Towards Large Language Models for Everyone: Instruction Following, Knowledge Retrieval and Multilingualism

Xi Victoria Lin

A dissertation

submitted in partial fulfillment of the

requirements for the degree of

Doctor of Philosophy

University of Washington
2024

Reading Committee:
Luke Zettlemoyer, Chair
Yejin Choi
Hannaneh Hajishirzi

Program Authorized to Offer Degree:
Computer Science and Engineering

University of Washington

Abstract

Towards Large Language Models for Everyone:
Instruction Following, Knowledge Retrieval and Multilingualism

Xi Victoria Lin

Chair of the Supervisory Committee:
Professor Luke Zettlemoyer
Computer Science and Engineering

Large language models (LLMs) have significantly advanced the field of Natural Language Processing and demonstrated the potential to fuel a variety of AI applications. Nonetheless, building them in a way that maximally benefits the very wide range of everyday use cases is challenging. Firstly, LLMs are pre-trained with the next-token prediction objective, which does not align well with specific user requests. Secondly, LLMs suffer from knowledge cut-off and tend to hallucinate about long-tail facts. Lastly, popular LLMs are trained on almost exclusively English text, making it difficult for non-English speakers to adopt them.

This thesis presents methodologies addressing all three challenges. We begin by studying the Instruction Meta-Learning (IML) approach, enabling LLMs to perform an array of tasks by fine-tuning them over pairs of natural language instructions and responses. Our study highlights the efficacy of scaling IML along three axes: fine-tuning task diversity, language diversity and model parameters. Next, we propose integrating LLMs with an external data store during IML (retrieval-augmented dual instruction tuning, RA-DIT). RA-DIT significantly improves LLM performance in scenarios that require access to large, external knowledge sources (e.g., answering information-seeking questions). Finally, we introduce a family of cross-lingual generative language models (XGLMs) pre-trained on a multilingual corpus exhibiting a heavy-tailed distribution. XGLMs demonstrate enhanced cross-lingual capabilities and few-shot generalization across medium- and low-resource languages. Together, these research strands provide core strategies for advancing

the boundaries of LLM capabilities and paving the way towards real-world deployment.

Acknowledgements

My PhD is a special chapter in my life which was filled with exhilaration, growth, personal challenges and transformation. Reflecting on this journey, I am struck by the swift passage of a decade and the evolution of my own identity from its onset to its conclusion. It is with a deep sense of gratitude that I acknowledge the abundance of blessings, both familiar and newfound, that have accompanied me through the trials and transformations.

I am deeply grateful to my advisor, Luke Zettlemoyer, for his tremendous support. His open-mindedness, approachable nature, unwavering dedication to science, and astute judgment in research have been truly inspirational. My experience at the Department of Computer Science and Engineering at the University of Washington (UW CSE) was mind-opening. Its vibrant and inclusive culture fosters innovation and collaboration, enabling me to pursue my PhD in two distinct phases: initially as a student within the department, and later as a full-time industrial researcher. I also express my heartfelt gratitude to my late former advisor, Ben Taskar, who had brought me into this dynamic community and an exceptional environment for conducting machine learning research.

I have had the privilege of being mentored by many outstanding researchers through sustained collaboration across various domains of natural language processing and artificial intelligence (AI). My gratitude extends to Ves Stoyanov, who opened my eyes to the landscape of large-scale language modeling (LLMs); to Scott Yih, for a venture into the wonder of open information retrieval and its synergistic integration with LLMs; to Richard Socher, for changing my view of neural networks as universal problem solvers; and to Caiming Xiong, for passing on acute insights into the integration of deep learning with diverse application challenges during the technology's infancy. Additionally, I am also indebted to Mike Ernst, my MSc thesis co-advisor, who encouraged my early work on the then-nascent topic of programming with natural language,

which has now evolved into a flourishing field; and to Sameer Singh, for his mentorship when I just started doing core machine learning research.

The majority of my thesis work was completed at the Fundamental AI Research team at Meta. I extend my sincere gratitude to my managers, Ves Stoyanov, Scott Yih and Daniel Li, for their firm support as I pursued my PhD concurrently with my full-time role. My gratitude also extends to Yejin Choi, Hannaneh Hajishirzi, Ves Stoyanov and Lucy Wang, for being willing to serve on my thesis committee and providing valuable feedback to this work.

Research collaboration has been an invaluable part of my PhD journey. I especially thank the following collaborators for the vibrant intellectual exchanges: Mikel Artetxe, Todor Mihaylov, Naman Goyal, Myle Ott, Stephen Roller, Srini Iyer, Ramakanth Pasunuru, Sida Wang, Tianlu Wang, Jingfei Du, Xilun Chen, Mingda Chen, Mona Diab, Omer Levy, Mike Lewis, Wang-Chiew Tan and Alon Halevy at Meta; Nazneen Rajani, Victor Zhong, Yingbo Zhou, Bryan McCann, Nitish Keskar, James Bradbury, Melvin Gruesbeck, Kazuma Hashimoto, Alex Trott, Romain Paulus, Stephen Merity, Chien-Sheng Wu, Xinyi Yang, Tian Xie, Wenpeng Yin, Tong Niu, Stephen Hoi and Mark Riedl at Salesforce; Chenglong Wang, Calvin Loncaric, Deric Pang and Kevin Vu for the partnership on Tellina; Tao Yu, Rui Zhang and Dragomir Radev for the quest on natural language interfaces to structured data and beyond. I am also grateful to my internship mentors at Microsoft Research, Kristina Toutanova, Scott Yih, Hoifung Poon, and Chris Quirk, as well as those at the Allen Institute for Artificial Intelligence (AI2), Tom Kwiatkowski, for their guidance and insights.

I was fortunate to be part of the Natural Language Processing Group at the University of Washington (UW NLP), a vibrant and diverse hub of researchers across various NLP disciplines. I thank my wonderful groupmates for their support and insightful research exchanges, especially Kenton Lee, Luheng He, Eunsol Choi, Nicholas FitzGerald, Mark Yatskar, Srini Iyer, Mandar Joshi, Omer Levy, Mike Lewis, Yannis Konstas, Chloé Kiddon, Sameer Singh, Victor Zhong, Weijia Shi, Sewon Min, Margaret Li, Ari Holtzman, Tim Dettmers, Suchin Gururangan, Huan Sun, Lingpeng Kong, Minjoon Seo, Aaron Jaech, Yi Luan, Rowan Zellers, Trang Tran, Tongshuang Wu, Akari Asai and Yizhong Wang. Additionally, a special thanks to Maarten Sap and Hannah Rashkin for preparing this widely adopted thesis template.

I also extend my gratitude to the wonderful staff at UW CSE, especially our graduate student service

(GSR) directors, Elise deGoede Dorough and Lindsay Michimoto, who have offered substantial help during each milestone of my PhD. My gradtitude also extends to my professors and mentors at the Department of Computer and Information Science at the University of Pennsylvania (UPenn CIS), who have provided invaluable guidance during the initial phase of my graduate studies.

My PhD journey was enriched by the invaluable friendship from many. I am grateful to Diyi Yang, Jessy Li, Wendy Shang, He He, Sheng Zha and Chang Liu for their enduring friendship and support through the tough times; to my officemates at UW, Ryan Mass, Dominik Moritz, Thierry Moreau, Kristi Morton, and Alex Colburn, for a dynamic working environment rich with interdisciplinary expertise; to Irene Zhang for her mentorship upon my arrival at UW, and for being an inspiring role model and advocate for women in computer science and engineering. I also thank Shu Liang, Chenglong Wang, Pavel Panchekha, Yanping Huang, Luheng He, Qiao Zhang, Xiaoyi Zhang, Shumo Chu, Tianqi Chen, Tianyi Zhou, Audrey Cook, Jennifer Gillenwater, David Weiss, Brian Dolhansky, Robert Rand, Meng Xu, Fei Miao, Congle Zhang, Yuyin Sun, Shirley Ren, Jinna Lei, Bing Xu, Mianmian Wen, Tim Shi, Navjot Matharu, Yufei Hu, Kerui Min and the graduate student communities at UW CSE and UPenn CIS for adding joy and fun to this ride.

Lastly, I am grateful to Anat Caspi and Aviv Taskar, who continue to inspire me daily. I am also thankful for Daisy, Evan, and Angela, my extended family in the Bay Area, whose companionship over the years has been invaluable in navigating the challenges of balancing a PhD and a demanding career in the industry environment of Silicon Valley. My deepest appreciation goes to my parents and grandparents, who bestowed me the freedom to pursue my dreams, all while providing unwavering love, support and inspiration.

DEDICATION

In memorial of Ben Taskar

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Large language models (LLMs) are neural architectures with billions of parameters pre-trained over tremendous amount of unlabeled data Brown et al. [2020]. They have shown substantial promise as general-purpose, multi-task learners [Bommasani et al., 2021], and have fueled successful products such as ChatGPT[1], Meta AI[2] and Google Bard[3]. This thesis tackles three key challenges in the usability of LLMs by proposing methodologies that enable them to follow natural language instructions, access external knowledge and process input in different languages.

1.1 The Rise of Large Language Models

Contextualized pre-training of language representations was introduced by the pioneering work of McCann et al. [2017], Peters et al. [2018] and Devlin et al. [2019]. These early work focus on encoder-only architectures, leaving practitioners to separately fine-tune prediction gates or a decoder for various downstream tasks. Raffel et al. [2020] and Lewis et al. [2020a] further extended this paradigm to pre-traing sequence-to-sequence architectures. Meanwhile, GPT [Radford and Narasimhan, 2018] and GPT-2 [Radford et al., 2021] demonstrated the potential for simple autoregressive language models trained with the next-token prediction objective to function as end-to-end multi-task learners. For example, given the question "*Who wrote the book the origin of species?*" as the prompt, the model generates "*Charles Darwin*" as the response.

[1] `https://chat.openai.com`
[2] `https://www.meta.ai`
[3] `https://bard.google.com`

GPT-3 [Brown et al., 2020] further showcased the power of scaling language model pre-training along two dimensions: the model size and the amount of pre-training data. With 175 billion parameters trained over 300 billion English tokens, the model acquired many "emergent capabilities", including being able to follow few-shot examples formatted with textual templates to perform new tasks [Min et al., 2022b].

Subsequent work has further improved LLMs by scaling up both model size and pre-training data [Chowdhery et al., 2022; Touvron et al., 2023a,b; Hoffmann et al., 2022]. However, scaling LLM pre-training is extremely costly, and to date, has only been done by a small number of organizations. While some of the most powerful LLMs are proprietary, the capabilities of open-sourced LLMs have consistently risen thanks to investments from multiple institutions. The work in this thesis are based on strong open-sourced LLMs such as OPT [Zhang et al., 2022] and Llama [Touvron et al., 2023a]. Additionally, we introduce two new LLMs: the instructable LLM (OPT-IML), presented in Chapter 2, and the multilingual LLM (XGLM), introduced in Chapter 4.

1.2 The Alignment Problem

While pre-trained LLMs are capable of multi-tasking in both zero- and few-shot settings [Brown et al., 2020], their performance is extremely sensitive to the prompt formulation [Gao et al., 2021; Liu et al., 2023]. Moreover, the self-supervised learning objective, which focuses on predicting the next token, inherently results in pre-trained LLMs being particularly good at completing sentences. As a result, they are unable to directly respond to most user requests specified in natural language (Figure 1.1). To address this problem, several concurrent work [Mishra et al., 2022; Wei et al., 2022b; Sanh et al., 2022] introduced *instruction-tuning*, an approach that fine-tunes LLMs using a collection of tasks expressed via instruction and ground-truth response pairs. This improves their problem-solving abilities when provided with a task description in natural language. The resulting LLMs, commonly referred to as *instructable LLMs*, demonstrated significant performance improvement following instructions in a zero-shot manner. This enhancement in usability made these models much more practical and useful for a wide range of applications.

Initial scaling-law curves from these studies suggest the potential for improved cross-task generalization through further scaling of both *model size* and *task diversity*. We therefore pose the following research question – ***can we enable an LLM to follow instructions to solve common natural language processing***

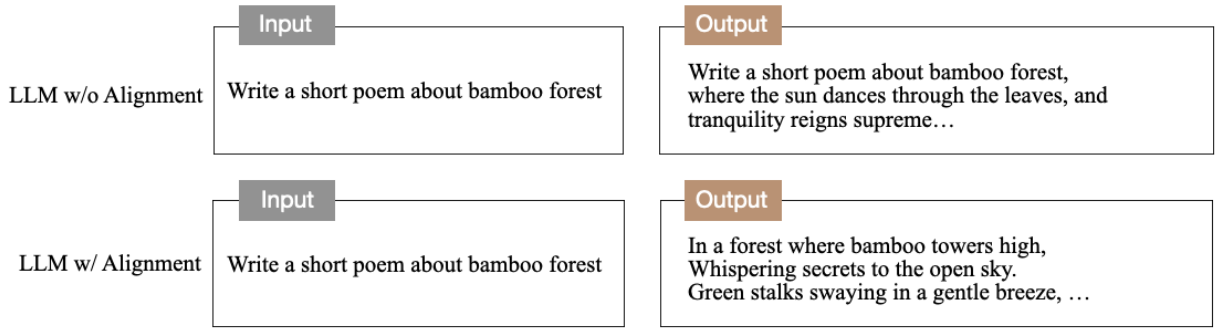

Figure 1.1: Pre-trained LLMs without instruction tuning tend to complete any input. Instruction tuning enables the model to generate content in response to the input specification (output generated using ChatGPT).

(NLP) tasks by fine-tuning it over a massive instruction dataset with high diversity? We consider NLP tasks that belong to a broad taxonomy, including *lexical tasks* (e.g. word analogy), *syntactic tasks* (e.g. part-of-speech tagging), *semantic tasks* (e.g. textual entailment) and *common NLP applications* (e.g. question answering, summarization and hate speech detection). Each level of the taxonomy includes more fine-grained task categories (with examples provided in parentheses). We use these tasks as the test bed to examine the generalization capabilities of instruction-tuned LLMs.[4]

To maximize the task and language diversity in our instruction tuning data, we compiled up to 2,000 NLP datasets from 8 instruction-tuning benchmarks previously released by the community (Table 2.1). In order to systematically evaluate the generalization capability of LLMs, we split the task collection into a training set and three evaluation sets designed to test three types of model generalization:

- to *unseen task categories (new instructions, new tasks, in- and out-of-domain examples);*
- to *unseen datasets from the same task category (new instructions, seen tasks, in- and out-of-domain examples);*
- to *unseen examples from the same dataset (seen instructions, in-domain examples).*

We fine-tuned OPT [Zhang et al., 2022] with 30B and 175B parameters on the training split of our benchmark, after carefully re-balancing the training data across multiple attributes such as task categories and annotation sources. We dubbed the resulting LLMs OPT-IML (OPT with *instruction meta learning*),

[4] Wang et al. [2022b] first introduced Super-NaturalInstructions, an instruction-tuning dataset consisting of 1,600+ NLP tasks. However, it only experimented with the encoder-decoder based T5 LMs [Raffel et al., 2020] of up to 13B parameters, and the instruction data were all taken from the same human annotation pipeline. Chung et al. [2022b] is the most closely related to our work. Our projects were initiated around the same time.

and the corresponding meta benchmark as OPT-IML Bench. ***Our experiments demonstrated the efficacy of developing instructable NLP task solvers by fine-tuning LLMs over a large and diverse instruction dataset.*** First, OPT-IML performs strongly when evaluated on unseen examples from a dataset included in the fine-tuning data. For example, OPT-IML-30B achieves > 0.85 ROUGE-L F1 metrics on SQuAD v1 [Rajpurkar et al., 2016], even though the fine-tuning data contains 1,500+ other datasets, and the model is effectively trained on only a small number of SQuAD v1 examples. This suggests that increasing the LLM size and engaging in extensive pre-training can effectively mitigate the task interference problem often observed in smaller multi-task models [McCann et al., 2018a], resulting in performance on par with state-of-the-art supervised learning models. Second, OPT-IML demonstrates strong generalization abilities over unseen instructions, both across different datasets and task categories. OPT-IML significantly improves over its base pre-trained model at both 30B and 175B scales on four different benchmarks: PromptSource [Sanh et al., 2022], FLAN [Wei et al., 2022b], Super-NaturalInstructions [Wang et al., 2022b], and UnifiedSKG [Xie et al., 2022]. Additionally, the OPT-IML models also perform competitively in comparison with each of the prior models individually tuned on these benchmarks in both zero and few-shot performance (§2.7).

We further ask the research question – ***what are the important factors that impact the effectiveness of instruction-tuning?*** Using our evaluation framework that considers multiple levels of generalization, we characterize the tradeoffs relating to different factors when scaling up instruction-tuning to the aggregate of 8 different benchmarks. We outline the tradeoffs of dataset and benchmark sampling strategies during tuning, the scaling laws with respect to tasks and categories, the effects of incorporating task demonstrations into instruction-tuning based on Min et al. [2022a], as well as instruction-tuning with specialized datasets that contain reasoning chains [Wei et al., 2022c] and dialogue. These experiments establish best practices for large-scale instruction-tuning of LLMs (§2.4).

While OPT-IML demonstrate a significant multi-tasking capability jump compared to its base model, we still observe model weaknesses similar to those observed in smaller multi-task learning models. For example, when evaluating on unseen instructions, the model performs better on instructions and tasks similar to those in the fine-tuning set. Adding more dissimilar instructions to the training set can hurt the performance. This highlights the importance of future research on multi-task learning and continuous learning using LLMs.

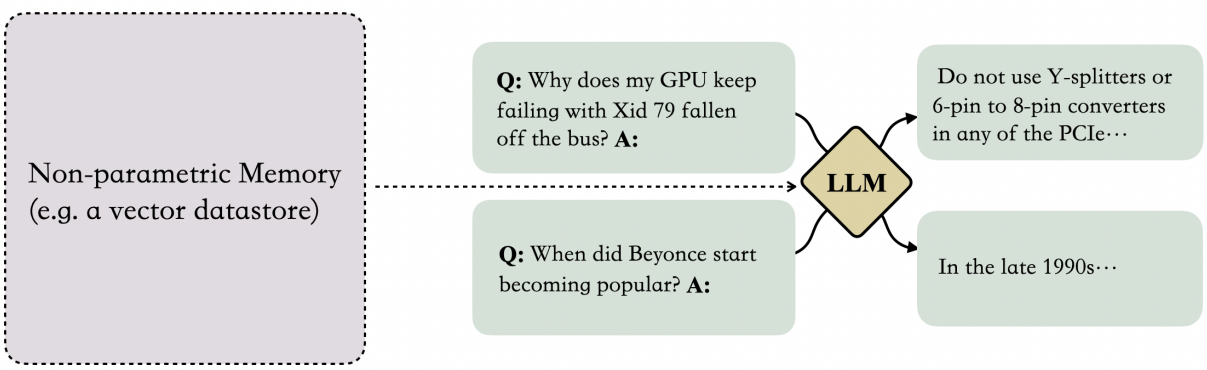

Figure 1.2: Non-parametric memory enables LLMs to solve a broader range of tasks, especially those requiring access to long-tail and private knowledge.

Two months prior to the publication of our work, Chung et al. [2022b] published FLAN-PaLM and FLAN-T5 along similar lines. FLAN-PaLM and FLAN-T5 achieve impressive gains on the challenging benchmarks of MMLU [Hendrycks et al., 2021a] and Big-Bench Hard [Suzgun et al., 2023] by instruction-tuning PaLM [Chowdhery et al., 2022] and T5 [Raffel et al., 2020] on a scaled-up collection of 1,800+ tasks. OPT-IML trained under similar settings still underperforms in comparison on these challenging benchmarks. In subsequent experiments, we determined that the primary factor contributing to the performance discrepancy between OPT-IML and FLAN-PalM is the inadequate training of the base LLM. The comparison with FLAN-T5, apart from differences in training data size, can also be attributed to architectural disparities between autoregressive LLMs and T5 [Artetxe et al., 2022b]. We believe that the insights gained from the experiments, particularly with regards to the importance of adequate pre-training, will be valuable for guiding future research in this area.

1.3 Retrieval Augmentation

Although LLMs post-alignment demonstrate strong instruction-following capabilities and are useful for various tasks, they suffer from the knowledge cutoff problem and cannot solve problems requiring knowledge beyond their pre-training cutoff date. Additionally, LLMs struggle with modeling long-tail knowledge and are inaccurate within this range. *Retrieval augmentation (RA)* enables LLMs to incorporate external knowledge for problem-solving by integrating them with non-parametric information retrieval (Figure 1.2) [Guu et al., 2020; Borgeaud et al., 2022; Shi et al., 2023b].

Previous work has applied retrieval augmentation to different language model architectures and in different model training stages. REALM [Guu et al., 2020] (encoder-only) and RETRO [Borgeaud et al., 2022] (autoregressive) opt for *end-to-end pre-training*, incorporating the retrieval component from the outset. Atlas [Izacard et al., 2022b] builds upon T5 [Raffel et al., 2020] with Fusion-in-Decoder [Izacard and Grave, 2021] modifications, and continuosly pre-trains the framework over unsupervised text. REPLUG [Shi et al., 2023b] and In-Context RALM [Ram et al., 2023] directly combine off-the-shelf LLMs and retrievers, showing that LLMs and retrievers, even when optimized independently, can be effectively fused through the emergent in-context learning capbabilities of LLMs.

However, incorporating retrieval during LLM pre-training significantly increases the system complexity and training cost, and the off-the-shelf fusion approach also has limitations, particularly as the LLMs are not inherently trained to incorporate retrieved content. Given fine-tuning is much cost-effective than pre-training, we pose the research question – ***is it possible to introduce retrieval augmentation during the LLM alignment stage while adapting the LLM to downstream tasks?*** To this end, we propose **R**etrieval-**A**ugmented **D**ual **I**nstruction **T**uning (RA-DIT), an approach that retrofits any LLM with retrieval capabilities via fine-tuning over a set of tasks selected to cultivate knowledge utilization and contextual awareness in the LM prediction.

The RA-DIT approach consists of two separate instruction tuning steps. For *language model fine-tuning*, we augment each fine-tuning prompt with a retrieved "background" field prepended to the instructions and fine-tune the LM using the label-loss objective [Chung et al., 2022b; Iyer et al., 2022]. By incorporating the background text during fine-tuning, we guide the LLM to optimally utilize the retrieved information and ignore distracting content [Shi et al., 2023a]. For *retriever fine-tuning*, we update the retriever using a generalized language-model-supervised retrieval objective [LSR, Shi et al., 2023b]. This way, we enable the retriever to yield more contextually relevant results, aligned with the preferences of the LLM. During inference, following [Shi et al., 2023b], we retrieve relevant text chunks based on the language model prompt. Each retrieved chunk is prepended to the prompt and the predictions from multiple chunks are computed in parallel and ensembled to produce the final output.

We demonstrate that each fine-tuning step leads to significant performance gains, and the fine-tuned LLM and retriever can be combined to achieve further improvements. We initialize the framework us-

ing pre-trained LLAMA [Touvron et al., 2023a] and a state-of-the-art dual-encoder based dense retriever, DRAGON+ [Lin et al., 2023a]. Our largest model, RA-DIT 65B, attains state-of-the-art performance in zero- and few-shot settings on knowledge intensive benchmarks, notably surpassing the un-tuned in-context RALM approach on datasets including MMLU [Hendrycks et al., 2021a] (+8.2% 0-shot; +0.7% 5-shot) and Natural Questions [Kwiatkowski et al., 2019] (+22% 0-shot; +3.8% 5-shot). RA-DIT 65B also substantially outperforms ATLAS 11B on eight knowledge-intensive tasks (+7.2% on average in the 64-shot fine-tuning setting). This suggests that language models and retrievers, when optimized independently and then fused through instruction-tuning, can compete effectively with RALMs that have undergone extensive continuous pre-training.

We further conducted a comprehensive model analysis, demonstrating the effectiveness of our approach across LLMs of varying sizes (7B, 13B and 65B). On average, smaller LLMs exhibited significantly larger relative performance improvements. Additionally, for simpler single-hop information retrieval tasks, both the 7B and 65B Llama models performed comparably with retrieval augmentation. However, in more complex tasks such as multi-hop question answering, retrieval augmentation failed to bridge the gap between smaller and larger LLMs, suggesting that parameter scaling confers unique capabilities to the model that cannot be replicated solely through non-parametric memory, as implemented in our approach.

We also found the retrieval component to be a fragile part of the framework, as the retriever frequently made errors and returned noisy text chunks (post fine-tuning). Through qualitative error analysis, a significant portion of our model's performance improvement can be attributed to its ability to effectively disregard irrelevant retrieval content, outperforming an LLM baseline fine-tuned with conventional instruction tuning methods. This observation highlights the importance of developing more robust and accurate retrieval mechanisms to further enhance the performance of retrieval-augmented language models. Additionally, we investigated the impact of different retrieval strategies on the overall performance, such as using different search algorithms or adjusting the number of retrieved documents. Our findings suggest that careful selection and optimization of the retrieval strategy can significantly contribute to the success of the approach.

1.4 Multilingualism

LLMs are impressive multi-task learners and can be used to solve a wide range of tasks. However, their pre-training data is predominantly English, making it difficult for non-English speakers to benefit from them. Although the training data of GPT-3 [Brown et al., 2020] contains a small percentage of non-English text (7%) allowing it to achieve some promising cross-lingual generalization, the model is almost exclusively deployed for use cases in English. Multilingual masked and sequence-to-sequence language models have been studied, including mBERT, XLM-R, mT5, and mBART Devlin et al. [2019]; Conneau et al. [2020]; Xue et al. [2021]; Liu et al. [2020]. These models are typically fine-tuned on large amount of labeled data in downstream tasks. In comparison, multilingualism for autoregressive LLMs is less well understood. This presents a significant challenge for developing LLMs that can effectively serve users across different languages and regions. Addressing this issue is crucial for ensuring that the benefits of LLM technology are accessible to a broader population and not limited to English speakers.

We pose the following research questions – ***can we train a multilingual LLM that performs competitively against state-of-the-art English LLMs while demonstrating strong multilingual and cross-lingual capabilities, particularly in medium- and low-resource languages?*** To address this, we train four multilingual generative language models (up to 7.5 billion parameters), dubbed XGLM's, and conducted a comprehensive study of multilingual zero- and in-context few-shot learning. We train the models using a large-scale corpus of 500 billion tokens comprising 30 diverse languages, upsampling the less-resourced languages to render a more balanced language representation.

We evaluate XGLMs on multiple multilingual natural language understanding (NLU) tasks, machine translation and a subset of English tasks demonstrated in [Brown et al., 2020]. The models demonstrate strong cross-lingual capabilities, with competitive zero- and few-shot learning performance when using English prompts alongside non-English examples. Our largest model (XGLM 7.5B) achieves strong zero- and few-shot learning performance on language completion and inference tasks, such as XStoryCloze (65.4% 0-shot, 66.5% 4-shot) and XNLI (46.3% 0-shot, 47.3% 4-shot). Additionally, it established a new state-of-the-art on few-shot machine translation across numerous language pairs in the FLORES-101 benchmark Goyal et al. [2022], significantly outperforming the GPT-3 model of comparable size (6.7 billion parameters).

However, we found that multilingual pre-training leads to performance drop on English. On 8 En-

glish natural language understanding tasks, XGLM 7.5B underperforms GPT-3 6.7B by an average of 10.9% in zero-shot learning. GPT-3 6.7B also outperforms XGLM 7.5B in machine translation on several high-resource language pairs, including WMT-14 English-French, WMT-16 English-German, and WMT-19 English-Chinese. There are multiple reasons why XGLM 7.5B underperforms English centric models on the English tasks. First, only 32.6% of XGLM 7.5B's 500B-token training data is English while both English-centric models are trained on close to 300B English tokens. Second, the model capacity of XGLM 7.5B is shared by 30 languages, and the "curse of multilinguality" can degrade the performance across all languages [Conneau et al., 2020]. We hypothesize that further scaling up the model capacity and training data can potentially close this gap.

We conduct an in-depth analysis of different multilingual prompting approaches and examine cross-lingual transfer through template and demonstration examples respectively. Our findings reveal that non-English templates sometimes yield unexpected low zero- and few-shot learning accuracy even when crafted by native speakers. However, using English templates or adding demonstration examples proved to be effective remedies. Interestingly, we found that using demonstration examples from another language often fails to further improve zero-shot learning performance when a strong prompting language like English is used. This suggests room for improvement in both cross-lingual pre-training and in-context transfer approaches. Our analysis highlights the importance of carefully selecting prompting languages and demonstration examples to optimize cross-lingual transfer and performance. By understanding these factors, we can develop more effective strategies for improving the multilingual capabilities of LLMs and ensuring their applicability across diverse language contexts.

1.5 Summary

In summary, this thesis focuses on addressing the challenges faced in building LLMs that can cater to a wide range of everyday use cases. The three main challenges include misalignment with user requests, external and tail-range knowledge access issues, and the predominance of English language in widely used LLMs. To tackle these challenges, we present methodologies based on instruction meta-learning (IML), retrieval-augmented dual instruction tuning (RA-DIT), and cross-lingual generative language pre-training (XGLMs). These approaches aim to enhance LLM capabilities by scaling IML along task and language

diversities, integrating external knowledge sources, and improving cross-lingual performance for medium- and low-resource languages. Together, these strategies pave the way for real-world deployment of advanced LLMs, enabling them to better serve diverse user needs.