

A Survey on Knowledge Distillation of Large Language Models

Xiaohan Xu¹, Ming Li², Chongyang Tao³, Tao Shen⁴, Reynold Cheng¹, Jinyang Li¹,
Can Xu⁵, Dacheng Tao⁶, Tianyi Zhou²

¹The University of Hong Kong ²University of Maryland ³Microsoft

⁴University of Technology Sydney ⁵Peking University ⁶The University of Sydney

{shawxxxh, chongyangtao, hishentao}@gmail.com {minglii, tianyi}@umd.edu
ckcheng@cs.hku.hk j10725@connect.hku.hk

Abstract—In the era of Large Language Models (LLMs), Knowledge Distillation (KD) emerges as a pivotal methodology for transferring advanced capabilities from leading proprietary LLMs, such as GPT-4, to their open-source counterparts like LLaMA and Mistral. Additionally, as open-source LLMs flourish, KD plays a crucial role in both compressing these models, and facilitating their self-improvement by employing themselves as teachers. This paper presents a comprehensive survey of KD’s role within the realm of LLM, highlighting its critical function in imparting advanced knowledge to smaller models and its utility in model compression and self-improvement. Our survey is meticulously structured around three foundational pillars: *algorithm*, *skill*, and *verticalization* – providing a comprehensive examination of KD mechanisms, the enhancement of specific cognitive abilities, and their practical implications across diverse fields. Crucially, the survey navigates the interaction between data augmentation (DA) and KD, illustrating how DA emerges as a powerful paradigm within the KD framework to bolster LLMs’ performance. By leveraging DA to generate context-rich, skill-specific training data, KD transcends traditional boundaries, enabling open-source models to approximate the contextual adeptness, ethical alignment, and deep semantic insights characteristic of their proprietary counterparts. This work aims to provide an insightful guide for researchers and practitioners, offering a detailed overview of current methodologies in knowledge distillation and proposing future research directions. By bridging the gap between proprietary and open-source LLMs, this survey underscores the potential for more accessible, efficient, and powerful AI solutions. Most importantly, we firmly advocate for compliance with the legal terms that regulate the use of LLMs, ensuring ethical and lawful application of KD of LLMs. An associated Github repository is available at <https://github.com/Tebmer/Awesome-Knowledge-Distillation-of-LLMs>.

Index Terms—Large language models, knowledge distillation, data augmentation, skill distillation, supervised fine-tuning



1 INTRODUCTION

In the evolving landscape of artificial intelligence (AI), proprietary¹ Large Language Models (LLMs) such as GPT-3.5 (Ouyang et al., 2022), GPT-4 (OpenAI et al., 2023), Gemini (Team et al., 2023) and Claude² have emerged as groundbreaking technologies, reshaping our understanding of natural language processing (NLP). These models, characterized by their vast scale and complexity, have unlocked new realms of possibility, from generating human-like text to offering sophisticated problem-solving capabilities. The core significance of these LLMs lies in their emergent abilities (Wei et al., 2022a,b; Xu et al., 2024a), a phenomenon where the models display capabilities beyond their explicit training objectives, enabling them to tackle a diverse array of tasks with remarkable proficiency. These models excel in understanding and generation, driving applications from creative generation to complex problem-solving (OpenAI et al., 2023; Liang et al., 2022). The potential of these models

extends far beyond current applications, promising to revolutionize industries, augment human creativity, and redefine our interaction with technology.

Despite the remarkable capabilities of proprietary LLMs like GPT-4 and Gemini, they are not without their shortcomings, particularly when viewed in light of the advantages offered by open-source models. A significant drawback is their limited accessibility and higher cost (OpenAI et al., 2023). These proprietary models often come with substantial usage fees and restricted access, making them less attainable for individuals and smaller organizations. In terms of data privacy and security (Wu et al., 2023a), using these proprietary LLMs frequently entails sending sensitive data to external servers, which raises concerns about data privacy and security. This aspect is especially critical for users handling confidential information. Moreover, the general-purpose design of proprietary LLMs, while powerful, may not always align with the specific needs of niche applications. The constraints of accessibility, cost, and adaptability thus present significant challenges in leveraging the full potential of proprietary LLMs.

In contrast to proprietary LLMs, open-source models like LLaMA (Touvron et al., 2023) and Mistral (Jiang et al., 2023a) bring several notable advantages. One of the primary

1. For simplicity, we use ‘proprietary’ to represent both versatile yet close-source LLMs like GPT-4 and open-source yet huge LLMs like LLaMA-2-70B, which encapsulate rich knowledge with a large number of parameters.

2. <https://www.anthropic.com/claude-in-slack>

benefits of open-source models is their accessibility and adaptability. Without the constraints of licensing fees or restrictive usage policies, these models are more readily available to a broader range of users, from individual researchers to smaller organizations. This openness fosters a more collaborative and inclusive AI research environment, encouraging innovation and diverse applications. Additionally, the customizable nature of open-source LLMs allows for more tailored solutions, addressing specific needs that generic, large-scale models may not meet.

However, the open-source LLMs also have their own set of drawbacks, primarily stemming from their relatively limited scale and resources compared to their proprietary counterparts. One of the most significant limitations is the smaller model scale, which often results in lower performance on real-world tasks with a bunch of instructions (Zheng et al., 2023a). These models, with fewer parameters, may struggle to capture the depth and breadth of knowledge embodied in larger models like GPT-4. Additionally, the pre-training investment in these open-source models is typically less substantial. This reduced investment can lead to a narrower range of pre-training data, potentially limiting the models’ understanding and handling of diverse or specialized topics (Liang et al., 2022; Sun et al., 2024a). Moreover, open-source models often undergo fewer fine-tuning steps due to resource constraints. Fine-tuning is crucial for optimizing a model’s performance for specific tasks or industries, and the lack thereof can hinder the model’s effectiveness in specialized applications. This limitation becomes particularly evident when these models are compared to the highly fine-tuned proprietary LLMs, which are often tailored to excel in a wide array of complex scenarios (OpenAI et al., 2023).

Primarily, recognizing the disparities between proprietary and open-source LLMs, KD techniques have surged as a means to bridge the performance gap between these models (Gou et al., 2021; Gupta and Agrawal, 2022). Knowledge distillation, in this context, involves leveraging the more advanced capabilities of leading proprietary models like GPT-4 or Gemini as a guiding framework to enhance the competencies of open-source LLMs. This process is akin to transferring the ‘knowledge’ of a highly skilled teacher to a student, wherein the student (e.g., open-source LLM) learns to mimic the performance characteristics of the teacher (e.g., proprietary LLM). Compared to traditional knowledge distillation algorithms (Gou et al., 2021), data augmentation (DA) (Feng et al., 2021) has emerged as a prevalent paradigm to achieve knowledge distillation of LLMs, where a small seed of knowledge is used to prompt the LLM to generate more data with respect to a specific skill or domain (Taori et al., 2023). Secondly, KD still retains its fundamental role in compressing LLMs, making them more efficient without significant loss in performance. (Gu et al., 2024; Agarwal et al., 2024). More recently, the strategy of employing open-source LLMs as teachers for their own self-improvement has emerged as a promising approach, enhancing their capabilities significantly (Yuan et al., 2024a; Chen et al., 2024a). Figure 1 provides an illustration of these three key roles played by KD in the context of LLMs.

A key aspect of the knowledge distillation is the enhancement of skills such as advanced context following

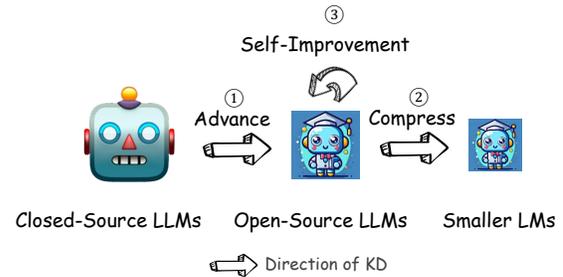


Fig. 1: KD plays three key roles in LLMs: 1) Primarily enhancing capabilities, 2) offering traditional compression for efficiency, and 3) an emerging trend of self-improvement via self-generated knowledge.

(e.g., in-context learning (Huang et al., 2022a) and instruction following (Taori et al., 2023)), improved alignment with user intents (e.g., human values/principles (Cui et al., 2023a), and thinking patterns like chain-of-thought (CoT) (Mukherjee et al., 2023)), and NLP task specialization (e.g., semantic understanding (Ding et al., 2023a), and code generation (Chaudhary, 2023)). These skills are crucial for the wide array of applications that LLMs are expected to perform, ranging from casual conversations to complex problem-solving in specialized domains. For instance, in vertical domains like healthcare (Wang et al., 2023a), law (LAW, 2023), or science (Zhang et al., 2024), where accuracy and context-specific knowledge are paramount, knowledge distillation allows open-source models to significantly improve their performance by learning from the proprietary models that have been extensively trained and fine-tuned in these areas.

The benefits of knowledge distillation in the era of LLMs are multifaceted and transformative (Gu et al., 2024). Through a suite of distillation techniques, the gap between proprietary and open-source models is significantly narrowed (Chiang et al., 2023; Xu et al., 2023a) and even filled (Zhao et al., 2023a). This process not only streamlines computational requirements but also enhances the environmental sustainability of AI operations, as open-source models become more proficient with lesser computational overhead. Furthermore, knowledge distillation fosters a more accessible and equitable AI landscape, where smaller entities and individual researchers gain access to state-of-the-art capabilities, encouraging wider participation and diversity in AI advancements. This democratization of technology leads to more robust, versatile, and accessible AI solutions, catalyzing innovation and growth across various industries and research domains.

The escalating need for a comprehensive survey on the knowledge distillation of LLMs stems from the rapidly evolving landscape of AI (OpenAI et al., 2023; Team et al., 2023) and the increasing complexity of these models. As AI continues to penetrate various sectors, the ability to efficiently and effectively distill knowledge from proprietary LLMs to open-source ones becomes not just a technical aspiration but a practical necessity. This need is driven by the growing demand for more accessible, cost-effective, and adaptable AI solutions that can cater to a diverse range

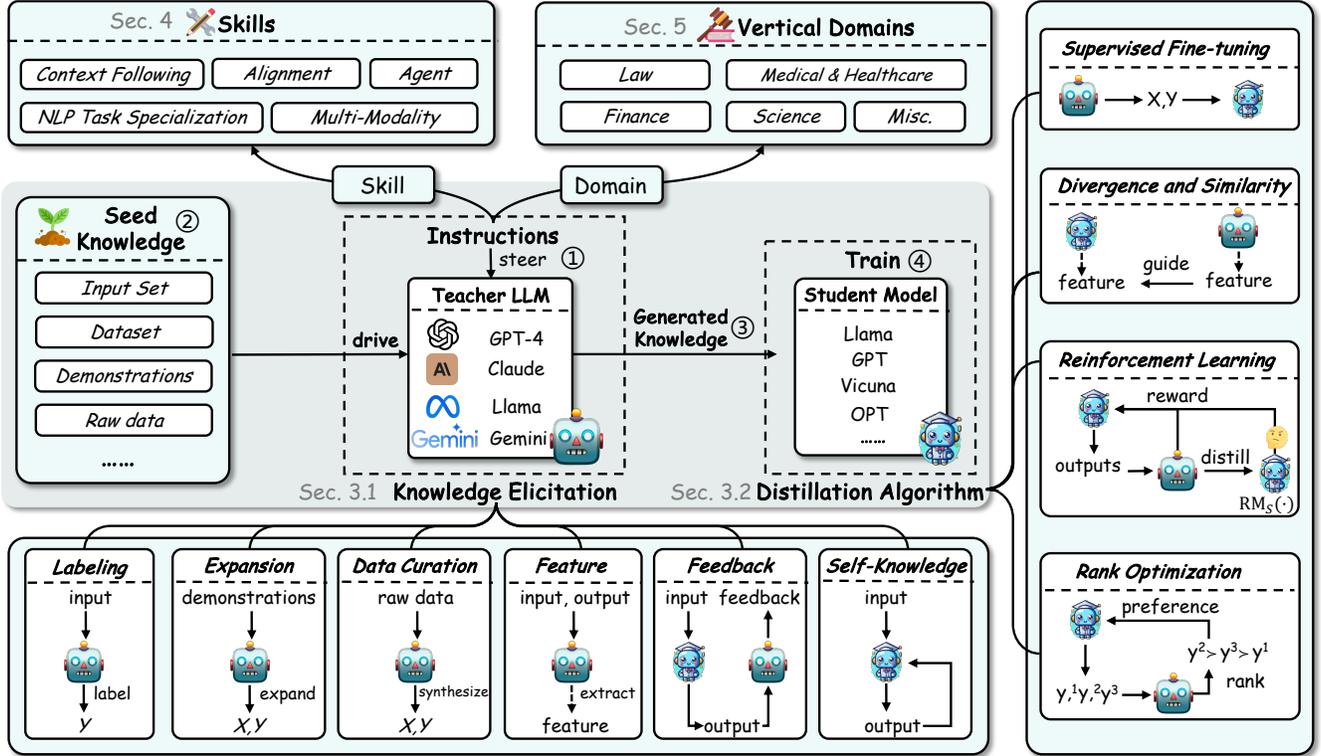


Fig. 2: An overview of this survey on knowledge distillation of large language models. Note that ‘Section’ is abbreviated as ‘Sec.’ in this figure. $RM_S(\cdot)$ denotes the student reward model. ①②③④ denote the steps in KD of LLMs.

of applications and users. A survey in this field is vital for synthesizing the current methodologies, challenges, and breakthroughs in knowledge distillation. It may serve as a beacon for researchers and practitioners alike, guiding them to distill complex AI capabilities into more manageable and accessible forms. Moreover, such a survey can illuminate the path forward, identifying gaps in current techniques and proposing directions for future research.

Survey Organization. The remainder of this survey is organized into several comprehensive sections, each designed to offer a deep dive into the multifaceted aspects of knowledge distillation within the realm of LLMs. Following this introduction, §2 provides a foundational overview of knowledge distillation, comparing traditional techniques with those emerging in the era of LLMs and highlighting the role of data augmentation (DA) in this context. §3 delves into the approaches to elicit knowledge from teacher LLMs and core distillation algorithms, examining methods from supervised fine-tuning to more complex strategies involving divergence and similarity, reinforcement learning, and ranking optimization. Then, §4 focuses on skill distillation, exploring how student models can be enhanced to improve context understanding, alignment with user intentions, and performance across a variety of NLP tasks. This includes discussions on natural language understanding (NLU), generation (NLG), information retrieval, recommendation systems, and the evaluation of text generation. In §5, we venture into domain-specific vertical distillation, showcasing how knowledge distillation techniques are applied within specialized fields such as law, healthcare, finance, and science,

illustrating the practical implications and transformative impact of these approaches. The survey suggests open problems in §6, identifying current challenges and gaps in knowledge distillation research that offer opportunities for future work. Finally, the conclusion and discussion in §7 synthesize the insights gained, reflecting on the implications for the broader AI and NLP research community and proposing directions for future research. Figure 2 shows an overview of this survey.

2 OVERVIEW

2.1 Comparing Traditional Recipe

The concept of knowledge distillation in the field of AI and deep learning (DL) refers to the process of transferring knowledge from a large, complex model (teacher) to a smaller, more efficient model (student) (Gou et al., 2021). This technique is pivotal in mitigating the challenges posed by the computational demands and resource constraints of deploying large-scale models in practical applications.

Historically, knowledge distillation techniques, prior to the era of LLMs, primarily concentrated on transferring knowledge from complex, often cumbersome neural networks to more compact and efficient architectures (Sanh et al., 2019; Kim and Rush, 2016). This process was largely driven by the need to deploy machine learning models in resource-constrained environments, such as mobile devices or edge computing platforms, where the computational power and memory are limited. The focus was predominantly on ad-hoc neural architecture selection and training objectives tailored for single tasks. These earlier methods

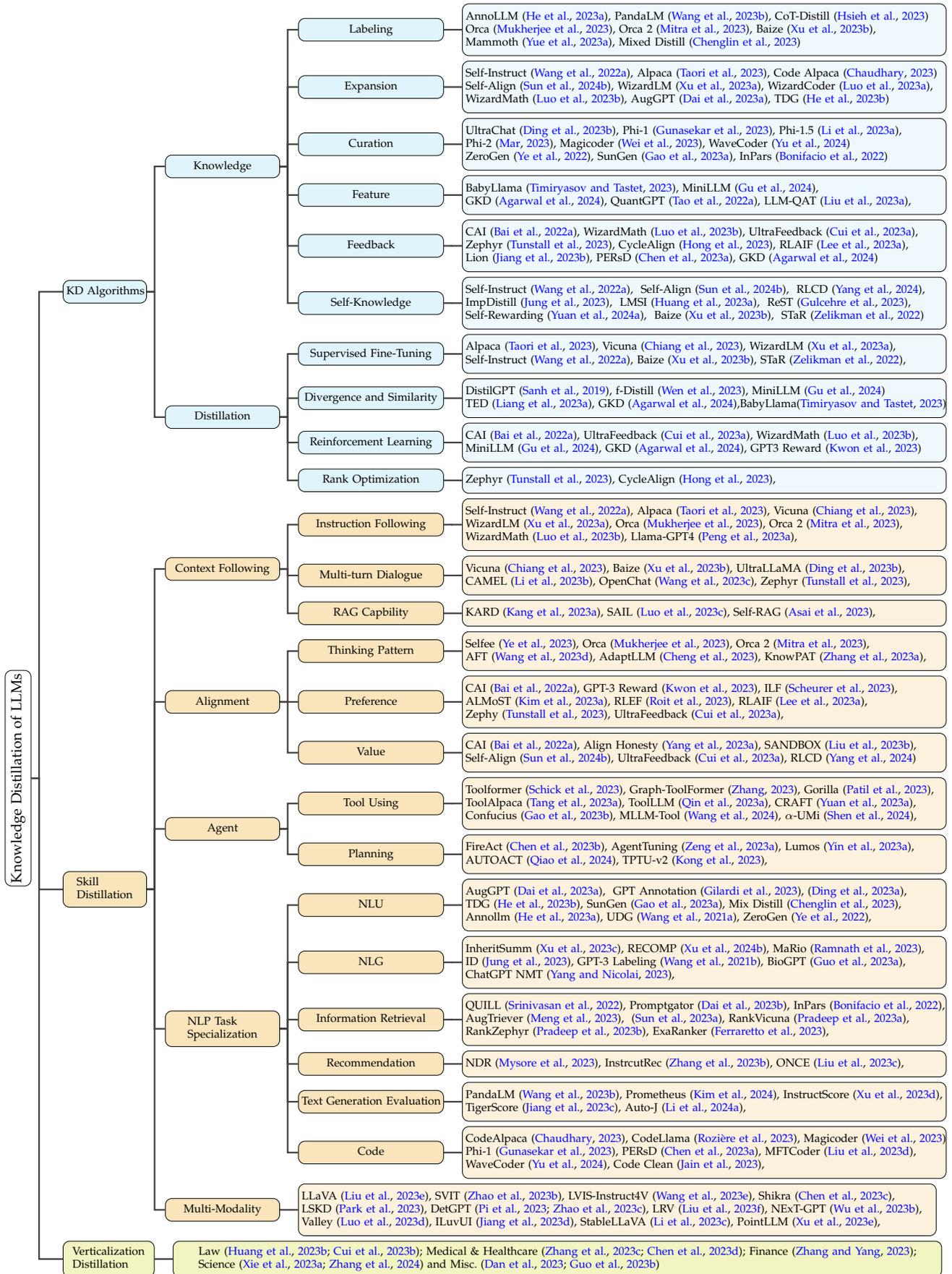


Fig. 3: Taxonomy of Knowledge Distillation of Large Language Models. The detailed taxonomy of Verticalization Distillation is shown in Figure 7.

involved training a smaller student network to mimic the output of a larger teacher network, often through techniques like soft target training, where the student learns from the softened softmax output of the teacher. Please refer to the survey (Gou et al., 2021) for more details on general knowledge distillation techniques in AI and DL.

In contrast, the advent of LLMs has revolutionized the knowledge distillation landscape. The current era of knowledge distillation in LLMs shifts the focus from mere architecture compression to knowledge elicitation and transfer (Taori et al., 2023; Chaudhary, 2023; Tunstall et al., 2023). This paradigm change is largely due to the expansive and deep-seated knowledge that LLMs like GPT-4 and Gemini possess. And the inaccessible parameters of LLMs make it hard to compress them by using pruning (Han et al., 2016) or quantization (Liu et al., 2023a) techniques. Unlike the earlier era, where the goal was to replicate the output behavior of the teacher model or reduce the model size, the current focus in LLM-based knowledge distillation is to elicit the specific knowledge these models have.

The key to this modern approach lies in heuristic and carefully designed prompts, which are used to elicit specific knowledge (Ding et al., 2023b) or capabilities (Chaudhary, 2023) from the LLMs. These prompts are crafted to tap into the LLM’s understanding and capabilities in various domains, ranging from natural language understanding (He et al., 2023a) to more complex cognitive tasks like reasoning (Hsieh et al., 2023) and problem-solving (Qiao et al., 2024). The use of prompts as a means of knowledge elicitation offers a more flexible and dynamic approach to distillation. It allows for a more targeted extraction of knowledge, focusing on specific skills or domains of interest. This method is particularly effective in harnessing the emergent abilities of LLMs, where the models exhibit capabilities beyond their explicit training objectives.

Furthermore, this era of knowledge distillation also emphasizes the transfer of more abstract qualities such as reasoning patterns (Mitra et al., 2023), preference alignment (Cui et al., 2023a), and value alignment (Sun et al., 2024b). This is in stark contrast to the earlier focus on output replication (Taori et al., 2023), indicating a shift towards a more holistic and comprehensive transfer of cognitive capabilities. The current techniques involve not just the replication of outputs, but also the emulation of the thought processes (Mitra et al., 2023) and decision-making (Asai et al., 2023) patterns of the teacher model. This involves complex strategies like chain-of-thought prompting, where the student model is trained to learn the reasoning process of the teacher, thereby enhancing its problem-solving and decision-making capabilities.

2.2 Relation to Data Augmentation (DA)

In the era of LLMs, Data Augmentation (DA) (Wang et al., 2022a; Ye et al., 2022) emerges as a critical paradigm integral to the process of knowledge distillation. Unlike traditional DA techniques such as paraphrasing (Gangal et al., 2022) or back-translation (Longpre et al., 2019), which primarily aim at expanding the training dataset in a somewhat mechanical manner, DA within the context of LLMs focuses on the generation of novel, context-rich training data tailored to specific domains and skills.

The relationship between DA and KD in LLMs is both symbiotic and foundational. By leveraging a set of seed knowledge, KD employs DA to prompt LLMs to produce explicit data that encapsulates specific skills or domain expertise (Chaudhary, 2023; West et al., 2022). This method stands out as a potent mechanism for bridging the knowledge and capability gap between proprietary and open-source models. Through DA, LLMs are prompted to create targeted, high-quality datasets that are not merely larger in volume but are also rich in diversity and specificity. This approach enables the distillation process to be more effective, ensuring that the distilled models not only replicate the teacher model’s output behavior but also embody its deep-seated understanding and cognitive strategies.

DA acts as a force multiplier, enabling the distilled models to acquire and refine capabilities that would otherwise require exponentially larger datasets and computational resources. It facilitates a more effective transfer of knowledge, focusing on the qualitative aspects of learning rather than quantitative expansion. This strategic use of DA within KD processes underscores a pivotal shift towards a more efficient, sustainable, and accessible approach to harnessing the power of LLMs. It empowers open-source models with the ability to approximate the contextual adeptness, ethical alignment, and deep semantic insights characteristic of their proprietary counterparts, thereby democratizing access to advanced AI capabilities and fostering innovation across a broader spectrum of applications and users.

2.3 Survey Scope

Building on the discussions introduced earlier, this survey aims to comprehensively explore the landscape of knowledge distillation within the context of LLMs, following a meticulously structured taxonomy as in Figure 3. The survey’s scope is delineated through three primary facets: KD Algorithms, Skill Distillation, and Verticalization Distillation. Each facet encapsulates a range of subtopics and methodologies. It’s important to note that KD algorithms provide the technical foundations for skill distillation and verticalization distillation.

KD Algorithms. This segment focuses on the technical foundations and methodologies of knowledge distillation. It includes an in-depth exploration of the processes involved in constructing knowledge from teacher models (e.g., proprietary LLMs) and integrating this knowledge into student models (e.g., open-source LLMs). Under the umbrella of ‘*knowledge*’, we delve into strategies such as labeling (Hsieh et al., 2023), expansion (Taori et al., 2023), curation (Gunasekar et al., 2023), feature understanding (Agarwal et al., 2024), feedback mechanisms (Tunstall et al., 2023), and self-knowledge generation (Wang et al., 2022a). This exploration seeks to uncover the various ways in which knowledge can be identified, expanded, and curated for effective distillation. The ‘*distillation*’ subsection examines learning approaches like supervised fine-tuning (SFT) (Wang et al., 2022a), divergence minimization (Agarwal et al., 2024), reinforcement learning techniques (Cui et al., 2023a), and rank optimization strategies (Tunstall et al., 2023). Together, these techniques demonstrate how KD enables open-source models to obtain knowledge from proprietary ones.

Skill Distillation. This facet examines the specific competencies and capabilities enhanced through KD. It encompasses detailed discussions on context following (Taori et al., 2023; Luo et al., 2023c), with subtopics like instruction following and retrieval-augmented generation (RAG) Capability. In the realm of alignment (Mitra et al., 2023; Tunstall et al., 2023), the survey investigates thinking patterns, persona/preference modeling, and value alignment. The ‘agent’ category delves into skills such as Tool Using and Planning. NLP task specialization (Dai et al., 2023a; Jung et al., 2023; Chaudhary, 2023) is scrutinized through lenses like natural language understanding (NLU), natural language generation (NLG), information retrieval, recommendation systems, text generation evaluation, and code generation. Finally, the survey addresses multi-modality (Liu et al., 2023e; Zhao et al., 2023b), exploring how KD enhances LLMs’ ability to integrate multiple forms of input.

Verticalization Distillation. This section assesses the application of KD across diverse vertical domains, offering insights into how distilled LLMs can be tailored for specialized fields such as Law (LAW, 2023), Medical & Healthcare (Wang et al., 2023a), Finance (Zhang and Yang, 2023), Science (Zhang et al., 2024), among others. This exploration not only showcases the practical implications of KD techniques but also highlights their transformative impact on domain-specific AI solutions.

Through these facets, this survey provides a comprehensive analysis of KD in LLMs, guiding researchers and practitioners through methodologies, challenges, and opportunities in this rapidly evolving domain.

Declaration. This survey represents our earnest effort to provide a comprehensive and insightful overview of knowledge distillation techniques applied to LLMs, focusing on algorithms, skill enhancement, and domain-specific applications. Given the vast and rapidly evolving nature of this field, especially with the prevalent practice of eliciting knowledge from training data across academia, we acknowledge that this manuscript may not encompass every pertinent study or development. Nonetheless, it endeavors to introduce the foundational paradigms of knowledge distillation, highlighting key methodologies and their impacts across a range of applications.

2.4 Distillation Pipeline in LLM Era

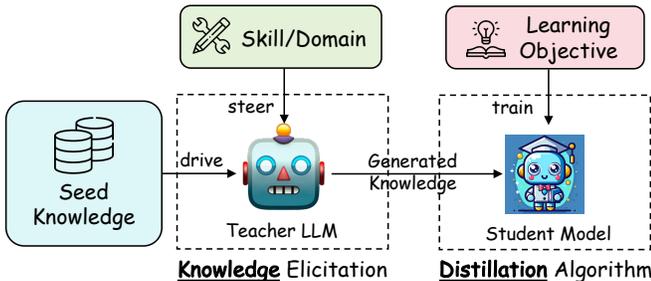


Fig. 4: An illustration of a general pipeline to distill knowledge from a large language model to a student model.

The general distillation pipeline of LLMs is a structured and methodical process aimed at transferring knowledge

from a sophisticated teacher model to a less complex student model. This pipeline is integral for leveraging the advanced capabilities of models like GPT-4 or Gemini in more accessible and efficient open-source counterparts. The outline of this pipeline can be broadly categorized into four distinct stages, each playing a crucial role in the successful distillation of knowledge. An illustration is shown in Figure 4. The detailed pipeline could also be seen in Figure 2.

I. Target Skill or Domain Steering Teacher LLM. The first stage involves directing the teacher LLM towards a specific target skill or domain. This is achieved through carefully crafted instructions or templates that guide the LLM’s focus. These instructions are designed to elicit responses that demonstrate the LLM’s proficiency in a particular area, be it a specialized domain like healthcare or law, or a skill such as reasoning or language understanding.

II. Seed Knowledge as Input. Once the target area is defined, the next step is to feed the teacher LLM with seed knowledge. This seed knowledge typically comprises a small dataset or specific data clues relevant to the elicited skill or domain knowledge from the teacher LLM. It acts as a catalyst, prompting the teacher LLM to generate more elaborate and detailed outputs based on this initial information. The seed knowledge is crucial as it provides a foundation upon which the teacher model can build and expand, thereby creating more comprehensive and in-depth knowledge examples.

III. Generation of Distillation Knowledge. In response to the seed knowledge and steering instructions, the teacher LLM generates knowledge examples. These examples are predominantly in the form of question-and-answer (QA) dialogues or narrative explanations, aligning with the natural language processing/understanding capabilities of the LLM. In certain specialized cases, the outputs may also include logits or hidden features, although this is less common due to the complexity and specific requirements of such data forms. The generated knowledge examples constitute the core of the distillation knowledge, encapsulating the advanced understanding and skills of the teacher LLM.

IV. Training the Student Model with a Specific Learning Objective. The final stage involves the utilization of the generated knowledge examples to train the student model. This training is guided by a loss function that aligns with the learning objectives. The loss function quantifies the student model’s performance in replicating or adapting the knowledge from the teacher model. By minimizing this loss, the student model learns to emulate the target skills or domain knowledge of the teacher, thereby acquiring similar capabilities. The process involves iteratively adjusting the student model’s parameters to reduce the discrepancy between its outputs and those of the teacher model, ensuring the effective transfer of knowledge.

In essential, the above four stages can be abstracted as two formulations. The first formulation represents the process of eliciting knowledge:

$$\mathcal{D}_I^{(kd)} = \{\text{Parse}(o, s) | o \sim p_T(o | I \oplus s), \forall s \sim \mathcal{S}\}, \quad (1)$$

where \oplus denotes fusing two pieces of text, I denotes an instruction or a template for a task, skill, or domain to steer the LLM and elicit knowledge, $s \sim \mathcal{S}$ denotes an

example of the seed knowledge, upon which the LLM can explore to generate novel knowledge, $\text{Parse}(o, s)$ stands for to parse the distillation example (e.g., (x, y)) from the teacher LLM’s output o (plus the input s in some cases), and p_T represents the teacher LLM with parameters θ_T . Given the datasets $\mathcal{D}_I^{(\text{kd})}$ built for distillation, we then define a learning objective as

$$\mathcal{L} = \sum_I \mathcal{L}_I(\mathcal{D}_I^{(\text{kd})}; \theta_S), \quad (2)$$

where \sum_I denotes there could be multiple tasks or skills being distilled into one student model, $\mathcal{L}_I(\cdot; \cdot)$ stands for a specific learning objective, and θ_S parameterizes the student model.

Following our exploration of the distillation pipeline and the foundational concepts underlying knowledge distillation in the LLM era, we now turn our focus to the specific algorithms that have gained prominence in this era.

3 KNOWLEDGE DISTILLATION ALGORITHMS

This section navigates through the process of knowledge distillation. According to Section 2.4, it is categorized into two principal steps: ‘Knowledge,’ focusing on eliciting knowledge from teacher LLMs (Eq.1), and ‘Distillation,’ centered on injecting this knowledge into student models (Eq.2). We will elaborate on these two processes in the subsequent sections.

3.1 Knowledge

This section focuses on the approaches to elicit knowledge from teacher LLMs. According to the manners to acquire knowledge, we divided them into *Labeling*, *Expansion*, *Data Curation*, *Feature*, *Feedback*, and *Self-Knowledge*. Figure 5 shows an illustration of these knowledge elicitation methods.

3.1.1 Labeling

Labeling knowledge refers to using a teacher LLM to label the output y for a given input x as the seed knowledge, according to the instruction I or demonstrations c , where $c = (x_1, y_1), \dots, (x_n, y_n)$. This method of eliciting knowledge from teacher LLMs is straightforward yet effective and has been widely applied across various tasks and applications. It requires only the collection of an input dataset and feeding it into LLMs to obtain the desired generations. Moreover, the generation of y is controllable through the predefined I and c . This process can be formulated as follows:

$$\mathcal{D}^{(\text{lab})} = \{x, y | x \sim \mathcal{X}, y \sim p_T(y | I \oplus c \oplus x)\}. \quad (3)$$

Input x could be sourced from existing NLP task datasets, which serve as typical reservoirs for distillation efforts. Numerous works have sought to harness the capabilities of powerful LLMs as teachers for annotating dataset samples across a range of tasks. For instance, efforts in natural language understanding involve using LLMs to categorize text (Gilardi et al., 2023; Ding et al., 2023a; He et al., 2023a), while in natural language generation, LLMs assist in generating sequences for outputs (Hsieh et al., 2023; Jung et al., 2023; Wang et al., 2021b). Text generation evaluation

tasks leverage LLMs to label evaluated results (Li et al., 2024b; Wang et al., 2023b), and reasoning tasks utilize LLMs for labeling Chains of Thought (CoT) explanations (Hsieh et al., 2023; Li et al., 2022; Ho et al., 2023; Magister et al., 2023; Fu et al., 2023; Ramnath et al., 2023; Li et al., 2023d; Liu et al., 2023g), among others. Rather than concentrating on specific tasks, many current works focus on labeling outputs based on instructions, thereby teaching student models to solve tasks in a more flexible way by following instructions. Collections of various NLP tasks, complemented by instructional templates, serve as valuable input sources for x . For instance, FLAN-v2 collections (Longpre et al., 2023) offers extensive publicly available sets of tasks with instructions, which are labeled with responses generated by teacher LLMs in Orca (Mukherjee et al., 2023; Mitra et al., 2023). The instructions from these NLP tasks are built from predefined templates, which lack diversity and may have gaps between human’s natural query. The real conversations between humans and chat models provide large-scale data with real queries and generations labeled by powerful LLMs, like ShareGPT. Additionally, Xu et al. (2023b) and Anand et al. (2023) label the real questions sampled from forums like Quora and Stack Overflow.

Moreover, the process of labeling could be guided by instructions I or demonstrations c . A commonly used instruction type for guiding labeling is chain-of-thought (CoT) prompt (Hsieh et al., 2023; Fu et al., 2023; Magister et al., 2023). Mukherjee et al. (2023) add multiple system messages (e.g. “You must generate a detailed and long answer.” or “explain like I’m five, think step-by-step”) to elicit rich signals. Yue et al. (2023a) and Chenglin et al. (2023) label a hybrid of knowledge of chain-of-thought (CoT) and program-of-thought (PoT) rationales. Xu et al. (2023b) propose a self-chat technique that two teacher LLMs simulate the real conversational to generate multi-turn dialogues for a question from Quora and Stack Overflow.

3.1.2 Expansion

While the labeling approach is simple and effective, it faces certain limitations. Primarily, it is constrained by the scale and variety of the input data. In real-world applications, especially those involving user conversations, there are also concerns regarding the privacy of the data involved. To address these limitations, various expansion methods have been proposed (Wang et al., 2022a; Taori et al., 2023; Chaudhary, 2023; Si et al., 2023; Ji et al., 2023a; Luo et al., 2023b,a; Wu et al., 2023c; Sun et al., 2024b; Xu et al., 2023a; Guo et al., 2023c; Rozière et al., 2023; West et al., 2022). These methods take the demonstrations as seed knowledge and aim to expand a large scale and various data by in-context learning.

A key characteristic of these expansion methods is the utilization of the in-context learning ability of LLMs to generate data similar to the provided demonstrations c . Unlike in the labeling approach, where the input x is sampled from the existing dataset, in the expansion approach, both x and y are generated by teacher LLMs. This process can be formulated as follows:

$$\mathcal{D}^{(\text{exp})} = \{(x, y) | x \sim p_T(x | I \oplus c), y \sim p_T(y | I \oplus x)\}. \quad (4)$$

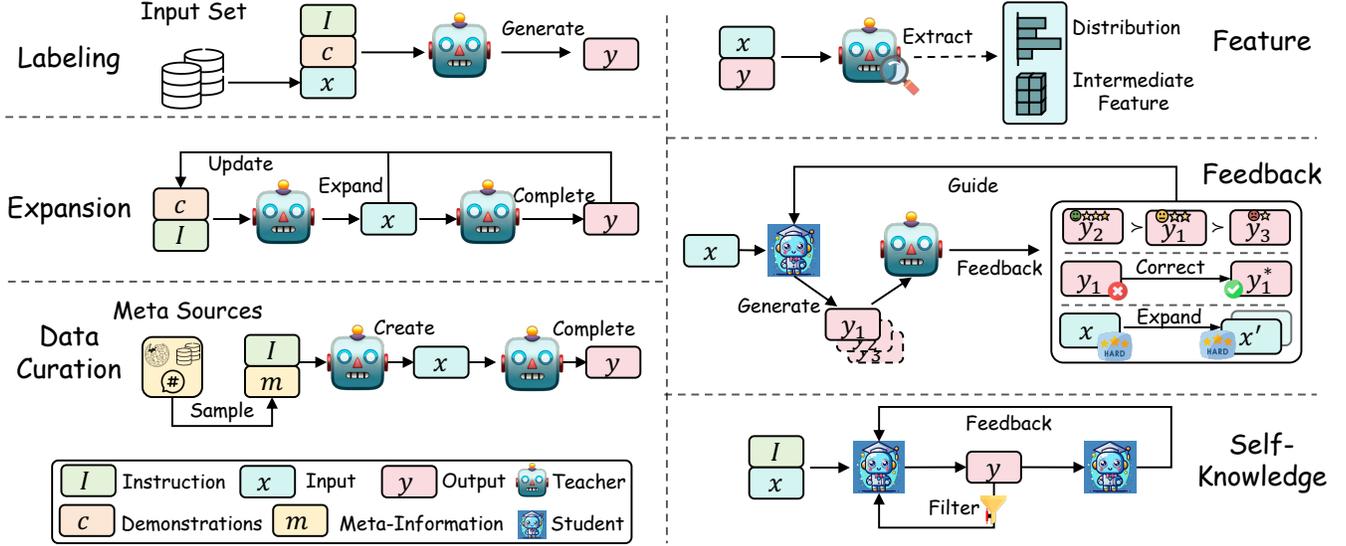


Fig. 5: An illustration of different knowledge elicitation methods from teacher LLMs. *Labeling*: The teacher generates the output from the input; *Expansion*: The teacher generates samples similar to the given demonstrations through in-context learning; *Data Curation*: The teacher synthesizes data according to meta-information, such as a topic or an entity; *Feature*: Feed the data into the teacher and extract its internal knowledge, such as logits and features; *Feedback*: The teacher provides feedback on the student’s generations, such as preferences, corrections, expansions of challenging samples, etc; *Self-Knowledge*: The student first generates outputs, which is then filtered for high quality or evaluated by the student itself.

In this formulation, x and y represent the new input-output pairs generated by the teacher LLM. The input x is generated based on a set of input-output demonstrations c . The output y is then generated in response to the new input x under the guidance of an instruction I . Note that the demonstrations could be predefined or dynamically updated by adding the newly generated samples.

Expansion techniques have been widely utilized to extract extensive instruction-following knowledge from teacher LLMs. Wang et al. (2022a) first introduces an iterative bootstrapping method, Self-Instruct, to utilize LLMs to generate a wide array of instructions based on several demonstrations sampled from 175 manually-written instructions. The newly generated instructions are then added back to the initial pool, benefiting subsequent expansion iterations. Subsequently, Taori et al. (2023) applies this expansion method to a more powerful teacher LLM, text-davinci-003, to distill 52K high-quality data. To improve the diversity and coverage during expansion, Wu et al. (2023c) and (Sun et al., 2024b) prompt the teacher LLM to generate instructions corresponding to some specific topics. Xu et al. (2023a) propose an Evol-Instruct method to expand the instructions from two dimensions: difficulty (e.g. rewriting the question to be more complex) and diversity (e.g. generating more long-tailed instructions). This Evol-Instruct method is domain-agnostic and has been used to expand the distillation of coding (Luo et al., 2023a) and math (Luo et al., 2023b). Additionally, expansion methods can significantly augment NLP task datasets with similar samples, thereby enhancing task performance. For instance, AugGPT (Dai et al., 2023a) leverages a teacher LLM to rephrase each sentence in the training samples into multiple conceptually similar, but semantically varied, samples to improve classification performance. Similarly, TDG (He

et al., 2023b) proposes the Targeted Data Generation (TDG) framework, which automatically identifies challenging subgroups within data and generates new samples for these subgroups using LLMs through in-context learning.

In summary, the expansion method leverages the in-context learning strengths of LLMs to produce more varied and extensive datasets with both inputs and outputs. However, the quality and diversity of the generated data are heavily reliant on the teacher LLMs and the initial seed demonstrations. This dependence can lead to a dataset with inherent bias from LLMs (Yu et al., 2023a; Wei et al., 2023) and a homogeneity issue where the generations may be prone to similarity ultimately, limiting the diversity this method seeks to achieve (Ding et al., 2023b). Moreover, the expansion process may inadvertently amplify any biases present in the seed data.

3.1.3 Data Curation

The pursuit of high-quality and scalable data generation in knowledge distillation from LLMs has led to the emergence of the Data Curation approach. This method arises in response to the limitations observed in both the Labeling and Expansion approaches. These methods often yield data of variable quality and face constraints in quantity. In Labeling, the seed knowledge is sourced from task datasets, leading to potential noise and dirty data. Meanwhile, in Expansion, the input x is derived from seed demonstrations, which can result in homogeneous data when generated in large quantities. To overcome these challenges, the Data Curation method curates high-quality or large-scale data by extensive meta-information as seed knowledge (Ding et al., 2023b; Gunasekar et al., 2023; Li et al., 2023a; Mar, 2023; Liu et al., 2023d; Wei et al., 2023; Yu et al., 2024; Ye et al., 2022; Gao et al., 2023a; Yang and Nicolai, 2023).

A distinct feature of Data Curation is its approach to synthesize data from scratch. Numerous diverse meta-information, such as topics or knowledge points, could be incorporated into this process to generate controllable x and y . Thus, this process can be meticulously controlled to yield datasets that are not only large in scale but also of high quality. The formulation for Data Curation can be represented as:

$$\mathcal{D}^{(\text{cur})} = \{(x, y) | x \sim p_T(x | I \oplus m), y \sim p_T(y | I \oplus x)\}. \quad (5)$$

In this formulation, m represents the diverse meta-information used to guide the synthesis of x , and I is the instruction guiding teacher LLMs to generate x or y .

Different studies primarily vary in their source and method of leveraging meta-information. UltraChat (Ding et al., 2023b) effectively demonstrates the process of curating both high-quality and diverse data by distilled knowledge. They collect extensive meta-information across three domains: *Questions about the World*, *Creation and Generation*, and *Assistance on Existing Materials*. For example, under *Questions about the World*, they explore 30 meta-topics like "Technology" and "Food and Drink." the teacher LLMs then use this meta-information to distill a broad array of instructions and conversations, achieving a substantial scale of 1.5 million instances. UltraChat stands out with its lexical and topical diversity. The UltraLLaMA model, finetuned on this data, consistently surpasses other open-source models. Another notable series, **phi** (Gunasekar et al., 2023; Li et al., 2023a; Mar, 2023), focuses on distilling smaller, high-quality datasets akin to "textbooks." **Phi-1** (Gunasekar et al., 2023) experiments with synthesizing "textbook quality" data in the coding domain. Their approach involves distilling clear, self-contained, instructive, and balanced content from LLMs, guided by random topics or function names to enhance diversity. The distilled data is a synthesis of 1 billion tokens of Python textbooks, complete with natural language explanations and code snippets, as well as 180 million tokens of Python exercises with solutions. Remarkably, the **phi-1** model, despite its smaller size, outperforms nearly all open-source models on coding benchmarks like HumanEval and MBPP while being 10 times smaller in model size and 100 times smaller in dataset size. MFTCoder (Liu et al., 2023d) utilizes hundreds of Python knowledge points as meta-information to create a CodeExercise Dataset. In contrast, Magicoder (Wei et al., 2023) and WaveCoder (Yu et al., 2024) get raw code collections from open-source code datasets, using this as meta-information for generating instructional data. In the context of NLU tasks, certain studies (Ye et al., 2022; Gao et al., 2023a; Wang et al., 2021a) explore the use of labels as meta-information to synthesize corresponding samples for data augmentation. Similarly, in information retrieval tasks, there are efforts to utilize documents as meta-information for generating potential queries, thereby constructing large-scale retrieval pairs (Bonifacio et al., 2022; Meng et al., 2023).

In conclusion, Data Curation through teacher LLMs has emerged as a promising technique for synthesizing datasets that are not only high-quality and diverse but also large in scale. The success of models like **phi-1** in specialized domains underscores the efficacy of this method. The ability

to create synthetic datasets will become a crucial technical skill and a key area of focus in AI (Li et al., 2023a).

3.1.4 Feature

The previously discussed knowledge elicitation methods are typically applied to powerful black-box models, which are expensive and somewhat unreproducible due to calling API. In contrast, white-box distillation offers a more transparent and accessible approach for researchers. It involves leveraging the output distributions, intermediate features, or activations from teacher LLMs, which we collectively refer to as *Feature* knowledge. White-box KD approaches have predominantly been studied for smaller encoder-based LMs, typically those with fewer than 1 billion parameters (cf. Gou et al. (2021) for detail). However, recent research has begun to explore white-box distillation in the context of generative LLMs (Timiryasov and Tastet, 2023; Liang et al., 2023a; Gu et al., 2024; Agarwal et al., 2024; Liu et al., 2023a; Wen et al., 2023; Wan et al., 2024a; Zhao and Zhu, 2023; Qin et al., 2023b; Boizard et al., 2024; Zhong et al., 2024).

The typical method for acquiring this feature knowledge involves teacher LLMs annotating the output sequence y with its internal representations. These annotations are then distilled into the student model using methods such as Kullback-Leibler Divergence (KLD). The process of eliciting feature knowledge can be formulated as follows:

$$\mathcal{D}^{(\text{feat})} = \{(x, y, \phi_{\text{feat}}(x, y; \theta_T)) | x \sim \mathcal{X}, y \sim \mathcal{Y}\}. \quad (6)$$

In this formulation, \mathcal{Y} is the output set, which can be generated by teacher LLMs, the student model, or directly sourced from the dataset. $\phi_{\text{feat}}(\cdot; \theta_T)$ represents the operation of extracting feature knowledge (such as output distribution) from the teacher LLM.

The most straightforward method to elicit feature knowledge of teacher is to label a fixed dataset of sequences with token-level probability distributions (Sanh et al., 2019; Wen et al., 2023). To leverage the rich semantic and syntactic knowledge in intermediate layers of the teacher model, TED (Liang et al., 2023a) designs task-aware layer-wise distillation. They align the student’s hidden representations with those of the teacher at each layer, selectively extracting knowledge pertinent to the target task. Gu et al. (2024) and Agarwal et al. (2024) introduce a novel approach where the student model first generates sequences, termed ‘self-generated sequences.’ The student then learns by using feedback (i.e. output distribution) from teacher on these sequences. This method is particularly beneficial when the student model lacks the capacity to mimic teacher’s distribution. Moreover, various LLM-quantization methods with distilling feature knowledge from teacher LLMs have been proposed (Tao et al., 2022a; Liu et al., 2023a; Kim et al., 2023b). These methods aim to preserve the original output distribution when quantizing the LLMs, ensuring minimal loss of performance. Additionally, feature knowledge could serve as a potent source for multi-teacher knowledge distillation. Timiryasov and Tastet (2023) leverages an ensemble of GPT-2 and LLaMA as teacher models to extract output distributions. Similarly, FuseLLM (Wan et al., 2024a) innovatively combines the capabilities of various LLMs through a weighted fusion of their output distributions, integrating them into a singular LLM. This approach has the potential

to significantly enhance the student model’s capabilities, surpassing those of any individual teacher LLM.

In summary, feature knowledge offers a more transparent alternative to black-box methods, allowing for deeper insight into and control over the distillation process. By utilizing feature knowledge from teacher LLMs, such as output distributions and intermediate layer features, white-box approaches enable richer knowledge transfer. While showing promise, especially in smaller models, its application is not suitable for black-box LLMs where internal parameters are inaccessible. Furthermore, student models distilled from white-box LLMs may underperform compared to their black-box counterparts, as the black-box teacher LLMs (e.g. GPT-4) tend to be more powerful.

3.1.5 Feedback

Most previous works predominantly focus on one-way knowledge transfer from the teacher to the student for imitation, without considering feedback from the teacher on the student’s generation. The feedback from the teacher typically offers guidance on student-generated outputs by providing preferences, assessments, or corrective information. For example, a common form of feedback involves teacher ranking the student’s generations and distilling this preference into the student model through Reinforcement Learning from AI Feedback (RLAIF) (Bai et al., 2022a). Here is a generalized formulation for eliciting feedback knowledge:

$$\mathcal{D}^{(\text{fb})} = \{(x, y, \phi_{\text{fb}}(x, y; \theta_T)) | x \sim \mathcal{X}, y \sim p_S(y|x)\}, \quad (7)$$

where y denotes the output generated by the student model in response to x , and $\phi_{\text{fb}}(\cdot; \theta_T)$ represents providing feedback from teacher LLMs. This operation evaluates the student’s output y given the input x , by offering assessment, corrective information, or other forms of guidance. This feedback knowledge can not only be distilled into the student to also generate feedback (such as creating a student preference model) but, more importantly, enable the student to refine its responses based on the feedback. Various methods have been explored to elicit this advanced knowledge (Bai et al., 2022a; Luo et al., 2023b; Cui et al., 2023a; Kwon et al., 2023; Jiang et al., 2023b; Chen et al., 2023a; Gu et al., 2024; Agarwal et al., 2024; Chen et al., 2024b; Guo et al., 2024; Ye et al., 2023; Hong et al., 2023; Lee et al., 2023a).

Preference, as previously discussed, represents a notable form of feedback knowledge from teacher models. Various knowledge of preferences could be distilled from teachers by prompting it with specific criteria. Bai et al. (2022a) introduce RLAIF for distilling harmlessness preferences from LLMs. This involves using an SFT-trained LLM to generate response pairs for each prompt, then ranking them for harmlessness to create a preference dataset. This dataset is distilled into a Preference Model (PM), which then guides the RL training of a more harmless LLM policy. Wizard-Math (Luo et al., 2023b) places emphasis on mathematical reasoning. They employ ChatGPT as teacher to directly provide process supervision and evaluate the correctness of each step in the generated solutions. To scale up high-quality distilled preference data, Cui et al. (2023a) develop a large-scale preference dataset for distilling better preference

models, UltraFeedback. It compiles various instructions and models to produce comparative data. Then, GPT-4 is used to score candidates from various aspects of preference, including instruction-following, truthfulness, honesty and helpfulness.

Beyond merely assessing student generations, teachers can also furnish extensive feedback on instances where students underperform. In Lion (Jiang et al., 2023b), teacher model pinpoints instructions that pose challenges to the student model, generating new, more difficult instructions aimed at bolstering the student’s abilities. PERsD (Chen et al., 2023a) showcases a method where teacher offers tailored refinement feedback on incorrect code snippets generated by students, guided by the specific execution errors encountered. Similarly, SelFee (Ye et al., 2023) leverages ChatGPT to generate feedback and revise the student’s answer based on the feedback. In contrast, FIGA (Guo et al., 2024) revises the student’s response by comparing it to the ground-truth response. Furthermore, teacher model’s distribution over the student’s generations can itself act as a form of feedback. MiniLLM (Gu et al., 2024) and GKD (Agarwal et al., 2024) present an innovative strategy wherein the student model initially generates sequences, followed by teacher model producing an output distribution as feedback. This method leverages the teacher’s insight to directly inform and refine the student model’s learning process.

3.1.6 Self-Knowledge

The knowledge could also be elicited from the student itself, which we refer to as *Self-Knowledge*. In this setting, the same model acts both as the teacher and the student, iteratively improving itself by distilling and refining its own previously generated outputs. This knowledge uniquely circumvents the need for an external, potentially proprietary, powerful teacher model, such as GPT-series LLMs. Furthermore, it allows the model to surpass the limitations or “ceiling” inherent in traditional teacher-student methods. Eliciting self-knowledge could be formulated as:

$$\mathcal{D}^{(\text{sk})} = \{(x, y, \phi_{\text{sk}}(x, y)) | x \sim \mathcal{S}, y \sim p_S(y|I \oplus x)\}, \quad (8)$$

where $\phi_{\text{sk}}(\cdot)$ is a generalized function that represents an additional process to the self-generated outputs y , which could include but is not limited to filtering, rewarding, or any other mechanisms for enhancing or evaluating y . It could be governed by external tools or the student itself θ_S . Recent research in this area has proposed various innovative methodologies to elicit self-knowledge, demonstrating its potential for creating more efficient and autonomous learning systems. (Allen-Zhu and Li, 2020; Wang et al., 2022a; Sun et al., 2024b; Yang et al., 2024; Jung et al., 2023; Huang et al., 2023a; Gulcehre et al., 2023; Yuan et al., 2024a; Xu et al., 2023b; Zelikman et al., 2022; Chen et al., 2024a; Zheng et al., 2024; Li et al., 2024c; Zhao et al., 2024; Singh et al., 2023; Chen et al., 2024c; Hosseini et al., 2024)

A notable example of this methodology is Self-Instruct (Wang et al., 2022a), which utilizes GPT-3 for data augmentation through the *Expansion* approach, generating additional data samples to enhance the dataset. This enriched dataset subsequently fine-tunes the original model. Other methods aim to elicit targeted knowledge

from student models by modifying prompts, and leveraging these data for further refinement. In Self-Align (Sun et al., 2024b), they find that models fine-tuned by Self-Instruct data tend to generate short or indirect responses. They prompt this model with verbose instruction to produce in-depth and detailed responses. Then, they employ context-distillation (Askell et al., 2021) to distill these responses paired with non-verbose instructions back to the model. Similarly, RLCD (Yang et al., 2024) introduces the use of contrasting prompts to generate preference pairs from an unaligned LLM, encompassing both superior and inferior examples. A preference model trained on these pairs then guides the enhancement of the unaligned model through reinforcement learning. Several other approaches employ filtering methods to refine self-generated data. For example, Impossible Distillation (Jung et al., 2023) targets sentence summarization tasks, implementing filters based on entailment, length, and diversity to screen self-generated summaries. LMSI (Huang et al., 2023a) generates multiple CoT reasoning paths and answers for each question, and then retains only those paths that lead to the most consistent answer.

Note that refined self-knowledge can be iteratively acquired as the student model continuously improves, further enhancing the student’s capabilities. This is Gulcehre et al. (2023) introduces a Reinforced Self-Training (ReST) framework that cyclically alternates between `Grow` and `Improve` stages to progressively obtain better self-knowledge and refine the student model. During the `Grow` stage, the student model generates multiple output predictions. Then, in the `Improve` stage, these self-generated outputs are ranked and filtered using a scoring function. Subsequently, the language model undergoes fine-tuning on this curated dataset, employing an offline RL objective. Self-Play (Chen et al., 2024a) introduces a framework resembling iterative DPO, where the language model is fine-tuned to differentiate the self-generated responses from the human-annotated data. These self-generated responses could be seen as “negative knowledge” to promote the student to better align with the target distribution. Self-Rewarding (Yuan et al., 2024a) explores a novel and promising approach by utilizing the language model itself as a reward model. It employs LLM-as-a-Judge prompting to autonomously assign rewards for the self-generated responses. The entire process can then be iterated, improving instruction following and reward modeling capabilities.

3.2 Distillation

This section focuses on the methodologies for effectively transferring the elicited knowledge from teacher LLMs into student models. We explore a range of distillation techniques, from the strategies that enhance imitation by *Supervised Fine-Tuning*, *Divergence* and *Similarity*, to advanced methods like *Reinforcement Learning* and *Rank Optimization*, as shown in Figure 3.

3.2.1 Supervised Fine-Tuning

Supervised Fine-Tuning (SFT), or called Sequence-Level KD (SeqKD) (Kim and Rush, 2016), is the simplest and one of the most effective methods for distilling powerful black-box

Divergence Type	$D(p, q)$ Function
Forward KLD	$\sum p(t) \log \frac{p(t)}{q(t)}$
Reverse KLD	$\sum q(t) \log \frac{q(t)}{p(t)}$
JS Divergence	$\frac{1}{2} \left(\sum p(t) \log \frac{2p(t)}{p(t)+q(t)} + \sum q(t) \log \frac{2q(t)}{p(t)+q(t)} \right)$

TABLE 1: Functional forms of D for various divergence types. p : reference

Similarity Function \mathcal{L}_F	Expression
L2-Norm Distance	$\ \Phi_T(f_T(x, y)) - \Phi_S(f_S(x, y))\ _2$
L1-Norm Distance	$\ \Phi_T(f_T(x, y)) - \Phi_S(f_S(x, y))\ _1$
Cross-Entropy Loss	$-\sum \Phi_T(f_T(x, y)) \log(\Phi_S(f_S(x, y)))$
Maximum Mean Discrepancy	$\text{MMD}(\Phi_T(f_T(x, y)), \Phi_S(f_S(x, y)))$

TABLE 2: Summary of similarity functions in knowledge distillation.

LLMs. SFT finetunes student model by maximizing the likelihood of sequences generated by the teacher LLMs, aligning the student’s predictions with those of the teacher. This process can be mathematically formulated as minimizing the objective function:

$$\mathcal{L}_{\text{SFT}} = \mathbb{E}_{x \sim \mathcal{X}, y \sim p_T(y|x)} [-\log p_S(y|x)], \quad (9)$$

where y is the output sequence produced by the teacher model. This simple yet highly effective technique forms the basis of numerous studies in the field. Numerous researchers have successfully employed SFT to train student models using sequences generated by teacher LLMs (Taori et al., 2023; Chiang et al., 2023; Wu et al., 2023c; Xu et al., 2023a; Luo et al., 2023b). Additionally, SFT has been explored in many self-distillation works (Wang et al., 2022a; Huang et al., 2023c; Xu et al., 2023b; Zelikman et al., 2022). Due to the large number of KD works applying SFT, we only list representative ones here. More detailed works can be found in §4.

3.2.2 Divergence and Similarity

This section mainly concentrates on algorithms designed for distilling feature knowledge from white-box teacher LLMs, including distributions and hidden state features. These algorithms can be broadly categorized into two groups: those minimizing divergence in probability distributions and those aimed at enhancing the similarity of hidden states.

Divergence. Divergence-based methods minimize divergence between the probability distributions of the teacher and student models, represented by a general divergence function D :

$$L_{\text{Div}} = \mathbb{E}_{x \sim \mathcal{X}, y \sim \mathcal{Y}} [D(p_T(y|x), p_S(y|x))], \quad (10)$$

The specific form of D varies depending on the type of divergence employed. Table 1 outlines the functional forms of D for different divergence measures. The commonly-used standard KD objectives essentially minimize the approximated forward Kullback-Leibler divergence (KLD) between the teacher and the student distribution (Sanh et al., 2019;

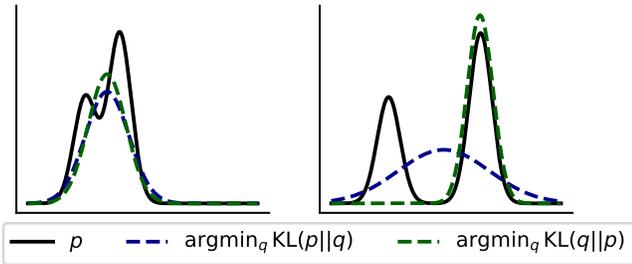


Fig. 6: **Comparison of Forward and Reverse KL Divergences in Approximating a Target Distribution.** Forward KL divergence approach tends to cover all modes of the target distribution but is less precise, i.e. “mode-covering” behavior. Reverse KL divergence method focuses predominantly on the most prominent mode, thereby exhibiting a “mode-seeking” behavior.

Wen et al., 2023; Timiryasov and Tastet, 2023; Liang et al., 2023a; Chen et al., 2024d), which forces p_S to cover all the modes of p_T . However, when a student model is unable to learn all modes of a highly complex teacher, the resultant “mode-covering” behavior might cause the student to assign probability mass to tokens with low probability under the teacher’s distribution (cf. Figure 6 blue curve). This mode-covering phenomenon can potentially lead to hallucinations and low-quality generations. Alternatively, mode-seeking divergences like reverse KL prioritize tokens where the teacher assigns high probabilities (cf. Figure 6 green curve). This approach can mitigate the risk of low-quality outputs, fostering more accurate generations. However, it often does so at the cost of reduced diversity. Gu et al. (2024) adopt reverse KL divergence to prevent students from overestimating low-probability regions of the teacher’s distribution, employing Policy Gradient methods for optimization. Both Agarwal et al. (2024) and Sason and Verdú (2016) assess the effect of different divergence functions in LLM distillation, finding the optimal divergence to be task-dependent. For instance, forward KL divergence is more suitable for tasks like Machine Translation, where the output has fewer modes or variations, while reverse KL divergence is preferable for tasks like dialogue generation and instruction tuning, which involve multiple modes and a wider range of potential responses. Thus, the nature of the task significantly influences the selection of the divergence function for optimal performance.

Similarity. Similarity-based methods in knowledge distillation aim to align the hidden states or features of the student model with those of the teacher. These methods use various similarity metrics to measure and optimize the congruence of internal representations between the two models. The objective is to ensure that the student model not only produces similar outputs to the teacher but also processes information in a comparable manner. The formulation for a similarity-based objective might look like this:

$$\mathcal{L}_{\text{Sim}} = \mathbb{E}_{x \sim \mathcal{X}, y \sim \mathcal{Y}} [\mathcal{L}_F(\Phi_T(f_T(x, y)), \Phi_S(f_S(x, y)))], \quad (11)$$

where $f_T(x, y)$ and $f_S(x, y)$ are the feature maps of the teacher and student models, respectively. The transforma-

tion functions Φ_T and Φ_S are applied to these feature maps to ensure they are in the same shape, facilitating direct comparison. The similarity function \mathcal{L}_F is used to match these transformed feature maps. Table 2 shows common choices for \mathcal{L}_F . Few works have employed similarity-based methods in the KD of LLMs. Among them, Liang et al. (2023a) propose Task-Aware Layer-Wise Distillation (TED), a method that utilizes task-aware filters. These filters are designed to selectively capture the most pertinent information for a specific task from the teacher model. The key objective is to minimize the discrepancy between the filtered representations in both teacher and student models. While similarity-based approaches are common in encoder-based LMs (Sun et al., 2019, 2020; Jiao et al., 2020; Hou et al., 2020; Zuo et al., 2022; Liang et al., 2021), their application in LLM knowledge distillation is not as widespread. However, considering their effectiveness, we anticipate an increase in research exploring these methods for LLM distillation in the near future.

3.2.3 Reinforcement Learning

This section explores advanced methods of distilling knowledge into student models using reinforcement learning (RL). This approach is especially relevant for leveraging the feedback from teacher to train student models (Bai et al., 2022a; Cui et al., 2023a; Luo et al., 2023b; Agarwal et al., 2024; Chen et al., 2024b; Ma et al., 2023a; Pang et al., 2023; Du et al., 2023a). The RL-based distillation process typically involves two main stages:

Distilled Reward Model Training. The first stage involves training a reward model r_ϕ using the feedback data $\mathcal{D}^{(\text{fd})}$ generated by teacher LLMs. Preference data, as one of the typical feedback, is employed to train the student reward model (Bai et al., 2022a; Cui et al., 2023a; Lee et al., 2023a; Kim et al., 2023a). They usually consist of input-output pairs (x, y_w, y_l) . Here, y_w and y_l represent “winning” and “losing” outputs relative to the teacher’s preferences. The loss function for the reward model is defined as:

$$\mathcal{L}_{\text{RM}}(r_\phi, \mathcal{D}^{(\text{fd})}) = - \mathbb{E}_{(x, y_w, y_l) \sim \mathcal{D}^{(\text{fd})}} [\log \sigma(r_\phi(x, y_w) - r_\phi(x, y_l))] \quad (12)$$

This formulation guides the reward model to correctly distinguish between more and less preferable outputs based on the teacher’s criteria. Instead of learning the instance-level rewards, RLMEC (Chen et al., 2024b) adopts a different approach by training a generative reward model. It is trained on an erroneous solution rewriting data distilled from a teacher LLM. This distilled reward model can produce token-level rewards for RL training.

Reinforcement Learning Optimization. In the second stage, the student model, represented by a policy π_θ , is optimized to maximize the expected reward as per the trained reward model. Simultaneously, it minimizes the divergence from a reference policy π_{ref} , typically the initial policy of the student model trained by SFT, controlled by a factor β . The RL objective is given by:

$$\max_{\pi_{\theta}} \mathbb{E}_{x \sim X, y \sim \pi_{\theta}(y|x)} [r_{\phi}(x, y)] - \beta D_{\text{KL}} [\pi_{\theta}(y | x) || \pi_{\text{ref}}(y | x)] \quad (13)$$

This RL framework not only ensures that the student model learns the explicit content from the teacher but also effectively adopts the teacher’s preference patterns. The use of RL, particularly with the PPO (Schulman et al., 2017) algorithm, offers a robust mechanism for aligning the student model’s outputs with the teacher. Alternatively, the teacher LLM can also serve as the reward model to directly assign rewards during RL, circumventing the need for training a reward model (Lee et al., 2023a; Kwon et al., 2023). While this approach may exhibit superior performance, it comes at a higher computational cost compared to employing a smaller distilled reward model.

3.2.4 Ranking Optimization

Ranking optimization presents a stable and computationally efficient alternative to RL for injecting preference feedback into language models (Rafailov et al., 2023; Song et al., 2023a; Yuan et al., 2023b). This method, diverging from traditional RL approaches, directly incorporates ranking information into language models from a fixed preference dataset during fine-tuning. Intuitively, it directly updates policy to increase the relative likelihood of preferred over less favored responses. This direct optimization of preferences, without the need for sampling outputs, makes the process more stable and efficient. Recently, some works have been proposed to explore using ranking optimization to distill teacher’s preferences into student models (Tunstall et al., 2023; Hong et al., 2023; Yuan et al., 2024a).

Zephyr (Tunstall et al., 2023) utilizes Direct Preference Optimization (DPO) (Rafailov et al., 2023) to distill the preference alignment in teacher LLMs. DPO streamlines the objective of reinforcement learning (as in Eq. 13), which involves reward maximization with a KL-divergence constraint, into a single-stage policy training. Specifically, DPO’s training goal is to maximize the following expectation:

$$\mathbb{E}_{(x, y_w, y_l) \sim D^{(td)}} \left[\log \sigma \left(\beta \log \frac{\pi_{\theta}(y_w|x)}{\pi_{\text{ref}}(y_w|x)} - \beta \log \frac{\pi_{\theta}(y_l|x)}{\pi_{\text{ref}}(y_l|x)} \right) \right], \quad (14)$$

where y_w is preferred over y_l according to the teacher LLM. Hong et al. (2023) (Hong et al., 2023) adopt two ranking-based optimization objectives, Rank Responses to align Human Feedback (RRHF) (Yuan et al., 2023b) and Preference Ranking Optimization (PRO) (Song et al., 2023a), for preference distillation. RRHF (Yuan et al., 2023b) focuses on a ranking loss defined as:

$$\mathcal{L}_{\text{RRHF}} = \sum_{r_i < r_j} \max(0, p_i - p_j), \quad (15)$$

where r_i and r_j are the reward scores assigned by the teacher LLM for responses y_i and y_j , respectively, and p_i, p_j are their corresponding conditional log probabilities under the policy π_{θ} . This approach emphasizes direct comparison and ranking of responses based on the teacher’s preferences. PRO (Song et al., 2023a) expands the concept of pairwise

comparison to handle preference rankings of any length. For a given instruction x and a sequence of responses ordered by teacher preference as $y_1 \succ y_2 \succ \dots \succ y_n$, the RPO training objective is:

$$\mathcal{L}_{\text{PRO}} = - \sum_{k=1}^{n-1} \log \frac{\exp(p_k)}{\sum_{i=k}^n \exp(p_i)}, \quad (16)$$

where p_k represents the conditional log probabilities for y_k under the student policy π_{θ} . By iteratively contrasting the likelihood of generating responses, PRO optimizes the student LM to prioritize the most preferred response while progressively ranking the rest in the order of diminishing preference.

4 SKILL DISTILLATION

Building upon the foundation laid out in Section 3 about eliciting knowledge and distillation algorithms, we shift our focus to how these techniques facilitate the distillation of specific skills in LLMs. Our exploration will encompass a diverse range of skills exhibited by LLMs, including *Context Following*, *Alignment*, *Agent*, *NLP Task Specialization* and *Multi-Modality*. *Context Following* focuses on the student’s ability to comprehend and respond effectively to input information. *Alignment* delves into the student’s capability to align its output with the teacher’s responses. Moving forward, *Agent* underscores the autonomous nature of language models. *NLP Task Specialization* highlights the LLM’s versatility in specializing across various Natural Language Processing tasks, demonstrating its adaptability. Finally, *Multi-Modality* encompasses the knowledge transfer from teacher LLMs to multi-modal models. Table 3 summarizes the representative works, encompassing details such as the skills involved, seed knowledge, teacher LLM, student model, knowledge elicitation method, and training objectives.

4.1 Context Following

This part concentrates on the distillation of context following skills from LLMs. This process involves transferring the ability of LLMs to handle a variety of complex contexts — such as few-shot demonstrations, intricate instructions, dialogue history, and retrieval-augmented information — into smaller models. Many research efforts in this domain aim to imbue smaller models with these sophisticated, context-following capabilities. Our discussion here will dissect this facet of skill distillation, categorizing it based on different types of context and elaborating on how each is distilled and incorporated into smaller, efficient models.

4.1.1 Instruction Following

Instruction-following capacity enables LLMs to understand and follow user-given instructions. This ability significantly enhances human-AI interaction, allowing for seamless understanding and execution of tasks as directed by users. A primary method for acquiring this skill involves constructing instruction-like prompt-response pairs and employing Supervised Fine Tuning (SFT) for model training. Data for this purpose can be manually curated by human experts or transformed from existing NLP tasks into instructional

Methods	Skill	Seed Knowledge	Teacher LLM	Student Model	Knowledge Elicitation	Objective
<i>Context Following</i>						
Self-Instruct (Wang et al., 2022a)	IF	175 human-curated tasks	GPT3	GPT3	Expansion + Self-Knowledge	SFT
Alpaca (Taori et al., 2023)	IF	175 human-curated tasks	GPT3	LLaMA	Expansion + Self-Knowledge	SFT
LaMini-LM (Wu et al., 2023c)	IF	3.5K Wikipedia Categories + Mixed Dataset	ChatGPT	Various Models	Expansion	SFT
WizardLM (Xu et al., 2023a)	IF	Alpaca Data	ChatGPT	LLaMA	Expansion	SFT
Lion (Jiang et al., 2023b)	IF	Alpaca Data	ChatGPT	LLaMA	Labeling + Expansion + Feedback	SFT
BabyLlama (Timiryasov and Tastet, 2023)	IF	10M-word BabyLM dataset	GPT2 + small LLaMA	58M-parameter LLaMA	Feature	D&S
MiniLLM (Gu et al., 2024)	IF	Dolly Dataset	GPT2 + OPT + LLaMA	GPT2 + OPT + LLaMA	Feature	D&S
Self-Align (Sun et al., 2024b)	IF	Human-written Principles	LLaMA	LLaMA	Expansion + Self-Knowledge	SFT
Self-Rewarding (Yuan et al., 2024a)	IF	Human-written Samples	LLaMA	LLaMA	Self-Knowledge	SFT + RL
STaR (Zelikman et al., 2022)	IF	Arithmetic + CommonsenseQA + GSM8K	GPT-J	GPT-J	Self-Knowledge	SFT
Llama-GPT4 (Peng et al., 2023a)	IF	Alpaca Dataset	GPT4	LLaMA	Labeling	SFT
Reflection-Tuning (Li et al., 2023e)	IF	Alpaca/WizardLM Dataset	ChatGPT	LLaMA	Labeling	SFT
Selective Reflection-Tuning (Li et al., 2024d)	IF	Alpaca/WizardLM Dataset	ChatGPT	LLaMA	Labeling	SFT
Vicuna (Chiang et al., 2023)	IF/MD	Human Conversation	ChatGPT + GPT4	LLaMA	Labeling	SFT
Koala (Geng et al., 2023)	IF/MD	Human Conversation	ChatGPT	LLaMA	Labeling	SFT
Baize (Xu et al., 2023b)	IF/MD	Quora + Stack Overflow	ChatGPT	LLaMA	Expansion + Self-Knowledge	SFT
UltraChat (Ding et al., 2023b)	IF/MD	Wikidata + Text Material + C4	ChatGPT	LLaMA	Curration	SFT
Orca (Mukherjee et al., 2023)	IF/TP	FLAN-v2	ChatGPT + GPT4	LLaMA	Labeling	SFT
Orca2 (Mittra et al., 2023)	IF/TP	FLAN-v2 + Few-Shot/Math/Synthetic	GPT4	LLaMA	Labeling	SFT
SelfFree (Ye et al., 2023)	IF/TP	Human Conv, Flan/Code/Math Collection	ChatGPT	LLaMA	Labeling	SFT
CoT-Distill (Hsieh et al., 2023)	IF/TP	e-SNLI + ANLI + CQA + SVAMP	PaLM	T5	Labeling	SFT
KnowPAT (Zhang et al., 2023a)	IF/TP	CPKG + QA Data	ChatGPT + ChatGLM + Vicuna-7B	LLaMA	Labeling	SFT
DEBATunE (Li et al., 2024e)	IF/TP	Controversial Topics	ChatGPT	LLaMA	Labeling	SFT
Phi-1 (Gunasekar et al., 2023)	IF/Code	-	GPT3.5	phi-1	Curration	SFT
Phi-1.5 (Li et al., 2023a)	IF/Code	20k Topics from Web	GPT3.5	phi-1	Curration + Labeling	SFT
SAIL (Luo et al., 2023c)	IF/RAG	Alpaca Data + Web Content	GPT4	LLaMA	Label	SFT
KARD (Kang et al., 2023b)	IF/RAG	MedQAUSMLE	ChatGPT	T5 + OPT	Label	SFT + D&S
Self-RAG (Asai et al., 2023)	IF/RAG	Open-Instruct	GPT4	LLaMA	Labeling	SFT
<i>Alignment</i>						
OpenChat (Wang et al., 2023c)	IF/Preference	Human Conversation	ChatGPT + GPT4	LLaMA	Labeling	SFT + RL
Zephyr (Nustall et al., 2023)	IF/Preference	Mixed Datasets	GPT4	Mistral	Labeling + Feedback	SFT + RO
ALMoST (Kim et al., 2023a)	IF/Preference	Human-written Prompts	LLaMA	LLaMA	Expansion + Labeling	SFT + RL
RLCD (Yang et al., 2024)	IF/Preference	Human-written Prompts	LLaMA	LLaMA	Labeling	SFT + RL
RLAIF (Lee et al., 2023a)	IF/Preference	Human-written Prompts	PaLM 2	PaLM 2	Labeling + Feedback	RL
GPT3 Reward (Kwon et al., 2023)	Preference	Human-written Prompts	GPT3	GPT3	Labeling	RL
ILF (Scheurer et al., 2023)	Preference	Task-specific Datasets	GPT3 + FeedME	GPT3	Labeling	RL
ULTRA FEEDBACK (Cui et al., 2023a)	Preference	Mixed Datasets	GPT4	LLaMA	Labeling	RL
Constitutional AI (Bai et al., 2022a)	Preference/Value	Human-written Prompts	Self-defined Student Model	Self-defined Model	Labeling + Expansion + Feedback	SFT + RL
SANDBOX (Liu et al., 2023b)	Value	Simulation	text-davinci-002-003 + GPT4 + ChatGPT	LLaMA	Data Curration	SFT + RL
<i>Agent</i>						
Toolformer (Schick et al., 2023)	Tool	CCNet	GPT-J	GPT-J	Labeling	SFT
Graph-ToolFormer (Zhang, 2023)	Tool	Mixed Graph Dataset	ChatGPT	GPT-J + LLaMA	Labeling	SFT
Gorilla (Patil et al., 2023)	Tool	Online API Documentation	GPT4	LLaMA	Labeling	SFT
GPT4Tools (Yang et al., 2023b)	Tool	Image Content	ChatGPT	LLaMA	Curration + Expansion	SFT
ToolAlpaca (Tang et al., 2023a)	Tool	Public-apis Repository	ChatGPT	LLaMA	Curration	SFT
ToolLLM (Qin et al., 2023a)	Tool	Real-world APIs	ChatGPT	LLaMA	Curration	SFT
MLLM-Tool (Wang et al., 2024)	Tool	HuggingFace Model Cards	GPT4	LLaMA	Curration	SFT
FireAct (Chen et al., 2023b)	Planning	Mixed QA Dataset	GPT4	LLaMA	Labeling	SFT
AgentTuning (Zeng et al., 2023a)	Planning	6 Agent Tasks	GPT4 + ChatGPT	LLaMA	Labeling + Expansion	SFT
Lumos (Yin et al., 2023a)	Planning	Mixed Interactive Tasks	GPT4	LLaMA	Labeling	SFT
AUTOACT (Qiao et al., 2024)	Planning	Mixed QA Tasks	LLaMA	LLaMA	Labeling	SFT
<i>NLP Task Specialization</i>						
AugGPT (Dai et al., 2023a)	NLU	Amazon/Symptoms/PubMed20k Dataset	ChatGPT	BERT	Label	SFT
TDG (He et al., 2023b)	NLU	SST + QQP + MNLI	GPT3	BERT	Expansion	SFT
SunGen (Gao et al., 2023a)	NLU	Text Classification Tasks	GPT2	DistilBERT	Curration	SFT
UDG (Wang et al., 2021a)	NLU	NLU Tasks	GPT3	BERT	Expansion	SFT
InheritSumm (Xu et al., 2023c)	NLG	Pile + ArXiv + CNN/DM + WikiHow	GPT3.5	ZCode++	Label	SFT
DIMSUM+ (Jung et al., 2023)	NLG	None	GPT2 + CTRL + BioGPT	T5	Curration + Self-Knowledge	SFT
Genie (Yehudai et al., 2024)	NLG	ELIS + ASQA + NQ + CNN/DM	Falcon + LLaMA	FLAN + LLaMA	Label	SFT
GKD (Agarwal et al., 2024)	NLG/NLU/IF	XSum+WMT14 en-de+GSM8K+FLAN2021	T5-LLaMA	T5	Feature + Feedback	D&S + RL
QUILL (Srinivasan et al., 2022)	IR	IR Datasets	T5	4-layer Transformer	Internal Knowledge	D&S
RankVicuna (Pradeep et al., 2023a)	IR	IR Datasets	ChatGPT	LLaMA	Labeling	SFT
RankZephyr (Pradeep et al., 2023b)	IR	IR Datasets	ChatGPT + GPT4	Mistral	Labeling	SFT
NDR (Mysore et al., 2023)	Recommendation	Recommendation Datasets	GPT3	MPnet-110M	Labeling	SFT
InstructRec (Zhang et al., 2023b)	Recommendation	39 instruction templates	ChatGPT	Flan-T5	Expansion + Self-Knowledge	SFT
ONCE (Liu et al., 2023c)	Recommendation	Recommendation Dataset	ChatGPT	LLaMA	Labeling	SFT
PandaLM (Wang et al., 2023b)	Evaluation	Alpaca Data	ChatGPT	LLaMA	Labeling	SFT
Prometheus (Kim et al., 2024)	Evaluation	50 Seed Rubrics	GPT4	LLaMA	Labeling	SFT
InstructScore (Xu et al., 2023d)	Evaluation	Mixed Dataset	GPT3	LLaMA	Labeling	SFT
WizardMath (Luo et al., 2023b)	Math	GSM8k + MATH	ChatGPT	LLaMA	Expansion + Feedback	SFT + RL
Mammoth (Yue et al., 2023a)	Math/TP	Mixed Math Dataset	GPT4	LLaMA	Labeling	SFT
Mixed Distill (Chenglin et al., 2023)	Math/TP	SVAMP + GSM8K + ASDIV + StrategyQA	ChatGPT	LLaMA	Labeling	SFT
WizardCoder (Luo et al., 2023a)	Code	Code Alpaca Data	ChatGPT	StarCoder	Expansion	SFT
Magicoder (Wei et al., 2023)	Code	Existing Source Codes	ChatGPT	LLaMA	Curration	SFT
WaveCoder (Yu et al., 2024)	Code	Existing Source Codes	GPT4	LLaMA	Curration	SFT
Code Alpaca (Chaudhary, 2023)	Code	Code Instructions	ChatGPT	LLaMA	Expansion + Self-Knowledge	SFT
Code Llama (Roziere et al., 2023)	Code	Human-written Instructions	LLaMA	LLaMA	Expansion + Self-Knowledge	SFT
Code Clean (Jain et al., 2023)	Code	Code Datasets	ChatGPT	LLaMA	Labeling	SFT
<i>Multi-Modality</i>						
LLaVA (Liu et al., 2023e)	Vision-Language	COCO	GPT4	LLaMA	Labeling	SFT
SVIT (Zhao et al., 2023b)	Vision-Language	Visual Genome + COCO	GPT4	LLaMA	Labeling	SFT
LVIS-Instruct4V (Wang et al., 2023e)	Vision-Language	LVIS	GPT4V	LLaMA	Labeling	SFT
LLaVAR (Zhang et al., 2023d)	Vision-Language	LAION	GPT4	LLaMA	Labeling	SFT
Macaw-LLM (Lyu et al., 2023)	Multiple Modalities	Image/Video with Caption	ChatGPT	LLaMA	Labeling	SFT
MIMIC-IT (Li et al., 2023f)	Multiple Modalities	Image/Video Dataset	ChatGPT	LLaMA	Labeling	SFT
ChatBridge (Zhao et al., 2023d)	Multiple Modalities	Task-Specific/Multimodal-Chat Data	GPT4 + ChatGPT	LLaMA	Labeling	SFT

TABLE 3: A summary of skill distillation works. IF: Instruction Following, MD: Multi-turn Dialogue, TP: Think Pattern, RAG: Retrieval-Augmented Generation, NLU: Natural Language Understanding, NLG: Natural Language Generation, IR: Information Retrieval, SFT: Supervised Fine-Tuning, D&S: Divergence and Similarity, RL: Reinforcement Learning, RO: Ranking Optimization.

formats with templates, such as prefacing machine translation data with *“Translate this sentence to Spanish:”*. However, these approaches have limitations. Manual data creation is labor-intensive, while template-based transformation lacks diversity in instructions and may not align well with natural human input. LLMs like GPT-4 offer an efficient alternative for creating diverse and controlled SFT data by their capabilities of in-context learning and instruction following. Most

relevant works use OpenAI’s GPT series models to generate prompt-response data pairs and then train the student LLMs by supervised fine-tuning (Wang et al., 2022a; Taori et al., 2023; Chiang et al., 2023; Wu et al., 2023c; Xu et al., 2023a; Mukherjee et al., 2023; Mittra et al., 2023; Luo et al., 2023b; Peng et al., 2023a).

Basic Instructions. Self-Instruct (Wang et al., 2022a) leverages the in-context learning capability of GPT-3 to expand

a seed pool of 175 tasks to 52K task-agnostic instructions, ensuring a broad spectrum of general instructions. Additionally, a filtering and post-processing stage is introduced to eliminate redundant or similar instructions. Notably, through training with this enriched dataset, GPT-3 acquires the ability to follow instructions, enabling it to perform comparably to InstructGPT in zero-shot instruction tasks and when provided with expert-written instructions for novel tasks. Based on the self-instruct method, Taori et al. (2023) train an Alpaca model using the Llama 7B model on 52K instruction-following demonstrations, generated in a similar style as self-instruct but utilizing the more robust text-davinci-003 model. To enhance the diversity of instructional data, Wu et al. (2023c) introduce a technique known as *Topic-Guided Instruction Generation*. This method involves gathering 3.5K common topics from Wikipedia to serve as guidance during the generation process.

Complex Instructions. Some works promote students to solve more complex instructions (Xu et al., 2023a; Luo et al., 2023b,a; Guo et al., 2023c). According to Xu et al. (2023a), instruction datasets derived from human-written seeds often exhibit low to moderate complexity. To enhance the complex instruction-following capabilities of smaller models, WizardLM (Xu et al., 2023a) introduces *Evol-Instruct*. This method gradually transforms instructions into more complex forms through a multi-step evolution process, focusing on both increasing difficulty levels and expanding the diversity of topics. They conducted four rounds of evolution using the OpenAI ChatGPT API, resulting in a dataset of 250k complex instructions. Subsequently, they trained the LLaMA 7B model, referred to as WizardLM, on this dataset. In the high-difficulty section of test instructions, WizardLM even outperformed ChatGPT, achieving a win rate 7.9% higher than ChatGPT. Zhao et al. (2023e) further conduct preliminary studies revealing the effectiveness of increasing instruction complexity. Instruction Fusion (Guo et al., 2023c) further uses teacher LLMs to increase the complexity by fusing two distinct evolved instructions. Furthermore, this concept of “evolving” instructions has been extended to distill specific skills such as coding (Luo et al., 2023a) and mathematics (Luo et al., 2023b).

Human Instructions. In contrast to works that rely on generating instructions from ChatGPT, which may lack diversity and have gaps with real human instructions, Vicuna (Chiang et al., 2023) and Koala (Geng et al., 2023) showcase impressive performance by using human conversations and natural instructions from community-contributed conversations. These conversations, found in platforms like ShareGPT, provide a forum for users to share their interactions with ChatGPT. It’s important to note, however, that models trained on such natural conversations might mimic the style but may not fully capture the reasoning process of the original teacher (Gudibande et al., 2023; Mukherjee et al., 2023).

System Instructions. To encourage student models to learn the reasoning process, Orca and Orca 2 (Mukherjee et al., 2023; Mitra et al., 2023) enhance the *prompt, response* data pairs by introducing a *system message* (e.g., “explain like I’m five, think step-by-step”) to encourage student models to grasp the reasoning process. This *system message*

prompts GPT-4 to provide explanation traces that elucidate the teacher’s reasoning process. Orca 2 (Mitra et al., 2023) further trains the student model to identify the most effective solution strategy for each task, guided by Orca’s performance. This approach significantly improves the ability of smaller models to follow instructions that involve reasoning.

High-Quality Instructions. As demonstrated in Zhou et al. (2023a) and (Li et al., 2024f), the data quality is crucial for instruction following training. UltraChat (Ding et al., 2023b) distills large-scale data with high-quality and diverse instructions from teacher LLMs by various meta-information. The UltraLLaMA model, fine-tuned on this data, consistently surpasses other open-source models. The Phi series models (Gunasekar et al., 2023; Li et al., 2023a; Mar, 2023) prioritize data quality and employ synthetic methods to generate data of “textbook quality” to enhance the learning experience for smaller models. Notably, Phi exhibits the ability to follow instructions effectively even without specific instruction fine-tuning. What’s particularly remarkable is that Phi-2, with just 2.7 billion parameters, outperforms Mistral and Llama-2 models with 7B and 13B parameters across various benchmark evaluations.

Improved Instructions. Another line of work focuses on improving the quality of existing instruction data, including both the improvement of instruction and corresponding response. SelfFee (Ye et al., 2023) utilizes the ChatGPT to iteratively improve the quality of responses. ExpertLLaMA (Xu et al., 2023f) improves the quality of responses by augmenting vanilla instructions with specialized Expert Identity descriptions. Reflection-Tuning (Li et al., 2023e) improves both the instruction and response sequentially by reflecting on specific criteria. DEITA (Liu et al., 2023h) proposes to enhance and score instructions in three directions including complexity, quality, and diversity to get high-quality distillation data. MUFFIN (Lou et al., 2023) proposes to scale the instruction according to the input by diversifying these tasks with various input facets. Selective Reflection-Tuning (Li et al., 2024d) first involves the student model in the data improvement pipeline with a novel student-selection module, in which the student model is able to decide the data learn from.

In summary, distilling instruction data from teachers presents a promising avenue for training cheap and reproducible instruction-following language models. Current small models have made strides in enhancing various aspects of instruction-following ability, like diversity, complexity and explanation. However, student models trained on instruction data expanded by ChatGPT often mimic ChatGPT’s style without replicating its factual accuracy (Gudibande et al., 2023). Achieving a more capable instruction-following capability requires a stronger teacher LLM (Gudibande et al., 2023) and access to diverse, high-quality instruction data, such as the one used in Orca (Mukherjee et al., 2023; Mitra et al., 2023), which incorporates extensive task instructions from the Flan 2022 Collection (Longpre et al., 2023).

4.1.2 Multi-turn Dialogue

While instruction following focuses on single-instance command execution, multi-turn dialogue extends this to comprehend and maintain context through ongoing interactions. This skill is vital for models to engage meaningfully in human-like conversations and respond coherently over successive dialogue turns. Some works have been dedicated to train to small chat models by distilling multi-turn knowledge from teacher LLMs (Chiang et al., 2023; Xu et al., 2023b; Ding et al., 2023b; Li et al., 2023b; Wang et al., 2023c; Tunstall et al., 2023).

ShareGPT serves as a platform for users to share their conversations with ChatGPT, offering a vast repository of multi-turn conversations readily available. Some small chat models are trained using this data to acquire the capability for engaging in multi-turn dialogues (Chiang et al., 2023; Ye et al., 2023; Wang et al., 2023c). For example, Vicuna (Chiang et al., 2023) is a chat model exclusively trained on ShareGPT data. Despite its sole training source being ShareGPT, Vicuna achieves a high MT-Bench (Zheng et al., 2023a) score assigned by GPT-4³. In the study conducted by Wang et al. (2023c), GPT-3.5 and GPT-4 are employed to generate mixed responses using ShareGPT data. They assign higher rewards to responses generated by GPT-4, aiming to incentivize student models to produce high-quality responses. Additionally, Ye et al. (2023) enhance the quality of multi-turn data from ShareGPT by generating self-feedback on model responses and iteratively refining the responses based on the received feedback.

To enhance the multi-turn capabilities of student models, another line of research focuses on expanding conversational datasets through self-chat and using them to train smaller models (Xu et al., 2023b; Ding et al., 2023b; Tunstall et al., 2023). For instance, Xu et al. (2023b) initiate their work by using questions sourced from Quora and Stack Overflow as seeds, resulting in the collection of 111.5k dialogues through self-chat. Subsequently, they employ parameter-efficient tuning to train a chat model named Baize. Ding et al. (2023b) first construct a significantly larger dataset called UltraChat, comprising 1.5 million high-quality multi-turn dialogues. They achieve this by distilling instructions and dialogues from ChatGPT. Notably, UltraChat encompasses a wide range of topics and instructions. Building upon the UltraChat dataset, they fine-tune a LLaMA model, resulting in the creation of a powerful chat model known as UltraLLaMA. UltraLLaMA consistently outperforms other open-source chat models, including Vicuna and Baize. Furthermore, UltraChat is employed in conjunction with an AI preference-aligned chat model named Zephyr (Tunstall et al., 2023). Zephyr enhances intent alignment through the application of distilled direct preference optimization (dDPO).

4.1.3 RAG Capability

LLMs are known to lack the ability to utilize up-to-date knowledge, and often produce responses containing factual inaccuracies due to their sole reliance on the parametric knowledge. Retrieval-Augmented Generation (RAG) is a

promising technique to decrease this issue. Handling the augmented context of retrieved information is also a non-trivial skill of LLMs. Several approaches to distill RAG capabilities have been proposed (Kang et al., 2023a; Luo et al., 2023c; Asai et al., 2023).

SAIL (Luo et al., 2023c) starts by retrieving search results for each training case using search APIs, creating search-augmented instructions that include both the instruction and grounding information. To encourage the language model to prioritize informative retrieval results, they input each retrieved passage along with the ground truth response into the entailment model to label each retrieval result for relevance. Subsequently, the search-augmented instructions and relevance labels are fed into teacher LLMs (like GPT-4) for generating responses. Following fine-tuning on this training set, the student model becomes proficient at denoising search results and generating accurate responses. KARD (Kang et al., 2023b) distills rationales r from the teacher LLM in response to questions x . These rationales are then utilized to train two models: a student LM and a Reranker. For training the student LM, the rationales serve as a means to retrieve relevant knowledge d , and the student LM is subsequently fine-tuned using the rationales alongside questions and knowledge. However, during inference, only questions are available. To address this, the Reranker is trained to mimic how the retriever scores passages with the rationale by minimizing the KL divergence between $Retriever(d|r)$ and $Reranker(d|x)$. However, the integration of a fixed number of passages in language models, without considering their necessity or relevance, can reduce versatility and lead to the generation of unhelpful responses. To equip student LMs with adaptive RAG capabilities, Self-Rag (Asai et al., 2023) distills this adaptive ability from teacher LLMs into a small critic model. This critic model determines whether retrieval is necessary and evaluates the quality of the retrieved results by generating ‘reflection tokens.’ For instance, Self-Rag initiates the retrieval operation when generating the reflection token `Retrieve`. To distill this critic data, GPT-4 is prompted to assess the need for retrieval using few-shot demonstrations I , the task input x , and output y to predict a reflection token r as follows: $p(r|I, x, y)$.

4.2 Alignment

4.2.1 Thinking Pattern

Most existing methods mainly focus on directly aligning the direct responses of the student models to the responses of teacher models (Taori et al., 2023). Though effective, these models might suffer the problems that they tend to learn to imitate the response style of the teacher models, but not the reasoning process (Mukherjee et al., 2023). Thus in order to better distill from the teacher models, methods are proposed that not only imitate the pure responses but some novel thinking patterns (Ye et al., 2023; Mukherjee et al., 2023; Mitra et al., 2023; Wang et al., 2023d; Cheng et al., 2023; Zhang et al., 2023a).

Motivated by the effectiveness of LLMs in generating their own feedback without relying on external models (Schick et al., 2022; Madaan et al., 2023; Saunders et al., 2022), SelfFee (Ye et al., 2023) proposes to train a

3. MT-Bench: a multi-turn question set, where the generations of models are evaluated by LLM, like GPT-4.

model that has been fine-tuned to continuously revise its own answer until it provides a high-quality response in a single inference. During training, it utilizes both the final response and feedback chain as the fitting target. This pattern, response with the revision process, shows a promising performance gain. Following SelfFee, Reflection-Tuning (Li et al., 2023e, 2024d) also utilizes the reflection process as the learning pattern. Noticing the lack of reasoning imitation of the previous methods, Orca (Mukherjee et al., 2023) first proposes Explanation tuning, which aims to learn the reasoning steps, including explanation traces, step-by-step thought processes, and other complex instructions, from the teacher model, rather than just the vanilla styles. Extensive experiments verify the effectiveness of distilling with this thinking pattern. The following Orca2 (Mitra et al., 2023) further presents to equip the student models with the ability to utilize different solution strategies for different tasks, motivated by the capability discrepancies between the smaller and larger models. By employing this training pattern, the student models are able to gain a better reasoning ability. Besides learning with the corresponding revision or reflection process, another thinking pattern that recently appeared is generating both responses and preferences. Zhang et al. (2023a) propose to learn both the knowledge and corresponding preference for domain-specific QA with LLMs. Recently, DEBATunE (Li et al., 2024e) proposes to improve the controllability of LLMs in generating statements on controversial topics. By engaging two agents in a structured multi-round debate on controversial topics, salient and in-depth statements can be obtained and further distilled into the student models.

4.2.2 Preference

The previously mentioned methods primarily focus on the basic capability of student models to produce outcomes that are strictly accurate but may not align with human preferences, reaching alignment at this level enables these models to aid in various tasks without meeting higher-level demands. Early methods mainly utilize human feedback for the alignment of human preferences (Ziegler et al., 2019; Stiennon et al., 2020; Wu et al., 2021; Ouyang et al., 2022; Bai et al., 2022b; Köpf et al., 2023; Yuan et al., 2023b). However, obtaining human feedback is costly and labor-intensive, thus methods that learn from AI feedback are also proposed to align with human preferences (Bai et al., 2022a; Kwon et al., 2023; Scheurer et al., 2023; Kim et al., 2023a; Roit et al., 2023; Yang et al., 2024; Lee et al., 2023a; Tunstall et al., 2023; Cui et al., 2023a; Wang et al., 2023f).

The concept of RLAIIF, introduced by Bai et al. (2022a), involves the integration of preferences labeled by LLMs with those labeled by humans. This approach is designed to simultaneously optimize two key objectives: ensuring the helpfulness of the output and minimizing any potential harm, making the responses of LLMs more aligned with Human preferences. Kwon et al. (2023) develop a proxy reward function using LLMs like GPT-3, which is created by first providing the LLM with a description of the behaviors desired by the user, along with a small number of examples. The LLM then produces rewards by evaluating how closely the outputs of a model align with the provided descriptions, essentially measuring their relevance to the estab-

lished ground truth. Scheurer et al. (2023) propose Imitation Learning from Language Feedback, in which a language model is utilized to improve various outputs generated by a model. This refinement is based on a reference provided by a human. Following this process, the most effectively refined output is chosen to be used in further supervised fine-tuning. As outlined by Kim et al. (2023a), ALMoST involves condensing human preferences into a set of heuristic guidelines. An example of such a rule is the idea that larger LLMs that utilize more comprehensive and higher-quality prompts are likely to yield superior responses. Based on these established guidelines, comparison data is generated using responses from LLMs of different sizes and with varying prompts. This data is then used to train a reward model. Yang et al. (2024) propose Reinforcement Learning from Contrast Distillation, which aims to align language models without relying on human feedback. This approach involves training a preference model using simulated pairs of preferences, including both high-quality and low-quality examples which are generated through contrasting prompts, positive and negative.

Lee et al. (2023a) further highlight the effectiveness of RLAIIF. This work proposes that RLAIIF not only matches but in some cases surpasses RLHF, and interestingly, RLAIIF can also enhance the performance of Supervised Fine-Tuning. Another notable discovery is that directly prompting the LLM for reward scores during reinforcement learning can be more effective than the conventional approach of training a reward model based on LLM preferences. Wang et al. (2023f) propose Conditioned-RLFT, which treats different data sources as coarse-grained reward labels and develops a class-conditioned policy to effectively utilize the varying qualities of data, which is a Reinforcement Learning-free supervised learning approach. Cui et al. (2023a) propose a large-scale, high-quality, and diversified preference dataset labeled by GPT4 for comprehensive feedback. Tunstall et al. (2023), by proposing distilled Direct Preference Optimization (Rafailov et al., 2023) on UltraFeedback, obtaining a small by powerful LLM.

4.2.3 Value

Attaining alignment with human preferences allows large models to optimize human satisfaction by operating in a manner that aligns with human preferences. However, to establish trustworthy LLMs, the notion of ‘aligning LLMs with human values’ is proposed and the key principles of alignment are often summarized as the “HHH” criteria: helpful, harmless, honest (Weidinger et al., 2021; Askell et al., 2021). Numerous methods have been undertaken for building trustworthy LLMs. However, due to the intrinsic difficulty of this aim, which is still an unsolved problem for proprietary models (Sun et al., 2024a), most existing methods rely on constructing high-quality human preference datasets (Ji et al., 2023b; Solaiman and Dennison, 2021; Bai et al., 2022b; Qiu et al., 2022; Kiesel et al., 2022; Liu et al., 2022a), utilizing human-written rules as constrains (Glaese et al., 2022; Sun et al., 2023b, 2024b), etc. For detailed progress on trustworthy LLMs, please further refer to Yao et al. (2023a); Liu et al. (2023i); Sun et al. (2024a).

Though slightly under-explored, aligning LLMs with human values by distilling is still possible (Bai et al., 2022a;

Cui et al., 2023a; Yang et al., 2024; Sun et al., 2024b). For instance, Bai et al. (2022a) propose RLAIIF, utilizing AI-generated labels to interactively improve both helpfulness and harmlessness. Sun et al. (2024b) prompt the student model with 16 principles as guidelines for generating helpful, ethical, and reliable responses. Similarly, both harmless and harmful generations could be elicited by modifying the prompts, and then are used to train the preference model (Yang et al., 2024). Cui et al. (2023a) utilize GPT-4 to rank generations regarding helpfulness, truthfulness, and honesty. Liu et al. (2023b) advance the alignment of LLMs with societal values by incorporating simulated social interactions into the training process. This approach encompasses a range of elements, including demonstrations that are both in alignment and in conflict with social norms, as well as collective ratings, in-depth feedback, and responses that are revised iteratively.

4.3 Agent

4.3.1 Tool Using

While recent LLMs have shown proficiency in solving various tasks, they still tend to make mistakes when handling large numerical values or executing intricate mathematical calculations (Qian et al., 2022; She et al., 2023; Manikandan et al., 2023; Liang et al., 2023b; Mialon et al., 2023). Thus equipping LLM agents with the capability to utilize tools has been increasingly focused on. Commonly used methods mainly relied on human-curated data for training (Parisi et al., 2022; Nakano et al., 2022; Qin et al., 2023c; Song et al., 2023b) or prompt designing (Cai et al., 2023; Shen et al., 2023a; Hao et al., 2024). Recently, distillation-based methods are also proposed (Schick et al., 2023; Zhang, 2023; Patil et al., 2023; Tang et al., 2023a; Qin et al., 2023a; Yuan et al., 2023a; Gao et al., 2023b; Wang et al., 2024; Shen et al., 2024; Yuan et al., 2024b).

Toolformer (Schick et al., 2023) utilizes a self-supervised manner, avoiding large human annotations, to obtain the most required APIs to use and further distill this capability to the model itself. The performance of the GPT-J-based Toolformer surpasses OPT (66B) (Zhang et al., 2022) and GPT3 (175B) (Brown et al., 2020) greatly. Graph-ToolFormer (Zhang, 2023) aims to equip LLMs with the ability to process and reason over complex graph data, which is designed to enhance LLMs with graph reasoning skills using external graph reasoning API tools by adopting ChatGPT to annotate and augment a larger graph reasoning statement dataset for training. Gorilla (Patil et al., 2023) addresses the limitations of current LLMs in generating accurate input arguments and reduces the problem of "hallucination" or generating incorrect API usage and it collects thousands of models from platforms like HuggingFace and Torch Hub as the API calls and utilizes GPT4 to generate synthetic instruction data for training. GPT4Tools (Yang et al., 2023b) introduces to enable open-source LLMs like LLaMA and OPT to use multimodal tools, a capability previously limited to advanced proprietary models like ChatGPT and GPT-4. The approach involves generating an instruction-following dataset by prompting an advanced teacher model with multimodal contexts, using the Low-Rank Adaptation optimization. ToolAlpaca (Tang et al., 2023a) proposes a framework

aimed at enhancing the tool-use capabilities of compact language models for embodied intelligence. It creates a dataset with 3938 instances from over 400 real-world tool APIs across 50 categories and utilizes ChatGPT to generate documentation for each prompt for later training. ToolLLM (Qin et al., 2023a) proposes a comprehensive framework for enhancing LLMs with tool-use proficiency, focusing on data creation, model training, and evaluation by distilling from chatGPT. Their ToolLLaMA shows impressive performance in executing complex instructions and handling new APIs, rivaling ChatGPT. CRAFT (Yuan et al., 2023a) builds a general tool creation and retrieval framework, which utilizes GPT4 to generate code snippets as the created tools. During the inference, other small LLMs could select and retrieve from the generated code snippets to execute or generate other methods conditioned on the given snippets. Confucius (Gao et al., 2023b) introduces a tiered training strategy for LLMs to master tool usage through a graduated curriculum and an innovative method called Iterative Self-instruction from Introspective Feedback (ISIF) for dynamic dataset enhancement to handle complex tools. MLLM-Tool (Wang et al., 2024) is a multi-modal tool agent capable of interpreting instructions embedded in visual or audio content through the integration of multi-modal encoders with open-source large language models. As a trainable method, the initial instruction-answer pairs are generated by utilizing GPT4. Shen et al. (2024) demonstrate that small LLMs are weak tool learners and proposes a multi-LLM framework that decomposes the tool-use ability of a single model into a planner, caller, and summarizer for the tool using, leading to a supreme performance. The two-stage training strategy introduced by this work is powered by ChatGPT and GPT4 for collecting execution trajectories for the training set. Yuan et al. (2024b) notice the potential issue of the current lengthy tool documentation, which hinders LLMs from understanding how to utilize a tool, thus proposing EASYTOOL to purify the important information from extensive documentation. The ground truth summarization of the training documents is obtained by using ChatGPT.

4.3.2 Planning

Another important aspect for LLM agents is the ability to decompose high-level tasks to a chosen set of actionable steps (Huang et al., 2022b), which is especially useful when acting in interactive environments. Huang et al. (2022b) first demonstrate that LLMs can generate plausible goal-driven action plans without training, introduces non-invasive tools to enhance model executability, and assesses these methods through human evaluation to balance executability and semantic accuracy. Most existing methods utilize prompting strategies for task planning (Singh et al., 2022; Zhou et al., 2023b; Song et al., 2023c; Wang et al., 2023g; Yao et al., 2023b; Liu et al., 2023j; Hao et al., 2023; Hu et al., 2023a), or building human-curated data for training (Lin et al., 2023a; Valmeekam et al., 2023). Recently, there have also been some distilling methods emerging (Chen et al., 2023b; Zeng et al., 2023a; Yin et al., 2023a; Qiao et al., 2024; Kong et al., 2023).

FireAct (Chen et al., 2023b) introduces an innovative approach for refining LLMs. This method involves fine-tuning smaller-scale LLMs using agent trajectories that are derived

from a variety of tasks and prompting techniques. Applying this method with trajectories generated by GPT4 has been shown to consistently enhance performance. AgentTuning (Zeng et al., 2023a) aims to enhance the performance of LLMs in executing agent tasks without sacrificing their wide-ranging capabilities. By utilizing a new dataset called AgentInstruct, which includes high-quality interaction trajectories, it applies a hybrid instruction-tuning approach that merges these trajectories with general domain instructions. Lumos (Yin et al., 2023a) pertains to a novel framework designed to train agents using a unified data format and modular architecture based on open-source LLMs. This system comprises three key modules: planning, grounding, and execution, enabling the decomposition of tasks into subgoals and actionable steps. TPTU-v2 (Kong et al., 2023) focuses on improving the task planning and tool usage abilities of LLMs in real-world scenarios, by utilizing data generated by human experts or LLMs. It introduces a framework comprising three components: an API Retriever, an LLM Finetuner, and a Demo Selector. AUTOACT (Qiao et al., 2024) proposes an agent learning framework that does not require large-scale annotated data or synthetic trajectories from high-resource models like GPT-4. Instead, it uses a self-instruct method to generate its own planning trajectories with limited initial data. It then applies a division-of-labor strategy, creating sub-agents specialized in different aspects of the task completion process.

Distillation also works out for the training of embodied multi-modal agents (Sumers et al., 2023; Yang et al., 2023c; Ma et al., 2023a; Du et al., 2023a; Sumers et al., 2023). For instance, Sumers et al. (2023) aim to enhance the ability of AI agents to follow instructions by using pretrained vision-language models to provide supervision for understanding and acting upon language within their operational environment, leveraging model distillation and hindsight experience replay to teach them contextually relevant interactions in a simulated 3D setting. Emma (Yang et al., 2023c) evaluates the challenges and inefficiency of training an embodied agent in a noisy visual world without expert guidance, and proposes to train them in a simulated environment using imitation learning, guided by an expert Language Model (like ChatGPT), which operates in a corresponding text-based simulation, focusing on the same tasks.

4.4 NLP Task Specialization

NLP tasks often grapple with challenges like data scarcity, interpretability issues, privacy concerns, and noisy data. The “Knowledge” section of our survey illustrates various methods for distilling knowledge from LLMs, effectively setting the stage for student models to adapt to a range of NLP tasks. This knowledge provides supervision for the training of student models through information augmentation (e.g., CoT and explanation), data augmentation, and semantic representation. By transferring the distilled knowledge from LLMs, student models can better handle diverse NLP challenges, improving task performance and addressing data limitations more robustly.

4.4.1 Natural Language Understanding

Natural Language Understanding (NLU) is a fundamental NLP task that involves comprehending and interpret-

ing human language. The knowledge distilled from LLMs, such as through data labeling or augmentation, is typically transferred into encoder-based language models like BERT (Vaswani et al., 2017) and RoBERTa (Liu et al., 2019).

Regarding the task of classification, certain studies have been noteworthy (Dai et al., 2023a; Gilardi et al., 2023; He et al., 2023b; Gao et al., 2023a; Chenglin et al., 2023; Li et al., 2023g). AugGPT (Dai et al., 2023a) focuses on both general and clinical domain text classification. To address the limitations of small-scale clinical datasets, which often lack expert annotation and are subject to stringent privacy regulations, AugGPT utilizes knowledge from teacher LLMs to rephrase each sentence in the training samples. This process creates multiple conceptually similar but semantically distinct samples, enhancing the dataset’s richness and diversity. Another approach is demonstrated by Gilardi et al. (2023), who employ ChatGPT as an annotator to categorize inputs. This method has been shown to outperform crowd-workers in several tasks, including relevance, stance, topics, and frame detection. Furthermore, He et al. (2023b) propose *Targeted Data Generation* (TDG), a novel approach for identifying challenging subgroups within a dataset. TDG leverages LLMs, along with human-in-the-loop, to generate new data specifically tailored for these subgroups, thereby enriching the dataset and improving model performance in sentiment analysis and natural language inference tasks. To facilitate the clinical information extraction task, Tang et al. (2023b) elicit diverse samples from LLMs by providing examples and different seeds of clinical entities, i.e. the *Curation* manner.

Several studies have also focused on multiple NLU tasks (Ding et al., 2023a; He et al., 2023a; Wang et al., 2021a; He et al., 2022; Ye et al., 2022; Meng et al., 2022). For example, He et al. (2023a) utilize the knowledge in GPT-3.5 to annotate inputs with labels and explanations for various NLU tasks, including user input and keyword relevance assessment, BoolQ, and WiC. Wang et al. (2021a) employ few-shot prompts to expand high-quality training data using GPT-3, i.e. the *Expansion* manner. Beyond merely employing a single approach to elicit NLP task knowledge, Ding et al. (2023a) explore a combination of *Labeling*, *Expansion*, and *Curation* methods to extract knowledge from GPT-3 for distilling data for both sequence- and token-level NLP tasks.

4.4.2 Natural Language Generation

Natural Language Generation (NLG) is a key aspect of evaluating the capabilities of LLMs, encompassing tasks such as summarization, machine translation, and other open-ended text generation tasks. Known for their potent generative abilities and creativity, LLMs excel in these areas, making them prime sources for distilling knowledge into student models tailored for NLG tasks (Xu et al., 2023c, 2024b; Ramnath et al., 2023; Agarwal et al., 2024). Additionally, the knowledge distilled from LLMs can be effectively used for NLG task-specific data augmentation (Jung et al., 2023; Wang et al., 2021b; Guo et al., 2023a; Yang and Nicolai, 2023; Wang et al., 2023h; Yang et al., 2023d). While the previous sections have focused on the works about open-ended generation and multi-turn dialogue, this part will

specifically highlight the distillation techniques relevant to other NLG tasks.

Although automatic metrics often favor smaller, fine-tuned models in summarization tasks, human evaluators tend to prefer the summaries generated by LLMs. Addressing this discrepancy, Xu et al. (2023c) develop a student summarization model by distilling a GPTSUMM dataset, which comprises over 4 million paragraph-summary pairs generated by querying GPT-3.5. In a different approach, Jung et al. (2023) introduce ‘Impossible Distillation,’ a method that creates high-quality summarization-specific dataset from weak teacher LLMs. This method involves training a student model on the generated dataset and enhancing its capabilities through Self-Knowledge. Turning to the task of machine translation, where creating parallel corpora is traditionally expensive and time-consuming, Yang and Nicolai (2023) propose a three-step distillation process. This process involves generating seeds of verbs and nouns, forming sentences, and then translating these sentences. Their findings suggest that while the distilled dataset may lack diversity, it effectively improves the translation signal for training student translation models. To distill high-quality content-grounded data automatically, Genie (Yehudai et al., 2024) proposes a general methodology containing three key steps: (a) preparation of the content, (b) distillation of responses from a teacher LLM corresponding to the content, and (c) filtering mechanism to ensure the quality and faithfulness of the generated data. Genie demonstrates that student models trained through this distilled data can match or even surpass models trained on human-generated data.

4.4.3 Information Retrieval

Information Retrieval (IR) represents a crucial branch of computer science, focused on efficiently retrieving information relevant to user queries from extensive repositories (Cai et al., 2022; Liu et al., 2022b; Feng et al., 2023; Shen et al., 2023b). A typical IR system encompasses three main components: the query rewriter, the retriever, and the reranker. Recent studies have highlighted the effectiveness of employing LLMs in IR systems, e.g. in enhancing the reranking stage through both point-wise and list-wise ranking methods (Ma et al., 2023b; Sun et al., 2023a; Qin et al., 2023d). However, the practical application of LLMs in IR systems faces challenges, primarily due to their slower generation speed, which conflicts with the low-latency requirements of IR tasks (Sun et al., 2023a). As a result, the KD of LLMs emerges as a more promising approach for IR, offering a way to infuse the distilled knowledge from LLMs into various stages of the IR pipeline without compromising on speed. There has been a significant body of work demonstrating how knowledge distilled from LLMs can benefit each component of the IR system, including the *Query Rewriter* (Srinivasan et al., 2022; Ma et al., 2023c), the *Retriever* (Dai et al., 2023b; Sachan et al., 2022, 2023; Schick and Schütze, 2021; Meng et al., 2023; Peng et al., 2023b), and the *Reranker* (Bonifacio et al., 2022; Sun et al., 2023a; Pradeep et al., 2023a,b; Saad-Falcon et al., 2023; Ferraretto et al., 2023; Jeronymo et al., 2023; Sun et al., 2023c).

Query Rewriter. The Query Rewriter (QR) is a pivotal component in IR systems, tasked with enhancing the precision

and expressiveness of user queries by refining or modifying the initial query to more accurately align with the user’s information needs. One notable approach is QUILL (Srinivasan et al., 2022), which introduces a two-stage distillation method for query intent understanding. Initially, a retrieval-augmented LLM, serving as the ‘professor,’ is distilled into a non-retrieval augmented teacher LLM, aiming to bolster its understanding capabilities. Subsequently, this enhanced teacher LLM is distilled into a final student model using a large dataset, further refining the process. Incorporating the QR into IR systems, Ma et al. (2023c) develop a ‘Rewrite-Retrieve-Read’ framework. This process begins with an LLM rewriting the queries via prompting, followed by a retrieval-augmented reading stage. To integrate the rewritten queries effectively into the IR system, the knowledge gleaned from the LLM is distilled into a compact student rewriter. This rewriter is then fine-tuned using feedback from the LLM reader through reinforcement learning.

Retriever and Reranker. In IR systems, the Retriever is designed to efficiently locate the top-k relevant texts from a large corpus. It encodes both queries and documents into vector representations and performs retrieval by computing the dot product between these vectors. The Reranker further refines the order of the retrieved documents to improve the overall quality of the output. This is achieved in two primary ways, including *Pointwise Reranker* and *Listwise Reranker*. Pointwise Reranker takes both the query and a single candidate document as input to directly generate a relevance score. Listwise Reranker directly reorders a list of input documents in terms of their relevance.

Retriever and Pointwise Reranker. For the retriever and pointwise reranker, a common application of KD from LLMs is the generation of pseudo-queries for given documents. This approach aims to expand the pairwise data, enhancing the training of dense retrievers or rerankers. For example, InPars (Bonifacio et al., 2022) utilizes GPT-3 to generate multiple pseudo-queries for an unlabeled document. To ensure the relevance of these queries, the system filters them based on the highest log probabilities of generating a query conditioned on the documents. Subsequently, InPars fine-tunes a reranker based on monoT5 (Raffel et al., 2020). Another similar approach, Promptagator (Dai et al., 2023b), introduces a few-shot dense retrieval method that leverages a small number of demonstrations from the target domain for pseudo-query generation. Diverging from the reliance on unlabeled documents, Sachan et al. (2022) distill knowledge from GPT-4 to curate diverse synthetic data for text embedding tasks across nearly 100 languages. They fine-tune powerful decoder-only LLMs, such as Mistral-7b (Jiang et al., 2023a), on this synthetic data using standard contrastive loss. Remarkably, this method demonstrates strong performance on text embedding and multilingual retrieval benchmarks without any labeled data. Beyond generating pseudo-queries, teacher LLMs can also be employed to generate relevance scores as soft labels. These scores are used to train the retriever by minimizing the KL-divergence loss between the teacher and student distributions, as explored by Sachan et al. (2023).

Listwise Reranker. A distinct set of studies focuses on listwise reranking, where its advantage lies in compar-

ing multiple documents simultaneously to determine the optimal reorder. RankGPT (Sun et al., 2023a) leverages GPT-4 to generate permutations for a group of candidate passages. To distill this listwise ranking knowledge into a pointwise student reranker, various training loss functions are employed, such as Listwise Cross-Entropy (Bruch et al., 2019), RankNet (Burges et al., 2005), and LambdaLoss (Wang et al., 2018). Building upon RankGPT’s framework, RankVicuna (Pradeep et al., 2023a) and RankZephyr (Pradeep et al., 2023b) further refine this approach by directly fine-tuning a listwise reranker using teacher-generated textual permutations. This enables the student reranker to produce sequences of ranked results directly, bypassing the intermediate step of calculating individual relevance scores.

4.4.4 Recommendation

Recommender systems are integral to enhancing user experience in various online services, providing personalized content based on user preferences and behaviors. Many works have demonstrated that LLMs could be directly used as recommenders without fine-tuning (Wang et al., 2023i; Dai et al., 2023c) or generate auxiliary textual features to benefit recommender systems (Xi et al., 2023; Ren et al., 2023; Wei et al., 2024). (Wang et al., 2023j; Ren et al., 2023; Wei et al., 2024). However, the real-time nature of online recommender systems demands rapid response times, posing a challenge with the inherent inference latency associated with LLMs. To address this, several studies have explored ways to distill and integrate the knowledge from LLMs into recommender systems, thereby leveraging their advanced capabilities while mitigating latency issues for efficient real-time recommendations (Mysore et al., 2023; Zhang et al., 2023b; Liu et al., 2023c).

Mysore et al. (2023) tackle data scarcity in narrative-driven recommendation (NDR), where users provide detailed descriptions of their preferences. They utilize GPT-3 to create synthetic narrative queries from user-item interactions via few-shot prompting, then distill this data into retrieval models for NDR. Similarly, GENRE (Liu et al., 2023c) employs GPT-3.5 to augment datasets with new knowledge about news summarization, user profiles, and personalized content, aiding the training of content-based recommendation models. To bridge the gap between language models and recommender systems, some research views behavior modeling as an extension of language modeling (Cui et al., 2022; Liu et al., 2023k). InstructRec (Zhang et al., 2023b), for instance, interprets recommendation as instruction following. They use ChatGPT to distill a wealth of user-personalized instruction data reflecting diverse preferences and intentions based on real historical interactions. This data is then used to fine-tune a 3B student language model specifically for recommendation purposes.

4.4.5 Text Generation Evaluation

Text generation evaluation, i.e. NLG evaluation, focuses on assessing the quality of generated content. Unlike traditional NLG evaluation metrics like BLEU (Papineni et al., 2002) or ROUGE (Lin, 2004), which primarily rely on surface-level text comparisons, LLMs, trained on extensive corpora and refined through techniques like RLHF, offer a more human-aligned assessment. This sophistication has led

to the increasing use of LLMs in NLG evaluation (detailed further in (Li et al., 2024b)). Through KD of LLMs, student evaluators could enhance inference efficiency and achieve more flexible and highly customized evaluation (Wang et al., 2023b; Kim et al., 2024; Xu et al., 2023d; Jiang et al., 2023c; Li et al., 2024a).

PandaLM (Wang et al., 2023b) concentrates on a pairwise evaluator designed to compare two pieces of generated content. It utilizes a teacher LLM (GPT-3.5) to judge which response is better for a given instruction and input, providing reasons for its decision. Addressing the need for customized and flexible criteria to meet realistic user demands, Prometheus (Kim et al., 2024) distills GPT-4 to construct a training dataset that includes reference answers and a variety of customized scoring rubrics. This dataset is then used to tune LLaMA for evaluating model-generated responses. Instructscore (Xu et al., 2023d) takes a more fine-grained approach by using GPT-4 to create detailed analysis data. This data is employed to tune LLaMA, enabling it to perform error analysis on generated texts compared to reference texts. The system further refines its evaluation capabilities through self-training with real model-generated response-reference pairs. For reference-free evaluation across diverse domains, TigerScore (Jiang et al., 2023c) samples data from a variety of text generation datasets, such as summarization, translation, and data-to-text. It distills error analysis knowledge from GPT-4 and uses this to fine-tune LLaMA. Lastly, to adapt evaluation to real-world scenarios beyond conventional NLP tasks, Auto-J (Li et al., 2024a) collects real-world user queries and their evaluations from a teacher LLM. This massive dataset of real-world scenarios is then used to distill evaluation knowledge into LLaMA through fine-tuning, enhancing its practical applicability.

4.4.6 Code

LLMs, trained on extensive corpora containing code, are highlighted for their proficiency in code-related tasks. Their capabilities extend beyond direct code generation to include the provision of external knowledge and data, which is crucial in distilling their expertise into smaller, more efficient models. Several works have successfully distilled code knowledge from LLMs into those compact and specialized code models (Chaudhary, 2023; Rozière et al., 2023; Gunasekar et al., 2023; Wei et al., 2023; Chen et al., 2023a; Liu et al., 2023d; Yu et al., 2024; Jain et al., 2023; Su and McMillan, 2023; Guo et al., 2023d).

A primary focus in these student code models is on code generation, a task of both common utility and practical significance. For instance, Code Alpaca (Chaudhary, 2023) fine-tunes Llama using self-instruct with ChatGPT-distilled instructions specifically for code generation tasks. Similarly, Code Llama-instruct (Rozière et al., 2023) is fine-tuned via self-instruct, prompting Llama-2 (Touvron et al., 2023) with coding problems, and further refined with unit tests. Phi-1 (Gunasekar et al., 2023) aims to enhance the quality of distilled code data by extracting “textbook quality” data from a teacher LLM, incorporating Python textbook and exercise data. Magicoder (Wei et al., 2023) addresses potential biases in teacher LLMs by referencing a wealth of open-source code, yielding more diverse and grounded data for code generation. To consider the capability of the student model

and leverage the feedback of the teacher, PERsD (Chen et al., 2023a) introduces a Personalized Distillation method where the teacher LLM refines the student’s generated code based on the execution feedback of the executor.

However, these models primarily target the code generation task, lacking generalizability across a broader range of code-related tasks. To address this issue, MFTCoder (Liu et al., 2023d) utilizes self-instruct to distill diverse code data from teacher LLMs for various tasks, such as code completion and text-to-code generation, training a student model via multi-task learning. WaveCoder (Yu et al., 2024), in contrast, creates a comprehensive instruction tuning dataset covering four universal code-related tasks distilled from GPT-3.5-turbo. WaveCoder first selects a diverse coreset of raw data using the KCenterGreedy (Sener and Savarese, 2018) clustering method, then employs the teacher LLM for generating task definitions and outputs. The teacher model also plays a role in evaluating and filtering this data. Notably, WaveCoder demonstrates superior generalization across different code-related tasks compared to other open-source models.

4.5 Multi-Modality

Multimodal Large Language Models (MLLMs) surpass traditional language-only LLMs by understanding and processing information across multiple modalities, more closely mirroring human perception and enabling a broader range of real-world applications. There is a growing trend towards developing MLLMs that follow multimodal instructions, facilitating tasks with enhanced levels of interactivity. To address the scarcity of multimodal instruction-following data and to harness the commonsense and world knowledge embedded in teacher LLMs, numerous studies have focused on multimodal knowledge distillation from LLMs (Liu et al., 2023e; Zhao et al., 2023b; Wang et al., 2023e; Chen et al., 2023c; Park et al., 2023; Pi et al., 2023; Zhao et al., 2023c; Liu et al., 2023f; Wu et al., 2023b; Luo et al., 2023d; Jiang et al., 2023d; Li et al., 2023c; Xu et al., 2023e).

Vision-Language. In the vision-language domain, LLaVA (Liu et al., 2023e) pioneers the extension of the Self-Instruct approach from the language to the multimodal field. It translates images into textual descriptions, including captions and bounding boxes, and distills GPT-4 for generating new data in the context of seed examples. This approach creates a LLaVA-Instruct-150k dataset, which serves as the foundation for further developments like LLaVA-1.5 (Liu et al., 2023l) and GPT4ROI (Zhang et al., 2023e), enhancing the instruction-following capabilities of MLLMs. To expand the dataset’s scale, SVIT (Zhao et al., 2023b) introduces a 4.2 million image dataset, distilled from GPT-4 by leveraging manual image annotations. It employs a novel data recipe to select an informative, diverse, and balanced subset of training data. LVIS-Instruct4V (Wang et al., 2023e) leverages GPT-4V (OpenAI, 2023), a powerful large multimodal model, as a teacher to distill a more accurate and context-aware instruction-following dataset, focusing on fine-grained understanding. Further advancements include integrating specific region referencing in image-based instruction following. For instance, Shikra (Chen et al., 2023c) uses

GPT-4 to distill referential question-answer pairs from the Flickr30K (Plummer et al., 2015) dataset, enhancing the understanding of referential regions within images. LSKD (Park et al., 2023) introduces localized references to specific image regions, prompting the teacher LLM to generate commonsense inferences about these areas. To enhance the visual instruction tuning pipeline with text-rich images, LLaVAR (Zhang et al., 2023d) employs the text-only GPT-4 as a teacher, using recognized texts and image captions to generate 16K conversation pairs for text-rich images. The resultant student MLLM demonstrates enhanced interaction skills in content that combines both text and imagery.

Multiple Modalities. To extend knowledge distillation of LLMs to encompass more modalities, such as audio and video, several innovative approaches have been introduced. These methods typically involve transforming these modalities into a textual format comprehensible to teacher LLMs, followed by the distillation of the teacher. Macaw-LLM (Lyu et al., 2023) leverages GPT-4 to generate instruction-response pairs corresponding to the content of images or videos. MIMIC-IT (Li et al., 2023f) aims to broaden the scope to language, image, and video understanding, creating a substantial dataset with 2.8 million multimodal instruction-response pairs distilled from ChatGPT. Chat-Bridge (Zhao et al., 2023d), on the other hand, represents a novel approach in multimodal language modeling. It translates various non-textual modalities into text, combining fine-grained and global descriptions. This information is then used to distill responses from ChatGPT or GPT-4 through an in-context learning process, effectively bridging the gap between different modalities.

Others. Beyond distilling instruction-following data, several methods have emerged that concentrate on harnessing different aspects of knowledge from LLMs. For instance, EMMA (Yang et al., 2023c) trains an MLLM to act as an embodied reflex agent within a visual environment. It achieves this by distilling GPT-4’s skills in a parallel textual world, generating actions and providing reflective feedback. Silkie (Li et al., 2023h) takes a unique approach by distilling preferences from GPT-4V, focusing on criteria like helpfulness and visual faithfulness. Ha et al. (2023) represent another innovative direction, where it generates, labels, and distills diverse robot-centric exploration experiences by LLMs into a multi-task visuo-linguo-motor policy.

5 DOMAIN-SPECIFIED VERTICAL DISTILLATION

This section shifts from skill distillation to examine KD of LLMs in various vertical domains, including Law, Medical & Healthcare, Finance, and Science, etc. It delves into customizing distilled LLMs for these fields, showing its significant role in enhancing domain-specific AI applications. The taxonomy of these works is shown in Figure 7.

5.1 Law

Law holds a crucial position in molding societies, overseeing human interactions, and ensuring justice prevails. Informed decision-making, legal interpretation, and the provision of legal advice by professionals hinge on precise

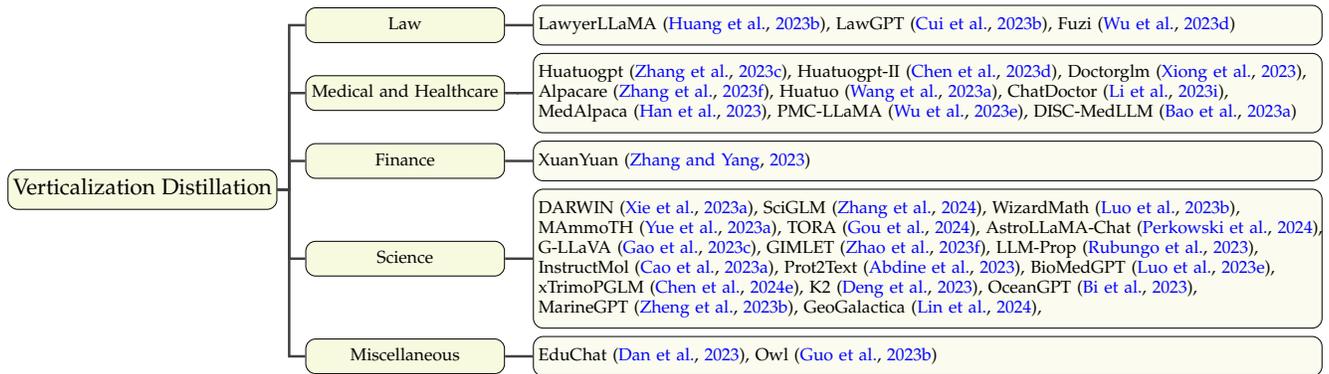


Fig. 7: Taxonomy of Verticalization Distillation.

and current information. Legal intelligent applications in different scenarios usually require combinations of multiple fundamental capabilities of legal text retrieval, understanding, reasoning and generating (Zhang et al., 2023g; Sun, 2023; Lai et al., 2023). To address challenges like legal terminology, subtle interpretations, and the constant evolution of legislation presents distinctive challenges that demand customized resolutions. To handle the above challenges, several studies have investigated the customization of LLMs for intelligent legal services (Cui et al., 2023b; Yue et al., 2023b; Huang et al., 2023b; Wu et al., 2023d). This involves a continued pre-training process on extensive legal corpora, followed by fine-tuning with self-constructed instructions or augmented data using advanced LLMs.

Huang et al. (2023b) have unveiled a Chinese legal large model named LawyerLLaMA. The model undergoes an initial pre-training phase on an extensive legal corpus, systematically assimilating knowledge of the Chinese legal system. Subsequently, fine-tuning occurs through the analysis of objective questions from the Chinese National Judicial Examination (Zhong et al., 2020) and the gathering of responses to legal consultations using ChatGPT. This process equips the model with the ability to apply legal knowledge to specific scenarios. Cui et al. (2023b) present LawGPT, built upon the foundation of OpenLLAMA. The model is trained using a construction process that incorporates real-world legal text, legal regulations, judicial interpretations, and actual legal consultation data. Additionally, the authors utilize the ChatGPT API for assisted construction, enabling the generation of supplementary data derived from the existing dataset. Wu et al. (2023d) have developed a large-scale Chinese legal model (named Fuzi) with ChatGLM as its foundation. This model undergoes training on an extensive Chinese legal corpus, which incorporates unsupervised judicial language data, including diverse judgment documents and legal regulations. Additionally, it undergoes supervised judicial fine-tuning with data encompassing legal QA and case retrieval. Fuzi’s training also involves both general instruction fine-tuning datasets, such as Alpaca, and domain-specific instruction fine-tuning datasets from LawyerLLaMA (Huang et al., 2023b) and LawGPT (Cui et al., 2023b).

5.2 Medical and Healthcare

The integration of LLMs holds great potential for transforming medicine and healthcare. Extensive research has focused on adapting general-purpose LLMs to the medical domain (Singhal et al., 2023), such as electronic health records, and healthcare applications like patient care (Zhu et al., 2023). Recent work has focused on enhancing medical instruction-following data with advanced teacher LLMs to better align with complex user instructions. Given the abundance of medical data, most studies combine real-world data with distilled instruction data from teacher LLMs (Zhang et al., 2023c; Xiong et al., 2023; Zhang et al., 2023f; Wang et al., 2023a; Li et al., 2023i; Han et al., 2023; Wu et al., 2023f; Bao et al., 2023a; Chen et al., 2023d).

While existing studies predominantly concentrate on training using dedicated medical dialogue datasets comprising medical textbooks (Wu et al., 2023e), biomedical papers (Luo et al., 2023e) medical knowledge-graphs (Bao et al., 2023b), or authentic doctor-patient interactions (Bao et al., 2023b), an expanding body of research is delving into the augmentation of medical instruction-following data with advanced LLMs to enhance the alignment with practical user instructions. Zhang et al. (2023c) introduce HuatuoGPT specifically tailored for medical consultations. The model leverages both *distilled data from ChatGPT* and *real-world data from doctors* during the supervised fine-tuning stage. In a parallel effort, Xiong et al. (2023) construct a dataset of medical dialogues in Chinese, employing ChatGPT’s assistance. Their methodology encompassed various techniques to train DoctorGLM, an easily deployable LLM designed for tasks such as diagnoses, drug recommendations, and other medical advice. Zhang et al. (2023f) fine-tune LLaMA-series models using 52k diverse, machine-generated, medical instruction-following data named MedInstruct-52k. This effort resulted in the development of AlpaCare, a model demonstrating robust medical proficiency and generalizability across both general and medical-specific domain free-form instruction evaluations. In a different vein, Wang et al. (2023a) propose HuaTuo, a LLaMA-based model that undergoes supervised fine-tuning with generated QA instances. This refinement process enhances the model’s possession of more reliable medical knowledge. Li et al. (2023i) introduce ChatDoctor, which was first trained as a generic conversation model based on LLaMA. It utilized 52K instruction-following data

from Stanford University’s Alpaca project (Taori et al., 2023). Subsequently, the conversation model underwent fine-tuning on a dataset of 100K patient-physician conversations collected from an online medical consultation website. This two-step training process underscores the model’s adaptability to diverse conversational contexts, particularly those specific to patient-physician interactions.

Built upon existing datasets, MedAlpaca (Han et al., 2023) proposes to reconstruct the data with GPT-3.5-Turbo, which is then used to fine-tune LLMs for effective medical applications. Furthermore, PMC-LLaMA (Wu et al., 2023f) proposes a training framework (i.e., continual pre-training and domain-specific multi-task supervised fine-tuning) to adapt a general LLM to the medicine domain, where GPT-4 is leveraged to write synonymous sentences for data augmentation in the SFT. To adapt LLMs to real-world medical consultation, DISC-MedLLM (Bao et al., 2023a) leverages GPT-3.5 to 1) construct 50K QA pairs in a few-shot manner and 2) re-generate the 420k dialogues based on real cases, which are then used to train LLMs in a supervised fine-tuning manner. More recently, HuatuoGPT-II (Chen et al., 2023d) proposes a one-stage training with instruction-formatting unification of domain data collection for medical adaption upon LLMs, where GPT-4 is used to formulate medical questions to fine-tuning instructions.

These diverse studies collectively contribute to the advancing field of the medical domain, facilitated by knowledge distillation from advanced LLMs. Through the exploration of various methodologies, these approaches provide valuable insights into the challenges and potential breakthroughs at the intersection of cutting-edge language models and medical applications.

5.3 Finance

The application of LLMs to the finance domain (Xue et al., 2023) significantly transforms how financial data is analyzed, decisions are made, and customer interactions are managed. In finance, LLMs offer unprecedented capabilities in understanding complex financial documents, predicting market trends, and automating risk assessment, thus enabling more informed and faster decision-making processes. By processing and analyzing vast amounts of unstructured financial data, such as news articles, reports, and real-time market feeds, LLMs can identify patterns and insights that were previously inaccessible, leading to more accurate forecasts and strategic financial planning. Furthermore, LLMs enhance customer experiences through personalized financial advice, automated customer service, and sophisticated chatbots that can handle complex queries. This level of automation and insight has the potential to increase efficiency, reduce operational costs, and improve compliance and risk management practices in financial institutions, making LLMs a transformative force in the finance sector. Knowledge distillation from a proprietary LLM is still under-explored, and most existing works focus on adapting LLMs to finance applications by continual pre-training on finance-specific corpora (Wu et al., 2023g; Lu et al., 2023) or fine-tuning in a supervised manner on multi-task finance-specific instructions (Yang et al., 2023e; Xie et al., 2023b; Wang et al., 2023k).

Specifically, XuanYuan (Zhang and Yang, 2023) leverages self-instruct over seed data and self-QA over structured/unstructured data to generate instruction data in the finance domain, which is used to train a finance LLM.

5.4 Science

The integration of LLMs into the science domain (Taylor et al., 2022; Yin et al., 2023b) represents a paradigm shift in research, knowledge discovery, and the dissemination of scientific information. In science, LLMs are leveraged to digest and synthesize vast amounts of literature, aiding in the identification of new research opportunities and the acceleration of scientific breakthroughs. They facilitate the understanding of complex scientific concepts by summarizing research papers, generating hypotheses, and even drafting research proposals and manuscripts, thus significantly reducing the time researchers spend on literature review and enabling them to focus more on experimental work. LLMs also democratize access to scientific knowledge by providing layperson summaries of complex research findings, making science more accessible to non-experts and fostering a broader public understanding of scientific advancements. By enhancing the efficiency of research workflows and fostering interdisciplinary collaborations, LLMs are poised to accelerate the pace of scientific discovery and innovation across various fields. To distill knowledge from an LLM, DARWIN Series (Xie et al., 2023a) utilizes a semi self-instruct for instruction generation for science papers, which is then used to fine-tune an LLM. SciGLM (Zhang et al., 2024) proposes to train a scientific LLM, which prompts a teacher LLM to generate detailed answers for unlabelled scientific questions, as well as a self-reflective critic-and-revise to improve data quality. Besides the above knowledge distillation methods to adapt LLMs to science, we will also delve into how the distillation happens in sub-domains, e.g., mathematics, astronautics, chemistry, etc.

Mathematics. The application of LLMs within the sub-domain of mathematics heralds a transformative era in mathematical research, education, and problem-solving (Azerbaiyev et al., 2023; Yu et al., 2023b). LLMs in mathematics facilitate the exploration and understanding of complex mathematical theories and problems by providing intuitive explanations, proofs, and solutions that can bridge the gap between advanced mathematical concepts and learners at various levels. These models have shown potential in conjecturing new mathematical theorems and patterns, thus opening new avenues for research and discovery that might not have been readily accessible to humans alone. In education, they serve as personalized tutors, offering students step-by-step guidance through mathematical problems and adapting explanations to the learner’s level of understanding. This democratizes access to high-quality mathematical education and fosters a deeper appreciation and understanding of mathematics among a broader audience. By enhancing collaborative efforts through the generation of new ideas and the simplification of complex concepts, LLMs are poised to significantly advance the field of mathematics, making it more accessible, efficient, and innovative. WizardMath (Luo et al., 2023b) enhances the mathematical reasoning capabilities of Llama-2 by applying the

novel Reinforcement Learning from Evol-Instruct Feedback (RLEIF) method, significantly outperforming other open-source LLMs on the GSM8k and MATH benchmarks, as well as surpassing several closed-source LLMs including ChatGPT-3.5 and Minerva. MAMmoTH (Yue et al., 2023a) is a series of open-source LLMs specifically developed for general math problem-solving, achieving superior performance on nine mathematical reasoning datasets. Utilizing a novel instruction tuning dataset called MathInstruct, which combines chain-of-thought and program-of-thought rationales, MAMmoTH models demonstrate substantial improvements over existing models. TORA (Gou et al., 2024), a series of Tool-integrated Reasoning Agents, significantly advances mathematical problem-solving by combining natural language reasoning with the use of external computational tools. It markedly outperforms existing open-source models on 10 mathematical reasoning datasets, showcasing notable improvements over both rationale-based and program-based approaches, and introduces innovative training techniques such as output space shaping to enhance model reasoning capabilities. G-LLaVA (Gao et al., 2023c) introduces a significant advancement in geometric problem-solving for LLMs by leveraging a multimodal approach that combines text and image data. This model, utilizing the Geo170K dataset comprising over 170,000 geometric image-caption and question-answer pairs, demonstrates remarkable improvements over GPT-4V on the MathVista benchmark.

Astronautics. The application of LLMs in astronautics (Nguyen et al., 2023) propels the field forward. AstroLLaMA-Chat (Perkowski et al., 2024) is an advancement of the AstroLLaMA model, leveraging a 7B-parameter LLaMA-2 model and targeted continual pre-training on a curated astronomy corpus to enhance performance in astronomy-focused question-answering. This model demonstrates significant improvements in specialized topic comprehension and introduces a chat-enabled version for the astronomy community, highlighting the effectiveness of domain-specific knowledge distillation in achieving superior performance on specialized topics.

Chemistry and Materials Science. The integration of LLMs into Chemistry and Materials Science has revolutionized the way researchers approach the discovery and development of new compounds and materials. By analyzing vast datasets and scientific literature, LLMs can predict the properties and behaviors of substances, significantly accelerating the innovation cycle.

GIMLET (Zhao et al., 2023f), Graph Instruction based MolecuLe zEro-shoT learning, is a novel approach to molecule property prediction that integrates graph and text data within a single language model framework, aiming to improve instruction-based zero-shot learning for molecular tasks. By leveraging a transformer mechanism with generalized position embedding and decoupled attention, GIMLET significantly outperforms traditional molecule-text baselines in zero-shot learning scenarios, demonstrating the model’s effectiveness in generalizing from instructions to a broad range of molecule-related tasks without prior explicit task-specific training. LLM-Prop (Rubungo et al., 2023), leveraging the T5 model, showcases how LLMs can outperform SoTA graph neural networks in predicting the

physical and electronic properties of crystalline solids from text descriptions. This approach underscores the potential of text-based methods in materials science, offering significant improvements in prediction accuracy while also contributing a benchmark dataset, TextEdge, to foster further research in this emerging field. InstructMol (Cao et al., 2023a) integrates multi-modal data, aligning molecular structures with natural language instructions for drug discovery tasks. Through a novel two-stage instruction-tuning approach, it significantly enhances performance in molecule-related tasks, establishing a reliable molecular assistant that outperforms existing LLMs and reduces the performance gap with specialized models. This demonstrates the value of multi-modal integration in developing versatile tools for complex domains like drug discovery.

Biology. In the field of Biology, particularly in the study of proteins, DNA, and RNA, LLMs are revolutionizing our understanding of the fundamental molecules of life. By analyzing vast datasets of biological sequences and structures, LLMs can predict the three-dimensional shapes of proteins, potential functions, and interactions at a scale and speed beyond traditional computational methods. This capability is critical for unraveling the complexities of biological systems, advancing drug discovery by identifying targets and designing molecules with high precision, and understanding genetic diseases through the interpretation of genomic variations.

Prot2Text (Abdine et al., 2023) introduces a novel multi-modal framework for generating protein function descriptions in free text by combining GNNs and LLMs. This approach, which integrates structural and sequential protein information, highlights the transformative impact of knowledge distillation through the fusion of GNNs and LLMs for accurate protein function prediction, potentially revolutionizing research in bioinformatics and biological sciences. BioMedGPT (Luo et al., 2023e) introduces a multimodal generative pre-trained transformer specifically designed for the biomedicine domain, emphasizing the significance of aligning molecular, protein, and natural language modalities to enhance biomedical question-answering, molecule, and protein QA tasks. This framework showcases the critical role of knowledge distillation in bridging the gap between complex biological data and human language, thereby facilitating groundbreaking advancements in drug discovery and therapeutic target identification. xTrimopGLM (Chen et al., 2024e), a unified 100B-scale pre-trained transformer model, addresses both protein understanding and generation tasks by integrating autoencoding and autoregressive pre-training objectives. Its significant advancements over existing models in 18 protein understanding benchmarks and its capability in de novo protein sequence generation highlight the model’s importance in advancing the field of protein science through knowledge distillation.

Geography, Geology, and Environmental Science. The integration of LLMs into Geography, Geology, and Environmental Science is revolutionizing these fields by enhancing data analysis, predictive modeling, and interdisciplinary research (Roberts et al., 2023; Lin et al., 2023b; Wang et al., 2023i).

K2 (Deng et al., 2023), the first-ever LLM specialized in the geoscience domain, demonstrates the significant impact

of knowledge distillation in vertical domain specialization. By adapting the general-domain LLaMA-7B model with a 5.5B token geoscience corpus and introducing the GeoSignal instruction tuning dataset, K2 showcases enhanced performance in geoscience knowledge understanding and utilization. The model’s development highlights a novel approach to efficiently gather domain-specific data and align model responses to specialized user queries. OceanGPT (Bi et al., 2023), introduced as the first LLM for ocean science tasks, underscores the vital role of knowledge distillation in the vertical domain of oceanography. It leverages DOINSTRUCT, a novel framework for generating domain-specific instruction data through multi-agent collaboration, and establishes OCEANBENCH, a benchmark for evaluating LLMs in the ocean domain. MarineGPT (Zheng et al., 2023b) showcases the transformative potential of knowledge distillation in the marine domain by leveraging a novel vision-language model tailored for marine science. Utilizing the Marine-5M dataset, which includes over 5 million marine image-text pairs, MarineGPT excels in providing detailed, accurate, and domain-specific responses. GeoGalactica (Lin et al., 2024) represents a pioneering step in specializing LLMs for geoscience, leveraging a 30 billion parameter model pre-trained on a vast geoscience corpus. This model is notable for being the largest of its kind within the geoscience domain.

5.5 Miscellaneous

Knowledge distillation of LLMs has vast potential across various verticals beyond the ones previously discussed, highlighting their versatility and transformative impact across different industries. For instance, in the education sector, EduChat (Dan et al., 2023) exemplifies a chatbot system that provides tailored support to teachers, students, and parents. KD is central to its design, leveraging pre-training on educational data followed by fine-tuning with custom instructions to deliver capabilities such as essay evaluation and emotional support. Similarly, Owl (Guo et al., 2023b), an LLM designed for IT operations, boosts operational efficiency using the Owl-Instruct dataset, which is distilled from ChatGPT. By applying a mixture-of-adapters strategy for domain-specific tuning, it enhances analysis and performance in IT-related tasks.

6 OPEN PROBLEMS

Further Data Selection How much data is required for LLM distillation and how to filter out the low-quality data remain open-domain questions. In the field of instruction tuning, one of the most commonly used methods for distillation, Zhou et al. (2023a) propose that only 1000 human-curated high-quality data is enough for the alignment of LLMs, hypothesizing that LLMs have learned the required knowledge from pretraining and only a small amount of data is required for the alignment. Its finding further raises a new question, how to automatically select the data for better distillation? Chen et al. (2023e) directly apply ChatGPT to rate each data sample together with explanations, and then the data is selected based on the rating. Cao et al. (2023b) split the existing instruction-tuning datasets and trains a

linear function to select the most effective data based on their statistical properties. Li et al. (2023j) propose a data selection pipeline similar to self-distillation, in which the LLM firstly learns from a small subset of the data to get the basic ability, and then further uses this learned model to rate for the original dataset. Du et al. (2023b) propose to consider three aspects including quality, coverage, and necessity for the filtering process. Li et al. (2023k) select instruction data by evaluating their one-shot improvement on a hold-out set. Li et al. (2024f) recently propose Superfiltering, which is able to utilize small language models like GPT2 to filter out the high-quality subset from a given high-quality dataset. Despite the emergence of these works working on data filtering, How to efficiently select the optimal distillation data for LLMs, and How much data is required for distillation are still unsolved.

Reduce the Distillation Cost (Lightweight Methods) Despite the remarkable abilities of the latest LLMs, their significant resource requirements underscore the urgent need to find efficient solutions to overcome these challenges. Common ways to further reduce the distillation cost include Model Compression and Efficient Fine-Tuning. In the realm of Model Compression, Quantization (Frantar et al., 2023; Dettmers et al., 2022; Kim et al., 2023c; Tao et al., 2022b; Yao et al., 2022; Xiao et al., 2023), Parameter Pruning (Ma et al., 2023d; Zhang et al., 2023h; Frantar and Alistarh, 2023), and Low-Rank Approximation (Xu et al., 2023g; Li et al., 2023l) are commonly utilized. In the realm of Efficient Fine-Tuning, Parameter Efficient Fine-Tuning (Hu et al., 2023b; Liu et al., 2022c; Wang et al., 2022b; Hu et al., 2021; Li and Liang, 2021; Liu et al., 2022d), and Memory Efficient Fine-Tuning (Dettmers et al., 2023; Kim et al., 2023d; Malladi et al., 2024) are utilized. A detailed survey on Efficient Large Language Models can be found here in Wan et al. (2024b). The problem that remains is how can we further compress the model and build effective distillation algorithms.

Multi-Teacher Distillation Most of the existing distilled models are distilled from a single teacher model, however, it is widely accepted that models trained with different sources of data have various capabilities. Thus a question arises: Is it possible to distill knowledge from different teacher models into one student model? BabyLlama (Timiryasov and Tastet, 2023) proposes to distill the knowledge from both the GPT2 and LLaMA into the small-size student models. Ensemble-Instruct (Lee et al., 2023b) tries to generate both instructions and responses ensembled from several different LLMs with RougeL as the indicator. FUSELLM (Wan et al., 2024a) externalizes the collective knowledge and unique strengths by leveraging the generative distributions of different LLMs aiming to train a student model beyond those of any individual source LLM. Despite the recent progress in this topic, it still remains an under-explored topic.

Explore Richer Knowledge from Teacher LLMs As indicated in Table 3, the majority of teacher LLMs are closed-source due to their advanced capabilities. Consequently, current methodologies primarily focus on using the generations from these models as hard labels, training student models through simple supervised fine-tuning. However, beyond

the straightforward imitation of output behaviors via hard labels, there is a growing interest in harnessing richer knowledge from teacher LLMs, including feedback and feature knowledge, as well as exploring diverse combinations of knowledge elicitation methods. As highlighted in the *Feedback* section, teachers can provide various types of feedback based on the student’s outputs (Lee et al., 2023a; Jiang et al., 2023b; Chen et al., 2023a). Similarly, the *Feature* section discusses how knowledge based on features, such as logits serving as soft labels, can offer deeper, intrinsic insights into the teacher model (Gu et al., 2024; Agarwal et al., 2024). These explorations have demonstrated promising outcomes, suggesting that access to a broader spectrum of knowledge can significantly enhance student model performance beyond what is achievable through simple SFT distillation alone. This highlights the critical need for further research into varied knowledge extraction methods from teacher LLMs to augment the effectiveness of KD processes.

Overcoming Catastrophic Forgetting During Distillation

Previous research has delved into the fine-tuning of LLMs to acquire the ability to follow instructions or transfer knowledge for forthcoming tasks, skills, or domains, leveraging advancements in LLM technology. Nevertheless, investigations have revealed that the continual fine-tuning of LLMs on particular datasets (skills, domains) can lead to a phenomenon known as catastrophic forgetting, wherein previously acquired knowledge and problem-solving abilities for earlier tasks are compromised (Chen et al., 2023f; Kotha et al., 2023; Koloski et al., 2023; Wu et al., 2024; Luo et al., 2023f). Earlier studies in machine learning and deep learning have investigated various techniques to help mitigate forgetting during the fine-tuning or continue learning process, such as rehearsal, which entails periodically revisiting and training on past data (Kirkpatrick et al., 2017; Rostami et al., 2019; Rolnick et al., 2019), as well as regularization methods like elastic weight consolidation (Lee et al., 2017), or dynamic architecture methods (Mallya et al., 2018; Wang et al., 2022c; Hu et al., 2023c; Chen et al., 2023f). To address the challenges of catastrophic forgetting and to enhance the diversity of generated instructions in knowledge distillation for LLMs, Jiang et al. (2023b) randomly sample an instruction from the easy instructions and also prompt the generator to generate a new instruction that belongs to the same domain as the sampled one. In a similar vein, Li et al. (2023m) study the problem of instruction-tuning in multi-modal LLMs knowledge distillation and introduce a competitive distillation framework. The model tries to produce new instructions that differ in content but are similar in difficulty to the original pictures in the multi-modal augmentation phase, so as to alleviate catastrophic forgetting of the model and enhance the diversity of the instruction tuning pool. Chen et al. (2023f) propose the Lifelong-MoE (Mixture-of Experts) architecture based on general language models, which dynamically adds model capacity via adding experts with regularized pretraining. Additionally, the model also introduces implicit regularization via distillation of the knowledge from old experts and gatings to effectively preserve old knowledge. Zeng et al. (2023b) propose a new generative-based rehearsal method as Dirichlet Continual Learning (DCL). This method com-

bines task distribution modeling and knowledge distillation to mitigate catastrophic forgetting without requiring access to the old data. To evaluate the effectiveness of instruction tuning in the context of continuous learning tasks, Zhang et al. (2023i) introduce a more challenging yet practical problem called Continual Instruction Tuning (CIT) and also establish a benchmark suite consisting of learning and evaluation protocols. Although current research has explored some simple methods to alleviate knowledge forgetting during model fine-tuning or knowledge distillation processes, effectively avoiding catastrophic forgetting across domains and skills remains a challenging issue. How to retain the original model’s capabilities effectively during knowledge distillation or transfer processes is still a challenging problem.

Trustworthy Knowledge Distillation Trustworthiness in LLMs is paramount, encompassing attributes such as truthfulness, safety, fairness, robustness, privacy, and adherence to machine ethics (Sun et al., 2024a). The rapid advancement of LLMs brings to the forefront concerns regarding their trustworthiness, stemming from their complex outputs, the biases present in vast training datasets, and the potential inclusion of private information. Current efforts in KD of LLMs primarily focus on distilling various skills from LLMs, with relatively little attention paid to trustworthiness aspects. Existing studies tend to concentrate on a subset of trustworthiness aspects, such as helpfulness, honesty, and harmlessness (Bai et al., 2022a; Yang et al., 2024; Cui et al., 2023a). Consequently, in the distillation process, student models may inherit issues related to trustworthiness from their teacher LLMs. As assessed in Sun et al. (2024a), smaller open-source LLMs generally fall short of their proprietary counterparts in trustworthiness metrics. Therefore, considering trustworthiness alongside the distillation of capabilities into student models is crucial. It is imperative that future research on KD not only enhances the capabilities of student models but also ensures that broader aspects of trustworthiness are meticulously addressed.

Weak-to-strong Distillation. The concept of “weak-to-strong generalization” in LLMs (Burns et al., 2023) emphasizes the potential to leverage weak supervision to elicit the advanced capabilities of more powerful models. This approach challenges the traditional distillation paradigm by suggesting that even with limited or imperfect supervision, it is possible to enhance the performance of LLMs significantly. This necessitates exploring innovative strategies that enable weaker models to guide the learning process of stronger ones effectively, highlighting the importance of developing methods that can bridge the gap between these models. Such research could unlock new avenues for improving LLMs’ efficiency and effectiveness, making the pursuit of “weak-to-strong distillation” a crucial area for future investigations in this LLM era. Initially, Burns et al. (2023) investigates whether weak model supervision can unlock the full capabilities of much stronger models. Through experiments with pre-trained language models in the GPT-4 family across NLP, chess, and reward modeling tasks, it finds that finetuning strong models on weak labels leads to better performance than their weak supervisors, demonstrating weak-to-strong generalization. Then, Li et al.

(2024g) introduce Superfiltering, a method that employs smaller, weaker models like GPT-2 to select high-quality data for fine-tuning larger, more capable models such as LLaMA2. This approach is rooted in discovering a strong consistency in evaluating instruction tuning data difficulty across models of varying sizes. More recently, Ji et al. (2024) introduce Aligner, a novel approach for aligning LLMs with human values and intentions by utilizing weak supervisory signals from smaller models to improve the performance of larger models. However, Burns et al. (2023) find that achieving the full capabilities of strong models requires more than naive finetuning, suggesting the need for further research in this area. Therefore, open questions still remain about 1) What are the theoretical and practical limits of weak-to-strong distillation? Can weak supervision reliably extract and enhance the full spectrum of capabilities in stronger models across all domains, or are there inherent limitations based on model architecture or task specificity? 2) How do we identify or design the optimal weak supervisors for distilling knowledge into stronger models? Is there a framework or criteria to predict which weak models would be most effective in guiding the learning process of more complex models for specific tasks? 3) To what extent are weak-to-strong distillation techniques transferable and scalable across different sizes and types of models? How can these methods be adapted to ensure efficacy and efficiency in distilling knowledge from very large models to significantly smaller ones, especially in resource-constrained environments?

Self-Alignment. Aligning LLMs traditionally relies heavily on human or teacher LLMs to supply extensive preference data. Consequently, the alignment of the student model is limited by the quantity of distilled preference data and the teacher’s capabilities. Self-alignment offers a promising alternative, aiming to enhance alignment beyond the constraints of teacher-provided preferences. In self-alignment, the student model endeavors to autonomously improve and align its responses with desired behaviors, including generating model-written feedback, critiques, and explanations. Several studies have explored utilizing the student model’s inherent capabilities to generate knowledge for alignment (Bai et al., 2022a; Sun et al., 2024b; Li et al., 2024c; Yuan et al., 2024a). Beyond merely producing improved responses (Bai et al., 2022a; Sun et al., 2024b), implementations of self-alignment include employing the student as its reward model to offer feedback (Yuan et al., 2024a), a strategy that merges *Self-Knowledge* with *Feedback* methods of eliciting knowledge. We advocate for increasingly leveraging the student model itself to provide feedback, thereby enhancing self-alignment capabilities. This approach not only facilitates moving beyond traditional human/teacher preference-based rewards but also opens avenues for continual self-improvement and alignment.

7 CONCLUSION AND DISCUSSION

This survey has explored the diverse landscape of knowledge distillation for LLMs, highlighting key techniques, applications, and challenges. KD plays a crucial role in democratizing access to advanced LLM capabilities, providing cutting-edge advancements without the high costs

of training and deployment. Our review emphasizes various KD approaches, from algorithmic innovations to skill enhancement and vertical distillation. Notably, data augmentation and synthesis within KD emerge as vital tools for improving distillation, revealing the powerful synergy between enriched training data and effective model distillation. As the AI landscape evolves, rapid advancements in model architectures and training methods present both challenges and research opportunities for KD of LLMs. Future innovation will need to focus on achieving efficiency, transparency, and ethics while maintaining model trustworthiness. Furthermore, promising areas such as weak-to-strong generalization, self-alignment, and multi-modal LLMs offer the potential to enhance the capabilities of distilled models. In conclusion, the KD of LLMs is set to play a pivotal role in the future of AI research. As highlighted in this survey, sustained research efforts will be critical in developing accessible, efficient, and responsible AI for all. Importantly, when conducting KD of LLMs like ChatGPT or Llama, it’s essential to comply with the model providers’ terms⁴, such as the restrictions on developing competitive products.

REFERENCES

- L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray et al., “Training language models to follow instructions with human feedback,” *Advances in Neural Information Processing Systems*, vol. 35, pp. 27730–27744, 2022.
- OpenAI, :, J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat, R. Avila, I. Babuschkin, S. Balaji, V. Balcom, P. Baltescu, H. Bao, M. Bavarian, J. Belgum, I. Bello, J. Berdine, G. Bernadett-Shapiro, C. Berner, L. Bogdonoff, O. Boiko, M. Boyd, A.-L. Brakman, G. Brockman, T. Brooks, M. Brundage, K. Button, T. Cai, R. Campbell, A. Cann, B. Carey, C. Carlson, R. Carmichael, B. Chan, C. Chang, F. Chantzis, D. Chen, S. Chen, R. Chen, J. Chen, M. Chen, B. Chess, C. Cho, C. Chu, H. W. Chung, D. Cummings, J. Currier, Y. Dai, C. Decareaux, T. Degry, N. Deutsch, D. Deville, A. Dhar, D. Dohan, S. Dowling, S. Dunning, A. Ecoffet, A. Eleti, T. Eloundou, D. Farhi, L. Fedus, N. Felix, S. P. Fishman, J. Forte, I. Fulford, L. Gao, E. Georges, C. Gibson, V. Goel, T. Gogineni, G. Goh, R. Gontijo-Lopes, J. Gordon, M. Grafstein, S. Gray, R. Greene, J. Gross, S. S. Gu, Y. Guo, C. Hallacy, J. Han, J. Harris, Y. He, M. Heaton, J. Heidecke, C. Hesse, A. Hickey, W. Hickey, P. Hoeschele, B. Houghton, K. Hsu, S. Hu, X. Hu, J. Huizinga, S. Jain, S. Jain, J. Jang, A. Jiang, R. Jiang, H. Jin, D. Jin, S. Jomoto, B. Jonn, H. Jun, T. Kafan, Łukasz Kaiser, A. Kamali, I. Kanitscheider, N. S. Keskar, T. Khan, L. Kilpatrick, J. W. Kim, C. Kim, Y. Kim, H. Kirchner, J. Kiros, M. Knight, D. Kokotajlo, Łukasz Kondraciuk, A. Kondrich, A. Konstantinidis, K. Kosic, G. Krueger, V. Kuo, M. Lampe, I. Lan, T. Lee, J. Leike, J. Leung, D. Levy, C. M. Li, R. Lim, M. Lin, S. Lin, M. Litwin, T. Lopez, R. Lowe, P. Lue, A. Makanju, K. Malfacini, S. Manning, T. Markov, Y. Markovski, B. Mar-

4. OpenAI Business Terms: <https://openai.com/policies/business-terms>

- tin, K. Mayer, A. Mayne, B. McGrew, S. M. McKinney, C. McLeavey, P. McMillan, J. McNeil, D. Medina, A. Mehta, J. Menick, L. Metz, A. Mishchenko, P. Mishkin, V. Monaco, E. Morikawa, D. Mossing, T. Mu, M. Murati, O. Murk, D. Mély, A. Nair, R. Nakano, R. Nayak, A. Nee-lakantan, R. Ngo, H. Noh, L. Ouyang, C. O’Keefe, J. Pachocki, A. Paino, J. Palermo, A. Pantuliano, G. Parascandolo, J. Parish, E. Parparita, A. Passos, M. Pavlov, A. Peng, A. Perelman, F. de Avila Belbute Peres, M. Petrov, H. P. de Oliveira Pinto, Michael, Pokorny, M. Pokrass, V. Pong, T. Powell, A. Power, B. Power, E. Proehl, R. Puri, A. Radford, J. Rae, A. Ramesh, C. Raymond, F. Real, K. Rimbach, C. Ross, B. Rotsted, H. Roussez, N. Ryder, M. Saltarelli, T. Sanders, S. Santurkar, G. Sastry, H. Schmidt, D. Schnurr, J. Schulman, D. Selsam, K. Sheppard, T. Sherbakov, J. Shieh, S. Shoker, P. Shyam, S. Sidor, E. Sigler, M. Simens, J. Sitkin, K. Slama, I. Sohl, B. Sokolowsky, Y. Song, N. Staudacher, F. P. Such, N. Summers, I. Sutskever, J. Tang, N. Tezak, M. Thompson, P. Tillet, A. Tootoonchian, E. Tseng, P. Tuggle, N. Turley, J. Tworek, J. F. C. Uribe, A. Vallone, A. Vijayvergiya, C. Voss, C. Wainwright, J. J. Wang, A. Wang, B. Wang, J. Ward, J. Wei, C. Weinmann, A. Welihinda, P. Welinder, J. Weng, L. Weng, M. Wiethoff, D. Willner, C. Winter, S. Wolrich, H. Wong, L. Workman, S. Wu, J. Wu, M. Wu, K. Xiao, T. Xu, S. Yoo, K. Yu, Q. Yuan, W. Zaremba, R. Zellers, C. Zhang, M. Zhang, S. Zhao, T. Zheng, J. Zhuang, W. Zhuk, and B. Zoph, “Gpt-4 technical report,” 2023.
- G. Team, R. Anil, S. Borgeaud, Y. Wu, J.-B. Alayrac, J. Yu, R. Soricut, J. Schalkwyk, A. M. Dai, A. Hauth *et al.*, “Gemini: a family of highly capable multimodal models,” *arXiv preprint arXiv:2312.11805*, 2023.
- J. Wei, Y. Tay, R. Bommasani, C. Raffel, B. Zoph, S. Borgeaud, D. Yogatama, M. Bosma, D. Zhou, D. Metzler, E. H. Chi, T. Hashimoto, O. Vinyals, P. Liang, J. Dean, and W. Fedus, “Emergent abilities of large language models,” *Trans. Mach. Learn. Res.*, vol. 2022, 2022. [Online]. Available: <https://openreview.net/forum?id=yzkSU5zdwD>
- J. Wei, X. Wang, D. Schuurmans, M. Bosma, F. Xia, E. Chi, Q. V. Le, D. Zhou *et al.*, “Chain-of-thought prompting elicits reasoning in large language models,” *Advances in Neural Information Processing Systems*, vol. 35, pp. 24824–24837, 2022.
- X. Xu, C. Tao, T. Shen, C. Xu, H. Xu, G. Long, and J. Guang Lou, “Re-reading improves reasoning in large language models,” 2024.
- P. Liang, R. Bommasani, T. Lee, D. Tsipras, D. Soylu, M. Yasunaga, Y. Zhang, D. Narayanan, Y. Wu, A. Kumar, B. Newman, B. Yuan, B. Yan, C. Zhang, C. Cosgrove, C. D. Manning, C. Ré, D. Acosta-Navas, D. A. Hudson, E. Zelikman, E. Durmus, F. Ladhak, F. Rong, H. Ren, H. Yao, J. Wang, K. Santhanam, L. J. Orr, L. Zheng, M. Yükeşgönül, M. Suzgun, N. Kim, N. Guha, N. S. Chatterji, O. Khattab, P. Henderson, Q. Huang, R. Chi, S. M. Xie, S. Santurkar, S. Ganguli, T. Hashimoto, T. Icard, T. Zhang, V. Chaudhary, W. Wang, X. Li, Y. Mai, Y. Zhang, and Y. Koreeda, “Holistic evaluation of language models,” *CoRR*, vol. abs/2211.09110, 2022. [Online]. Available: <https://doi.org/10.48550/arXiv.2211.09110>
- X. Wu, R. Duan, and J. Ni, “Unveiling security, privacy, and ethical concerns of chatgpt,” *Journal of Information and Intelligence*, 2023.
- 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.
- A. Q. Jiang, A. Sablayrolles, A. Mensch, C. Bamford, D. S. Chaplot, D. de las Casas, F. Bressand, G. Lengyel, G. Lample, L. Saulnier, L. R. Lavaud, M.-A. Lachaux, P. Stock, T. L. Scao, T. Lavril, T. Wang, T. Lacroix, and W. E. Sayed, “Mistral 7b,” 2023.
- L. Zheng, W. Chiang, Y. Sheng, S. Zhuang, Z. Wu, Y. Zhuang, Z. Lin, Z. Li, D. Li, E. P. Xing, H. Zhang, J. E. Gonzalez, and I. Stoica, “Judging llm-as-a-judge with mt-bench and chatbot arena,” *CoRR*, vol. abs/2306.05685, 2023. [Online]. Available: <https://doi.org/10.48550/arXiv.2306.05685>
- L. Sun, Y. Huang, H. Wang, S. Wu, Q. Zhang, C. Gao, Y. Huang, W. Lyu, Y. Zhang, X. Li, Z. Liu, Y. Liu, Y. Wang, Z. Zhang, B. Kailkhura, C. Xiong, C. Xiao, C. Li, E. Xing, F. Huang, H. Liu, H. Ji, H. Wang, H. Zhang, H. Yao, M. Kellis, M. Zitnik, M. Jiang, M. Bansal, J. Zou, J. Pei, J. Liu, J. Gao, J. Han, J. Zhao, J. Tang, J. Wang, J. Mitchell, K. Shu, K. Xu, K.-W. Chang, L. He, L. Huang, M. Backes, N. Z. Gong, P. S. Yu, P.-Y. Chen, Q. Gu, R. Xu, R. Ying, S. Ji, S. Jana, T. Chen, T. Liu, T. Zhou, W. Wang, X. Li, X. Zhang, X. Wang, X. Xie, X. Chen, X. Wang, Y. Liu, Y. Ye, Y. Cao, Y. Chen, and Y. Zhao, “Trustllm: Trustworthiness in large language models,” 2024.
- J. Gou, B. Yu, S. J. Maybank, and D. Tao, “Knowledge distillation: A survey,” *International Journal of Computer Vision*, vol. 129, pp. 1789–1819, 2021.
- M. Gupta and P. Agrawal, “Compression of deep learning models for text: A survey,” *ACM Transactions on Knowledge Discovery from Data (TKDD)*, vol. 16, no. 4, pp. 1–55, 2022.
- S. Y. Feng, V. Gangal, J. Wei, S. Chandar, S. Vosoughi, T. Mitamura, and E. Hovy, “A survey of data augmentation approaches for nlp,” *arXiv preprint arXiv:2105.03075*, 2021.
- 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,” https://github.com/tatsu-lab/stanford_alpaca, 2023.
- Y. Gu, L. Dong, F. Wei, and M. Huang, “MiniLLM: Knowledge distillation of large language models,” in *The Twelfth International Conference on Learning Representations*, 2024. [Online]. Available: <https://openreview.net/forum?id=5h0qf71BZZ>
- R. Agarwal, N. Vieillard, Y. Zhou, P. Stanczyk, S. R. Garea, M. Geist, and O. Bachem, “On-policy distillation of language models: Learning from self-generated mistakes,” in *The Twelfth International Conference on Learning Representations*, 2024. [Online]. Available: <https://openreview.net/forum?id=5h0qf71BZZ>

- [//openreview.net/forum?id=3zKtaqxLhW](https://openreview.net/forum?id=3zKtaqxLhW)
- W. Yuan, R. Y. Pang, K. Cho, S. Sukhbaatar, J. Xu, and J. Weston, "Self-rewarding language models," 2024.
- Z. Chen, Y. Deng, H. Yuan, K. Ji, and Q. Gu, "Self-play fine-tuning converts weak language models to strong language models," 2024.
- Y. Huang, Y. Chen, Z. Yu, and K. McKeown, "In-context learning distillation: Transferring few-shot learning ability of pre-trained language models," 2022.
- G. Cui, L. Yuan, N. Ding, G. Yao, W. Zhu, Y. Ni, G. Xie, Z. Liu, and M. Sun, "Ultrafeedback: Boosting language models with high-quality feedback," *arXiv preprint arXiv:2310.01377*, 2023.
- S. Mukherjee, A. Mitra, G. Jawahar, S. Agarwal, H. Palangi, and A. Awadallah, "Orca: Progressive learning from complex explanation traces of gpt-4," *arXiv preprint arXiv:2306.02707*, 2023.
- B. Ding, C. Qin, L. Liu, Y. K. Chia, B. Li, S. Joty, and L. Bing, "Is GPT-3 a good data annotator?" in *ACL (1)*. Association for Computational Linguistics, 2023, pp. 11 173–11 195.
- S. Chaudhary, "Code alpaca: An instruction-following llama model for code generation," <https://github.com/sahil280114/codealpaca>, 2023.
- H. Wang, C. Liu, N. Xi, Z. Qiang, S. Zhao, B. Qin, and T. Liu, "Huatuo: Tuning llama model with chinese medical knowledge," *arXiv preprint arXiv:2304.06975*, 2023.
- LawGPT*. GitHub, 2023.
- D. Zhang, Z. Hu, S. Zhoubian, Z. Du, K. Yang, Z. Wang, Y. Yue, Y. Dong, and J. Tang, "Sciglm: Training scientific language models with self-reflective instruction annotation and tuning," *CoRR*, vol. abs/2401.07950, 2024. [Online]. Available: <https://doi.org/10.48550/arXiv.2401.07950>
- 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/>
- C. Xu, Q. Sun, K. Zheng, X. Geng, P. Zhao, J. Feng, C. Tao, and D. Jiang, "Wizardlm: Empowering large language models to follow complex instructions," *arXiv preprint arXiv:2304.12244*, 2023.
- W. X. Zhao, K. Zhou, J. Li, T. Tang, X. Wang, Y. Hou, Y. Min, B. Zhang, J. Zhang, Z. Dong, Y. Du, C. Yang, Y. Chen, Z. Chen, J. Jiang, R. Ren, Y. Li, X. Tang, Z. Liu, P. Liu, J.-Y. Nie, and J.-R. Wen, "A survey of large language models," 2023.
- X. He, Z. Lin, Y. Gong, A. Jin, H. Zhang, C. Lin, J. Jiao, S. M. Yiu, N. Duan, W. Chen *et al.*, "Annollm: Making large language models to be better crowdsourced annotators," *arXiv preprint arXiv:2303.16854*, 2023.
- Y. Wang, Z. Yu, Z. Zeng, L. Yang, C. Wang, H. Chen, C. Jiang, R. Xie, J. Wang, X. Xie, W. Ye, S. Zhang, and Y. Zhang, "Pandalm: An automatic evaluation benchmark for llm instruction tuning optimization," 2023.
- C. Hsieh, C. Li, C. Yeh, H. Nakhost, Y. Fujii, A. Ratner, R. Krishna, C. Lee, and T. Pfister, "Distilling step-by-step! outperforming larger language models with less training data and smaller model sizes," in *ACL (Findings)*. Association for Computational Linguistics, 2023, pp. 8003–8017.
- A. Mitra, L. D. Corro, S. Mahajan, A. Codas, C. Simoes, S. Agarwal, X. Chen, A. Razdaibiedina, E. Jones, K. Aggarwal, H. Palangi, G. Zheng, C. Rosset, H. Khanpour, and A. Awadallah, "Orca 2: Teaching small language models how to reason," 2023.
- C. Xu, D. Guo, N. Duan, and J. J. McAuley, "Baize: An open-source chat model with parameter-efficient tuning on self-chat data," in *EMNLP*. Association for Computational Linguistics, 2023, pp. 6268–6278.
- X. Yue, X. Qu, G. Zhang, Y. Fu, W. Huang, H. Sun, Y. Su, and W. Chen, "Mammoth: Building math generalist models through hybrid instruction tuning," *arXiv preprint arXiv:2309.05653*, 2023.
- L. Chenglin, C. Qianglong, W. Caiyu, and Z. Yin, "Mixed distillation helps smaller language model better reasoning," 2023.
- 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," *arXiv preprint arXiv:2212.10560*, 2022.
- Z. Sun, Y. Shen, Q. Zhou, H. Zhang, Z. Chen, D. Cox, Y. Yang, and C. Gan, "Principle-driven self-alignment of language models from scratch with minimal human supervision," *Advances in Neural Information Processing Systems*, vol. 36, 2024.
- Z. Luo, C. Xu, P. Zhao, Q. Sun, X. Geng, W. Hu, C. Tao, J. Ma, Q. Lin, and D. Jiang, "Wizardcoder: Empowering code large language models with evol-instruct," *arXiv preprint arXiv:2306.08568*, 2023.
- H. Luo, Q. Sun, C. Xu, P. Zhao, J. Lou, C. Tao, X. Geng, Q. Lin, S. Chen, and D. Zhang, "Wizardmath: Empowering mathematical reasoning for large language models via reinforced evol-instruct," *arXiv preprint arXiv:2308.09583*, 2023.
- H. Dai, Z. Liu, W. Liao, X. Huang, Y. Cao, Z. Wu, L. Zhao, S. Xu, W. Liu, N. Liu, S. Li, D. Zhu, H. Cai, L. Sun, Q. Li, D. Shen, T. Liu, and X. Li, "Auggpt: Leveraging chatgpt for text data augmentation," 2023.
- Z. He, M. T. Ribeiro, and F. Khani, "Targeted data generation: Finding and fixing model weaknesses," in *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, A. Rogers, J. Boyd-Graber, and N. Okazaki, Eds. Toronto, Canada: Association for Computational Linguistics, Jul. 2023, pp. 8506–8520. [Online]. Available: <https://aclanthology.org/2023.acl-long.474>
- N. Ding, Y. Chen, B. Xu, Y. Qin, S. Hu, Z. Liu, M. Sun, and B. Zhou, "Enhancing chat language models by scaling high-quality instructional conversations," in *EMNLP*. Association for Computational Linguistics, 2023, pp. 3029–3051.
- S. Gunasekar, Y. Zhang, J. Aneja, C. C. T. Mendes, A. D. Giorno, S. Gopi, M. Javaheripi, P. Kauffmann, G. de Rosa, O. Saarikivi, A. Salim, S. Shah, H. S. Behl, X. Wang, S. Bubeck, R. Eldan, A. T. Kalai, Y. T. Lee, and Y. Li, "Textbooks are all you need," 2023.
- Y. Li, S. Bubeck, R. Eldan, A. Del Giorno, S. Gunasekar, and Y. T. Lee, "Textbooks are all you need ii: phi-1.5 technical report," *arXiv preprint arXiv:2309.05463*, 2023.
- Phi-2: The surprising power of small lan-*

- guage models, December 2023. [Online]. Available: <https://www.microsoft.com/en-us/research/blog/phi-2-the-surprising-power-of-small-language-models/>
- Y. Wei, Z. Wang, J. Liu, Y. Ding, and L. Zhang, "Magicoder: Source code is all you need," 2023.
- Z. Yu, X. Zhang, N. Shang, Y. Huang, C. Xu, Y. Zhao, W. Hu, and Q. Yin, "Wavecoder: Widespread and versatile enhanced instruction tuning with refined data generation," 2024.
- J. Ye, J. Gao, Q. Li, H. Xu, J. Feng, Z. Wu, T. Yu, and L. Kong, "Zerogen: Efficient zero-shot learning via dataset generation," in *EMNLP*. Association for Computational Linguistics, 2022, pp. 11 653–11 669.
- J. Gao, R. Pi, Y. Lin, H. Xu, J. Ye, Z. Wu, W. Zhang, X. Liang, Z. Li, and L. Kong, "Self-guided noise-free data generation for efficient zero-shot learning," in *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*, 2023. [Online]. Available: https://openreview.net/pdf?id=h5OpjGd_lo6
- L. H. Bonifacio, H. Q. Abonizio, M. Fadaee, and R. F. Nogueira, "Inpars: Data augmentation for information retrieval using large language models," *CoRR*, vol. abs/2202.05144, 2022. [Online]. Available: <https://arxiv.org/abs/2202.05144>
- I. Timiryasov and J.-L. Tastet, "Baby llama: knowledge distillation from an ensemble of teachers trained on a small dataset with no performance penalty," in *Proceedings of the BabyLM Challenge at the 27th Conference on Computational Natural Language Learning*, A. Warstadt, A. Mueller, L. Choshen, E. Wilcox, C. Zhuang, J. Ciro, R. Mosquera, B. Paranjabe, A. Williams, T. Linzen, and R. Cotterell, Eds. Singapore: Association for Computational Linguistics, Dec. 2023, pp. 279–289. [Online]. Available: <https://aclanthology.org/2023.conll-babylm.24>
- C. Tao, L. Hou, W. Zhang, L. Shang, X. Jiang, Q. Liu, P. Luo, and N. Wong, "Compression of generative pre-trained language models via quantization," *arXiv preprint arXiv:2203.10705*, 2022.
- Z. Liu, B. Oguz, C. Zhao, E. Chang, P. Stock, Y. Mehdad, Y. Shi, R. Krishnamoorthi, and V. Chandra, "Llm-qat: Data-free quantization aware training for large language models," *arXiv preprint arXiv:2305.17888*, 2023.
- Y. Bai, S. Kadavath, S. Kundu, A. Askell, J. Kernion, A. Jones, A. Chen, A. Goldie, A. Mirhoseini, C. McKinnon, C. Chen, C. Olsson, C. Olah, D. Hernandez, D. Drain, D. Ganguli, D. Li, E. Tran-Johnson, E. Perez, J. Kerr, J. Mueller, J. Ladish, J. Landau, K. Ndousse, K. Lukosuite, L. Lovitt, M. Sellitto, N. Elhage, N. Schiefer, N. Mercado, N. Das-Sarma, R. Lasenby, R. Larson, S. Ringer, S. Johnston, S. Kravec, S. E. Showk, S. Fort, T. Lanham, T. Telleen-Lawton, T. Conerly, T. Henighan, T. Hume, S. R. Bowman, Z. Hatfield-Dodds, B. Mann, D. Amodei, N. Joseph, S. McCandlish, T. Brown, and J. Kaplan, "Constitutional ai: Harmlessness from ai feedback," 2022.
- L. Tunstall, E. Beeching, N. Lambert, N. Rajani, K. Rasul, Y. Belkada, S. Huang, L. von Werra, C. Fourrier, N. Habib *et al.*, "Zephyr: Direct distillation of lm alignment," *arXiv preprint arXiv:2310.16944*, 2023.
- J. Hong, Q. Tu, C. Chen, X. Gao, J. Zhang, and R. Yan, "Cyclealign: Iterative distillation from black-box llm to white-box models for better human alignment," *arXiv preprint arXiv:2310.16271*, 2023.
- H. Lee, S. Phatale, H. Mansoor, K. Lu, T. Mesnard, C. Bishop, V. Carbune, and A. Rastogi, "Rlaif: Scaling reinforcement learning from human feedback with ai feedback," *arXiv preprint arXiv:2309.00267*, 2023.
- Y. Jiang, C. Chan, M. Chen, and W. Wang, "Lion: Adversarial distillation of closed-source large language model," *arXiv preprint arXiv:2305.12870*, 2023.
- H. Chen, A. Saha, S. Hoi, and S. Joty, "Personalized distillation: Empowering open-sourced LLMs with adaptive learning for code generation," in *The 2023 Conference on Empirical Methods in Natural Language Processing*, 2023. [Online]. Available: <https://openreview.net/forum?id=alxWMBcNVN>
- K. Yang, D. Klein, A. Celikyilmaz, N. Peng, and Y. Tian, "RLCD: Reinforcement learning from contrastive distillation for LM alignment," in *The Twelfth International Conference on Learning Representations*, 2024. [Online]. Available: <https://openreview.net/forum?id=v3XXtxWKi6>
- J. Jung, P. West, L. Jiang, F. Brahman, X. Lu, J. Fisher, T. Sorensen, and Y. Choi, "Impossible distillation: from low-quality model to high-quality dataset & model for summarization and paraphrasing," 2023.
- J. Huang, S. Gu, L. Hou, Y. Wu, X. Wang, H. Yu, and J. Han, "Large language models can self-improve," in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, H. Bouamor, J. Pino, and K. Bali, Eds. Singapore: Association for Computational Linguistics, Dec. 2023, pp. 1051–1068. [Online]. Available: <https://aclanthology.org/2023.emnlp-main.67>
- C. Gulcehre, T. L. Paine, S. Srinivasan, K. Konyushkova, L. Weerts, A. Sharma, A. Siddhant, A. Ahern, M. Wang, C. Gu, W. Macherey, A. Doucet, O. Firat, and N. de Freitas, "Reinforced self-training (rest) for language modeling," 2023.
- E. Zelikman, Y. Wu, J. Mu, and N. D. Goodman, "Star: Bootstrapping reasoning with reasoning," in *NeurIPS*, 2022.
- V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter," *arXiv preprint arXiv:1910.01108*, 2019.
- Y. Wen, Z. Li, W. Du, and L. Mou, "f-divergence minimization for sequence-level knowledge distillation," in *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, A. Rogers, J. Boyd-Graber, and N. Okazaki, Eds. Toronto, Canada: Association for Computational Linguistics, Jul. 2023, pp. 10 817–10 834. [Online]. Available: <https://aclanthology.org/2023.acl-long.605>
- C. Liang, S. Zuo, Q. Zhang, P. He, W. Chen, and T. Zhao, "Less is more: Task-aware layer-wise distillation for language model compression," in *International Conference on Machine Learning*. PMLR, 2023, pp. 20 852–20 867.
- M. Kwon, S. M. Xie, K. Bullard, and D. Sadigh, "Reward design with language models," in *ICLR*. OpenReview.net, 2023.
- B. Peng, C. Li, P. He, M. Galley, and J. Gao, "Instruction tuning with gpt-4," 2023.
- G. Li, H. A. A. K. Hammoud, H. Itani, D. Khizbullin, and B. Ghanem, "Camel: Communicative agents for" mind" exploration of large scale language model society," *arXiv*

- preprint *arXiv:2303.17760*, 2023.
- G. Wang, S. Cheng, X. Zhan, X. Li, S. Song, and Y. Liu, "OpenChat: Advancing Open-source Language Models with Mixed-Quality Data," Sep. 2023, *arXiv:2309.11235* [cs]. [Online]. Available: <http://arxiv.org/abs/2309.11235>
- M. Kang, S. Lee, J. Baek, K. Kawaguchi, and S. J. Hwang, "Knowledge-augmented reasoning distillation for small language models in knowledge-intensive tasks," *arXiv preprint arXiv:2305.18395*, 2023.
- H. Luo, Y.-S. Chuang, Y. Gong, T. Zhang, Y. Kim, X. Wu, D. Fox, H. Meng, and J. Glass, "Sail: Search-augmented instruction learning," *arXiv preprint arXiv:2305.15225*, 2023.
- A. Asai, Z. Wu, Y. Wang, A. Sil, and H. Hajishirzi, "Self-rag: Learning to retrieve, generate, and critique through self-reflection," *arXiv preprint arXiv:2310.11511*, 2023.
- S. Ye, Y. Jo, D. Kim, S. Kim, H. Hwang, and M. Seo, "Selfee: Iterative self-revising llm empowered by self-feedback generation," Blog post, May 2023. [Online]. Available: <https://kaistai.github.io/SelfFee/>
- P. Wang, L. Li, L. Chen, F. Song, B. Lin, Y. Cao, T. Liu, and Z. Sui, "Making large language models better reasoners with alignment," 2023.
- D. Cheng, S. Huang, and F. Wei, "Adapting large language models via reading comprehension," 2023.
- Y. Zhang, Z. Chen, Y. Fang, L. Cheng, Y. Lu, F. Li, W. Zhang, and H. Chen, "Knowledgeable preference alignment for llms in domain-specific question answering," 2023.
- J. Scheurer, J. A. Campos, T. Korbak, J. S. Chan, A. Chen, K. Cho, and E. Perez, "Training language models with language feedback at scale," 2023.
- S. Kim, S. Bae, J. Shin, S. Kang, D. Kwak, K. Yoo, and M. Seo, "Aligning large language models through synthetic feedback," in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, H. Bouamor, J. Pino, and K. Bali, Eds. Singapore: Association for Computational Linguistics, Dec. 2023, pp. 13 677–13 700. [Online]. Available: <https://aclanthology.org/2023.emnlp-main.844>
- P. Roit, J. Ferret, L. Shani, R. Aharoni, G. Cideron, R. Dadashi, M. Geist, S. Girgin, L. Hussenot, O. Keller, N. Momchev, S. Ramos Garea, P. Stanczyk, N. Vieillard, O. Bachem, G. Elidan, A. Hassidim, O. Pietquin, and I. Szpektor, "Factually consistent summarization via reinforcement learning with textual entailment feedback," in *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, A. Rogers, J. Boyd-Graber, and N. Okazaki, Eds. Toronto, Canada: Association for Computational Linguistics, Jul. 2023, pp. 6252–6272. [Online]. Available: <https://aclanthology.org/2023.acl-long.344>
- Y. Yang, E. Chern, X. Qiu, G. Neubig, and P. Liu, "Alignment for honesty," *arXiv preprint arXiv:2312.07000*, 2023.
- R. Liu, R. Yang, C. Jia, G. Zhang, D. Zhou, A. M. Dai, D. Yang, and S. Vosoughi, "Training socially aligned language models on simulated social interactions," 2023.
- T. Schick, J. Dwivedi-Yu, R. Dessi, R. Raileanu, M. Lomeli, L. Zettlemoyer, N. Cancedda, and T. Scialom, "Toolformer: Language models can teach themselves to use tools," 2023.
- J. Zhang, "Graph-toolformer: To empower llms with graph reasoning ability via prompt augmented by chatgpt," *arXiv preprint arXiv:2304.11116*, 2023.
- S. G. Patil, T. Zhang, X. Wang, and J. E. Gonzalez, "Gorilla: Large language model connected with massive apis," 2023.
- Q. Tang, Z. Deng, H. Lin, X. Han, Q. Liang, B. Cao, and L. Sun, "Toolalpaca: Generalized tool learning for language models with 3000 simulated cases," 2023.
- Y. Qin, S. Liang, Y. Ye, K. Zhu, L. Yan, Y. Lu, Y. Lin, X. Cong, X. Tang, B. Qian, S. Zhao, L. Hong, R. Tian, R. Xie, J. Zhou, M. Gerstein, D. Li, Z. Liu, and M. Sun, "Toolllm: Facilitating large language models to master 16000+ real-world apis," 2023.
- L. Yuan, Y. Chen, X. Wang, Y. R. Fung, H. Peng, and H. Ji, "Craft: Customizing llms by creating and retrieving from specialized toolsets," 2023.
- S. Gao, Z. Shi, M. Zhu, B. Fang, X. Xin, P. Ren, Z. Chen, J. Ma, and Z. Ren, "Confucius: Iterative tool learning from introspection feedback by easy-to-difficult curriculum," 2023.
- C. Wang, W. Luo, Q. Chen, H. Mai, J. Guo, S. Dong, Xiaohua, Xuan, Z. Li, L. Ma, and S. Gao, "Mllm-tool: A multimodal large language model for tool agent learning," 2024.
- W. Shen, C. Li, H. Chen, M. Yan, X. Quan, H. Chen, J. Zhang, and F. Huang, "Small llms are weak tool learners: A multi-llm agent," 2024.
- B. Chen, C. Shu, E. Shareghi, N. Collier, K. Narasimhan, and S. Yao, "Fireact: Toward language agent fine-tuning," 2023.
- A. Zeng, M. Liu, R. Lu, B. Wang, X. Liu, Y. Dong, and J. Tang, "Agenttuning: Enabling generalized agent abilities for llms," 2023.
- D. Yin, F. Brahman, A. Ravichander, K. Chandu, K.-W. Chang, Y. Choi, and B. Y. Lin, "Lumos: Learning agents with unified data, modular design, and open-source llms," 2023.
- S. Qiao, N. Zhang, R. Fang, Y. Luo, W. Zhou, Y. E. Jiang, C. Lv, and H. Chen, "Autoact: Automatic agent learning from scratch via self-planning," 2024.
- Y. Kong, J. Ruan, Y. Chen, B. Zhang, T. Bao, S. Shi, G. Du, X. Hu, H. Mao, Z. Li, X. Zeng, and R. Zhao, "Tptu-v2: Boosting task planning and tool usage of large language model-based agents in real-world systems," 2023.
- F. Gilardi, M. Alizadeh, and M. Kubli, "Chatgpt outperforms crowd workers for text-annotation tasks," *Proceedings of the National Academy of Sciences*, vol. 120, no. 30, Jul. 2023. [Online]. Available: <http://dx.doi.org/10.1073/pnas.2305016120>
- Z. Wang, A. W. Yu, O. Firat, and Y. Cao, "Towards zero-label language learning," 2021.
- Y. Xu, R. Xu, D. Iter, Y. Liu, S. Wang, C. Zhu, and M. Zeng, "InheritSumm: A general, versatile and compact summarizer by distilling from GPT," in *Findings of the Association for Computational Linguistics: EMNLP 2023*, H. Bouamor, J. Pino, and K. Bali, Eds. Singapore: Association for Computational Linguistics, Dec. 2023, pp. 13 879–13 892. [Online]. Available: <https://aclanthology.org/2023.findings-emnlp.927>
- F. Xu, W. Shi, and E. Choi, "RECOMP: Improving retrieval-augmented LMs with context compression and selective augmentation," in *The Twelfth International Conference on Learning Representations*, 2024. [Online]. Available:

- <https://openreview.net/forum?id=mlJLVigNHp>
- S. Ramnath, B. Joshi, S. Hallinan, X. Lu, L. H. Li, A. Chan, J. Hessel, Y. Choi, and X. Ren, "Tailoring self-rationalizers with multi-reward distillation," 2023.
- S. Wang, Y. Liu, Y. Xu, C. Zhu, and M. Zeng, "Want to reduce labeling cost? GPT-3 can help," in *Findings of the Association for Computational Linguistics: EMNLP 2021*, M.-F. Moens, X. Huang, L. Specia, and S. W.-t. Yih, Eds. Punta Cana, Dominican Republic: Association for Computational Linguistics, Nov. 2021, pp. 4195–4205. [Online]. Available: <https://aclanthology.org/2021.findings-emnlp.354>
- Z. Guo, P. Wang, Y. Wang, and S. Yu, "Improving small language models on pubmedqa via generative data augmentation," 2023.
- W. Yang and G. Nicolai, "Neural machine translation data generation and augmentation using chatgpt," 2023.
- K. Srinivasan, K. Raman, A. Samanta, L. Liao, L. Bertelli, and M. Bendersky, "QUILL: Query intent with large language models using retrieval augmentation and multi-stage distillation," in *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing: Industry Track*, Y. Li and A. Lazaridou, Eds. Abu Dhabi, UAE: Association for Computational Linguistics, Dec. 2022, pp. 492–501. [Online]. Available: <https://aclanthology.org/2022.emnlp-industry.50>
- Z. Dai, V. Y. Zhao, J. Ma, Y. Luan, J. Ni, J. Lu, A. Bakalov, K. Guu, K. B. Hall, and M. Chang, "Promptagator: Few-shot dense retrieval from 8 examples," in *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*, 2023. [Online]. Available: <https://openreview.net/pdf?id=gmlL46YMpu2j>
- R. Meng, Y. Liu, S. Yavuz, D. Agarwal, L. Tu, N. Yu, J. Zhang, M. Bhat, and Y. Zhou, "Augtriever: Unsupervised dense retrieval by scalable data augmentation," 2023.
- W. Sun, L. Yan, X. Ma, S. Wang, P. Ren, Z. Chen, D. Yin, and Z. Ren, "Is chatgpt good at search? investigating large language models as re-ranking agents," 2023.
- R. Pradeep, S. Sharifmoghaddam, and J. Lin, "Rankvicuna: Zero-shot listwise document reranking with open-source large language models," 2023.
- , "Rankzephyr: Effective and robust zero-shot listwise reranking is a breeze!" 2023.
- F. Ferraretto, T. Laitz, R. Lotufo, and R. Nogueira, "Exaranker: Synthetic explanations improve neural rankers," in *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR '23. New York, NY, USA: Association for Computing Machinery, 2023, p. 2409–2414. [Online]. Available: <https://doi.org/10.1145/3539618.3592067>
- S. Mysore, A. McCallum, and H. Zamani, "Large language model augmented narrative driven recommendations," in *Proceedings of the 17th ACM Conference on Recommender Systems*, ser. RecSys '23. New York, NY, USA: Association for Computing Machinery, 2023, p. 777–783. [Online]. Available: <https://doi.org/10.1145/3604915.3608829>
- J. Zhang, R. Xie, Y. Hou, W. X. Zhao, L. Lin, and J.-R. Wen, "Recommendation as instruction following: A large language model empowered recommendation approach," 2023.
- Q. Liu, N. Chen, T. Sakai, and X.-M. Wu, "Once: Boosting content-based recommendation with both open- and closed-source large language models," 2023.
- S. Kim, J. Shin, Y. Cho, J. Jang, S. Longpre, H. Lee, S. Yun, S. Shin, S. Kim, J. Thorne, and M. Seo, "Prometheus: Inducing evaluation capability in language models," in *The Twelfth International Conference on Learning Representations*, 2024. [Online]. Available: <https://openreview.net/forum?id=8euJaTveKw>
- W. Xu, D. Wang, L. Pan, Z. Song, M. Freitag, W. Wang, and L. Li, "INSTRUCTSCORE: Towards explainable text generation evaluation with automatic feedback," in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, H. Bouamor, J. Pino, and K. Bali, Eds. Singapore: Association for Computational Linguistics, Dec. 2023, pp. 5967–5994. [Online]. Available: <https://aclanthology.org/2023.emnlp-main.365>
- D. Jiang, Y. Li, G. Zhang, W. Huang, B. Y. Lin, and W. Chen, "Tigerscore: Towards building explainable metric for all text generation tasks," 2023.
- J. Li, S. Sun, W. Yuan, R.-Z. Fan, hai zhao, and P. Liu, "Generative judge for evaluating alignment," in *The Twelfth International Conference on Learning Representations*, 2024. [Online]. Available: <https://openreview.net/forum?id=gtkFw6sZGS>
- B. Rozière, J. Gehring, F. Gloeckle, S. Sootla, I. Gat, X. E. Tan, Y. Adi, J. Liu, T. Remez, J. Rapin, A. Kozhevnikov, I. Evtimov, J. Bitton, M. Bhatt, C. C. Ferrer, A. Grattafiori, W. Xiong, A. Défossez, J. Copet, F. Azhar, H. Touvron, L. Martin, N. Usunier, T. Scialom, and G. Synnaeve, "Code llama: Open foundation models for code," 2023.
- B. Liu, C. Chen, C. Liao, Z. Gong, H. Wang, Z. Lei, M. Liang, D. Chen, M. Shen, H. Zhou, H. Yu, and J. Li, "Mftcoder: Boosting code llms with multitask fine-tuning," 2023.
- N. Jain, T. Zhang, W. Chiang, J. E. Gonzalez, K. Sen, and I. Stoica, "Llm-assisted code cleaning for training accurate code generators," *CoRR*, vol. abs/2311.14904, 2023. [Online]. Available: <https://doi.org/10.48550/arXiv.2311.14904>
- H. Liu, C. Li, Q. Wu, and Y. J. Lee, "Visual instruction tuning," in *NeurIPS*, 2023.
- B. Zhao, B. Wu, M. He, and T. Huang, "Svit: Scaling up visual instruction tuning," 2023.
- J. Wang, L. Meng, Z. Weng, B. He, Z. Wu, and Y.-G. Jiang, "To see is to believe: Prompting gpt-4v for better visual instruction tuning," 2023.
- K. Chen, Z. Zhang, W. Zeng, R. Zhang, F. Zhu, and R. Zhao, "Shikra: Unleashing multimodal llm's referential dialogue magic," 2023.
- J. S. Park, J. Hessel, K. R. Chandu, P. P. Liang, X. Lu, P. West, Y. Yu, Q. Huang, J. Gao, A. Farhadi, and Y. Choi, "Localized symbolic knowledge distillation for visual commonsense models," 2023.
- R. Pi, J. Gao, S. Diao, R. Pan, H. Dong, J. Zhang, L. Yao, J. Han, H. Xu, L. Kong, and T. Zhang, "DetGPT: Detect what you need via reasoning," in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, H. Bouamor, J. Pino, and K. Bali, Eds. Singapore: Association for Computational Linguistics, Dec. 2023, pp. 14 172–14 189. [Online]. Available: <https://aclanthology.org/2023.emnlp-main.876>

- L. Zhao, E. Yu, Z. Ge, J. Yang, H. Wei, H. Zhou, J. Sun, Y. Peng, R. Dong, C. Han, and X. Zhang, "Chatspot: Bootstrapping multimodal llms via precise referring instruction tuning," 2023.
- F. Liu, K. Lin, L. Li, J. Wang, Y. Yacoob, and L. Wang, "Mitigating hallucination in large multi-modal models via robust instruction tuning," 2023.
- S. Wu, H. Fei, L. Qu, W. Ji, and T.-S. Chua, "Next-gpt: Any-to-any multimodal llm," 2023.
- R. Luo, Z. Zhao, M. Yang, J. Dong, D. Li, P. Lu, T. Wang, L. Hu, M. Qiu, and Z. Wei, "Valley: Video assistant with large language model enhanced ability," 2023.
- Y. Jiang, E. Schoop, A. Swearngin, and J. Nichols, "Iluvui: Instruction-tuned language-vision modeling of uis from machine conversations," 2023.
- Y. Li, C. Zhang, G. Yu, Z. Wang, B. Fu, G. Lin, C. Shen, L. Chen, and Y. Wei, "Stablellava: Enhanced visual instruction tuning with synthesized image-dialogue data," 2023.
- R. Xu, X. Wang, T. Wang, Y. Chen, J. Pang, and D. Lin, "Pointllm: Empowering large language models to understand point clouds," 2023.
- Q. Huang, M. Tao, Z. An, C. Zhang, C. Jiang, Z. Chen, Z. Wu, and Y. Feng, "Lawyer llama technical report," *arXiv preprint arXiv:2305.15062*, 2023.
- J. Cui, Z. Li, Y. Yan, B. Chen, and L. Yuan, "Chatlaw: Open-source legal large language model with integrated external knowledge bases," *arXiv preprint arXiv:2306.16092*, 2023.
- H. Zhang, J. Chen, F. Jiang, F. Yu, Z. Chen, G. Chen, J. Li, X. Wu, Z. Zhiyi, Q. Xiao, X. Wan, B. Wang, and H. Li, "HuatuogPT, towards taming language model to be a doctor," in *Findings of the Association for Computational Linguistics: EMNLP 2023*, H. Bouamor, J. Pino, and K. Bali, Eds. Singapore: Association for Computational Linguistics, Dec. 2023, pp. 10859–10885. [Online]. Available: <https://aclanthology.org/2023.findings-emnlp.725>
- J. Chen, X. Wang, A. Gao, F. Jiang, S. Chen, H. Zhang, D. Song, W. Xie, C. Kong, J. Li, X. Wan, H. Li, and B. Wang, "Huatuogpt-ii, one-stage training for medical adaption of llms," *CoRR*, vol. abs/2311.09774, 2023. [Online]. Available: <https://doi.org/10.48550/arXiv.2311.09774>
- X. Zhang and Q. Yang, "Xuanyuan 2.0: A large chinese financial chat model with hundreds of billions parameters," in *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management, CIKM 2023, Birmingham, United Kingdom, October 21-25, 2023*, I. Frommholz, F. Hopfgartner, M. Lee, M. Oakes, M. Lalmas, M. Zhang, and R. L. T. Santos, Eds. ACM, 2023, pp. 4435–4439. [Online]. Available: <https://doi.org/10.1145/3583780.3615285>
- T. Xie, Y. Wan, W. Huang, Z. Yin, Y. Liu, S. Wang, Q. Linghu, C. Kit, C. Grazian, W. Zhang, I. Razzak, and B. Hoex, "DARWIN series: Domain specific large language models for natural science," *CoRR*, vol. abs/2308.13565, 2023. [Online]. Available: <https://doi.org/10.48550/arXiv.2308.13565>
- Y. Dan, Z. Lei, Y. Gu, Y. Li, J. Yin, J. Lin, L. Ye, Z. Tie, Y. Zhou, Y. Wang, A. Zhou, Z. Zhou, Q. Chen, J. Zhou, L. He, and X. Qiu, "Educhat: A large-scale language model-based chatbot system for intelligent education," *CoRR*, vol. abs/2308.02773, 2023. [Online]. Available: <https://doi.org/10.48550/arXiv.2308.02773>
- H. Guo, J. Yang, J. Liu, L. Yang, L. Chai, J. Bai, J. Peng, X. Hu, C. Chen, D. Zhang, X. Shi, T. Zheng, L. Zheng, B. Zhang, K. Xu, and Z. Li, "OWL: A large language model for IT operations," *CoRR*, vol. abs/2309.09298, 2023. [Online]. Available: <https://doi.org/10.48550/arXiv.2309.09298>
- Y. Kim and A. M. Rush, "Sequence-level knowledge distillation," *arXiv preprint arXiv:1606.07947*, 2016.
- S. Han, H. Mao, and W. J. Dally, "Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding," *International Conference on Learning Representations (ICLR)*, 2016.
- V. Gangal, S. Y. Feng, M. Alikhani, T. Mitamura, and E. Hovy, "Nareor: The narrative reordering problem," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 10, 2022, pp. 10645–10653.
- S. Longpre, Y. Lu, Z. Tu, and C. DuBois, "An exploration of data augmentation and sampling techniques for domain-agnostic question answering," in *Proceedings of the 2nd Workshop on Machine Reading for Question Answering*, A. Fisch, A. Talmor, R. Jia, M. Seo, E. Choi, and D. Chen, Eds. Hong Kong, China: Association for Computational Linguistics, Nov. 2019, pp. 220–227. [Online]. Available: <https://aclanthology.org/D19-5829>
- P. West, C. Bhagavatula, J. Hessel, J. Hwang, L. Jiang, R. Le Bras, X. Lu, S. Welleck, and Y. Choi, "Symbolic knowledge distillation: from general language models to commonsense models," in *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, M. Carpuat, M.-C. de Marneffe, and I. V. Meza Ruiz, Eds. Seattle, United States: Association for Computational Linguistics, Jul. 2022, pp. 4602–4625. [Online]. Available: <https://aclanthology.org/2022.naacl-main.341>
- Z. Li, X. Xu, T. Shen, C. Xu, J.-C. Gu, and C. Tao, "Leveraging large language models for nlg evaluation: A survey," 2024.
- S. Li, J. Chen, Y. Shen, Z. Chen, X. Zhang, Z. Li, H. Wang, J. Qian, B. Peng, Y. Mao, W. Chen, and X. Yan, "Explanations from large language models make small reasoners better," 2022.
- N. Ho, L. Schmid, and S. Yun, "Large language models are reasoning teachers," in *ACL (1)*. Association for Computational Linguistics, 2023, pp. 14852–14882.
- L. C. Magister, J. Mallinson, J. Adamek, E. Malmi, and A. Severyn, "Teaching small language models to reason," in *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, A. Rogers, J. Boyd-Graber, and N. Okazaki, Eds. Toronto, Canada: Association for Computational Linguistics, Jul. 2023, pp. 1773–1781. [Online]. Available: <https://aclanthology.org/2023.acl-short.151>
- Y. Fu, H. Peng, L. Ou, A. Sabharwal, and T. Khot, "Specializing smaller language models towards multi-step reasoning," 2023.
- L. H. Li, J. Hessel, Y. Yu, X. Ren, K.-W. Chang, and Y. Choi, "Symbolic chain-of-thought distillation: Small models can also "think" step-by-step," in *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, A. Rogers, J. Boyd-Graber,

- and N. Okazaki, Eds. Toronto, Canada: Association for Computational Linguistics, Jul. 2023, pp. 2665–2679. [Online]. Available: <https://aclanthology.org/2023.acl-long.150>
- W. Liu, G. Li, K. Zhang, B. Du, Q. Chen, X. Hu, H. Xu, J. Chen, and J. Wu, “Mind’s mirror: Distilling self-evaluation capability and comprehensive thinking from large language models,” 2023.
- S. Longpre, L. Hou, T. Vu, A. Webson, H. W. Chung, Y. Tay, D. Zhou, Q. V. Le, B. Zoph, J. Wei *et al.*, “The flan collection: Designing data and methods for effective instruction tuning,” *arXiv preprint arXiv:2301.13688*, 2023.
- Y. Anand, Z. Nussbaum, B. Duderstadt, B. Schmidt, and A. Mulyar, “Gpt4all: Training an assistant-style chatbot with large scale data distillation from gpt-3.5-turbo,” *GitHub*, 2023.
- Q. Si, T. Wang, Z. Lin, X. Zhang, Y. Cao, and W. Wang, “An empirical study of instruction-tuning large language models in chinese,” in *EMNLP (Findings)*. Association for Computational Linguistics, 2023, pp. 4086–4107.
- Y. Ji, Y. Deng, Y. Gong, Y. Peng, Q. Niu, L. Zhang, B. Ma, and X. Li, “Exploring the impact of instruction data scaling on large language models: An empirical study on real-world use cases,” 2023.
- M. Wu, A. Waheed, C. Zhang, M. Abdul-Mageed, and A. F. Aji, “Lamini-lm: A diverse herd of distilled models from large-scale instructions,” 2023.
- W. Guo, J. Yang, K. Yang, X. Li, Z. Rao, Y. Xu, and D. Niu, “Instruction fusion: Advancing prompt evolution through hybridization,” 2023.
- Y. Yu, Y. Zhuang, J. Zhang, Y. Meng, A. Ratner, R. Krishna, J. Shen, and C. Zhang, “Large language model as attributed training data generator: A tale of diversity and bias,” 2023.
- F. Wan, X. Huang, D. Cai, X. Quan, W. Bi, and S. Shi, “Knowledge fusion of large language models,” in *The Twelfth International Conference on Learning Representations*, 2024. [Online]. Available: <https://openreview.net/forum?id=jiDsk12qcz>
- Q. Zhao and B. Zhu, “Towards the fundamental limits of knowledge transfer over finite domains,” in *NeurIPS 2023 Workshop on Mathematics of Modern Machine Learning*, 2023. [Online]. Available: <https://openreview.net/forum?id=9qxoXqxa0N>
- C. Qin, W. Xia, F. Jiao, and S. Joty, “Improving in-context learning via bidirectional alignment,” 2023.
- N. Boizard, K. El-Haddad, C. Hudelot, and P. Colombo, “Towards cross-tokenizer distillation: the universal logit distillation loss for llms,” *arXiv preprint arXiv:2402.12030*, 2024.
- Q. Zhong, L. Ding, L. Shen, J. Liu, B. Du, and D. Tao, “Revisiting knowledge distillation for autoregressive language models,” 2024.
- M. Kim, S. Lee, J. Lee, S. Hong, D.-S. Chang, W. Sung, and J. Choi, “Token-scaled logit distillation for ternary weight generative language models,” *arXiv preprint arXiv:2308.06744*, 2023.
- Z. Chen, K. Zhou, W. X. Zhao, J. Wan, F. Zhang, D. Zhang, and J.-R. Wen, “Improving large language models via fine-grained reinforcement learning with minimum editing constraint,” 2024.
- G. Guo, R. Zhao, T. Tang, X. Zhao, and J.-R. Wen, “Beyond imitation: Leveraging fine-grained quality signals for alignment,” in *The Twelfth International Conference on Learning Representations*, 2024. [Online]. Available: <https://openreview.net/forum?id=LNLjU5C5dK>
- Z. Allen-Zhu and Y. Li, “Towards understanding ensemble, knowledge distillation and self-distillation in deep learning,” *arXiv preprint arXiv:2012.09816*, 2020.
- T. Zheng, S. Guo, X. Qu, J. Guo, W. Zhang, X. Du, C. Lin, W. Huang, W. Chen, J. Fu *et al.*, “Kun: Answer polishment for chinese self-alignment with instruction back-translation,” *arXiv preprint arXiv:2401.06477*, 2024.
- X. Li, P. Yu, C. Zhou, T. Schick, O. Levy, L. Zettlemoyer, J. E. Weston, and M. Lewis, “Self-alignment with instruction backtranslation,” in *The Twelfth International Conference on Learning Representations*, 2024. [Online]. Available: <https://openreview.net/forum?id=1oijHJBRsT>
- B. Zhao, H. Hajishirzi, and Q. Cao, “Apt: Adaptive pruning and tuning pretrained language models for efficient training and inference,” *arXiv preprint arXiv:2401.12200*, 2024.
- A. Singh, J. D. Co-Reyes, R. Agarwal, A. Anand, P. Patil, P. J. Liu, J. Harrison, J. Lee, K. Xu, A. Parisi *et al.*, “Beyond human data: Scaling self-training for problem-solving with language models,” *arXiv preprint arXiv:2312.06585*, 2023.
- W. Chen, D. Song, and B. Li, “Grath: Gradual self-truthifying for large language models,” 2024.
- A. Hosseini, X. Yuan, N. Malkin, A. Courville, A. Sordoni, and R. Agarwal, “V-star: Training verifiers for self-taught reasoners,” 2024.
- A. Askell, Y. Bai, A. Chen, D. Drain, D. Ganguli, T. Henighan, A. Jones, N. Joseph, B. Mann, N. DasSarma, N. Elhage, Z. Hatfield-Dodds, D. Hernandez, J. Kernion, K. Ndousse, C. Olsson, D. Amodei, T. Brown, J. Clark, S. McCandlish, C. Olah, and J. Kaplan, “A general language assistant as a laboratory for alignment,” 2021.
- J. Huang, S. Gu, L. Hou, Y. Wu, X. Wang, H. Yu, and J. Han, “Large language models can self-improve,” in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, H. Bouamor, J. Pino, and K. Bali, Eds. Singapore: Association for Computational Linguistics, Dec. 2023, pp. 1051–1068. [Online]. Available: <https://aclanthology.org/2023.emnlp-main.67>
- H. Chen, X. Quan, H. Chen, M. Yan, and J. Zhang, “Knowledge distillation for closed-source language models,” *arXiv preprint arXiv:2401.07013*, 2024.
- I. Sason and S. Verdú, “ f -divergence inequalities,” *IEEE Transactions on Information Theory*, vol. 62, no. 11, pp. 5973–6006, 2016.
- S. Sun, Y. Cheng, Z. Gan, and J. Liu, “Patient knowledge distillation for bert model compression,” 2019.
- Z. Sun, H. Yu, X. Song, R. Liu, Y. Yang, and D. Zhou, “MobileBERT: a compact task-agnostic BERT for resource-limited devices,” in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, D. Jurafsky, J. Chai, N. Schlueter, and J. Tetreault, Eds. Online: Association for Computational Linguistics, Jul. 2020, pp. 2158–2170. [Online]. Available: <https://aclanthology.org/2020.acl-main.195>
- X. Jiao, Y. Yin, L. Shang, X. Jiang, X. Chen, L. Li, F. Wang, and Q. Liu, “TinyBERT: Distilling BERT for natural language understanding,” in *Findings of the Association for*

- Computational Linguistics: EMNLP 2020*, T. Cohn, Y. He, and Y. Liu, Eds. Online: Association for Computational Linguistics, Nov. 2020, pp. 4163–4174. [Online]. Available: <https://aclanthology.org/2020.findings-emnlp.372>
- L. Hou, Z. Huang, L. Shang, X. Jiang, X. Chen, and Q. Liu, “Dynabert: Dynamic bert with adaptive width and depth,” *Advances in Neural Information Processing Systems*, vol. 33, pp. 9782–9793, 2020.
- S. Zuo, Q. Zhang, C. Liang, P. He, T. Zhao, and W. Chen, “Moebert: from bert to mixture-of-experts via importance-guided adaptation,” *arXiv preprint arXiv:2204.07675*, 2022.
- K. J. Liang, W. Hao, D. Shen, Y. Zhou, W. Chen, C. Chen, and L. Carin, “Mixkd: Towards efficient distillation of large-scale language models,” in *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. [Online]. Available: <https://openreview.net/forum?id=UFGEeljkLu5>
- Y. J. Ma, W. Liang, G. Wang, D.-A. Huang, O. Bastani, D. Jayaraman, Y. Zhu, L. Fan, and A. Anandkumar, “Eureka: Human-level reward design via coding large language models,” 2023.
- J.-C. Pang, P. Wang, K. Li, X.-H. Chen, J. Xu, Z. Zhang, and Y. Yu, “Language model self-improvement by reinforcement learning contemplation,” 2023.
- Y. Du, O. Watkins, Z. Wang, C. Colas, T. Darrell, P. Abbeel, A. Gupta, and J. Andreas, “Guiding pretraining in reinforcement learning with large language models,” in *Proceedings of the 40th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, A. Krause, E. Brunskill, K. Cho, B. Engelhardt, S. Sabato, and J. Scarlett, Eds., vol. 202. PMLR, 23–29 Jul 2023, pp. 8657–8677. [Online]. Available: <https://proceedings.mlr.press/v202/du23f.html>
- J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, “Proximal policy optimization algorithms,” 2017.
- R. Rafailov, A. Sharma, E. Mitchell, S. Ermon, C. D. Manning, and C. Finn, “Direct preference optimization: Your language model is secretly a reward model,” 2023.
- F. Song, B. Yu, M. Li, H. Yu, F. Huang, Y. Li, and H. Wang, “Preference ranking optimization for human alignment,” *arXiv preprint arXiv:2306.17492*, 2023.
- Z. Yuan, H. Yuan, C. Tan, W. Wang, S. Huang, and F. Huang, “Rrhf: Rank responses to align language models with human feedback without tears,” *arXiv preprint arXiv:2304.05302*, 2023.
- M. Li, L. Chen, J. Chen, S. He, and T. Zhou, “Reflection-tuning: Recycling data for better instruction-tuning,” in *NeurIPS 2023 Workshop on Instruction Tuning and Instruction Following*, 2023. [Online]. Available: <https://openreview.net/forum?id=xaqoZZqkPU>
- M. Li, L. Chen, J. Chen, S. He, J. Gu, and T. Zhou, “Selective reflection-tuning: Student-selected data recycling for llm instruction-tuning,” 2024. [Online]. Available: <https://api.semanticscholar.org/CorpusID:267682220>
- X. Geng, A. Gudibande, H. Liu, E. Wallace, P. Abbeel, S. Levine, and D. Song, “Koala: A dialogue model for academic research,” Blog post, April 2023. [Online]. Available: <https://bair.berkeley.edu/blog/2023/04/03/koala/>
- M. Li, J. Chen, L. Chen, and T. Zhou, “Can llms speak for diverse people? tuning llms via debate to generate controllable controversial statements,” 2024.
- M. Kang, S. Lee, J. Baek, K. Kawaguchi, and S. J. Hwang, “Knowledge-augmented reasoning distillation for small language models in knowledge-intensive tasks,” 2023.
- R. Yang, L. Song, Y. Li, S. Zhao, Y. Ge, X. Li, and Y. Shan, “Gpt4tools: Teaching large language model to use tools via self-instruction,” 2023.
- A. Yehudai, B. Carmeli, Y. Mass, O. Arviv, N. Mills, A. Toledo, E. Shnarch, and L. Choshen, “Genie: Achieving human parity in content-grounded datasets generation,” 2024.
- Y. Zhang, R. Zhang, J. Gu, Y. Zhou, N. Lipka, D. Yang, and T. Sun, “Llavar: Enhanced visual instruction tuning for text-rich image understanding,” 2023.
- C. Lyu, M. Wu, L. Wang, X. Huang, B. Liu, Z. Du, S. Shi, and Z. Tu, “Macaw-llm: Multi-modal language modeling with image, audio, video, and text integration,” *arXiv preprint arXiv:2306.09093*, 2023.
- B. Li, Y. Zhang, L. Chen, J. Wang, F. Pu, J. Yang, C. Li, and Z. Liu, “Mimic-it: Multi-modal in-context instruction tuning,” 2023.
- Z. Zhao, L. Guo, T. Yue, S. Chen, S. Shao, X. Zhu, Z. Yuan, and J. Liu, “Chatbridge: Bridging modalities with large language model as a language catalyst,” 2023.
- Y. Zhao, B. Yu, B. Hui, H. Yu, F. Huang, Y. Li, and N. L. Zhang, “A preliminary study of the intrinsic relationship between complexity and alignment,” 2023.
- A. Gudibande, E. Wallace, C. Snell, X. Geng, H. Liu, P. Abbeel, S. Levine, and D. Song, “The false promise of imitating proprietary llms,” *arXiv preprint arXiv:2305.15717*, 2023.
- C. Zhou, P. Liu, P. Xu, S. Iyer, J. Sun, Y. Mao, X. Ma, A. Efrat, P. Yu, L. YU, S. Zhang, G. Ghosh, M. Lewis, L. Zettlemoyer, and O. Levy, “LIMA: Less is more for alignment,” in *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. [Online]. Available: <https://openreview.net/forum?id=KBMOKmX2he>
- M. Li, Y. Zhang, S. He, Z. Li, H. Zhao, J. Wang, N. Cheng, and T. Zhou, “Superfiltering: Weak-to-strong data filtering for fast instruction-tuning,” 2024. [Online]. Available: <https://api.semanticscholar.org/CorpusID:267365346>
- B. Xu, A. Yang, J. Lin, Q. Wang, C. Zhou, Y. Zhang, and Z. Mao, “Expertprompting: Instructing large language models to be distinguished experts,” 2023.
- W. Liu, W. Zeng, K. He, Y. Jiang, and J. He, “What makes good data for alignment? a comprehensive study of automatic data selection in instruction tuning,” 2023.
- R. Lou, K. Zhang, J. Xie, Y. Sun, J. Ahn, H. Xu, Y. Su, and W. Yin, “Muffin: Curating multi-faceted instructions for improving instruction-following,” 2023.
- T. Schick, J. Dwivedi-Yu, Z. Jiang, F. Petroni, P. Lewis, G. Izacard, Q. You, C. Nalmpantis, E. Grave, and S. Riedel, “Peer: A collaborative language model,” 2022.
- A. Madaan, N. Tandon, P. Gupta, S. Hallinan, L. Gao, S. Wiegrefe, U. Alon, N. Dziri, S. Prabhunoye, Y. Yang, S. Gupta, B. P. Majumder, K. Hermann, S. Welleck, A. Yazdanbakhsh, and P. Clark, “Self-refine: Iterative refinement with self-feedback,” 2023.
- W. Saunders, C. Yeh, J. Wu, S. Bills, L. Ouyang, J. Ward, and J. Leike, “Self-critiquing models for assisting human

- evaluators," 2022.
- D. M. Ziegler, N. Stiennon, J. Wu, T. B. Brown, A. Radford, D. Amodei, P. Christiano, and G. Irving, "Fine-tuning language models from human preferences," *arXiv preprint arXiv:1909.08593*, 2019.
- N. Stiennon, L. Ouyang, J. Wu, D. Ziegler, R. Lowe, C. Voss, A. Radford, D. Amodei, and P. F. Christiano, "Learning to summarize with human feedback," *Advances in Neural Information Processing Systems*, vol. 33, pp. 3008–3021, 2020.
- J. Wu, L. Ouyang, D. M. Ziegler, N. Stiennon, R. Lowe, J. Leike, and P. Christiano, "Recursively summarizing books with human feedback," 2021.
- Y. Bai, A. Jones, K. Ndousse, A. Askell, A. Chen, N. DasSarma, D. Drain, S. Fort, D. Ganguli, T. Henighan *et al.*, "Training a helpful and harmless assistant with reinforcement learning from human feedback," *arXiv preprint arXiv:2204.05862*, 2022.
- A. Köpf, Y. Kilcher, D. von Rütte, S. Anagnostidis, Z.-R. Tam, K. Stevens, A. Barhoum, N. M. Duc, O. Stanley, R. Nagyfi, S. ES, S. Suri, D. Glushkov, A. Dantuluri, A. Maguire, C. Schuhmann, H. Nguyen, and A. Mattick, "Openassistant conversations – democratizing large language model alignment," 2023.
- G. Wang, S. Cheng, X. Zhan, X. Li, S. Song, and Y. Liu, "Openchat: Advancing open-source language models with mixed-quality data," 2023.
- L. Weidinger, J. Mellor, M. Rauh, C. Griffin, J. Uesato, P.-S. Huang, M. Cheng, M. Glaese, B. Balle, A. Kasirzadeh, Z. Kenton, S. Brown, W. Hawkins, T. Stepleton, C. Biles, A. Birhane, J. Haas, L. Rimell, L. A. Hendricks, W. Isaac, S. Legassick, G. Irving, and I. Gabriel, "Ethical and social risks of harm from language models," 2021.
- J. Ji, M. Liu, J. Dai, X. Pan, C. Zhang, C. Bian, C. Zhang, R. Sun, Y. Wang, and Y. Yang, "Beavertails: Towards improved safety alignment of llm via a human-preference dataset," 2023.
- I. Solaiman and C. Dennison, "Process for adapting language models to society (palms) with values-targeted datasets," *Advances in Neural Information Processing Systems*, vol. 34, pp. 5861–5873, 2021.
- L. Qiu, Y. Zhao, J. Li, P. Lu, B. Peng, J. Gao, and S.-C. Zhu, "Valuenet: A new dataset for human value driven dialogue system," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 10, 2022, pp. 11 183–11 191.
- J. Kiesel, M. Alshomary, N. Handke, X. Cai, H. Wachsmuth, and B. Stein, "Identifying the human values behind arguments," in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, S. Muresan, P. Nakov, and A. Villavicencio, Eds. Dublin, Ireland: Association for Computational Linguistics, May 2022, pp. 4459–4471. [Online]. Available: <https://aclanthology.org/2022.acl-long.306>
- R. Liu, G. Zhang, X. Feng, and S. Vosoughi, "Aligning generative language models with human values," in *Findings of the Association for Computational Linguistics: NAACL 2022*, M. Carpuat, M.-C. de Marneffe, and I. V. Meza Ruiz, Eds. Seattle, United States: Association for Computational Linguistics, Jul. 2022, pp. 241–252. [Online]. Available: <https://aclanthology.org/2022.findings-naacl.18>
- A. Glaese, N. McAleese, M. Trebacz, J. Aslanides, V. Firoiu, T. Ewalds, M. Rauh, L. Weidinger, M. Chadwick, P. Thacker *et al.*, "Improving alignment of dialogue agents via targeted human judgements," *arXiv preprint arXiv:2209.14375*, 2022.
- H. Sun, Z. Zhang, F. Mi, Y. Wang, W. Liu, J. Cui, B. Wang, Q. Liu, and M. Huang, "MoralDial: A framework to train and evaluate moral dialogue systems via moral discussions," in *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, A. Rogers, J. Boyd-Graber, and N. Okazaki, Eds. Toronto, Canada: Association for Computational Linguistics, Jul. 2023, pp. 2213–2230. [Online]. Available: <https://aclanthology.org/2023.acl-long.123>
- J. Yao, X. Yi, X. Wang, J. Wang, and X. Xie, "From instructions to intrinsic human values – a survey of alignment goals for big models," 2023.
- Y. Liu, Y. Yao, J.-F. Ton, X. Zhang, R. G. H. Cheng, Y. Klochkov, M. F. Taufiq, and H. Li, "Trustworthy llms: a survey and guideline for evaluating large language models' alignment," *arXiv preprint arXiv:2308.05374*, 2023.
- J. Qian, H. Wang, Z. Li, S. Li, and X. Yan, "Limitations of language models in arithmetic and symbolic induction," 2022.
- X. She, Y. Liu, Y. Zhao, Y. He, L. Li, C. Tantithamthavorn, Z. Qin, and H. Wang, "Pitfalls in language models for code intelligence: A taxonomy and survey," 2023.
- H. Manikandan, Y. Jiang, and J. Z. Kolter, "Language models are weak learners," 2023.
- Y. Liang, C. Wu, T. Song, W. Wu, Y. Xia, Y. Liu, Y. Ou, S. Lu, L. Ji, S. Mao, Y. Wang, L. Shou, M. Gong, and N. Duan, "Taskmatrix.ai: Completing tasks by connecting foundation models with millions of apis," 2023.
- G. Mialon, R. Dessi, M. Lomeli, C. Nalmpantis, R. Pasunuru, R. Raileanu, B. Rozière, T. Schick, J. Dwivedi-Yu, A. Celikyilmaz, E. Grave, Y. LeCun, and T. Scialom, "Augmented language models: a survey," 2023.
- A. Parisi, Y. Zhao, and N. Fiedel, "Talm: Tool augmented language models," 2022.
- R. Nakano, J. Hilton, S. Balaji, J. Wu, L. Ouyang, C. Kim, C. Hesse, S. Jain, V. Kosaraju, W. Saunders, X. Jiang, K. Cobbe, T. Eloundou, G. Krueger, K. Button, M. Knight, B. Chess, and J. Schulman, "Webgpt: Browser-assisted question-answering with human feedback," 2022.
- Y. Qin, Z. Cai, D. Jin, L. Yan, S. Liang, K. Zhu, Y. Lin, X. Han, N. Ding, H. Wang, R. Xie, F. Qi, Z. Liu, M. Sun, and J. Zhou, "WebCPM: Interactive web search for Chinese long-form question answering," in *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, A. Rogers, J. Boyd-Graber, and N. Okazaki, Eds. Toronto, Canada: Association for Computational Linguistics, Jul. 2023, pp. 8968–8988. [Online]. Available: <https://aclanthology.org/2023.acl-long.499>
- Y. Song, W. Xiong, D. Zhu, W. Wu, H. Qian, M. Song, H. Huang, C. Li, K. Wang, R. Yao, Y. Tian, and S. Li, "Restgpt: Connecting large language models with real-world restful apis," 2023.
- T. Cai, X. Wang, T. Ma, X. Chen, and D. Zhou, "Large language models as tool makers," 2023.

- Y. Shen, K. Song, X. Tan, D. Li, W. Lu, and Y. Zhuang, "Hugginggpt: Solving ai tasks with chatgpt and its friends in hugging face," 2023.
- S. Hao, T. Liu, Z. Wang, and Z. Hu, "Toolkengpt: Augmenting frozen language models with massive tools via tool embeddings," 2024.
- S. Yuan, K. Song, J. Chen, X. Tan, Y. Shen, R. Kan, D. Li, and D. Yang, "Easytool: Enhancing llm-based agents with concise tool instruction," 2024.
- S. Zhang, S. Roller, N. Goyal, M. Artetxe, M. Chen, S. Chen, C. Dewan, M. Diab, X. Li, X. V. Lin, T. Mihaylov, M. Ott, S. Shleifer, K. Shuster, D. Simig, P. S. Koura, A. Sridhar, T. Wang, and L. Zettlemoyer, "Opt: Open pre-trained transformer language models," 2022.
- T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell *et al.*, "Language models are few-shot learners," *Advances in neural information processing systems*, vol. 33, pp. 1877–1901, 2020.
- W. Huang, P. Abbeel, D. Pathak, and I. Mordatch, "Language models as zero-shot planners: Extracting actionable knowledge for embodied agents," in *International Conference on Machine Learning*. PMLR, 2022, pp. 9118–9147.
- I. Singh, V. Blukis, A. Mousavian, A. Goyal, D. Xu, J. Tremblay, D. Fox, J. Thomason, and A. Garg, "Progprompt: Generating situated robot task plans using large language models," 2022.
- D. Zhou, N. Schärli, L. Hou, J. Wei, N. Scales, X. Wang, D. Schuurmans, C. Cui, O. Bousquet, Q. Le, and E. Chi, "Least-to-most prompting enables complex reasoning in large language models," 2023.
- C. H. Song, J. Wu, C. Washington, B. M. Sadler, W.-L. Chao, and Y. Su, "Llm-planner: Few-shot grounded planning for embodied agents with large language models," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 2998–3009.
- Z. Wang, S. Cai, A. Liu, X. Ma, and Y. Liang, "Describe, explain, plan and select: Interactive planning with large language models enables open-world multi-task agents," *arXiv preprint arXiv:2302.01560*, 2023.
- S. Yao, D. Yu, J. Zhao, I. Shafran, T. L. Griffiths, Y. Cao, and K. Narasimhan, "Tree of thoughts: Deliberate problem solving with large language models," *arXiv preprint arXiv:2305.10601*, 2023.
- B. Liu, Y. Jiang, X. Zhang, Q. Liu, S. Zhang, J. Biswas, and P. Stone, "Llm+ p: Empowering large language models with optimal planning proficiency," *arXiv preprint arXiv:2304.11477*, 2023.
- S. Hao, Y. Gu, H. Ma, J. J. Hong, Z. Wang, D. Z. Wang, and Z. Hu, "Reasoning with language model is planning with world model," *arXiv preprint arXiv:2305.14992*, 2023.
- M. Hu, Y. Mu, X. Yu, M. Ding, S. Wu, W. Shao, Q. Chen, B. Wang, Y. Qiao, and P. Luo, "Tree-planner: Efficient close-loop task planning with large language models," *arXiv preprint arXiv:2310.08582*, 2023.
- B. Y. Lin, C. Huang, Q. Liu, W. Gu, S. Sommerer, and X. Ren, "On grounded planning for embodied tasks with language models," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 37, no. 11, 2023, pp. 13 192–13 200.
- K. Valmeekam, M. Marquez, S. Sreedharan, and S. Kambhampati, "On the planning abilities of large language models - a critical investigation," in *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. [Online]. Available: <https://openreview.net/forum?id=X6dEqXIsEW>
- T. Sumers, K. Marino, A. Ahuja, R. Fergus, and I. Dasgupta, "Distilling internet-scale vision-language models into embodied agents," in *Proceedings of the 40th International Conference on Machine Learning*, ser. ICML'23. JMLR.org, 2023.
- Y. Yang, T. Zhou, K. Li, D. Tao, L. Li, L. Shen, X. He, J. Jiang, and Y. Shi, "Embodied multi-modal agent trained by an llm from a parallel textworld," 2023.
- A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.
- Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized bert pretraining approach," 2019.
- J. Li, L. Gui, Y. Zhou, D. West, C. Aloisi, and Y. He, "Distilling chatgpt for explainable automated student answer assessment," in *EMNLP (Findings)*. Association for Computational Linguistics, 2023, pp. 6007–6026.
- R. Tang, X. Han, X. Jiang, and X. Hu, "Does synthetic data generation of llms help clinical text mining?" *arXiv preprint arXiv:2303.04360*, 2023.
- X. He, I. Nassar, J. Kiros, G. Haffari, and M. Norouzi, "Generate, annotate, and learn: NLP with synthetic text," *Trans. Assoc. Comput. Linguistics*, vol. 10, pp. 826–842, 2022. [Online]. Available: <https://transacl.org/ojs/index.php/tacl/article/view/3811>
- Y. Meng, J. Huang, Y. Zhang, and J. Han, "Generating training data with language models: Towards zero-shot language understanding," in *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022. [Online]. Available: http://papers.nips.cc/paper_files/paper/2022/hash/0346c148ba1c21c6b4780a961ea141dc-Abstract-Conference.html
- J. Wang, Z. Yao, A. Mitra, S. Osebe, Z. Yang, and H. Yu, "UMASS_BioNLP at MEDIQA-chat 2023: Can LLMs generate high-quality synthetic note-oriented doctor-patient conversations?" in *Proceedings of the 5th Clinical Natural Language Processing Workshop*, T. Naumann, A. Ben Abacha, S. Bethard, K. Roberts, and A. Rumshisky, Eds. Toronto, Canada: Association for Computational Linguistics, Jul. 2023, pp. 460–471. [Online]. Available: <https://aclanthology.org/2023.clinicalnlp-1.49>
- Z. Yang, S. Cherian, and S. Vucetic, "Data augmentation for radiology report simplification," in *Findings of the Association for Computational Linguistics: EACL 2023*, A. Vlachos and I. Augenstein, Eds. Dubrovnik, Croatia: Association for Computational Linguistics, May 2023, pp. 1922–1932. [Online]. Available: <https://aclanthology.org/2023.findings-eacl.144>
- Z. Cai, C. Tao, T. Shen, C. Xu, X. Geng, X. A. Lin, L. He, and D. Jiang, "Hyper: Multitask hyper-prompted training enables large-scale retrieval generalization," in *The Eleventh*

- International Conference on Learning Representations*, 2022.
- C. Liu, C. Tao, X. Geng, T. Shen, D. Zhao, C. Xu, B. Jiao, and D. Jiang, "Adam: Dense retrieval distillation with adaptive dark examples," *arXiv preprint arXiv:2212.10192*, 2022.
- J. Feng, C. Tao, X. Geng, T. Shen, C. Xu, G. Long, D. Zhao, and D. Jiang, "Knowledge refinement via interaction between search engines and large language models," *arXiv preprint arXiv:2305.07402*, 2023.
- T. Shen, G. Long, X. Geng, C. Tao, T. Zhou, and D. Jiang, "Large language models are strong zero-shot retriever," *arXiv preprint arXiv:2304.14233*, 2023.
- X. Ma, X. Zhang, R. Pradeep, and J. Lin, "Zero-shot listwise document reranking with a large language model," 2023.
- Z. Qin, R. Jagerman, K. Hui, H. Zhuang, J. Wu, J. Shen, T. Liu, J. Liu, D. Metzler, X. Wang, and M. Bendersky, "Large language models are effective text rankers with pairwise ranking prompting," 2023.
- X. Ma, Y. Gong, P. He, H. Zhao, and N. Duan, "Query rewriting in retrieval-augmented large language models," in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, H. Bouamor, J. Pino, and K. Bali, Eds. Singapore: Association for Computational Linguistics, Dec. 2023, pp. 5303–5315. [Online]. Available: <https://aclanthology.org/2023.emnlp-main.322>
- D. Sachan, M. Lewis, M. Joshi, A. Aghajanyan, W.-t. Yih, J. Pineau, and L. Zettlemoyer, "Improving passage retrieval with zero-shot question generation," in *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, Y. Goldberg, Z. Kozareva, and Y. Zhang, Eds. Abu Dhabi, United Arab Emirates: Association for Computational Linguistics, Dec. 2022, pp. 3781–3797. [Online]. Available: <https://aclanthology.org/2022.emnlp-main.249>
- D. S. Sachan, M. Lewis, D. Yogatama, L. Zettlemoyer, J. Pineau, and M. Zaheer, "Questions are all you need to train a dense passage retriever," *Transactions of the Association for Computational Linguistics*, vol. 11, pp. 600–616, 2023. [Online]. Available: <https://aclanthology.org/2023.tacl-1.35>
- T. Schick and H. Schütze, "Generating datasets with pretrained language models," in *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, M.-F. Moens, X. Huang, L. Specia, and S. W.-t. Yih, Eds. Online and Punta Cana, Dominican Republic: Association for Computational Linguistics, Nov. 2021, pp. 6943–6951. [Online]. Available: <https://aclanthology.org/2021.emnlp-main.555>
- Z. Peng, X. Wu, and Y. Fang, "Soft prompt tuning for augmenting dense retrieval with large language models," *arXiv preprint arXiv:2307.08303*, 2023.
- J. Saad-Falcon, O. Khattab, K. Santhanam, R. Florian, M. Franz, S. Roukos, A. Sil, M. A. Sultan, and C. Potts, "UDAPDR: unsupervised domain adaptation via LLM prompting and distillation of rerankers," in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, 2023, pp. 11 265–11 279. [Online]. Available: <https://aclanthology.org/2023.emnlp-main.693>
- V. Jeronimo, L. Bonifacio, H. Abonizio, M. Fadaee, R. Lotufo, J. Zavrel, and R. Nogueira, "Inpars-v2: Large language models as efficient dataset generators for information retrieval," *arXiv preprint arXiv:2301.01820*, 2023.
- W. Sun, Z. Chen, X. Ma, L. Yan, S. Wang, P. Ren, Z. Chen, D. Yin, and Z. Ren, "Instruction distillation makes large language models efficient zero-shot rankers," 2023.
- C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," *J. Mach. Learn. Res.*, vol. 21, no. 1, jan 2020.
- S. Bruch, X. Wang, M. Bendersky, and M. Najork, "An analysis of the softmax cross entropy loss for learning-to-rank with binary relevance," in *Proceedings of the 2019 ACM SIGIR International Conference on Theory of Information Retrieval, ICTIR 2019, Santa Clara, CA, USA, October 2-5, 2019*, 2019, pp. 75–78. [Online]. Available: <https://doi.org/10.1145/3341981.3344221>
- C. Burges, T. Shaked, E. Renshaw, A. Lazier, M. Deeds, N. Hamilton, and G. Hullender, "Learning to rank using gradient descent," in *Proceedings of the 22nd International Conference on Machine Learning*, ser. ICML '05. New York, NY, USA: Association for Computing Machinery, 2005, p. 89–96. [Online]. Available: <https://doi.org/10.1145/1102351.1102363>
- X. Wang, C. Li, N. Golbandi, M. Bendersky, and M. Najork, "The lambdaloss framework for ranking metric optimization," in *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, ser. CIKM '18. New York, NY, USA: Association for Computing Machinery, 2018, p. 1313–1322. [Online]. Available: <https://doi.org/10.1145/3269206.3271784>
- W. Wang, X. Lin, F. Feng, X. He, and T.-S. Chua, "Generative recommendation: Towards next-generation recommender paradigm," 2023.
- S. Dai, N. Shao, H. Zhao, W. Yu, Z. Si, C. Xu, Z. Sun, X. Zhang, and J. Xu, "Uncovering chatgpt's capabilities in recommender systems," in *Proceedings of the 17th ACM Conference on Recommender Systems*, ser. RecSys '23. New York, NY, USA: Association for Computing Machinery, 2023, p. 1126–1132. [Online]. Available: <https://doi.org/10.1145/3604915.3610646>
- Y. Xi, W. Liu, J. Lin, X. Cai, H. Zhu, J. Zhu, B. Chen, R. Tang, W. Zhang, R. Zhang, and Y. Yu, "Towards open-world recommendation with knowledge augmentation from large language models," 2023.
- X. Ren, W. Wei, L. Xia, L. Su, S. Cheng, J. Wang, D. Yin, and C. Huang, "Representation learning with large language models for recommendation," 2023.
- W. Wei, X. Ren, J. Tang, Q. Wang, L. Su, S. Cheng, J. Wang, D. Yin, and C. Huang, "Llmrec: Large language models with graph augmentation for recommendation," 2024.
- L. Wang, S. Zhang, Y. Wang, E.-P. Lim, and Y. Wang, "LLM4Vis: Explainable visualization recommendation using ChatGPT," in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: Industry Track*, M. Wang and I. Zitouni, Eds. Singapore: Association for Computational Linguistics, Dec. 2023, pp. 675–692. [Online]. Available: <https://aclanthology.org/2023.emnlp-industry.64>
- Z. Cui, J. Ma, C. Zhou, J. Zhou, and H. Yang, "M6-rec: Generative pretrained language models are open-ended recommender systems," 2022.

- P. Liu, L. Zhang, and J. A. Gulla, "Pre-train, prompt and recommendation: A comprehensive survey of language modelling paradigm adaptations in recommender systems," 2023.
- K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "Bleu: a method for automatic evaluation of machine translation," in *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ser. ACL '02. USA: Association for Computational Linguistics, 2002, p. 311–318. [Online]. Available: <https://doi.org/10.3115/1073083.1073135>
- C.-Y. Lin, "ROUGE: A package for automatic evaluation of summaries," in *Text Summarization Branches Out*. Barcelona, Spain: Association for Computational Linguistics, Jul. 2004, pp. 74–81. [Online]. Available: <https://aclanthology.org/W04-1013>
- C. Su and C. McMillan, "Distilled GPT for source code summarization," *CoRR*, vol. abs/2308.14731, 2023. [Online]. Available: <https://doi.org/10.48550/arXiv.2308.14731>
- W. Guo, J. Yang, K. Yang, X. Li, Z. Rao, Y. Xu, and D. Niu, "Instruction fusion: Advancing prompt evolution through hybridization," *CoRR*, vol. abs/2312.15692, 2023. [Online]. Available: <https://doi.org/10.48550/arXiv.2312.15692>
- O. Sener and S. Savarese, "Active learning for convolutional neural networks: A core-set approach," in *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*, 2018. [Online]. Available: <https://openreview.net/forum?id=H1aIuk-RW>
- H. Liu, C. Li, Y. Li, and Y. J. Lee, "Improved baselines with visual instruction tuning," 2023.
- S. Zhang, P. Sun, S. Chen, M. Xiao, W. Shao, W. Zhang, Y. Liu, K. Chen, and P. Luo, "Gpt4roi: Instruction tuning large language model on region-of-interest," 2023.
- OpenAI, "Gpt-4v(ision) system card," 2023. [Online]. Available: <https://api.semanticscholar.org/CorpusID:263218031>
- B. A. Plummer, L. Wang, C. M. Cervantes, J. C. Caicedo, J. Hockenmaier, and S. Lazebnik, "Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 2641–2649.
- L. Li, Z. Xie, M. Li, S. Chen, P. Wang, L. Chen, Y. Yang, B. Wang, and L. Kong, "Silkie: Preference distillation for large visual language models," *arXiv preprint arXiv:2312.10665*, 2023.
- H. Ha, P. Florence, and S. Song, "Scaling up and distilling down: Language-guided robot skill acquisition," in *Conference on Robot Learning*. PMLR, 2023, pp. 3766–3777.
- S. Wu, Z. Liu, Z. Zhang, Z. Chen, W. Deng, W. Zhang, J. Yang, Z. Yao, Y. Lyu, X. Xin, S. Gao, P. Ren, Z. Ren, and Z. Chen, "fuzi.mingcha," <https://github.com/irlab-sdu/fuzi.mingcha>, 2023.
- H. Xiong, S. Wang, Y. Zhu, Z. Zhao, Y. Liu, Q. Wang, and D. Shen, "Doctorglm: Fine-tuning your chinese doctor is not a herculean task," *arXiv preprint arXiv:2304.01097*, 2023.
- X. Zhang, C. Tian, X. Yang, L. Chen, Z. Li, and L. R. Petzold, "Alpacare: Instruction-tuned large language models for medical application," *arXiv preprint arXiv:2310.14558*, 2023.
- Y. Li, Z. Li, K. Zhang, R. Dan, S. Jiang, and Y. Zhang, "Chatdoctor: A medical chat model fine-tuned on a large language model meta-ai (llama) using medical domain knowledge," *Cureus*, vol. 15, no. 6, 2023.
- T. Han, L. C. Adams, J. Papaioannou, P. Grundmann, T. Oberhauser, A. Löser, D. Truhn, and K. K. Bressen, "Medalpaca - an open-source collection of medical conversational AI models and training data," *CoRR*, vol. abs/2304.08247, 2023. [Online]. Available: <https://doi.org/10.48550/arXiv.2304.08247>
- C. Wu, W. Lin, X. Zhang, Y. Zhang, Y. Wang, and W. Xie, "Pmc-llama: Towards building open-source language models for medicine," *arXiv preprint arXiv:2305.10415*, vol. 6, 2023.
- Z. Bao, W. Chen, S. Xiao, K. Ren, J. Wu, C. Zhong, J. Peng, X. Huang, and Z. Wei, "Disc-medllm: Bridging general large language models and real-world medical consultation," *CoRR*, vol. abs/2308.14346, 2023. [Online]. Available: <https://doi.org/10.48550/arXiv.2308.14346>
- Z. Gou, Z. Shao, Y. Gong, yelong shen, Y. Yang, M. Huang, N. Duan, and W. Chen, "ToRA: A tool-integrated reasoning agent for mathematical problem solving," in *The Twelfth International Conference on Learning Representations*, 2024. [Online]. Available: <https://openreview.net/forum?id=EpOTtjVoap>
- E. Perkowski, R. Pan, T. D. Nguyen, Y. Ting, S. Kruk, T. Zhang, C. O'Neill, M. Jablonska, Z. Sun, M. J. Smith, H. Liu, K. Schawinski, K. Iyer, I. Ciuca, and UniverseTBD, "Astrollama-chat: Scaling astrollama with conversational and diverse datasets," *CoRR*, vol. abs/2401.01916, 2024. [Online]. Available: <https://doi.org/10.48550/arXiv.2401.01916>
- J. Gao, R. Pi, J. Zhang, J. Ye, W. Zhong, Y. Wang, L. Hong, J. Han, H. Xu, Z. Li, and L. Kong, "G-llava: Solving geometric problem with multi-modal large language model," *CoRR*, vol. abs/2312.11370, 2023. [Online]. Available: <https://doi.org/10.48550/arXiv.2312.11370>
- H. Zhao, S. Liu, C. Ma, H. Xu, J. Fu, Z.-H. Deng, L. Kong, and Q. Liu, "GIMLET: A unified graph-text model for instruction-based molecule zero-shot learning," in *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. [Online]. Available: <https://openreview.net/forum?id=Tt6DrRCgJV>
- A. N. Rubungo, C. Arnold, B. P. Rand, and A. B. Dieng, "Llm-prop: Predicting physical and electronic properties of crystalline solids from their text descriptions," *CoRR*, vol. abs/2310.14029, 2023. [Online]. Available: <https://doi.org/10.48550/arXiv.2310.14029>
- H. Cao, Z. Liu, X. Lu, Y. Yao, and Y. Li, "Instructmol: Multi-modal integration for building a versatile and reliable molecular assistant in drug discovery," *CoRR*, vol. abs/2311.16208, 2023. [Online]. Available: <https://doi.org/10.48550/arXiv.2311.16208>
- H. Abdine, M. Chatzianastasis, C. Bouyioukos, and M. Vazirgiannis, "Prot2text: Multimodal protein's function generation with GNNs and transformers," in *Deep Generative Models for Health Workshop NeurIPS 2023*, 2023. [Online]. Available: <https://openreview.net/forum?id=EJ7YNgWYFj>
- Y. Luo, J. Zhang, S. Fan, K. Yang, Y. Wu, M. Qiao,

- and Z. Nie, "Biomedgpt: Open multimodal generative pre-trained transformer for biomedicine," *arXiv preprint arXiv:2308.09442*, 2023.
- B. Chen, X. Cheng, P. Li, Y. Geng, J. Gong, S. Li, Z. Bei, X. Tan, B. Wang, X. Zeng, C. Liu, A. Zeng, Y. Dong, J. Tang, and L. Song, "xtrimopglm: Unified 100b-scale pre-trained transformer for deciphering the language of protein," *CoRR*, vol. abs/2401.06199, 2024. [Online]. Available: <https://doi.org/10.48550/arXiv.2401.06199>
- C. Deng, T. Zhang, Z. He, Y. Xu, Q. Chen, Y. Shi, L. Fu, W. Zhang, X. Wang, C. Zhou, Z. Lin, and J. He, "K2: A foundation language model for geoscience knowledge understanding and utilization," 2023.
- Z. Bi, N. Zhang, Y. Xue, Y. Ou, D. Ji, G. Zheng, and H. Chen, "Oceangpt: A large language model for ocean science tasks," *CoRR*, vol. abs/2310.02031, 2023. [Online]. Available: <https://doi.org/10.48550/arXiv.2310.02031>
- Z. Zheng, J. Zhang, T. Vu, S. Diao, Y. H. W. Tim, and S. Yeung, "Marinegpt: Unlocking secrets of ocean to the public," *CoRR*, vol. abs/2310.13596, 2023. [Online]. Available: <https://doi.org/10.48550/arXiv.2310.13596>
- Z. Lin, C. Deng, L. Zhou, T. Zhang, Y. Xu, Y. Xu, Z. He, Y. Shi, B. Dai, Y. Song, B. Zeng, Q. Chen, T. Shi, T. Huang, Y. Xu, S. Wang, L. Fu, W. Zhang, J. He, C. Ma, Y. Zhu, X. Wang, and C. Zhou, "Geogalactica: A scientific large language model in geoscience," *CoRR*, vol. abs/2401.00434, 2024. [Online]. Available: <https://doi.org/10.48550/arXiv.2401.00434>
- D. Zhang, A. Petrova, D. Trautmann, and F. Schilder, "Unleashing the power of large language models for legal applications," in *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, 2023, pp. 5257–5258.
- Z. Sun, "A short survey of viewing large language models in legal aspect," *arXiv preprint arXiv:2303.09136*, 2023.
- J. Lai, W. Gan, J. Wu, Z. Qi, and P. S. Yu, "Large language models in law: A survey," *arXiv preprint arXiv:2312.03718*, 2023.
- S. Yue, W. Chen, S. Wang, B. Li, C. Shen, S. Liu, Y. Zhou, Y. Xiao, S. Yun, W. Lin *et al.*, "Disc-lawllm: Fine-tuning large language models for intelligent legal services," *arXiv preprint arXiv:2309.11325*, 2023.
- H. Zhong, C. Xiao, C. Tu, T. Zhang, Z. Liu, and M. Sun, "Jec-qa: a legal-domain question answering dataset," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 05, 2020, pp. 9701–9708.
- K. Singhal, T. Tu, J. Gottweis, R. Sayres, E. Wulczyn, L. Hou, K. Clark, S. Pfohl, H. Cole-Lewis, D. Neal, M. Schaeckermann, A. Wang, M. Amin, S. Lachgar, P. A. Mansfield, S. Prakash, B. Green, E. Dominowska, B. A. y Arcas, N. Tomasev, Y. Liu, R. Wong, C. Semturs, S. S. Mahdavi, J. K. Barral, D. R. Webster, G. S. Corrado, Y. Matias, S. Azizi, A. Karthikesalingam, and V. Natarajan, "Towards expert-level medical question answering with large language models," *CoRR*, vol. abs/2305.09617, 2023. [Online]. Available: <https://doi.org/10.48550/arXiv.2305.09617>
- W. Zhu, X. Wang, H. Zheng, M. Chen, and B. Tang, "Promptblue: A chinese prompt tuning benchmark for the medical domain," *arXiv preprint arXiv:2310.14151*, 2023.
- C. Wu, X. Zhang, Y. Zhang, Y. Wang, and W. Xie, "Pmc-llama: Further finetuning llama on medical papers," *CoRR*, vol. abs/2304.14454, 2023. [Online]. Available: <https://doi.org/10.48550/arXiv.2304.14454>
- Z. Bao, W. Chen, S. Xiao, K. Ren, J. Wu, C. Zhong, J. Peng, X. Huang, and Z. Wei, "Disc-medllm: Bridging general large language models and real-world medical consultation," *arXiv preprint arXiv:2308.14346*, 2023.
- S. Xue, F. Zhou, Y. Xu, H. Zhao, S. Xie, Q. Dai, C. Jiang, J. Zhang, J. Zhou, D. Xiu, and H. Mei, "Weaverbird: Empowering financial decision-making with large language model, knowledge base, and search engine," *CoRR*, vol. abs/2308.05361, 2023. [Online]. Available: <https://doi.org/10.48550/arXiv.2308.05361>
- S. Wu, O. Irsoy, S. Lu, V. Dabravolski, M. Dredze, S. Gehrmann, P. Kambadur, D. S. Rosenberg, and G. Mann, "Bloomberggpt: A large language model for finance," *CoRR*, vol. abs/2303.17564, 2023. [Online]. Available: <https://doi.org/10.48550/arXiv.2303.17564>
- D. Lu, H. Wu, J. Liang, Y. Xu, Q. He, Y. Geng, M. Han, Y. Xin, and Y. Xiao, "Bbt-fin: Comprehensive construction of chinese financial domain pre-trained language model, corpus and benchmark," *CoRR*, vol. abs/2302.09432, 2023. [Online]. Available: <https://doi.org/10.48550/arXiv.2302.09432>
- Y. Yang, Y. Tang, and K. Y. Tam, "Investlm: A large language model for investment using financial domain instruction tuning," *CoRR*, vol. abs/2309.13064, 2023. [Online]. Available: <https://doi.org/10.48550/arXiv.2309.13064>
- Q. Xie, W. Han, X. Zhang, Y. Lai, M. Peng, A. Lopez-Lira, and J. Huang, "PIXIU: A large language model, instruction data and evaluation benchmark for finance," *CoRR*, vol. abs/2306.05443, 2023. [Online]. Available: <https://doi.org/10.48550/arXiv.2306.05443>
- N. Wang, H. Yang, and C. D. Wang, "Fingpt: Instruction tuning benchmark for open-source large language models in financial datasets," *CoRR*, vol. abs/2310.04793, 2023. [Online]. Available: <https://doi.org/10.48550/arXiv.2310.04793>
- R. Taylor, M. Kardas, G. Cucurull, T. Scialom, A. Hartshorn, E. Saravia, A. Poulton, V. Kerkez, and R. Stojnic, "Galactica: A large language model for science," *CoRR*, vol. abs/2211.09085, 2022. [Online]. Available: <https://doi.org/10.48550/arXiv.2211.09085>
- J. Yin, S. Dash, F. Wang, and M. Shankar, "FORGE: pre-training open foundation models for science," in *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis, SC 2023, Denver, CO, USA, November 12-17, 2023*, D. Arnold, R. M. Badia, and K. M. Mohror, Eds. ACM, 2023, pp. 81:1–81:13. [Online]. Available: <https://doi.org/10.1145/3581784.3613215>
- Z. Azerbayev, H. Schoelkopf, K. Paster, M. D. Santos, S. McAleer, A. Q. Jiang, J. Deng, S. Biderman, and S. Welleck, "Llemma: An open language model for mathematics," *CoRR*, vol. abs/2310.10631, 2023. [Online]. Available: <https://doi.org/10.48550/arXiv.2310.10631>
- F. Yu, A. Gao, and B. Wang, "Outcome-supervised verifiers for planning in mathematical reasoning," *CoRR*, vol. abs/2311.09724, 2023. [Online]. Available: <https://doi.org/10.48550/arXiv.2311.09724>

- T. D. Nguyen, Y. Ting, I. Ciuca, C. O'Neill, Z. Sun, M. Jablonska, S. Kruk, E. Perkowski, J. W. Miller, J. Li, J. Peek, K. Iyer, T. Rózanski, P. Khetarpal, S. Zaman, D. Brodrick, S. J. R. Méndez, T. Bui, A. Goodman, A. Accomazzi, J. P. Naiman, J. Cranney, K. Schawinski, and UniverseTBD, "Astrollama: Towards specialized foundation models in astronomy," *CoRR*, vol. abs/2309.06126, 2023. [Online]. Available: <https://doi.org/10.48550/arXiv.2309.06126>
- J. Roberts, T. Lüddecke, S. Das, K. Han, and S. Albanie, "Gpt4geo: How a language model sees the world's geography," 2023.
- Z. Lin, C. Deng, L. Zhou, T. Zhang, Y. Xu, Y. Xu, Z. He, Y. Shi, B. Dai, Y. Song, B. Zeng, Q. Chen, T. Shi, T. Huang, Y. Xu, S. Wang, L. Fu, W. Zhang, J. He, C. Ma, Y. Zhu, X. Wang, and C. Zhou, "Geogalactica: A scientific large language model in geoscience," 2023.
- C. Wang, D. Engler, X. Li, J. Hou, D. J. Wald, K. Jaiswal, and S. Xu, "Near-real-time earthquake-induced fatality estimation using crowdsourced data and large-language models," 2023.
- L. Chen, S. Li, J. Yan, H. Wang, K. Gunaratna, V. Yadav, Z. Tang, V. Srinivasan, T. Zhou, H. Huang, and H. Jin, "Alpagasus: Training a better alpaca with fewer data," 2023.
- Y. Cao, Y. Kang, and L. Sun, "Instruction mining: High-quality instruction data selection for large language models," 2023.
- M. Li, Y. Zhang, Z. Li, J. Chen, L. Chen, N. Cheng, J. Wang, T. Zhou, and J. Xiao, "From quantity to quality: Boosting llm performance with self-guided data selection for instruction tuning," *ArXiv*, vol. abs/2308.12032, 2023. [Online]. Available: <https://api.semanticscholar.org/CorpusID:261076515>
- Q. Du, C. Zong, and J. Zhang, "Mods: Model-oriented data selection for instruction tuning," 2023.
- Y. Li, B. Hui, X. Xia, J. Yang, M. Yang, L. Zhang, S. Si, J. Liu, T. Liu, F. Huang, and Y. Li, "One shot learning as instruction data prospector for large language models," 2023.
- E. Frantar, S. P. Singh, and D. Alistarh, "Optimal brain compression: A framework for accurate post-training quantization and pruning," 2023.
- T. Dettmers, M. Lewis, Y. Belkada, and L. Zettlemoyer, "GPT3.int8(): 8-bit matrix multiplication for transformers at scale," in *Advances in Neural Information Processing Systems*, A. H. Oh, A. Agarwal, D. Belgrave, and K. Cho, Eds., 2022. [Online]. Available: <https://openreview.net/forum?id=dXiGWqBoxaD>
- Y. J. Kim, R. Henry, R. Fahim, and H. H. Awadalla, "Finequant: Unlocking efficiency with fine-grained weight-only quantization for llms," 2023.
- C. Tao, L. Hou, W. Zhang, L. Shang, X. Jiang, Q. Liu, P. Luo, and N. Wong, "Compression of generative pre-trained language models via quantization," in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, S. Muresan, P. Nakov, and A. Villavicencio, Eds. Dublin, Ireland: Association for Computational Linguistics, May 2022, pp. 4821–4836. [Online]. Available: <https://aclanthology.org/2022.acl-long.331>
- Z. Yao, R. Yazdani Aminabadi, M. Zhang, X. Wu, C. Li, and Y. He, "Zeroquant: Efficient and affordable post-training quantization for large-scale transformers," *Advances in Neural Information Processing Systems*, vol. 35, pp. 27 168–27 183, 2022.
- G. Xiao, J. Lin, M. Seznec, H. Wu, J. Demouth, and S. Han, "Smoothquant: Accurate and efficient post-training quantization for large language models," 2023.
- X. Ma, G. Fang, and X. Wang, "Llm-pruner: On the structural pruning of large language models," 2023.
- M. Zhang, H. Chen, C. Shen, Z. Yang, L. Ou, X. Yu, and B. Zhuang, "Loraprune: Pruning meets low-rank parameter-efficient fine-tuning," 2023.
- E. Frantar and D. Alistarh, "Sparsegpt: Massive language models can be accurately pruned in one-shot," 2023.
- M. Xu, Y. L. Xu, and D. P. Mandic, "Tensorgpt: Efficient compression of the embedding layer in llms based on the tensor-train decomposition," 2023.
- Y. Li, Y. Yu, Q. Zhang, C. Liang, P. He, W. Chen, and T. Zhao, "Lospase: Structured compression of large language models based on low-rank and sparse approximation," 2023.
- Z. Hu, L. Wang, Y. Lan, W. Xu, E.-P. Lim, L. Bing, X. Xu, S. Poria, and R. K.-W. Lee, "Llm-adapters: An adapter family for parameter-efficient fine-tuning of large language models," 2023.
- H. Liu, D. Tam, M. Mohammed, J. Mohta, T. Huang, M. Bansal, and C. Raffel, "Few-shot parameter-efficient fine-tuning is better and cheaper than in-context learning," in *Advances in Neural Information Processing Systems*, A. H. Oh, A. Agarwal, D. Belgrave, and K. Cho, Eds., 2022. [Online]. Available: <https://openreview.net/forum?id=rBCvMG-JsPd>
- Y. Wang, S. Agarwal, S. Mukherjee, X. Liu, J. Gao, A. H. Awadallah, and J. Gao, "AdaMix: Mixture-of-adaptations for parameter-efficient model tuning," in *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, Y. Goldberg, Z. Kozareva, and Y. Zhang, Eds. Abu Dhabi, United Arab Emirates: Association for Computational Linguistics, Dec. 2022, pp. 5744–5760. [Online]. Available: <https://aclanthology.org/2022.emnlp-main.388>
- E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, "Lora: Low-rank adaptation of large language models," 2021.
- X. L. Li and P. Liang, "Prefix-tuning: Optimizing continuous prompts for generation," in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, C. Zong, F. Xia, W. Li, and R. Navigli, Eds. Online: Association for Computational Linguistics, Aug. 2021, pp. 4582–4597. [Online]. Available: <https://aclanthology.org/2021.acl-long.353>
- X. Liu, K. Ji, Y. Fu, W. Tam, Z. Du, Z. Yang, and J. Tang, "P-tuning: Prompt tuning can be comparable to fine-tuning across scales and tasks," in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, S. Muresan, P. Nakov, and A. Villavicencio, Eds. Dublin, Ireland: Association for Computational Linguistics, May 2022, pp. 61–68. [Online].

- Available: <https://aclanthology.org/2022.acl-short.8>
- T. Dettmers, A. Pagnoni, A. Holtzman, and L. Zettlemoyer, "Qlora: Efficient finetuning of quantized llms," 2023.
- J. Kim, J. H. Lee, S. Kim, J. Park, K. M. Yoo, S. J. Kwon, and D. Lee, "Memory-efficient fine-tuning of compressed large language models via sub-4-bit integer quantization," 2023.
- S. Malladi, T. Gao, E. Nichani, A. Damian, J. D. Lee, D. Chen, and S. Arora, "Fine-tuning language models with just forward passes," 2024.
- Z. Wan, X. Wang, C. Liu, S. Alam, Y. Zheng, J. Liu, Z. Qu, S. Yan, Y. Zhu, Q. Zhang, M. Chowdhury, and M. Zhang, "Efficient large language models: A survey," 2024.
- Y.-S. Lee, M. Sultan, Y. El-Kurdi, T. Naseem, A. Munawar, R. Florian, S. Roukos, and R. Astudillo, "Ensemble-instruct: Instruction tuning data generation with a heterogeneous mixture of LMs," in *Findings of the Association for Computational Linguistics: EMNLP 2023*, H. Bouamor, J. Pino, and K. Bali, Eds. Singapore: Association for Computational Linguistics, Dec. 2023, pp. 12 561–12 571. [Online]. Available: <https://aclanthology.org/2023.findings-emnlp.836>
- W. Chen, Y. Zhou, N. Du, Y. Huang, J. Laudon, Z. Chen, and C. Cui, "Lifelong language pretraining with distribution-specialized experts," in *International Conference on Machine Learning*. PMLR, 2023, pp. 5383–5395.
- S. Kotha, J. M. Springer, and A. Raghunathan, "Understanding catastrophic forgetting in language models via implicit inference," *arXiv preprint arXiv:2309.10105*, 2023.
- B. Koloski, B. Škrlić, M. Robnik-Šikonja, and S. Pollak, "Measuring catastrophic forgetting in cross-lingual transfer paradigms: Exploring tuning strategies," *arXiv preprint arXiv:2309.06089*, 2023.
- T. Wu, L. Luo, Y.-F. Li, S. Pan, T.-T. Vu, and G. Haffari, "Continual learning for large language models: A survey," *arXiv preprint arXiv:2402.01364*, 2024.
- Y. Luo, Z. Zhang, F. Meng, Y. Li, J. Zhou, and Y. Zhang, "An empirical study of catastrophic forgetting in large language models during continual fine-tuning," *arXiv preprint arXiv:2308.08747*, 2023.
- J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska *et al.*, "Overcoming catastrophic forgetting in neural networks," *Proceedings of the national academy of sciences*, vol. 114, no. 13, pp. 3521–3526, 2017.
- M. Rostami, S. Kolouri, and P. K. Pilly, "Complementary learning for overcoming catastrophic forgetting using experience replay," *arXiv preprint arXiv:1903.04566*, 2019.
- D. Rolnick, A. Ahuja, J. Schwarz, T. Lillicrap, and G. Wayne, "Experience replay for continual learning," *Advances in Neural Information Processing Systems*, vol. 32, 2019.
- S.-W. Lee, J.-H. Kim, J. Jun, J.-W. Ha, and B.-T. Zhang, "Overcoming catastrophic forgetting by incremental moment matching," *Advances in neural information processing systems*, vol. 30, 2017.
- A. Mallya, D. Davis, and S. Lazebnik, "Piggyback: Adapting a single network to multiple tasks by learning to mask weights," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 67–82.
- Z. Wang, Z. Zhang, C.-Y. Lee, H. Zhang, R. Sun, X. Ren, G. Su, V. Perot, J. Dy, and T. Pfister, "Learning to prompt for continual learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 139–149.
- Z. Hu, Y. Li, J. Lyu, D. Gao, and N. Vasconcelos, "Dense network expansion for class incremental learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 11 858–11 867.
- X. Li, L. Lin, S. Wang, and C. Qian, "Unlock the power: Competitive distillation for multi-modal large language models," *arXiv preprint arXiv:2311.08213*, 2023.
- M. Zeng, W. Xue, Q. Liu, and Y. Guo, "Continual learning with dirichlet generative-based rehearsal," *arXiv preprint arXiv:2309.06917*, 2023.
- Z. Zhang, M. Fang, L. Chen, and M.-R. Namazi-Rad, "Citb: A benchmark for continual instruction tuning," *arXiv preprint arXiv:2310.14510*, 2023.
- C. Burns, P. Izmailov, J. H. Kirchner, B. Baker, L. Gao, L. Aschenbrenner, Y. Chen, A. Ecoffet, M. Joglekar, J. Leike, I. Sutskever, and J. Wu, "Weak-to-strong generalization: Eliciting strong capabilities with weak supervision," *CoRR*, vol. abs/2312.09390, 2023. [Online]. Available: <https://doi.org/10.48550/arXiv.2312.09390>
- M. Li, Y. Zhang, S. He, Z. Li, H. Zhao, J. Wang, N. Cheng, and T. Zhou, "Superfiltering: Weak-to-strong data filtering for fast instruction-tuning," *CoRR*, vol. abs/2402.00530, 2024. [Online]. Available: <https://doi.org/10.48550/arXiv.2402.00530>
- J. Ji, B. Chen, H. Lou, D. Hong, B. Zhang, X. Pan, J. Dai, and Y. Yang, "Aligner: Achieving efficient alignment through weak-to-strong correction," *CoRR*, vol. abs/2402.02416, 2024. [Online]. Available: <https://doi.org/10.48550/arXiv.2402.02416>