank adaptation," 2023.

[17] N. Ligeti-Nagy, G. Ferenczi, E. Héja, K. Jelencsik-Mátyus, L. J. Laki, N. Vadász, Z. Gy. Yang, and T. Váradi, "HuLU: magyar nyelvű benchmark adatbázis kiépítése a neurális nyelvmodellek kiértékelése céljából," in *XVIII. Magyar Számítógépes Nyelvészeti Konferencia*. Szeged: JATEPress, 2022, p. 431–446.

[18] N. Ligeti-Nagy, E. Héja, L. J. Laki, D. Takács, Z. Gy. Yang, and T. Váradi, "Hát te mekkorát nőttél! - A HuLU első életéve új adatbázisokkal és webszolgáltatással ," in *XIX. Magyar Számítógépes Nyelvészeti Konferencia*. Szeged: JATEPress, 2023, p. 217–230.

[19] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, and S. Bowman, "GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding," in *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*. Brussels, Belgium: Association for Computational Linguistics, Nov. 2018, pp. 353–355. [Online]. Available: https://aclanthology.org/W18-5446

[20] A. Wang, Y. Pruksachatkun, N. Nangia, A. Singh, J. Michael, F. Hill, O. Levy, and S. R. Bowman, "SuperGLUE: A Stickier Benchmark for General-Purpose Language Understanding Systems," 2020.

[21] Z. Csaki, B. Li, J. Li, Q. Xu, P. Pawakapan, L. Zhang, Y. Du, H. Zhao, C. Hu, and U. Thakker, "Sambalingo: Teaching large language models new languages," 2024.

[22] T. Nguyen, C. V. Nguyen, V. D. Lai, H. Man, N. T. Ngo, F. Dernoncourt, R. A. Rossi, and T. H. Nguyen, "Culturax: A cleaned, enormous, and multilingual dataset for large language models in 167 languages," 2023.