A Survey on Symbolic Knowledge Distillation of Large Language Models

Kamal Acharya, *Graduate Student Member, IEEE*, Alvaro Velasquez, *Member, IEEE* and Houbing Herbert Song, *Fellow, IEEE*

***Abstract*—This survey paper delves into the emerging and critical area of symbolic knowledge distillation in Large Language Models (LLMs). As LLMs like Generative Pre-trained Transformer-3 (GPT-3) and Bidirectional Encoder Representations from Transformers (BERT) continue to expand in scale and complexity, the challenge of effectively harnessing their extensive knowledge becomes paramount. This survey concentrates on the process of distilling the intricate, often implicit knowledge contained within these models into a more symbolic, explicit form. This transformation is crucial for enhancing the interpretability, efficiency, and applicability of LLMs. We categorize the existing research based on methodologies and applications, focusing on how symbolic knowledge distillation can be used to improve the transparency and functionality of smaller, more efficient Artificial Intelligence (AI) models. The survey discusses the core challenges, including maintaining the depth of knowledge in a comprehensible format, and explores the various approaches and techniques that have been developed in this field. We identify gaps in current research and potential opportunities for future advancements. This survey aims to provide a comprehensive overview of symbolic knowledge distillation in LLMs, spotlighting its significance in the progression towards more accessible and efficient AI systems.**

***Impact Statement*—There is burgeoning interest in the potential of symbolic knowledge to enhance the interpretability, efficiency, and application scope of LLMs, transforming them into more robust, understandable, and versatile tools. Despite the recognition of its importance, there remains a notable dearth of comprehensive research that thoroughly examines and evaluates the process and implications of this integration. Existing literature predominantly focuses on either the advancements in LLMs or content of the knowledge in the LLMs , with less emphasis on the symbolic knowledge distillation of LLMs. This survey aims to fill this critical gap by offering an extensive review of the current state of symbolic knowledge disitllation in LLMs by highlighting the methodologies, challenges, and advancements in this field.**

***Index Terms*—Large Language Models, Symbolic Knowledge, Symbolic Knowledge Distillation**

I. Introduction

LARGE Language Models (LLMs) are a prominent topic in Artificial Intelligence(AI), with significant breakthroughs occurring frequently. Trained on extensive data sets including websites, research papers, and books, LLMs encapsulate knowledge within their numerous parameters. They can serve as knowledge bases[1], from which information can be extracted and formatted for various purposes, such as fine-tuning other models for specific tasks[2], validating actions[3], or generating larger and more accurate datasets[4]. However, the knowledge embedded in LLMs is not immediately accessible and requires careful extraction and efficient utilization to yield effective results.

The knowledge within LLMs, stored in the weights of their parameters, can be converted into a more interpretable symbolic form through the process of symbolic knowledge distillation. The core challenge here lies in translating the implicit, distributed knowledge encoded in the neural networks of LLMs into explicit, symbolic representations. This transformation is essential for several reasons: to improve the transparency and interpretability of the models, to facilitate knowledge transfer to smaller, more efficient models, and to enable more robust and explainable AI systems. By converting the knowledge into symbolic form, it becomes possible to understand the reasoning behind the model's decisions. This is crucial for applications where understanding the 'why' behind predictions or recommendations is as important as the outcomes themselves. The process is fraught with complexities, including preserving the nuance and depth of the learned knowledge while making it comprehensible and utilizable in a symbolic format.

In this paper, we introduce a detailed framework dedicated to symbolic knowledge distillation of LLMs, initiating our discussion with a historical overview of symbolic knowledge distillation and its evolutionary path to its current state. Following this, we delve into an analysis of various traditional knowledge distillation methods and their comparison with symbolic knowledge distillation approaches. We further explore LLM architectures, including their training and fine-tuning mechanisms. We classify symbolic knowledge distillation techniques into three distinct categories: Direct, Multilevel, and Distillation via Reinforcement Learning. Additionally, we have compiled research papers focused on symbolic knowledge, as well as those specifically addressing symbolic knowledge distillation of LLMs. Our survey provides a thorough examination of the latest developments in symbolic knowledge distillation of LLMs, highlighting the methodologies, challenges, and progress in the field, thereby offering valuable insights for the research community interested in further exploration of this domain.

The rapid expansion of LLMs has led to the production

Manuscript received January 6, 2024. This work was supported in part by the U.S. National Science Foundation under Grant No. 2309760 and Grant No. 2317117.

K. Acharya and H. Song are with the Security and Optimization for Networked Globe Laboratory (SONG Lab), Department of Information Systems, University of Maryland, Baltimore County, Baltimore, MD 21250 USA (e-mail: kamala2@umbc.edu; h.song@ieee.org).

A. Velasquez is with the Department of Computer Science, University of Colorado, Boulder, CO 80309 USA (e-mail: alvaro.velasquez@colorado.edu).

of numerous survey papers. All the previous survey papers on LLMs cover different aspects except for the symbolic knowledge. Further exploring we find that no survey paper has been published related to the symbolic knowledge distillation. The focus areas of existing survey papers on LLMs include:

- Comprehensive overviews of LLMs[5], [6], [7]
- Evaluation of LLMs[8]
- Code generation[9]
- LLMs in education[10]
- LLM as Knowledge Base[11], [12]
- Reasoning Knowledge in LLMs[13]
- Explainability in LLMs[14]
- Aligning LLMs with human[15]
- Instruction tuning for LLM[16]
- Model Compression in LLM[17]
- Trustworthiness evaluation of LLM[18]
- LLM for software engineering[19]
- Hallucination in LLM[20]
- Multimodal LLM[21]
- LLMs for Robotics[22]
- LLMs for Information Retrieval[23]

Our work stands in contrast to existing approaches in several key aspects. While traditional methods primarily focus on either the performance enhancement of smaller models or the interpretability aspect of knowledge distillation, our framework synergizes these objectives.

The remainder of this paper is structured as follows: Section II reviews the milestones in knowledge distillation and LLM, establishing the context and background for our work. Section III details the preliminaries about symbolic knowledge distillation and LLM, followed by Section IV, which presents a thorough process of symbolic knowledge distillation in LLM. Section V discusses the related research work that has been carried out. In Section VI, we discuss opportunities that have emerged from Symbolic Knowledge Distillation. Section VII is devoted to the challenges of implementing proposed Symbolic knowledge distillation applications. We identify the obstacles and challenges that may arise. Section VIII highlights the Lesson Learned and Key Takeaways and finally, in Section IX, we offer concluding remarks on our survey paper.

II. Milestones in Knowledge Distillation and Large Language Models

Over the last seven decades, language technology has advanced significantly. The Turing Test[24], conducted in 1950, was one of the earliest milestones in this field, which laid the foundation for the concept that machines can perform at the level of humans and demonstrate the intelligence. In the same year Shannon used concept of entropy and provided the way of prediction of the next letter when the preceding text is known[25]. In 1964, ELIZA[26] was introduced as a Natural Language Processing (NLP) computer program which was designed to mimic the conversational style of a psychotherapist. SHRDLU[27], introduced in 1968, was an early example of an interactive natural language understanding system which can understand and respond to natural language commands related to a simplified world of objects. Following year was the dominance of the Statistical Language Model(SLM). Notable works that lead the way were "Introduction of Stochastic Approach for Parsing"[28] in 1986 and "Statistical Approach to machine translation"[29] in 1990. Due to the problem like Brittleness Across Domains, False Independence Assumption and Shannon-Style Experiments, there was downfall of the SLMs[30].

With the introduction of Long Short-Term Memory(LSTM)[31] in 1997, we entered into the era of Neural Language Model(NLM). These models helped in language processing by capturing the long term dependencies and successfully handling the vanishing gradients. In 2001, the first neural language model was introduced which can be trained using Stochastic Gradient Descent(SGD) algorithm and proved to be computationally efficient and scalable to larger dataset.[32]. Neural Networks not only increased in scope and functionality but also in terms of the size[33]. The concept of model compression[34] was introduced in 2006. Model compression and acceleration techniques was divided into four different approaches[35]: parameter pruning and sharing[36][37][38][39][40], low-rank factorization[41][42], transferred/compact convolutional layers[43] and knowledge distillation[44].

In 2011, IBM Watson made significant strides in language processing by winning a Jeopardy game against human competitors[45]. Two years later, in 2013, the Word2Vec algorithm[46] was introduced, which enabled computers to understand the context of a word and its relationship with other words using dense vector representation where similar words are located close to each other. In 2014, seq2seq[47] was introduced which used encoder to represent variable length input sequence into fixed length vector and decoder to generate output sequence. In the same year, Global Vectors for Word Representation(GloVe)[48] was introduced, which used co-occurance matrix to capture relationship between the words in corpus and was successful in capturing the local and global context informaiton. Knowledge distillation is a model compression technique introduced in 2015 that transfers knowledge from a high-capacity teacher model to a more compact student model. Later in that year FitNets[49] was introduced that add an additional term along with the knowledge distillation loss. In 2016, study[50] instead of utilizing representations from a specific point in the network, employed attention maps as hints, comparing the mean squared error (MSE) between the attention maps of the student and teacher models. In same year, SQuAD (Standford Question Answering Dataset)[51] was introduced, which facilitated the development of question-answering systems by being benchmark dataset for evaluating machine reading comprehension.

In 2017, the Transformer[52] model was introduced, which enabled the development of advanced language models that can learn relationships between words in a sentence more efficiently by using the concept of self-attention. In the following year, 2017 [53] employed a similar approach. However, instead of utilizing representations or attention maps, they provided hints by using Gram matrices. In 2018, a supplementary module called the paraphraser[54] is incorporated into the model.

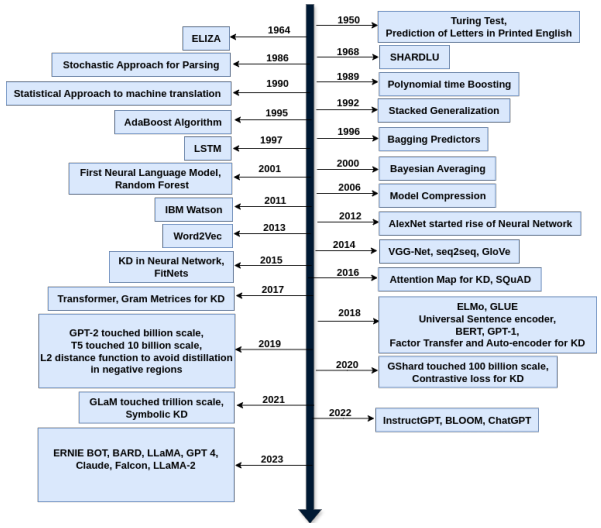


Fig. 1. Milestones in history of LLM and Knowledge Distillation

In same year, ELMo (Embedding from Language Model)[55], context dependent representation of word was introduced which uses different embeddings for same word in different context. Universal Sentence Encoder[56] was also introduced in same year, which further enhanced language processing by introducing embeddings for sentence representations and can handle multiple languages.

General Language Understanding Evaluation(GLUE)[57], a benchmark to evaluate the performance of NLP models on a range of language understanding tasks, became a standard evaluation framework for comparing different language models. Bidirectional Encoder Representations from Transformers(BERT)[58] and Generative Pre-Training-1(GPT-1)[59] were introduced in the same year, 2018 which begin the era of Pre-trained Language Model(PLM). In 2019, GPT-2[60] became the first language model to touch a billion scale of parameters. Later that year, T5[61] became the first language model to touch the 10 billion parameter scale. According to [62] published in 2019, the current approach of extracting hints may not be optimal due to the loss of information caused by the ReLU transformation. To address this, they introduced a modified activation function called marginReLU. In [63] published in 2020, the student model learns from the intermediate representations of the teacher model by employing a contrastive loss over these representations. As like the way human way of learning, knowledge distillation was applied in the model; self-learning[64], mutual learning[65], teacher student learning[44], teacher assistant[66] and continual learning[67]. Moreover, the application of knowledge distillation extends beyond transferring knowledge between models. It can also be utilized in various other tasks, including adversarial attacks [68], data augmentation [69][70], data privacy and security [71], as well as dataset distillation [72][73]. Between 2010 and 2020, the domain of transfer learning experienced significant expansion, with numerous transfer learning models achieving state-of-the-art results across various disciplines[74].

Google Shard (GShard)[75], introduced in 2020, became the first language model to touch the 100 billion parameter scale. And in 2021, the Generalist Language Model (GLaM)[76] became the first language model to touch the trillion parameter scale. Concept of symbolic knowledge distillation[2] was introduced in the same year which is a technique for training smaller models using larger models as teachers and involves distilling knowledge symbolically. Since then symbolic knowledge distillation has been used in various areas such as reference free sentence summarization[3], comparative knowledge acquisition[77]. The scaling laws for neural language models[78], reveal that model performance improves predictably with increases in model size, dataset size, and computational resources, following a power-law relationship. This means that larger models are significantly more efficient in learning from data. In 2022 and 2023, this trend persisted, with various industry leaders introducing new large-scale language models that leveraged these principles to achieve enhanced performance, demonstrating the continued advancement and efficacy of scaling up model size and computational power in the development of language models. Major technology companies are investing heavily in developing their own LLMs because they recognize the immense potential of these systems to revolutionize various industries, such as healthcare, finance, and customer service. Also, LLMs can help these companies maintain their position as leaders in the field of AI and keep up with competitors. Given the swift advancements in this field, there is a pressing need to steer AI towards paths that prioritize safety and responsibility[1].

The study[79] concludes that for compute-optimal training, both the model size and the number of training tokens should be scaled equally; specifically, each doubling of the model size should be accompanied by a doubling of the number of training tokens. Conversely, study[80] suggest that the supply of high-quality language data will likely be depleted by 2026. In contrast, low-quality language data and image data are projected to be exhausted between 2030 and 2050 for low-quality language data, and between 2030 and 2060 for image data. The current trajectory of rapidly increasing the parameters of LLMs, which depend on vast datasets, may decelerate unless there are significant improvements in data efficiency or new data sources are discovered. These findings have influenced the development of next-generation LLMs towards models capable of generating their own training data for self-improvement. Furthermore, LLMs will need to incorporate self-fact-checking capabilities. These scenarios underscore the importance of symbolic knowledge distillation and suggest a potential shift of LLMs towards this approach.

It has been utilized for labeling[81][82], where the teacher model generates outputs based on the provided input, and for expansion[83][84], where the teacher model produces samples akin to given demonstrations through in-context learning. For data generation[85] which involves synthesizing data according to specific meta-information, such as a topic or entity, feedback[86] which involves providing guidance on the student's outputs, encompassing preferences, corrections, and expansions of challenging samples. Finally, for self-checking[87]

[1]https://www.whitehouse.gov/briefing-room/presidential-actions/2023/10/30/executive-order-on-the-safe-secure-and-trustworthy-development-and-use-of-artificial-intelligence/(last accessed on: [28/02/2024])

which entails the student model generating outputs, which are subsequently filtered for high quality or self-evaluated by the student model.

III. Background and Preliminaries

For understanding the process of symbolic knowledge distillation of LLMs, we need to dive deeper into the two different technical theory of knowledge distillation followed by LLMs. Following sub-section will focus on that part.

A. Knowledge Distillation

Knowledge distillation is a technique used to transfer knowledge from a larger, more complex model (teacher) to a smaller, simpler model (student) with the goal of retaining much of the teacher model's performance[117]. This process is crucial in scenarios where computational resources are limited or where deployment requires lightweight models. There are various types of traditional knowledge distillation techniques: response-based, feature-based and relation-based and one modern symbolic knowledge distillation, each with its unique approach and area of application:

1) Response-based Knowledge Distillation: Response-based knowledge distillation involves transferring knowledge from the teacher model's final output layer to the student model, aiming to mimic the teacher's final predictions. This approach is straightforward and has proven effective across various tasks, employing a loss function based on the divergence between the teacher's and student's logits. It's widely applied in model compression and has been adapted for different types of model predictions, including object detection and human pose estimation, where the teacher's output may include additional information like bounding box offsets[118] or heatmaps for landmarks[119]. A key application of response-based knowledge distillation is in image classification[44], where "soft targets" – the probabilities assigned to each class by the teacher model – play a crucial role. These probabilities are adjusted using a temperature factor to control the softness of the targets, allowing the transfer of knowledge from the teacher to the student. The distillation process typically employs the Kullback-Leibler divergence loss to optimize the similarity between the teacher's and student's probability distributions.

This method is praised for its simplicity and effectiveness, particularly in leveraging knowledge for training. However, its reliance on the final layer's output means it may not fully utilize intermediate-level supervision from the teacher, an aspect crucial for representation learning in deep neural networks.

2) Feature-based Knowledge Distillation: Feature-based knowledge distillation taps into the strength of deep neural networks to learn hierarchical feature representations, a process central to representation learning[120]. Unlike response-based knowledge distillation, which focuses on the outputs of the last layer, feature-based distillation utilizes the outputs from intermediate layers, or feature maps, to guide the student model. This approach is particularly beneficial for training models that are both narrower and deeper, as it provides a richer set of training signals.

The concept was first introduced with Fitnets[49], aiming to improve student model training by matching feature activations between the teacher and student directly. Following this, several methodologies have been developed to facilitate this matching process, either directly or indirectly[121]. Notable contributions include the derivation of "attention maps" to express the use of neuron selectivity transfer[122], matching probability distributions in feature space[123], and introducing "factors" for more interpretable intermediate representations[54]. Techniques like route constrained hint learning[124] and the use of activation boundaries[125] have been proposed to minimize the performance gap between teacher and student models, alongside innovative strategies like cross-layer knowledge distillation[121] which adaptively matches teacher and student layers.

Despite the effectiveness of feature-based knowledge transfer in enriching the student model's learning, challenges remain in selecting appropriate layers for hints and guidance due to the size discrepancies between teacher and student models. This necessitates further exploration into how best to match the feature representations between teacher and student models effectively.

3) Relation-based Knowledge Distillation: Relation-based knowledge distillation goes beyond the scope of response-based and feature-based methods by examining the relationships between different layers or data samples within the teacher model. This approach delves into the dynamics between feature maps, layers, and even the relationships between different teachers or data samples, offering a more nuanced form of knowledge transfer.

Flow of solution process (FSP)[53] utilizes the Gram matrix between two layers to encapsulate the relationships between pairs of feature maps through inner product calculations. Knowledge distillation via singular value decomposition[126] distill essential information from these relationships. [127] explored multi-teacher scenarios by constructing graphs based on logits and features from each teacher, modeling their importance and relationships. [128] proposed a multi-head graph-based distillation technique that leverages intra-data relations between feature maps through a multi-head attention network. [129] focused on pairwise hint information, allowing the student model to mimic mutual information flows from pairs of hint layers in the teacher model.

The distillation loss in relation-based knowledge distillation is formulated based on the similarity and correlation functions between the feature representations of teacher and student models, aiming to capture and transfer the intricate relationships present in the teacher's architecture. Relation-based knowledge can also encompass structured knowledge of data, privileged information about input features, and various other categories, each represented by different loss functions like Earth Mover distance, Huber loss, Angle-wise loss, and Frobenius norm. While recent advancements have introduced several types of relation-based knowledge, the challenge remains in effectively modeling the relational information from feature maps or data samples for knowledge transfer. This area

TABLE I
TECHNICAL COMPANIES WITH THEIR LLM

Companies	LLM	Year	Parameters(in billions)	Corpus Size
Google	T5[61]	2019	11	1 trillion tokens
	GShard[75]	2020	600	1 trillion tokens
	mT5[88]	2021	13	1 trillion tokens
	GLaM[76]	2021	1200	1.6 trillion tokens
	FLAN[89]	2021	137	Not Available
	LaMDA[90]	2022	137	1.56T words, 168 billion tokens
	Minerva[91]	2022	540	38.5B tokens
	UL2 [92]	2022	20	1 trillion tokens
	PaLM[93]	2022	540	768 billion tokens
	FLAN-T5[94]	2022	11	Not Available
	FLAN-PaLM[94]	2022	540	Not Available
	Gemini(https://gemini.google.com/app)	2024	Not Available	Not Available
OpenAI	GPT-2[95]	2019	1.5	40GB (∼10 billion tokens)
	GPT-3[96]	2020	175	499 billion tokens
	Codex[97]	2021	12	100 billion tokens
	WebGPT[98]	2021	175	Not Available
	InstructGPT[99]	2022	175	Not Available
	ChatGPT(https://openai.com/blog/chatgpt)	2022	Not Available	Not Available
	GPT-4[100]	2023	Not Available	Not Available
EleutherAI	GPT-J[101]	2021	6	825 GiB
	GPT-Neo[102]	2021	2.7	825 GiB
	GPT-NeoX[103]	2022	20	825 GiB
DeepMind	Gopher[104]	2021	280	300 billion tokens
	AlphaCode[105]	2022	41	967 billion tokens
	Chinchilla[79]	2022	70	1.4 trillion tokens
	Sparrow[106]	2022	70	Not Available
Meta	Galactica[107]	2022	120	106 billion tokens
	OPT[108]	2022	175	180 billion tokens
	OPT-IML[109]	2022	175	Not Available
	LLaMA[110]	2023	65	1.4 trillion
Hugging Face	T0[111]	2021	11	Not Available
	BLOOM[112]	2022	175	350 billion tokens (1.6TB)
	mT0[113]	2022	13	Not Available
Baidu	Ernie 2.0 Large[114]	2019	1.5	Not Available
	Ernie 3.0[115]	2021	10	375 billion tokens
	Ernie 3.0 Titan[116]	2021	260	300 billion tokens
	Ernie Bot (https://yiyan.baidu.com/)	2023	Not Available	Not Available

continues to be ripe for further research and exploration to enhance the efficacy of knowledge distillation techniques.

4) Symbolic Knowledge Distillation: Contrary to the methods discussed earlier, symbolic knowledge distillation is centered on the distillation and transmission of knowledge in a symbolic format, including rules, logic, or symbolic representations. This method integrates structured knowledge bases and rules with machine learning models to boost their performance and clarity. It encodes intricate, structured information in a manner that allows for manipulation in reasoning, inference, and decision-making processes. The importance of this approach lies in its alignment with human methods of interpreting and reasoning with knowledge, thus providing enhanced transparency and interpretability.

Symbolic knowledge distillation represents a technique within machine learning where knowledge is extracted from a complex, typically less transparent model (like a deep neural network) and converted into a symbolic, more understandable format. This methodology merges the principles of conventional knowledge distillation with those of symbolic AI, aiming to improve the interpretability, transparency, and possibly the efficiency of machine learning models. It serves as a bridge between the often "black box" nature of deep learning models and the necessity for models that can be comprehended and trusted by humans. Such a requirement is especially critical in sectors demanding high levels of responsibility and explainability, including healthcare, finance, and autonomous driving. Although the specific mathematical model employed may vary based on the approach and the symbolic representation chosen, the overall process typically includes several defined steps.

Training the Teacher Model: A complex model (teacher) is trained on a dataset to achieve high performance. This model can be a deep neural network, and its architecture and training process depend on the specific task (e.g., image recognition, NLP).

Extracting Knowledge: The subsequent phase involves deriving insights from the teacher model, achievable through multiple approaches, including: examining the neuron activation patterns within the network; employing methods like Layer-wise Relevance Propagation (LRP)[130] or SHapley Additive exPlanations(SHAP)[131] to assess the significance of various inputs in the network's decision-making process; and identifying rules or patterns based on the decision boundaries established by the network.

Symbolic Representation: The gathered knowledge is subsequently converted into a symbolic representation. This process includes: developing decision trees or compiling sets of logical rules that mimic the neural network's behavior, and utilizing graphical models or alternative structured forms to

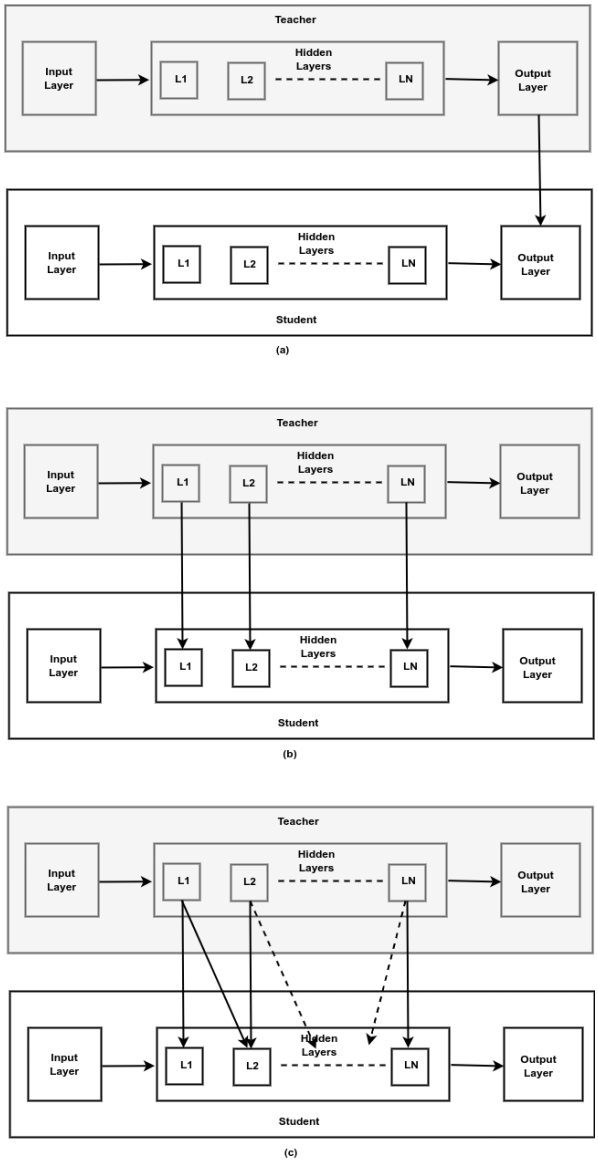


Fig. 2. Types of Traditional Knowledge Distillation (a) Response-based, (b) Feature-based and (c) Relation-based

encapsulate the relationships and dependencies deciphered by the network.

Training the Student Model: Following the translation of extracted knowledge into a symbolic form, a simpler and more interpretable 'student' model is trained to mimic this symbolic representation. The training process involves two key strategies. The symbolic representation may be used directly as a comprehensive set of rules for decision-making, allowing the student model to replicate decision processes based on predefined logical rules or the student model is trained to approximate the symbolic representation itself. This approach often incorporates conventional supervised learning techniques, with the significant distinction that the symbolic knowledge extracted from the teacher model acts as a guide or target.

Evaluation and Refinement: Once the student model has been trained to mimic the symbolic representation, it undergoes evaluation to verify that it retains the critical knowledge and performance attributes of the teacher model. This assessment might reveal the need for adjustments either to the symbolic representation itself or to the training methodology of the student model. Such refinements are crucial for ensuring that the student not only approximates the teacher's performance but does so in a way that is both interpretable and transparent. This emphasis on interpretability and transparency is key, as it aims to produce a student model that not only performs well but also provides insights into its decision-making processes, making it more understandable and trustworthy to users.

B. Large Language Models

LLMs are the foundation model for the language and has been the hot topic for past few years. Alot of opportunities has been created in one hand and due to ineffective use, it has also created some kind of fear among the users. In this section we will focus on the architecture of LLM followed by the training process.

1) Architecture: Transformer[52] architecture is the backbone of all the LLMs. Due to its features like parallelizable computation, attention based mechanism it has been able to reduced reliance in hand-crafted features and also improved the performance in NLP tasks. All the LLMs are directly or in-directly has the root the in the transformer architecture. Existing all the LLMs can be found to be belonging into one of the following architecture:

Encoder-Decoder Architecture: The underlying principle of this architecture involves transforming the input sequence into a fixed-length vector form, and subsequently, transforming this representation into the output sequence. The architecture is composed of two sets of Transformer blocks: one serving as the encoder and the other as the decoder. The encoder is tasked with processing the input sequence, utilizing a series of multi-head self-attention layers to convert it into latent representations. These representations are then leveraged by the decoder, which, through an autoregressive process, generates the output sequence by employing cross-attention mechanisms to focus on the latent representations provided by the encoder. PLM like T5[61], BART[132] and Flan-T5[94] uses this architecture.

Casual Decoder Architecture: The causal decoder architecture is a type of decoder-only architecture used in language modeling, where the input and output tokens are processed in the same fashion through the decoder. This architecture incorporates a unidirectional attention mask, which ensures that each input token can only attend to past tokens and itself by masking all future attentions to zeros. The GPT-series models, including GPT-1[59], GPT-2[60], and GPT-3[96], are representative language models of this architecture. Many other LLMs, such as OPT[108], BLOOM[133], and Gopher[104], have also adopted the causal decoder architecture.

Prefix Decoder Architecture: The prefix decoder architecture, also known as a non-causal decoder, is another type of decoder-only architecture which revises the masking mechanism of causal decoders to enable bidirectional attention over

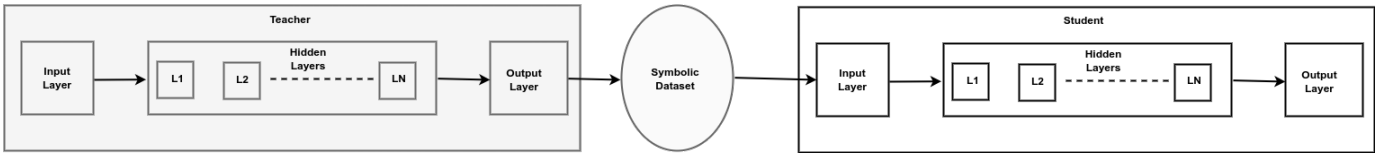


Fig. 3. Symbolic Knowledge Distillation

TABLE II
COMPARISON OF TRADITIONAL AND SYMBOLIC KNOWLEDGE DISTILLATION PROCESS

Parameters	Traditional Knowledge Distillation	Symbolic Knowledge Distillation
Nature of Knowledge Transfer	Soft outputs or logits which represent the teacher's learned probability distribution	Human-readable representations such as logical rules, decision trees, or graphical models
Interpretability and Transparency	Student model remains a black-box neural network	Student model, guided by symbolic representations offer insights into the decision-making process
Methods Used for Distillation	Techniques such as temperature scaling are used to soften the teacher's outputs	Involve methods like Layer-wise Relevance Propagation (LRP) or SHAP
Student Model	Mimic the teacher model	Can be tune to behave differently than teacher model
Data Generation	No	Yes
Layerwise Dependency	Differnet layers have different influences	No such dependency

the prefix tokens, while maintaining unidirectional attention only on generated tokens. This allows the prefix decoders to bidirectionally encode the prefix sequence and predict the output tokens autoregressively, where the same parameters are shared during encoding and decoding. Unlike the causal decoder architecture, the prefix decoder architecture can incorporate bidirectional information into the decoding process, making it more suitable for tasks that require understanding the context of the entire input sequence. Existing representative LLMs based on prefix decoders include GLM-130B[134] and U-PaLM[135].

2) Training Process of Large Language Models: The whole training process of LLM can be divided into two phases:

Pre-trainning: Pre-training LLMs involves training on extensive unlabeled text datasets to learn general language patterns and insights. The success of pre-training hinges on both the scale and quality of the training corpus, with large, diverse datasets allowing models to capture a wide array of language patterns and generalize effectively to new data.

The pre-training process unfolds in phases, starting with data collection, which is divided into general and specialized data sources. General data encompasses a wide range of text, including webpages, conversations, Q&A portals, and books, while specialized data targets more niche content like research papers, code, and multilingual texts. The second phase, data pre-processing, focuses on refining the dataset by eliminating noisy, redundant, and irrelevant content. Techniques employed include quality filtering, deduplication (at sentence, document, and dataset levels), privacy protection (removing personal information), and tokenization (splitting text into manageable units for the model). Given that LLMs are not typically retrained frequently, the pre-training phase must be approached with precision, prioritizing a balanced mix of source materials[104], and ensuring both the quantity[110] and quality[136] of the data are optimal. Pre-training tasks may involve language modeling[95], favored by decoder-only architectures for predicting subsequent tokens, or de-noising autoencoding[132], which focuses on correcting or replacing corrupted tokens.

Fine tuning or Adaptive tuning: The fine-tuning stage is crucial for adapting pre-trained LLMs to specific domains or tasks, leveraging labeled examples or reinforcement learning to refine the model's understanding and predictive capabilities. It encompasses two main strategies: instruction tuning and alignment tuning.

Instruction tuning entails the fine-tuning of a language model by incorporating explicit instructions or demonstrations during training. This approach is designed to direct the model towards desired behaviors and outcomes, facilitating a more targeted response to tasks. The instructions for this tuning can be derived from existing datasets reformatted to include clear directives or crafted to reflect specific human needs. Alignment tuning, on the other hand, aims to adjust the LLM's outputs to match human expectations accurately, a process that may involve a trade-off known as the alignment tax[106]. This concept refers to potential compromises in the model's capabilities as it is fine-tuned to prioritize outputs that are deemed more acceptable or beneficial from a human perspective. The most commonly used alignment criterias are helpfulness, honesty, and harmlessness[106][99]. Few other criteria are also mentioned like behavior, intent, incentive, and inner aspects[137].

IV. SYMBOLIC KNOWLEDGE DISTILLATION OF LARGE LANGUAGE MODELS

Symbolic Knowledge Distillation of LLMs aimed at distilling the extensive knowledge encapsulated within LLMs into more interpretable and efficient forms. It's central methodology revolves around transforming the latent knowledge of models like GPT-3 into symbolic or rule-based representations. It involves a sophisticated process designed to transform the latent, complex knowledge within these models into explicit, structured, and interpretable forms. This process begins with the careful crafting of customised prompts that guide LLMs to generate outputs rich in specific knowledge types. Following this, NLP techniques like Named Entity Recognition (NER), Part-Of-Speech (POS) tagging, and dependency parsing, are

employed to analyze and structure the responses. This step extract meaningful information and identify patterns within the text, which are then transformed into structured knowledge formats such as logic rules, knowledge graphs, or semantic frames. It derives explicit rules and patterns from the LLMs' responses, thereby facilitating the encoding of this information into symbolic representations that can be easily understood and manipulated.

The subsequent phase of this process involves the refinement and validation of the generated symbolic representations to preserve depth of knowledge and to ensure their accuracy, consistency, and practical utility. This includes refining the symbolic knowledge using the human experts or using the trained models to classify the generated knowledge on the basis of quality. The refined symbolic knowledge base undergoes validation against established benchmarks, allowing for the assessment of enhancements and ensuring the symbolic representations meet the required standards of quality and utility.

The creation of a high-quality knowledge base facilitates the training of smaller models, demonstrating that a quality dataset can significantly improve the performance of models that are 100 times smaller than their teacher counterparts[2]. This highlights the efficacy of integrating symbolic knowledge into language models, presenting a viable alternative to scaling up LLMs. Symbolic knowledge distillation generates smaller, yet more efficient models, making them suitable for deployment in everyday practical applications, offering a more resource-efficient pathway to achieving high-quality outputs in language models.

Various approaches that are used to distill the symbolic knowledge of LLMs can be categorised as:

A. Direct Distillation

The distillation of symbolic knowledge from LLMs like GPT-3 begins with the construction of a specific prompt. This prompt is designed to elicit responses that encapsulate commonsense or factual understanding. It could involve scenarios, questions, or statements that require the application of general knowledge about the world. The effectiveness of this step hinges on the ability to craft prompts that are both clear and contextually rich enough to guide the LLM towards producing relevant and insightful outputs. Upon receiving the prompt, the LLM generates a response based on its training and the intricacies of the provided context. These models, have been exposed to extensive and varied textual data, encompassing a wide array of commonsense situations and factual knowledge. This extensive training enables them to generate responses that are not only contextually appropriate but also rich in commonsense and factual knowledge. The model's response is a complex interplay of its learned patterns, linguistic understanding, and the implicit knowledge embedded within its training corpus. This step translates the implicit knowledge within the model into explicit textual responses that can be further analyzed and utilized for knowledge extraction.

The generated text is then analyzed to extract knowledge. This can be in the form of statements, inferences, or relationships that are implicitly or explicitly expressed in the text.

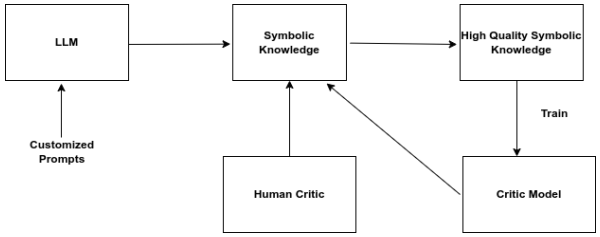


Fig. 4. Overview of Direct Distillation process LLMs

The extraction process might involve additional processing steps like parsing the text to identify relevant information or using templates to format the knowledge in a structured way. The knowledge base derived from this process can be further improved with the assistance of critics, who may be human evaluators providing feedback on the quality and acceptability of the generated content. Once a substantial volume of high-quality generated data has been accumulated, this data can be utilized to train a critic model like RoBERTa, which can be used to evaluate the generated text for accuracy, relevance, and coherence. The critic model can filter out lower-quality outputs, ensuring that only high-quality commonsense knowledge is retained. The high-quality knowledge can then be distilled into structured formats like knowledge graphs or further trained into specialized models. This process involves organizing the knowledge in a way that can be easily utilized by other systems or models.

B. Multilevel distillation of symbolic knowledge

This approach iteratively refines the knowledge transfer from a larger, pre-trained teacher model to a smaller, more efficient student model. The process begins with the teacher model, typically a LLM like GPT-3, generating initial knowledge base. The generated knowledge base is then filtered for quality, focusing on aspects like accuracy and length. The smaller student model, such as GPT2-Large, is initially trained on this filtered dataset. Subsequently, the student model generates new knowledge base, which are again filtered to enhance quality. This cycle of generation and refining through filtering is repeated iteratively, with each iteration aiming to improve fidelity and succinctness of the distilled knowledge.

During each iteration, various filters are applied to ensure the quality which are fidelity filter, length filter or contextual filter. The Fidelity Filter ensures a true representation of the input sentence, verified using an off-the-shelf Natural Language Inference (NLI) model. The Length Filter controls the length to fit within a predefined compression ratio, gradually guiding the model to produce increasingly concise output. A Contextual Filter is used in some cases, focusing on the coherence in the larger context of the text. The process results in the development of increasingly efficient student models that inherit the distillation ability of the teacher model but with enhanced control over quality. This method allows for the creation of high-quality, succinct dataset with diverse compression ratios, without relying on pre-existing annotated datasets.

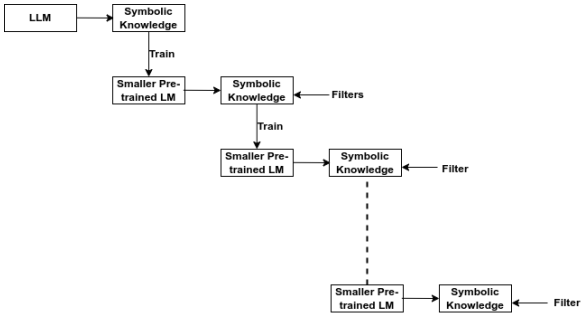


Fig. 5. Overview of Multilevel Distillation process LLMs

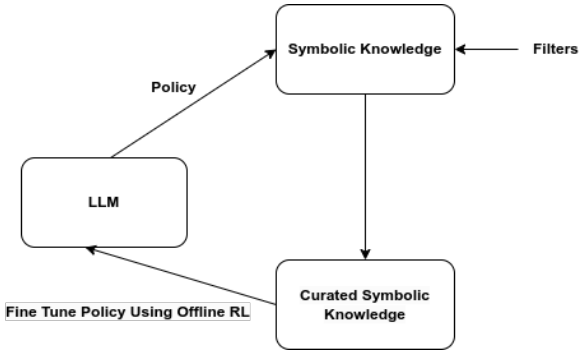


Fig. 6. Overview of Distillation process using RL

C. Distillation using Reinforcement Learning policy

The approach refines the policy of a LLM through a two-step iterative process: generating and filtering data. The first step, involves using the current LLM policy to generate a range of output predictions for given contexts, effectively augmenting the training dataset. Initially, this policy might be based on a supervised learning model, and the generated outputs may not be perfectly aligned with human preferences. However, this step is essential for creating a diverse set of potential outputs for further refinement. The generated data forms the basis for the next critical phase of the process.

In the second step, the data produced is ranked and filtered using a filters like scoring function, typically a learned reward model trained on human preferences. This step is pivotal in selecting the best outputs that align with the desired human outcomes, as determined by the scores from the reward model. The filtering threshold can be incrementally increased in subsequent iterations, ensuring that only the top-performing outputs are selected for further training. The language model is then fine-tuned on this curated dataset with an offline RL objective, adjusting its policy to produce outputs that are more likely to receive high scores. This process of generating and filtering, repeated iteratively, serves as a feedback loop, continuously refining the model's policy towards outputs increasingly aligned with human preferences.

All three techniques mentioned have been successfully applied to various research areas, including commonsense reasoning[2], translation[4], summarisation[3] , and mathematical reasoning[138], among others, yielding significant results. *Fig.*7 provides an overview of all the areas explored so far, with detailed discussions presented in the related works section. *Table.III* offers insights into each research area, categorizing them based on the techniques discussed above.

V. Related Works

In this segment, we begin by exploring the foundational work that positions LLMs as a knowledge base and then delve into research focused on analyzing the knowledge contained within LLMs. Lastly, we review efforts aimed at distilling this knowledge into a symbolic form. An overview of this concept is presented in *Fig.*7.

A. Knowledge Base of LLM

LLM can act as a knowledge base or oracle that performs well on open-domain question answering without fine-tuning[1]. LLM can also function as the domain-specific KBs in biomedical field however they are highly influenced by prompt bias and synonym variance[139]. It rapidly and stably acquires linguistic knowledge, including syntax, grammar, and parts of speech, predominantly in the early stages of pre-training, showing little variation across different domains. In contrast, the assimilation of factual and commonsense knowledge is slower, more sensitive to the domain of the training data, and exhibits a more gradual progression throughout the pre-training period[140].

B. Consistency of Knowledge In LM

The research[141] sheds light on the consistency of knowledge in PLMs like BERT and RoBERTa. Their findings reveal a concerning lack of consistency in these models, particularly when responding to paraphrased queries with factual content. The study[142] adds another layer of complexity to this issue by highlighting the challenges PLMs face in accurately processing negated facts and their susceptibility to being misled by contextually irrelevant or misleading information.

C. Editing the Knowledge in LLM

Editing knowledge in LLMs has become a prominent area of research with several innovative approaches proposed to address this challenge. Constrained layer-wise fine-tuning[143] formulates knowledge modification as a constrained optimization problem and allows for fine-tuning specific layers to update knowledge while retaining existing information. [144] introduced the concept of Knowledge Neurons, enabling pinpointing specific components responsible for factual knowledge within LLMs and providing the means to manipulate them for altering model output. The KNOWLEDGEEDITOR[145] offers an efficient way to update factual knowledge in pre-trained LLMs without extensive retraining.The paper[146] introduces methods for detecting, updating, and visualizing beliefs in LLM by using the Sequential Local and Generalizing (SLAG) update objective. Model Editor Networks with Gradient Decomposition (MEND)[147] efficiently edit large-scale pre-trained models by transforming gradients during fine-tuning. Continual Knowledge Learning (CKL)[148] addresses the challenge of updating and maintaining the relevancy of world knowledge in LLMs.

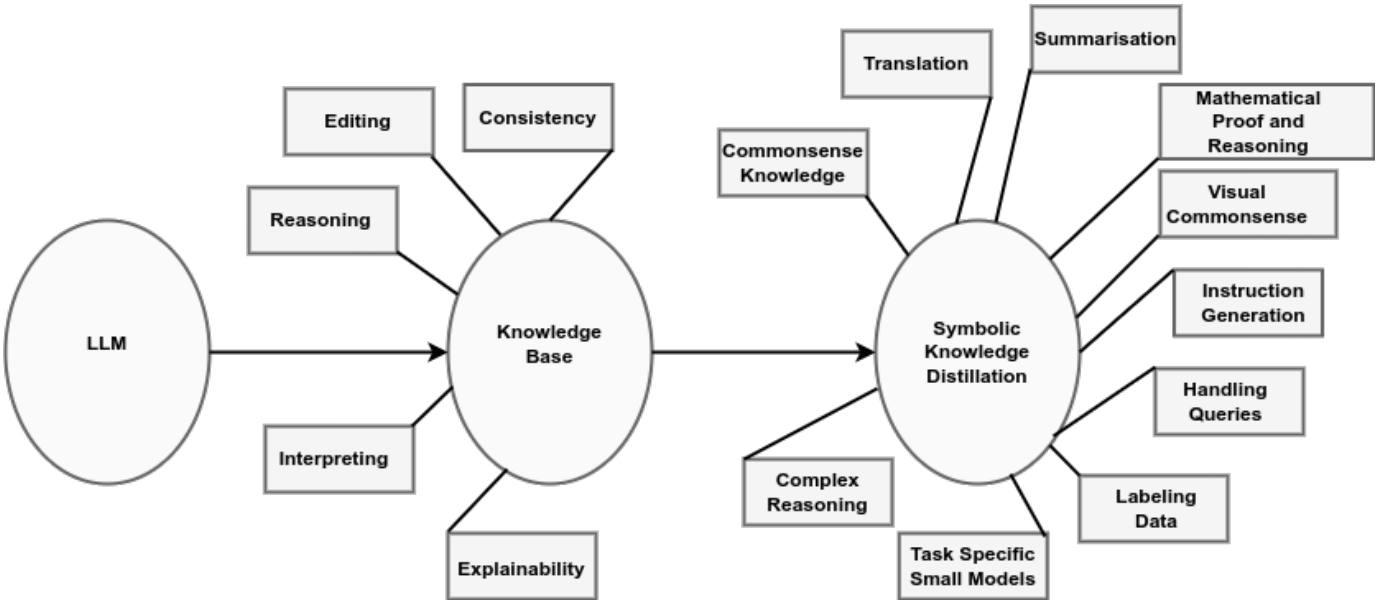


Fig. 7. Overview of Related Works

D. Reasoning with Knowledge in LLM

The research landscape concerning reasoning abilities in PLMs and transformers, has seen significant exploration and development. The paper[149] found that while BERT could learn simpler one-hop rules, it struggled with more complex two-hop rules and distinguishing between symmetric and non-symmetric relations. [150] demonstrates that transformers can effectively emulate reasoning over language, achieving high accuracy on various synthetic datasets that require different depths of inference and can act as limited "soft theorem provers". PROVER[151] extended [150] to answer binary questions over rule-bases while generating corresponding proofs for enhanced interpretability. ProofWriter[152] stands out for its ability to produce implications and corresponding natural language proofs from given theories, using the T5 transformer architecture. The paper[153] explores the capability of Transformer Language Models (TLMs) in logical reasoning with natural language focusing on first-order logic proofs. The paper[154] explore the capacity of transformer models to perform deductive reasoning on logical theories expressed in natural language by introducing a method for generating challenging reasoning datasets whereas the paper[155] enhance the deductive reasoning abilities of PLMs using soft Horn rules and achieved high performance on unseen logical rules and showed improved understanding of logical properties like negation and symmetry. The paper[156] introduces a novel dataset to evaluate the mathematical reasoning capabilities of neural networks, focusing on problems across arithmetic, algebra, probability, and calculus.

The paper[157] integrates commonsense reasoning on natural language question-answering tasks by employing smaller language models,and demonstrate competitive performance against large PLMs. RICA (Robust Inference using Commonsense Axioms)[158],found that PLMs are vulnerable to perturbation attacks, where minor changes in input data drastically alter their conclusions. The paper[159] presents the Common Sense Explanations (CoS-E) dataset and the Commonsense Auto-Generated Explanation (CAGE) framework, which leverages natural language explanations(human-like explanations) to improve model's reasoning capabilities.

E. Interpreting the Knowledge of LLM

Interpreting the knowledge encoded in LLMs has been advanced through various studies, each contributing unique insights into how these models capture and process linguistic information. [160] argue that attention weights often don't align with other feature importance measures and can produce similar predictions despite different attention distributions. This view is nuanced by [161], who suggest that attention can serve as an explanation, but its validity depends on the context and testing methods. [162] also investigate attention in text classification, finding that while there is some correlation between attention weights and model predictions, attention weights alone are not definitive indicators of input importance and propose that gradient-based attention weight rankings provide a deeper understanding.

The study[163] include method for quantifying non-linearity in transformers, particularly in feed-forward networks. They reveal a non-distinct feature extraction process in BERT layers, influenced by skip connections. [164] demonstrate that transformer layers function as key-value memories, capturing textual patterns and inducing distributions over the output vocabulary, with lower layers focusing on shallow patterns and upper layers on semantic ones. [165] show that factual associations in GPT models are tied to localized computations, particularly in middle-layer feed-forward modules.

F. Explainability in LLM

The study[166] investigates the application of Influence Functions (IFs) to identify artifacts in models, comparing their

effectiveness with that of common word-saliency methods. Researchers in study [167] compare IFs with simpler retrieval-based methods and suggest that despite the complexity of IFs, simpler methods can achieve comparable performance. Exploring further in study[168], they introduce Training-feature attribution (TFA), which synergizes saliency maps and instance attribution to effectively uncover artifacts. Researcher in [169] propose Human In the Loop Debugging using Influence Functions (HILDIF), a pipeline that employs influence functions for debugging deep text classifiers, allowing human involvement in enhancing model performance.

In a different approach, study [170] presents a novel method for training language models to generate natural text explanations alongside their predictions, utilizing the text-to-text framework[61]. Addressing the challenge of inconsistency in natural language explanations, [171] introduces an adversarial framework to identify and measure these inconsistencies. The Proto-Trex model[172] uses prototypical examples to explain model predictions, thus mitigating the opacity often associated with complex models. Research[173] enhances interpretability by extracting key text segments, termed "rationales", serving as justifications for model predictions. Study[174] works on improving commonsense reasoning by employing contrastive explanations generated through specialized prompts, aligning model reasoning more closely with human cognitive patterns.

G. Symbolic Knowledge Distillation

The conducted research works in this area can be categorised as follows:

1) Commonsense Knowledge: The study[2] introduces a transformative shift in the conventional practice, transitioning from the traditional 'from-human-to-corpus-to-machine' approach to an innovative 'from-machine-to-corpus-to-machine' paradigm through the introduction of symbolic knowledge distillation. In their research, the authors not only succeed in creating a substantially larger common-sense dataset from ATOMIC resource[175], approximately ten times larger than previously manually synthesized datasets, but also enhance its diversity and quality. Their novel approach involves training the common-sense model using this newly generated knowledge graph. Despite being only 1/100th of its predecessor model, it outperforms the previous model, showcasing the effectiveness of their approach. The paper[176] introduces NOVACOMET, an innovative open commonsense knowledge model that merges the strengths of both knowledge and general task models. This model, built upon symbolic knowledge distilled from proprietary models like GPT-3, creates an auditable discrete knowledge graph, NOVATOMIC, which facilitates open-format training and application to a wide array of reasoning tasks. It demonstrates superior performance in commonsense reasoning, outperforming comparable models in various benchmarks. The model's training involves novel techniques like commonsense field masking for enhanced flexibility in knowledge handling. Iterative Imitation and Decoding for Distillation(I2D2)[177] framework employs a four-stage process that includes prompt construction, constrained decoding using NeuroLogic Decoding, critic filtering, and self-imitation learning, where the model is iteratively refined based on its own high-quality outputs. A new corpus, Gen-A-tomic, was created to provide diverse and accurate commonsense knowledge. I2D2 demonstrated superior performance in accuracy and precision over larger models like GPT-3, with GPT-2 XL showing significant improvements through self-imitation learning iterations.

2) Translation: Reinforced Self-Training (ReST)[4] is a method to align LLMs with human preferences in the realm of machine translation. This approach incorporates reinforcement learning from human feedback (RLHF) to enhance the output quality. ReST initiates by generating a dataset through sampling from the initial LLM policy, followed by the application of offline reinforcement learning algorithms to refine the policy. This method is identified as more efficient than traditional online RLHF techniques, primarily because it facilitates the creation of the training dataset in an offline manner, promoting the reuse of data. The effectiveness of ReST is demonstrated through significant improvements in translation quality, validated by both automated metrics and human evaluations across various machine translation benchmarks.

3) Summarisation: REFEREE[3] is a framework for reference-free sentence summarization that allows for direct control of compression ratio. It uses Symbolic Knowledge Distillation to distill latent knowledge from PLMs, resulting in smaller but better summarizers with sharper controllability. The framework employs iterative distillation of knowledge, where student models from previous iterations serve as teacher models in the next iteration. This iterative process also generates a high-quality dataset of sentence-summary pairs with varying compression ratios. The final student models outperform the larger GPT3-Instruct model in terms of compression ratio controllability without compromising the quality of the summarization.

4) Mathematical Proof and Reasoning: The paper[138] presents a method called expert iteration, which combines proof search with learning to improve language modeling in formal mathematics. The method involves finding new original proofs for the same statements and closing marginally harder statements at each iteration, which in turn provides more useful training data for the next iteration. By interleaving proof search with learning, expert iteration is able to dramatically outperform proof search only. The paper demonstrates the effectiveness of expert iteration on a manually curated set of problem statements and achieves state-of-the-art results on the miniF2F benchmark, a set of formalized statements of mathematical problems from various competitions.The paper[178] explores the concept of distilling abilities from LLMs into smaller ones, specifically for enhancing their performance in multi-step math reasoning tasks. The process begins with generating a dataset using a larger model (like GPT-3.5) employing chain-of-thought reasoning, where the model details the steps leading to a solution. This dataset is then used to fine-tune a smaller T5 model, with the aim of specializing its abilities in the specific area of multi-step reasoning. This fine-tuning process allows the smaller model to learn the complex reasoning patterns demonstrated by the larger model.

5) Visual Commonsense: Localized Symbolic Knowledge Distillation (LSKD)[179] enhances vision-language models by focusing on localized regions within images. This method addresses a significant limitation in existing models, which interpret images as a whole, by introducing Localized Visual Commonsense models that can specify and reason about multiple distinct regions in an image. The authors develop a scalable framework for generating localized visual commonsense statements and establish the Localized Commonsense Knowledge Corpus, which aids in expanding the capabilities of vision+language models to include references-as-input. The paper highlights the state-of-the-art zero-shot performance of these models on three localized visual reasoning tasks and showcases the superiority of the student model over the teacher model through human evaluation.

6) Instruction Generation: Traditional instruction-tuned models, reliant on human-written instruction data, often lack diversity and creativity, constraining the generality of the model. SELF-INSTRUCT[180] mitigates this by enabling models to generate their own instructions, inputs, and outputs, which are then used for fine-tuning. This process involves generating task instructions, classifying them, creating instances via input-first or output-first approaches, and filtering out low-quality data. The approach significantly reduces the need for human-labeled data, fostering a broader and more creative instructional capability in LMs. The performance evaluation shows that the GPT3SELF-INST model, fine-tuned on this self-generated data, substantially outperforms the vanilla GPT-3 in instruction-following tasks and closely matches the performance of models like InstructGPT001. Alpaca[181] enhance the SELF-INSTRUCT data generation pipeline by employing the more advanced text-davinci-003 model for instruction data generation that explicitly defines the requirements for instruction generation, aiming for more focused and relevant outputs. The adoption of aggressive batch decoding, producing 20 instructions simultaneously, significantly reduces data generation costs and simplifying the pipeline by eliminating the distinction between classification and non-classification instructions and generating only a single instance per instruction, instead of 2 to 3, streamlines the process. Evol-Instruct[182] is a novel method that uses LLMs to automatically generate a vast array of complex instructional data. This approach begins with simple initial instructions and employs the LLM to evolve these into more sophisticated and diverse instructions through in-depth and in-breadth evolution processes. It enhances instructions by adding constraints, increasing reasoning complexity, and diversifying topics, thus creating a rich dataset for fine-tuning LLMs. This dataset is used to train the LLaMA model, resulting in WizardLM, a model demonstrating superior performance in following complex instructions compared to human-generated datasets and existing models like ChatGPT.

7) Handling queries: Vicuna-13B[183] is an open-source chatbot developed by fine-tuning the LLaMA model with around 70,000 user-shared ChatGPT conversations from ShareGPT. It demonstrates superior performance, achieving over 90% of ChatGPT's quality, and surpassing other models like LLaMA and Stanford Alpaca. The training, which cost approximately $300, utilized advanced techniques for handling multi-turn conversations. Despite its advancements, Vicuna-13B shares common LLM limitations, such as challenges in reasoning or math tasks, and has potential issues with factual accuracy and safety. Koala[184], a chatbot model developed by fine-tuning Meta's LLaMA with web-sourced dialogue data, including interactions with large models like ChatGPT. Koala demonstrates competitive performance against established models such as ChatGPT and Stanford's Alpaca, particularly in handling real user queries. ASK ME ANYTHING PROMPTING (AMA)[185] is a prompting method for improving the performance of LLMs like GPT-3. AMA leverages multiple effective but imperfect prompts, aggregating them using weak supervision to enhance prediction quality. This method primarily utilizes open-ended question-answering formats, which are found to be more effective than restrictive prompts. AMA's recursive use of the LLM to transform task inputs into these formats, combined with the aggregation of diverse prompts, demonstrates significant improvements in LLM predictions. QAMELEON[186] is an innovative approach to multilingual question answering (QA) systems, leveraging PLMs within a few-shot learning framework. PLMs generate QA pairs in multiple languages, significantly reducing the need for extensive, language-specific training datasets. By requiring only a minimal number of examples (as few as five per language), QAMELEON efficiently fine-tunes QA models, overcoming traditional constraints of resource-intensive data annotation. This approach not only simplifies and accelerates the development of multilingual QA systems but also achieves superior accuracy and efficiency, demonstrating its potential as a scalable and effective solution in NLP.

8) Labeling Data: The research paper[81] examines the efficacy of using GPT-3 for data labeling in NLP tasks, highlighting its cost-effectiveness compared to traditional human labeling. The study reveals that GPT-3 can reduce labeling costs by 50% to 96% across various tasks, including sentiment analysis, text classification, and summarization. The paper introduces a novel framework that combines GPT-3 generated pseudo labels with human labels, improving performance under limited budgets. Furthermore, an active labeling strategy is explored, where low-confidence labels by GPT-3 are re-annotated by humans, enhancing label quality. Despite these benefits, the paper notes that GPT-3 is more suited for low-stakes labeling tasks, as its reliability in high-stakes scenarios remains limited. The research[82] presents a novel method for utilizing PLMs in tasks with scarce labeled training data. This technique involves prompting the LM with multiple queries about an example, and the model's responses are then interpreted as votes for specific labels or as abstentions. This process, integrated within a weak supervision framework, leverages the capabilities of the LM as a labeling function. The Snorkel system is subsequently employed to clean and refine these noisy label sources, culminating in the creation of enhanced training data for an end classifier.

9) Task Specific Small Models: The method, "Distilling step-by-step"[187], involves extracting rationales from LLMs alongside output labels. These rationales, serving as detailed explanations for model predictions, are then used in a multi-

task learning framework to train smaller models on both label and rationale prediction tasks. This technique significantly reduces the data and model size required, enabling smaller models to surpass the performance of LLMs more efficiently. The paper demonstrates the effectiveness of this approach across multiple datasets and tasks, showcasing it as a resource-efficient alternative to standard finetuning and traditional distillation methods.

10) Complex Reasoning: Orca [188] is designed to enhance the capabilities of smaller models through imitation learning from large foundation models (LFMs). Traditional methods faced issues like limited imitation signals, small-scale homogeneous training data, and inadequate evaluation, leading to an overestimation of the small models' capabilities. These models often imitated the style but not the reasoning process of LFMs. Orca addresses these challenges by learning from GPT-4's rich signals, including explanation traces, step-by-step thought processes, and complex instructions, with guidance from ChatGPT as a teacher. This approach enables progressive learning through large-scale and diverse imitation data. Orca significantly outperforms state-of-the-art instruction-tuned models like Vicuna-13B in complex zero-shot reasoning benchmarks, achieving more than a 100% improvement in Big-Bench Hard (BBH) and a 42% improvement in AGIEval. Orca reaches parity with ChatGPT in BBH and exhibits competitive performance in professional and academic exams like the SAT, LSAT, GRE, and GMAT, in zero-shot settings without Chain of Thought (CoT), though it still trails behind GPT-4. Orca 2[189] builds upon the Orca project, focusing on enhancing smaller LMs' reasoning abilities. Orca 2 continues exploration, particularly addressing the limitations of imitation learning, which had been the primary method for training small LMs. This method, while effective in replicating the output of larger models, often fell short in reasoning and comprehension skills. It introduces various reasoning techniques (e.g., step-by-step processing, recall-then-generate, recall-reason-generate, extract-generate, direct-answer methods) and focuses on teaching small LMs to choose the most effective reasoning strategy for a given task. This approach aims to enable small LMs to perform at their best, regardless of their size, by utilizing more nuanced data and training strategies. The system is described as a "Cautious Reasoner," learning to execute specific reasoning steps and strategize at a higher level how to approach particular tasks.

TABLE III
RELATED WORKS IN SYMBOLIC KNOWLEDGE DISTILLATION

Research	Types	Application
[2]	Direct	Commonsense Reasoning
[3]	Multi-level	Summarisation
[4]	RL based	Translation
[176]	Direct	Commonsense Reasoning
[177]	Direct	Commonsense Reasoning
[138]	Direct	Mathematical Proof and Reasoning
[178]	Direct	Mathematical Proof and Reasoning
[179]	Direct	Visual Commonsense Reasoning
[180]	Direct	Instruction Generation
[181]	Direct	Instruction Generation
[182]	Direct	Instruction Generation
[183]	Direct	Handling Queries
[184]	Direct	Handling Queries
[185]	Direct	Handling Queries
[186]	Direct	Handling Queries
[81]	Direct	Labeling Data
[82]	Direct	Labeling Data
[187]	Direct	Generating Task Specific Small Models
[188]	Direct	Complex Reasoning
[189]	Direct	Complex Reasoning

VI. OPPORTUNITIES

Symbolic Knowledge distillation of LLM has been one of the heated topics and has been gaining rapid popularity. Among the various areas, the most prominent areas where it can be applied are:

A. Creation of larger, diversified and qualitative dataset

It offers significant potential in enhancing dataset quality and diversity. This process involves extracting structured knowledge from LLMs to create datasets that are not only larger in scale but also exhibit a broader range of qualities and characteristics. These enriched datasets can be pivotal in training more robust and efficient machine learning models, leading to advancements in various domains such as NLP, image recognition, and beyond. The ability to generate high-quality datasets from LLMs accelerates the development of more sophisticated AI systems, contributing to advances in both academic research and practical applications.

B. Reduction in the cost by utilising machines in the low level task under guidance on humans

Implementing symbolic knowledge distillation in low-level tasks allows for the effective delegation of routine and repetitive tasks to machines, significantly reducing operational costs. By leveraging the distilled knowledge from LLMs, machines can perform these tasks with a high degree of accuracy and efficiency, under the supervision of human experts. This collaboration between human intelligence and machine capabilities leads to optimized resource utilization, where humans focus on more complex, creative, or decision-making tasks while machines handle the routine aspects, thereby enhancing overall productivity and cost-effectiveness.

C. Smaller and more powerful models than LLMs for summarization, translation, common sense etc

Distilling knowledge from LLMs into smaller models presents a promising avenue for creating compact yet powerful AI tools. These distilled models retain the core capabilities of their larger counterparts but with reduced computational requirements. This makes them particularly suitable for applications like text summarization, language translation, and common sense reasoning, where efficiency and speed are crucial. These smaller models offer the dual benefits of lower resource consumption and faster processing times, making them ideal for deployment in environments with limited computational resources or for applications requiring real-time responses.

TABLE IV
RELATED WORKS IN SYMBOLIC KNOWLEDGE DISTILLATION WITH THEIR MAJOR COMPONENTS

Research	Teacher	Student	Dataset Generated	Size of Dataset
[2]	GPT-3(175B)	$COMET^{distil}(1.5B)$	Commonsense Knowledge Graph	6.5M
[3]	GPT-3	REFEREE-CONTROL	Sentence-summary pairs	100K
[4]	Encoder-Decoder Architecture	Teacher Itself	Translation Dataset	N/A
[176]	GPT-3	NOVACOMET	NOVATOMIC	2.2M
[177]	GPT-3	GPT-2	Gen-A-tomic	7M
[138]	Decoder Only Architecture	Teacher Itself	Tactic Dataset	N/A
[178]	GPT-3.5	FlanT5	Math Reasoning	N/A
[179]	ChatGPT	BLIP-2	Localized Commonsense Knowledge	1M
[180]	GPT-3	Teacher Itself	Instruction Dataset	82K
[181]	GPT-3.5	7B LLaMA	Instruction Dataset	52K
[182]	ChatGPT	WizardLM	Instruction Dataset	250K
[183]	ChatGPT	Vicuna-13B	Conversational Dataset	70K
[184]	ChatGPT	Koala-13B	Conversational Dataset	N/A
[185]	GPT3-175B	GPT-J-6B	Prompt Dataset	N/A
[186]	PaLM-540B	mT5-XL	Multilingual QA	47173
[81]	GPT-3	RoBERTa	Labeled Data	5.1K
[82]	GPT-3	T0++	Labeled Data	N/A
[187]	540B PaLM	770M T5	Rationales	N/A
[188]	GPT-4	Orca(13B)	Zero shot queries	5M
[189]	GPT-4	Orca-2	Progressive queries	817K

D. *Instruction tuning*

Instruction tuning, in the context of symbolic knowledge distillation from LLMs, refers to the process of refining and optimizing AI models to better understand and execute specific instructions. This approach enhances the model's ability to interpret and act upon user commands accurately, leading to more intuitive and user-friendly AI systems. Instruction tuning is particularly relevant in applications where user interaction is key, such as virtual assistants, educational tools, and interactive AI systems. By focusing on instruction tuning, developers can create AI models that are not only powerful in their capabilities but also align closely with user expectations and needs, facilitating more effective and seamless human-AI interactions.

E. *Novel Algorithm and Evaluation Benchmark*

Size alone does not determine the quality of language generation. Innovative approaches, such as those seen in I2D2[177], present a viable option, particularly in scenarios where utilizing massive models like GPT-3 is impractical. Given that this field is in its infancy, the evaluation benchmarks are quite intricate and require significant refinement. Current evaluation techniques are from traditional knowledge distillation benchmarks and must be updated to fit this novel area of study.Symbolic Knowledge Distillation of LLMs involves two components: the neural aspect (LLMs) and the symbolic aspect (distilled symbolic knowledge). Together, these form a Neurosymbolic model, which necessitates the development of new benchmarks for evaluation, testing, and validation[190].

F. *Creation of Open source data and open model*

The concept of symbolic distillation presents an intriguing avenue for creating open source data and models within the realm of LLMs. Currently, many LLMs are proprietary and trained on closed-source data, limiting accessibility and transparency. Symbolic distillation involves extracting symbolic knowledge and representations from LLMs, which can then be used to generate open source data. This open data can serve as the foundation for training new models that are open source, thereby democratizing access to advanced language models. By transitioning from closed source to open source, we can promote transparency, collaboration, and innovation in the field of NLP, aligning with the principles of open science and open AI.

G. *Self Improvement of LLMs*

Reinforcement Learning from Human Feedback (RLHF) has emerged as a prevalent method for refining LLMs. However, the involvement of human input inherently constrains its efficacy and outcomes to the limits of human capabilities. Upon undergoing fine-tuning, LLMs can surpass human performance levels. Leveraging these enhanced models to autonomously fine-tune themselves, either via rewards[87] or prompt tuning or alternative mechanisms, presents a viable strategy for eliminating the limitations imposed by human intervention opening the gateway for Superintelligence. When employing Reinforcement Learning (RL) for fine-tuning LLMs by themselves, opting for Neurosymbolic RL approaches is often more advantageous. This is because Neurosymbolic RL not only aids in the tuning process but also enhances the model with the ability to interpret and explain its decision-making process comprehensively[191].

H. *Cross-domain Symbiosis*

Symbolic Knowledge extracted from LLMs extends its utility beyond the linguistic domain. Studies, such [179], demonstrate that textual knowledge can augment visual models by offering explanations and enhancing efficiency. This interdisciplinary application can be further leveraged in diverse fields such as medical imaging, autonomous driving, and surveillance, serving not only to elucidate model outputs but also to improve transfer from one domain to another(simulation

to real) by providing the semantic anchors[192]. This cross-domain synergy highlights the potential of Symbolic Knowledge in broadening the applicability and understanding of complex AI systems.

I. Industrial Applications

Symbolic knowledge distillation reveals a critical insight: the effectiveness of LLMs is significantly influenced not only by their size (number of parameters) but more importantly by the quality of the datasets on which they are trained. It highlights the significant role of symbolic knowledge distillation in enhancing domain-specific AI applications by fine-tuning LLMs with specialized corpora and instruction-following data. Notable implementations include LawyerLLaMA[193] and LawGPT[194] for legal services, HuatuoGPT[195] and ChatDoctor[196] for medical applications, XuanYuan[197] for finance, DARWIN Series[198] and SciGLM[199] for scientific research. These tailored models demonstrate substantial improvements in accuracy, efficiency, and usability, showcasing the transformative potential of symbolic knowledge distillation in various industries.

VII. Challenges

A. Ensuring Data Quality and Diversity in Datasets

While symbolic knowledge distillation from LLMs promises to enhance dataset quality, a major challenge is ensuring the high quality and representativeness of the generated data. The datasets derived from LLMs may inherit biases or inaccuracies present in the original training data of these models. This can lead to the propagation of errors and skewed perspectives in the new datasets, affecting the reliability and fairness of AI systems trained on them. Ensuring data quality requires rigorous validation processes and mechanisms to identify and mitigate biases, which can be resource-intensive, complex, is still an not so explored area.

B. Balancing Automation and Human Oversight in Dataset Generation

While utilizing machines under human guidance can reduce costs, achieving the right balance between automation and human oversight is challenging. Over-reliance on automation may lead to oversight of nuanced or exceptional cases that require human judgment. Conversely, excessive human intervention can negate the efficiency gains from automation. Establishing effective protocols and systems for human-machine collaboration, where machines handle routine tasks while humans oversee and intervene as needed, is crucial but difficult to optimize.

C. Developing Compact Models Without Compromising Performance

Creating smaller models from LLMs that maintain high performance levels is a significant challenge.There are research efforts to quantize LLMs to ultra-low bit sizes, their performance has been found lacking and does not meet the standards required for industrial applications[200][201]. Symbolic Knowledge Distillation has shown promise in specific, narrower fields such as translation, summarization, and commonsense reasoning. However, it must evolve into a comprehensive symbolic knowledge base capable of generalizing across all domains. Developing these compact models requires sophisticated techniques to compress and optimize the knowledge transfer without losing the nuances and depth of the original model.

D. Effective Instruction Tuning for Diverse Applications

Instruction tuning in AI models poses the challenge of adapting to a wide range of instructions and use cases. Models must be versatile enough to understand and execute a variety of commands accurately across different domains and contexts. This requires extensive training and fine-tuning, which can be resource-intensive. Moreover, ensuring that the models remain adaptable and up-to-date with evolving user needs and language usage is an ongoing challenge, necessitating continuous monitoring and updates.

E. Adaptability and Continuous Learning

Ensuring that distilled models can adapt to new information and evolving data landscapes is challenging. Continuous learning mechanisms that allow models to update their knowledge without compromising efficiency or requiring complete retraining are essential for keeping distilled models relevant and effective.

VIII. Lesson Learned and Key Takeaways

A. Efficiency Through Distillation

Symbolic knowledge distillation demonstrates a powerful method to enhance the efficiency of LLMs. By distilling complex, large-scale models into smaller, more manageable versions without significant loss in performance, researchers can achieve remarkable efficiency gains. This approach not only reduces computational requirements but also makes advanced AI capabilities more accessible for applications with limited resources.

B. Advancement in Commonsense Reasoning

The transition to a 'from-machine-to-corpus-to-machine' paradigm marks a significant advancement in commonsense reasoning. This innovative approach, through the creation of extensive and diverse datasets like ATOMIC and models like NOVACOMET, underscores the potential of machine-generated knowledge in improving AI's understanding and application of commonsense knowledge.

C. Innovation in Data Generation and Use by Collaborating Human Intelligence and Machine Capabilities

LLMs has the potential in generating high-quality, diverse datasets. These datasets serve as a foundation for training more robust models, emphasizing the importance of data quality, diversity, and the innovative use of symbolic knowledge in dataset creation. The effective collaboration between human

oversight and automated processes in dataset generation and task execution highlights the synergistic potential of combining human intelligence with machine efficiency. This collaboration is key to overcoming current limitations and unlocking new capabilities in AI systems.

D. Cross-Domain Applications

The applications of symbolic knowledge distillation extend beyond NLP into areas such as visual commonsense reasoning and mathematical proof solving. This cross-domain applicability showcases the versatility of distilled models and their potential to revolutionize various fields by enhancing model performance and understanding.

E. Instruction Tuning and Generation

The development and refinement of techniques for instruction tuning and generation signify a leap towards creating more user-friendly and intuitive AI systems. Models capable of generating their own instructions or being finely tuned to understand and execute specific commands can lead to more natural and effective human-AI interactions.

F. Challenges and Opportunities

While the advancements are notable, they also underscore challenges such as ensuring data quality, balancing automation with human oversight, and developing compact models without compromising performance. Addressing these challenges presents opportunities for further research and innovation in model training, dataset creation, and the development of algorithms for enhanced capabilities and benchmark for the evaluation.

To address the identified gaps in current research on symbolic knowledge distillation, it is crucial to first ensure the quality and diversity of datasets through rigorous validation to identify and mitigate biases inherited from LLMs, ensuring the trustworthy knowledge distillation. Balancing automation and human oversight is also essential; effective protocols for human-machine collaboration can optimize efficiency while ensuring nuanced cases are handled appropriately.Though the size of data required for efficient distillation is still unknown, research[202] propose that only 1000 high quality human curated data is enough. Another challenge is developing compact models without compromising performance, which requires sophisticated techniques to compress and optimize knowledge transfer while maintaining the depth of the original models. Effective instruction tuning for diverse applications demands extensive training and fine-tuning to ensure models can accurately execute various commands across domains. Ensuring adaptability and continuous learning in distilled models is vital, necessitating mechanisms for ongoing updates without compromising efficiency. Addressing these areas will advance symbolic knowledge distillation towards more reliable and practical applications.

IX. Conclusion

This survey paper has explored the emerging and crucial domain of symbolic knowledge distillation in LLMs. As LLMs continue to grow in scale and complexity, the need to effectively extract and represent their extensive knowledge becomes paramount. By categorizing existing research based on methodologies and applications, we have highlighted how symbolic knowledge distillation can enhance the transparency and functionality of smaller, more efficient AI models. This comprehensive overview underscores the significance of symbolic knowledge distillation in advancing more accessible and efficient AI systems. While there is a notable lack of comprehensive research in this area, our survey paper fills this crucial gap by offering an extensive review of the current state of symbolic knowledge distillation in LLMs, shedding light on methodologies, challenges, and advancements in this field.

References

[1] F. Petroni, T. Rocktäschel, P. Lewis, A. Bakhtin, Y. Wu, A. H. Miller, and S. Riedel, "Language models as knowledge bases?" *arXiv preprint arXiv:1909.01066*, 2019.

[2] P. West, C. Bhagavatula, J. Hessel, J. D. Hwang, L. Jiang, R. L. Bras, X. Lu, S. Welleck, and Y. Choi, "Symbolic knowledge distillation: from general language models to commonsense models," *arXiv preprint arXiv:2110.07178*, 2021.

[3] M. Sclar, P. West, S. Kumar, Y. Tsvetkov, and Y. Choi, "Referee: Reference-free sentence summarization with sharper controllability through symbolic knowledge distillation," *arXiv preprint arXiv:2210.13800*, 2022.

[4] C. Gulcehre, T. L. Paine, S. Srinivasan, K. Konyushkova, L. Weerts, A. Sharma, A. Siddhant, A. Ahern, M. Wang, C. Gu *et al.*, "Reinforced self-training (rest) for language modeling," *arXiv preprint arXiv:2308.08998*, 2023.

[5] W. X. Zhao, K. Zhou, J. Li, T. Tang, X. Wang, Y. Hou, Y. Min, B. Zhang, J. Zhang, Z. Dong *et al.*, "A survey of large language models," *arXiv preprint arXiv:2303.18223*, 2023.

[6] B. Min, H. Ross, E. Sulem, A. P. B. Veyseh, T. H. Nguyen, O. Sainz, E. Agirre, I. Heintz, and D. Roth, "Recent advances in natural language processing via large pre-trained language models: A survey," *ACM Computing Surveys*, vol. 56, no. 2, pp. 1–40, 2023.

[7] M. U. Hadi, R. Qureshi, A. Shah, M. Irfan, A. Zafar, M. B. Shaikh, N. Akhtar, J. Wu, S. Mirjalili *et al.*, "Large language models: a comprehensive survey of its applications, challenges, limitations, and future prospects," *Authorea Preprints*, 2023.

[8] Y. Chang, X. Wang, J. Wang, Y. Wu, K. Zhu, H. Chen, L. Yang, X. Yi, C. Wang, Y. Wang *et al.*, "A survey on evaluation of large language models," *arXiv preprint arXiv:2307.03109*, 2023.

[9] D. Zan, B. Chen, F. Zhang, D. Lu, B. Wu, B. Guan, W. Yongji, and J.-G. Lou, "Large language models meet nl2code: A survey," in *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2023, pp. 7443–7464.

[10] E. Kasneci, K. Sessler, S. Küchemann, M. Bannert, D. Dementieva, F. Fischer, U. Gasser, G. Groh, S. Günnemann, E. Hüllermeier, S. Krusche, G. Kutyniok, T. Michaeli, C. Nerdel, J. Pfeffer, O. Poquet, M. Sailer, A. Schmidt, T. Seidel, M. Stadler, J. Weller, J. Kuhn, and G. Kasneci, "Chatgpt for good? on opportunities and challenges of large language models for education," *Learning and Individual Differences*, vol. 103, p. 102274, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1041608023000195

[11] B. AlKhamissi, M. Li, A. Celikyilmaz, M. Diab, and M. Ghazvininejad, "A review on language models as knowledge bases," 2022. [Online]. Available: https://arxiv.org/abs/2204.06031

[12] S. Razniewski, A. Yates, N. Kassner, and G. Weikum, "Language models as or for knowledge bases," *arXiv preprint arXiv:2110.04888*, 2021.

[13] J. Huang and K. C.-C. Chang, "Towards reasoning in large language models: A survey," *arXiv preprint arXiv:2212.10403*, 2022.

[14] H. Zhao, H. Chen, F. Yang, N. Liu, H. Deng, H. Cai, S. Wang, D. Yin, and M. Du, "Explainability for large language models: A survey," *arXiv preprint arXiv:2309.01029*, 2023.

[15] Y. Wang, W. Zhong, L. Li, F. Mi, X. Zeng, W. Huang, L. Shang, X. Jiang, and Q. Liu, "Aligning large language models with human: A survey," *arXiv preprint arXiv:2307.12966*, 2023.

[16] S. Zhang, L. Dong, X. Li, S. Zhang, X. Sun, S. Wang, J. Li, R. Hu, T. Zhang, F. Wu *et al.*, "Instruction tuning for large language models: A survey," *arXiv preprint arXiv:2308.10792*, 2023.

[17] X. Zhu, J. Li, Y. Liu, C. Ma, and W. Wang, "A survey on model compression for large language models," *arXiv preprint arXiv:2308.07633*, 2023.

[18] 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.

[19] A. Fan, B. Gokkaya, M. Harman, M. Lyubarskiy, S. Sengupta, S. Yoo, and J. M. Zhang, "Large language models for software engineering: Survey and open problems," *arXiv preprint arXiv:2310.03533*, 2023.

[20] Y. Zhang, Y. Li, L. Cui, D. Cai, L. Liu, T. Fu, X. Huang, E. Zhao, Y. Zhang, Y. Chen *et al.*, "Siren's song in the ai ocean: A survey on hallucination in large language models," *arXiv preprint arXiv:2309.01219*, 2023.

[21] J. Wu, W. Gan, Z. Chen, S. Wan, and P. S. Yu, "Multimodal large language models: A survey," *arXiv preprint arXiv:2311.13165*, 2023.

[22] F. Zeng, W. Gan, Y. Wang, N. Liu, and P. S. Yu, "Large language models for robotics: A survey," *arXiv preprint arXiv:2311.07226*, 2023.

[23] Y. Zhu, H. Yuan, S. Wang, J. Liu, W. Liu, C. Deng, Z. Dou, and J.-R. Wen, "Large language models for information retrieval: A survey," *arXiv preprint arXiv:2308.07107*, 2023.

[24] A. M. Turing, *Computing machinery and intelligence*. Springer, 2009.

[25] C. E. Shannon, "Prediction and entropy of printed english," *Bell system technical journal*, vol. 30, no. 1, pp. 50–64, 1951.

[26] J. Weizenbaum, "Eliza—a computer program for the study of natural language communication between man and machine," *Communications of the ACM*, vol. 9, no. 1, pp. 36–45, 1966.

[27] T. Winograd, "Procedures as a representation for data in a computer program for understanding natural language," MASSACHUSETTS INST OF TECH CAMBRIDGE PROJECT MAC, Tech. Rep., 1971.

[28] G. Sampson, "A stochastic approach to parsing," in *Coling 1986 Volume 1: The 11th International Conference on Computational Linguistics*, 1986.

[29] P. F. Brown, J. Cocke, S. A. Della Pietra, V. J. Della Pietra, F. Jelinek, J. Lafferty, R. L. Mercer, and P. S. Roossin, "A statistical approach to machine translation," *Computational linguistics*, vol. 16, no. 2, pp. 79–85, 1990.

[30] R. Rosenfeld, "Two decades of statistical language modeling: Where do we go from here?" *Proceedings of the IEEE*, vol. 88, no. 8, pp. 1270–1278, 2000.

[31] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[32] Y. Bengio, R. Ducharme, and P. Vincent, "A neural probabilistic language model," *Advances in neural information processing systems*, vol. 13, 2000.

[33] Y. Idelbayev and M. Á. Carreira-Perpiñán, "Lc: A flexible, extensible open-source toolkit for model compression," in *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, 2021, pp. 4504–4514.

[34] C. Buciluǎ, R. Caruana, and A. Niculescu-Mizil, "Model compression," in *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2006, pp. 535–541.

[35] Y. Cheng, D. Wang, P. Zhou, and T. Zhang, "Model compression and acceleration for deep neural networks: The principles, progress, and challenges," *IEEE Signal Processing Magazine*, vol. 35, no. 1, pp. 126–136, 2018.

[36] J. Wu, C. Leng, Y. Wang, Q. Hu, and J. Cheng, "Quantized convolutional neural networks for mobile devices," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 4820–4828.

[37] M. Courbariaux, Y. Bengio, and J.-P. David, "Binaryconnect: Training deep neural networks with binary weights during propagations," *Advances in neural information processing systems*, vol. 28, 2015.

[38] V. Sindhwani, T. Sainath, and S. Kumar, "Structured transforms for small-footprint deep learning," *Advances in Neural Information Processing Systems*, vol. 28, 2015.

[39] S. Han, J. Pool, J. Tran, and W. Dally, "Learning both weights and connections for efficient neural network," *Advances in neural information processing systems*, vol. 28, 2015.

[40] Y. Wang, C. Xu, C. Xu, and D. Tao, "Packing convolutional neural networks in the frequency domain," *IEEE transactions on pattern analysis and machine intelligence*, vol. 41, no. 10, pp. 2495–2510, 2018.

[41] X. Yu, T. Liu, X. Wang, and D. Tao, "On compressing deep models by low rank and sparse decomposition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 7370–7379.

[42] E. L. Denton, W. Zaremba, J. Bruna, Y. LeCun, and R. Fergus, "Exploiting linear structure within convolutional networks for efficient evaluation," *Advances in neural information processing systems*, vol. 27, 2014.

[43] Y. Cheng, D. Wang, P. Zhou, and T. Zhang, "A survey of model compression and acceleration for deep neural networks," *arXiv preprint arXiv:1710.09282*, 2017.

[44] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," *arXiv preprint arXiv:1503.02531*, 2015.

[45] R. High, "The era of cognitive systems: An inside look at ibm watson and how it works," *IBM Corporation, Redbooks*, vol. 1, p. 16, 2012.

[46] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013.

[47] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," *Advances in neural information processing systems*, vol. 27, 2014.

[48] J. Pennington, R. Socher, and C. Manning, "GloVe: Global vectors for word representation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Doha, Qatar: Association for Computational Linguistics, Oct. 2014, pp. 1532–1543. [Online]. Available: https://aclanthology.org/D14-1162

[49] A. Romero, N. Ballas, S. E. Kahou, A. Chassang, C. Gatta, and Y. Bengio, "Fitnets: Hints for thin deep nets," *arXiv preprint arXiv:1412.6550*, 2014.

[50] S. Zagoruyko and N. Komodakis, "Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer," *arXiv preprint arXiv:1612.03928*, 2016.

[51] P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang, "Squad: 100,000+ questions for machine comprehension of text," *arXiv preprint arXiv:1606.05250*, 2016.

[52] 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.

[53] J. Yim, D. Joo, J. Bae, and J. Kim, "A gift from knowledge distillation: Fast optimization, network minimization and transfer learning," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4133–4141.

[54] J. Kim, S. Park, and N. Kwak, "Paraphrasing complex network: Network compression via factor transfer," *Advances in neural information processing systems*, vol. 31, 2018.

[55] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, "Deep contextualized word representations," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. New Orleans, Louisiana: Association for Computational Linguistics, Jun. 2018, pp. 2227–2237. [Online]. Available: https://aclanthology.org/N18-1202

[56] D. Cer, Y. Yang, S.-y. Kong, N. Hua, N. Limtiaco, R. S. John, N. Constant, M. Guajardo-Cespedes, S. Yuan, C. Tar *et al.*, "Universal sentence encoder," *arXiv preprint arXiv:1803.11175*, 2018.

[57] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, and S. R. Bowman, "Glue: A multi-task benchmark and analysis platform for natural language understanding," *arXiv preprint arXiv:1804.07461*, 2018.

[58] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[59] A. Radford, K. Narasimhan, T. Salimans, I. Sutskever *et al.*, "Improving language understanding by generative pre-training," 2018.

[60] I. Solaiman, M. Brundage, J. Clark, A. Askell, A. Herbert-Voss, J. Wu, A. Radford, G. Krueger, J. W. Kim, S. Kreps *et al.*, "Release strategies and the social impacts of language models," *arXiv preprint arXiv:1908.09203*, 2019.

[61] 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," *The Journal of Machine Learning Research*, vol. 21, no. 1, pp. 5485–5551, 2020.

[62] B. Heo, J. Kim, S. Yun, H. Park, N. Kwak, and J. Y. Choi, "A comprehensive overhaul of feature distillation," in *Proceedings of the*

IEEE/CVF International Conference on Computer Vision, 2019, pp. 1921–1930.

[63] Y. Tian, D. Krishnan, and P. Isola, "Contrastive representation distillation," *arXiv preprint arXiv:1910.10699*, 2019.

[64] L. Yuan, F. E. Tay, G. Li, T. Wang, and J. Feng, "Revisit knowledge distillation: a teacher-free framework," 2019.

[65] Y. Zhang, T. Xiang, T. M. Hospedales, and H. Lu, "Deep mutual learning," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 4320–4328.

[66] S. I. Mirzadeh, M. Farajtabar, A. Li, N. Levine, A. Matsukawa, and H. Ghasemzadeh, "Improved knowledge distillation via teacher assistant," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 34, no. 04, 2020, pp. 5191–5198.

[67] M. Zhai, L. Chen, F. Tung, J. He, M. Nawhal, and G. Mori, "Lifelong gan: Continual learning for conditional image generation," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 2759–2768.

[68] N. Papernot, P. McDaniel, X. Wu, S. Jha, and A. Swami, "Distillation as a defense to adversarial perturbations against deep neural networks," in *2016 IEEE symposium on security and privacy (SP)*. IEEE, 2016, pp. 582–597.

[69] H. Lee, S. J. Hwang, and J. Shin, "Self-supervised label augmentation via input transformations," in *International Conference on Machine Learning*. PMLR, 2020, pp. 5714–5724.

[70] M. A. Gordon and K. Duh, "Explaining sequence-level knowledge distillation as data-augmentation for neural machine translation," *arXiv preprint arXiv:1912.03334*, 2019.

[71] J. Wang, W. Bao, L. Sun, X. Zhu, B. Cao, and S. Y. Philip, "Private model compression via knowledge distillation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 01, 2019, pp. 1190–1197.

[72] T. Wang, J.-Y. Zhu, A. Torralba, and A. A. Efros, "Dataset distillation," *arXiv preprint arXiv:1811.10959*, 2018.

[73] O. Bohdal, Y. Yang, and T. Hospedales, "Flexible dataset distillation: Learn labels instead of images," *arXiv preprint arXiv:2006.08572*, 2020.

[74] S. Niu, Y. Liu, J. Wang, and H. Song, "A decade survey of transfer learning (2010–2020)," *IEEE Transactions on Artificial Intelligence*, vol. 1, no. 2, pp. 151–166, 2020.

[75] D. Lepikhin, H. Lee, Y. Xu, D. Chen, O. Firat, Y. Huang, M. Krikun, N. Shazeer, and Z. Chen, "Gshard: Scaling giant models with conditional computation and automatic sharding," *arXiv preprint arXiv:2006.16668*, 2020.

[76] N. Du, Y. Huang, A. M. Dai, S. Tong, D. Lepikhin, Y. Xu, M. Krikun, Y. Zhou, A. W. Yu, O. Firat *et al.*, "Glam: Efficient scaling of language models with mixture-of-experts," in *International Conference on Machine Learning*. PMLR, 2022, pp. 5547–5569.

[77] P. Howard, J. Wang, V. Lal, G. Singer, Y. Choi, and S. Swayamdipta, "Neurocomparatives: Neuro-symbolic distillation of comparative knowledge," *arXiv preprint arXiv:2305.04978*, 2023.

[78] J. Kaplan, S. McCandlish, T. Henighan, T. B. Brown, B. Chess, R. Child, S. Gray, A. Radford, J. Wu, and D. Amodei, "Scaling laws for neural language models," *arXiv preprint arXiv:2001.08361*, 2020.

[79] J. Hoffmann, S. Borgeaud, A. Mensch, E. Buchatskaya, T. Cai, E. Rutherford, D. d. L. Casas, L. A. Hendricks, J. Welbl, A. Clark *et al.*, "Training compute-optimal large language models," *arXiv preprint arXiv:2203.15556*, 2022.

[80] P. Villalobos, J. Sevilla, L. Heim, T. Besiroglu, M. Hobbhahn, and A. Ho, "Will we run out of data? an analysis of the limits of scaling datasets in machine learning," *arXiv preprint arXiv:2211.04325*, 2022.

[81] S. Wang, Y. Liu, Y. Xu, C. Zhu, and M. Zeng, "Want to reduce labeling cost? gpt-3 can help," *arXiv preprint arXiv:2108.13487*, 2021.

[82] R. Smith, J. A. Fries, B. Hancock, and S. H. Bach, "Language models in the loop: Incorporating prompting into weak supervision," *arXiv preprint arXiv:2205.02318*, 2022.

[83] S. Chaudhary, "Code alpaca: An instruction-following llama model for code generation," *Code alpaca: An instruction-following llama model for code generation*, 2023.

[84] 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.

[85] N. Ding, Y. Chen, B. Xu, Y. Qin, Z. Zheng, S. Hu, Z. Liu, M. Sun, and B. Zhou, "Enhancing chat language models by scaling high-quality instructional conversations," *arXiv preprint arXiv:2305.14233*, 2023.

[86] Y. Jiang, C. Chan, M. Chen, and W. Wang, "Lion: Adversarial distillation of closed-source large language model," *arXiv preprint arXiv:2305.12870*, 2023.

[87] W. Yuan, R. Y. Pang, K. Cho, S. Sukhbaatar, J. Xu, and J. Weston, "Self-rewarding language models," *arXiv preprint arXiv:2401.10020*, 2024.

[88] L. Xue, N. Constant, A. Roberts, M. Kale, R. Al-Rfou, A. Siddhant, A. Barua, and C. Raffel, "mt5: A massively multilingual pre-trained text-to-text transformer," *arXiv preprint arXiv:2010.11934*, 2020.

[89] J. Wei, M. Bosma, V. Y. Zhao, K. Guu, A. W. Yu, B. Lester, N. Du, A. M. Dai, and Q. V. Le, "Finetuned language models are zero-shot learners," *arXiv preprint arXiv:2109.01652*, 2021.

[90] R. Thoppilan, D. De Freitas, J. Hall, N. Shazeer, A. Kulshreshtha, H.-T. Cheng, A. Jin, T. Bos, L. Baker, Y. Du *et al.*, "Lamda: Language models for dialog applications," *arXiv preprint arXiv:2201.08239*, 2022.

[91] A. Lewkowycz, A. Andreassen, D. Dohan, E. Dyer, H. Michalewski, V. Ramasesh, A. Slone, C. Anil, I. Schlag, T. Gutman-Solo *et al.*, "Solving quantitative reasoning problems with language models," *Advances in Neural Information Processing Systems*, vol. 35, pp. 3843–3857, 2022.

[92] Y. Tay, M. Dehghani, V. Q. Tran, X. Garcia, J. Wei, X. Wang, H. W. Chung, D. Bahri, T. Schuster, S. Zheng *et al.*, "Ul2: Unifying language learning paradigms," in *The Eleventh International Conference on Learning Representations*, 2022.

[93] A. Chowdhery, S. Narang, J. Devlin, M. Bosma, G. Mishra, A. Roberts, P. Barham, H. W. Chung, C. Sutton, S. Gehrmann *et al.*, "Palm: Scaling language modeling with pathways," *Journal of Machine Learning Research*, vol. 24, no. 240, pp. 1–113, 2023.

[94] H. W. Chung, L. Hou, S. Longpre, B. Zoph, Y. Tay, W. Fedus, Y. Li, X. Wang, M. Dehghani, S. Brahma *et al.*, "Scaling instruction-finetuned language models," *arXiv preprint arXiv:2210.11416*, 2022.

[95] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever *et al.*, "Language models are unsupervised multitask learners," *OpenAI blog*, vol. 1, no. 8, p. 9, 2019.

[96] 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.

[97] M. Chen, J. Tworek, H. Jun, Q. Yuan, H. P. d. O. Pinto, J. Kaplan, H. Edwards, Y. Burda, N. Joseph, G. Brockman *et al.*, "Evaluating large language models trained on code," *arXiv preprint arXiv:2107.03374*, 2021.

[98] R. Nakano, J. Hilton, S. Balaji, J. Wu, L. Ouyang, C. Kim, C. Hesse, S. Jain, V. Kosaraju, W. Saunders *et al.*, "Webgpt: Browser-assisted question-answering with human feedback," *arXiv preprint arXiv:2112.09332*, 2021.

[99] 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. 27 730–27 744, 2022.

[100] R. OpenAI, "Gpt-4 technical report. arxiv 2303.08774," *View in Article*, vol. 2, p. 13, 2023.

[101] B. Wang and A. Komatsuzaki, "Gpt-j-6b: A 6 billion parameter autoregressive language model," 2021.

[102] S. Black, L. Gao, P. Wang, C. Leahy, and S. Biderman, "GPT-Neo: Large Scale Autoregressive Language Modeling with Mesh-Tensorflow," Mar. 2021, If you use this software, please cite it using these metadata. [Online]. Available: https://doi.org/10.5281/zenodo.5297715

[103] S. Black, S. Biderman, E. Hallahan, Q. Anthony, L. Gao, L. Golding, H. He, C. Leahy, K. McDonell, J. Phang *et al.*, "Gpt-neox-20b: An open-source autoregressive language model," *arXiv preprint arXiv:2204.06745*, 2022.

[104] J. W. Rae, S. Borgeaud, T. Cai, K. Millican, J. Hoffmann, F. Song, J. Aslanides, S. Henderson, R. Ring, S. Young *et al.*, "Scaling language models: Methods, analysis & insights from training gopher," *arXiv preprint arXiv:2112.11446*, 2021.

[105] Y. Li, D. Choi, J. Chung, N. Kushman, J. Schrittwieser, R. Leblond, T. Eccles, J. Keeling, F. Gimeno, A. Dal Lago *et al.*, "Competition-level code generation with alphacode," *Science*, vol. 378, no. 6624, pp. 1092–1097, 2022.

[106] A. Glaese, N. McAleese, M. Trębacz, 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.

[107] 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," *arXiv preprint arXiv:2211.09085*, 2022.

[108] S. Zhang, S. Roller, N. Goyal, M. Artetxe, M. Chen, S. Chen, C. Dewan, M. Diab, X. Li, X. V. Lin *et al.*, "Opt: Open pre-trained transformer language models," *arXiv preprint arXiv:2205.01068*, 2022.

[109] S. Iyer, X. V. Lin, R. Pasunuru, T. Mihaylov, D. Simig, P. Yu, K. Shuster, T. Wang, Q. Liu, P. S. Koura *et al.*, "Opt-iml: Scaling language model instruction meta learning through the lens of generalization," *arXiv preprint arXiv:2212.12017*, 2022.

[110] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar *et al.*, "Llama: Open and efficient foundation language models," *arXiv preprint arXiv:2302.13971*, 2023.

[111] V. Sanh, A. Webson, C. Raffel, S. H. Bach, L. Sutawika, Z. Alyafeai, A. Chaffin, A. Stiegler, T. L. Scao, A. Raja, M. Dey, M. S. Bari, C. Xu, U. Thakker, S. Sharma, E. Szczechla, T. Kim, G. Chhablani, N. V. Nayak, D. Datta, J. Chang, M. T. Jiang, H. Wang, M. Manica, S. Shen, Z. X. Yong, H. Pandey, R. Bawden, T. Wang, T. Neeraj, J. Rozen, A. Sharma, A. Santilli, T. Févry, J. A. Fries, R. Teehan, S. Biderman, L. Gao, T. Bers, T. Wolf, and A. M. Rush, "Multitask prompted training enables zero-shot task generalization," *CoRR*, vol. abs/2110.08207, 2021. [Online]. Available: https://arxiv.org/abs/2110.08207

[112] B. Workshop, T. L. Scao, A. Fan, C. Akiki, E. Pavlick, S. Ilić, D. Hesslow, R. Castagné, A. S. Luccioni, F. Yvon *et al.*, "Bloom: A 176b-parameter open-access multilingual language model," *arXiv preprint arXiv:2211.05100*, 2022.

[113] N. Muennighoff, T. Wang, L. Sutawika, A. Roberts, S. Biderman, T. L. Scao, M. S. Bari, S. Shen, Z.-X. Yong, H. Schoelkopf *et al.*, "Crosslingual generalization through multitask finetuning," *arXiv preprint arXiv:2211.01786*, 2022.

[114] Y. Sun, S. Wang, Y. Li, S. Feng, H. Tian, H. Wu, and H. Wang, "ERNIE 2.0: A continual pre-training framework for language understanding," *CoRR*, vol. abs/1907.12412, 2019. [Online]. Available: http://arxiv.org/abs/1907.12412

[115] Y. Sun, S. Wang, S. Feng, S. Ding, C. Pang, J. Shang, J. Liu, X. Chen, Y. Zhao, Y. Lu *et al.*, "Ernie 3.0: Large-scale knowledge enhanced pre-training for language understanding and generation," *arXiv preprint arXiv:2107.02137*, 2021.

[116] S. Wang, Y. Sun, Y. Xiang, Z. Wu, S. Ding, W. Gong, S. Feng, J. Shang, Y. Zhao, C. Pang *et al.*, "Ernie 3.0 titan: Exploring larger-scale knowledge enhanced pre-training for language understanding and generation," *arXiv preprint arXiv:2112.12731*, 2021.

[117] 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.

[118] G. Chen, W. Choi, X. Yu, T. Han, and M. Chandraker, "Learning efficient object detection models with knowledge distillation," *Advances in neural information processing systems*, vol. 30, 2017.

[119] F. Zhang, X. Zhu, and M. Ye, "Fast human pose estimation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 3517–3526.

[120] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.

[121] D. Chen, J.-P. Mei, Y. Zhang, C. Wang, Z. Wang, Y. Feng, and C. Chen, "Cross-layer distillation with semantic calibration," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 8, 2021, pp. 7028–7036.

[122] Z. Huang and N. Wang, "Like what you like: Knowledge distill via neuron selectivity transfer," *arXiv preprint arXiv:1707.01219*, 2017.

[123] N. Passalis and A. Tefas, "Learning deep representations with probabilistic knowledge transfer," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 268–284.

[124] X. Jin, B. Peng, Y. Wu, Y. Liu, J. Liu, D. Liang, J. Yan, and X. Hu, "Knowledge distillation via route constrained optimization," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 1345–1354.

[125] B. Heo, M. Lee, S. Yun, and J. Y. Choi, "Knowledge transfer via distillation of activation boundaries formed by hidden neurons," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 01, 2019, pp. 3779–3787.

[126] S. H. Lee, D. H. Kim, and B. C. Song, "Self-supervised knowledge distillation using singular value decomposition," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 335–350.

[127] C. Zhang and Y. Peng, "Better and faster: knowledge transfer from multiple self-supervised learning tasks via graph distillation for video classification," *arXiv preprint arXiv:1804.10069*, 2018.

[128] S. Lee and B. C. Song, "Graph-based knowledge distillation by multi-head attention network," *arXiv preprint arXiv:1907.02226*, 2019.

[129] N. Passalis, M. Tzelepi, and A. Tefas, "Heterogeneous knowledge distillation using information flow modeling," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 2339–2348.

[130] G. Montavon, A. Binder, S. Lapuschkin, W. Samek, and K.-R. Müller, "Layer-wise relevance propagation: an overview," *Explainable AI: interpreting, explaining and visualizing deep learning*, pp. 193–209, 2019.

[131] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," *Advances in neural information processing systems*, vol. 30, 2017.

[132] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, and L. Zettlemoyer, "Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension," *arXiv preprint arXiv:1910.13461*, 2019.

[133] T. L. Scao, A. Fan, C. Akiki, E. Pavlick, S. Ilić, D. Hesslow, R. Castagné, A. S. Luccioni, F. Yvon, M. Gallé *et al.*, "Bloom: A 176b-parameter open-access multilingual language model," *arXiv preprint arXiv:2211.05100*, 2022.

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

[135] Y. Tay, J. Wei, H. W. Chung, V. Q. Tran, D. R. So, S. Shakeri, X. Garcia, H. S. Zheng, J. Rao, A. Chowdhery *et al.*, "Transcending scaling laws with 0.1% extra compute," *arXiv preprint arXiv:2210.11399*, 2022.

[136] D. Hernandez, T. Brown, T. Conerly, N. DasSarma, D. Drain, S. El-Showk, N. Elhage, Z. Hatfield-Dodds, T. Henighan, T. Hume *et al.*, "Scaling laws and interpretability of learning from repeated data," *arXiv preprint arXiv:2205.10487*, 2022.

[137] Z. Kenton, T. Everitt, L. Weidinger, I. Gabriel, V. Mikulik, and G. Irving, "Alignment of language agents," *arXiv preprint arXiv:2103.14659*, 2021.

[138] S. Polu, J. M. Han, K. Zheng, M. Baksys, I. Babuschkin, and I. Sutskever, "Formal mathematics statement curriculum learning," *arXiv preprint arXiv:2202.01344*, 2022.

[139] M. Sung, J. Lee, S. Yi, M. Jeon, S. Kim, and J. Kang, "Can language models be biomedical knowledge bases?" *arXiv preprint arXiv:2109.07154*, 2021.

[140] L. Z. Liu, Y. Wang, J. Kasai, H. Hajishirzi, and N. A. Smith, "Probing across time: What does roberta know and when?" *arXiv preprint arXiv:2104.07885*, 2021.

[141] Y. Elazar, N. Kassner, S. Ravfogel, A. Ravichander, E. Hovy, H. Schütze, and Y. Goldberg, "Measuring and improving consistency in pretrained language models," *Transactions of the Association for Computational Linguistics*, vol. 9, pp. 1012–1031, 2021.

[142] N. Kassner and H. Schütze, "Negated and misprimed probes for pretrained language models: Birds can talk, but cannot fly," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics, Jul. 2020, pp. 7811–7818. [Online]. Available: https://aclanthology.org/2020.acl-main.698

[143] C. Zhu, A. S. Rawat, M. Zaheer, S. Bhojanapalli, D. Li, F. Yu, and S. Kumar, "Modifying memories in transformer models," *arXiv preprint arXiv:2012.00363*, 2020.

[144] D. Dai, L. Dong, Y. Hao, Z. Sui, B. Chang, and F. Wei, "Knowledge neurons in pretrained transformers," *arXiv preprint arXiv:2104.08696*, 2021.

[145] N. De Cao, W. Aziz, and I. Titov, "Editing factual knowledge in language models," *arXiv preprint arXiv:2104.08164*, 2021.

[146] P. Hase, M. Diab, A. Celikyilmaz, X. Li, Z. Kozareva, V. Stoyanov, M. Bansal, and S. Iyer, "Do language models have beliefs? methods for detecting, updating, and visualizing model beliefs," *arXiv preprint arXiv:2111.13654*, 2021.

[147] E. Mitchell, C. Lin, A. Bosselut, C. Finn, and C. D. Manning, "Fast model editing at scale," *arXiv preprint arXiv:2110.11309*, 2021.

[148] J. Jang, S. Ye, S. Yang, J. Shin, J. Han, G. Kim, S. J. Choi, and M. Seo, "Towards continual knowledge learning of language models," *arXiv preprint arXiv:2110.03215*, 2021.

[149] N. Kassner, B. Krojer, and H. Schütze, "Are pretrained language models symbolic reasoners over knowledge?" *arXiv preprint arXiv:2006.10413*, 2020.

[150] P. Clark, O. Tafjord, and K. Richardson, "Transformers as soft reasoners over language," *arXiv preprint arXiv:2002.05867*, 2020.

[151] S. Saha, S. Ghosh, S. Srivastava, and M. Bansal, "Prover: Proof generation for interpretable reasoning over rules," *arXiv preprint arXiv:2010.02830*, 2020.

[152] O. Tafjord, B. D. Mishra, and P. Clark, "Proofwriter: Generating implications, proofs, and abductive statements over natural language," *arXiv preprint arXiv:2012.13048*, 2020.

[153] N. Gontier, K. Sinha, S. Reddy, and C. Pal, "Measuring systematic generalization in neural proof generation with transformers," *Advances in Neural Information Processing Systems*, vol. 33, pp. 22 231–22 242, 2020.

[154] K. Richardson and A. Sabharwal, "Pushing the limits of rule reasoning in transformers through natural language satisfiability," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 10, 2022, pp. 11 209–11 219.

[155] M. Saeed, N. Ahmadi, P. Nakov, and P. Papotti, "Rulebert: Teaching soft rules to pre-trained language models," *arXiv preprint arXiv:2109.13006*, 2021.

[156] D. Saxton, E. Grefenstette, F. Hill, and P. Kohli, "Analysing mathematical reasoning abilities of neural models," *arXiv preprint arXiv:1904.01557*, 2019.

[157] P. Banerjee, S. Mishra, K. K. Pal, A. Mitra, and C. Baral, "Commonsense reasoning with implicit knowledge in natural language," in *3rd Conference on Automated Knowledge Base Construction*, 2021.

[158] P. Zhou, R. Khanna, S. Lee, B. Y. Lin, D. Ho, J. Pujara, and X. Ren, "Rica: Evaluating robust inference capabilities based on commonsense axioms," *arXiv preprint arXiv:2005.00782*, 2020.

[159] N. F. Rajani, B. McCann, C. Xiong, and R. Socher, "Explain yourself! leveraging language models for commonsense reasoning," *arXiv preprint arXiv:1906.02361*, 2019.

[160] S. Jain and B. C. Wallace, "Attention is not explanation," *arXiv preprint arXiv:1902.10186*, 2019.

[161] S. Wiegreffe and Y. Pinter, "Attention is not not explanation," *arXiv preprint arXiv:1908.04626*, 2019.

[162] S. Serrano and N. A. Smith, "Is attention interpretable?" *arXiv preprint arXiv:1906.03731*, 2019.

[163] S. Zhao, D. Pascual, G. Brunner, and R. Wattenhofer, "Of non-linearity and commutativity in bert," in *2021 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2021, pp. 1–8.

[164] M. Geva, R. Schuster, J. Berant, and O. Levy, "Transformer feed-forward layers are key-value memories," *arXiv preprint arXiv:2012.14913*, 2020.

[165] K. Meng, D. Bau, A. Andonian, and Y. Belinkov, "Locating and editing factual associations in gpt," *Advances in Neural Information Processing Systems*, vol. 35, pp. 17 359–17 372, 2022.

[166] X. Han, B. C. Wallace, and Y. Tsvetkov, "Explaining black box predictions and unveiling data artifacts through influence functions," *arXiv preprint arXiv:2005.06676*, 2020.

[167] P. Pezeshkpour, S. Jain, B. C. Wallace, and S. Singh, "An empirical comparison of instance attribution methods for nlp," *arXiv preprint arXiv:2104.04128*, 2021.

[168] P. Pezeshkpour, S. Jain, S. Singh, and B. C. Wallace, "Combining feature and instance attribution to detect artifacts," *arXiv preprint arXiv:2107.00323*, 2021.

[169] H. Zylberajch, P. Lertvittayakumjorn, and F. Toni, "Hildif: Interactive debugging of nli models using influence functions," in *Proceedings of the First Workshop on Interactive Learning for Natural Language Processing*, 2021, pp. 1–6.

[170] S. Narang, C. Raffel, K. Lee, A. Roberts, N. Fiedel, and K. Malkan, "Wt5?! training text-to-text models to explain their predictions," *arXiv preprint arXiv:2004.14546*, 2020.

[171] O.-M. Camburu, B. Shillingford, P. Minervini, T. Lukasiewicz, and P. Blunsom, "Make up your mind! adversarial generation of inconsistent natural language explanations," *arXiv preprint arXiv:1910.03065*, 2019.

[172] F. Friedrich, P. Schramowski, C. Tauchmann, and K. Kersting, "Interactively providing explanations for transformer language models," *arXiv preprint arXiv:2110.02058*, 2021.

[173] T. Lei, R. Barzilay, and T. Jaakkola, "Rationalizing neural predictions," *arXiv preprint arXiv:1606.04155*, 2016.

[174] B. Paranjape, J. Michael, M. Ghazvininejad, L. Zettlemoyer, and H. Hajishirzi, "Prompting contrastive explanations for commonsense reasoning tasks," *arXiv preprint arXiv:2106.06823*, 2021.

[175] M. Sap, R. Le Bras, E. Allaway, C. Bhagavatula, N. Lourie, H. Rashkin, B. Roof, N. A. Smith, and Y. Choi, "Atomic: An atlas of machine commonsense for if-then reasoning," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 33, no. 01, 2019, pp. 3027–3035.

[176] P. West, R. L. Bras, T. Sorensen, B. Y. Lin, L. Jiang, X. Lu, K. Chandu, J. Hessel, A. Baheti, C. Bhagavatula *et al.*, "Novacomet: Open commonsense foundation models with symbolic knowledge distillation," *arXiv preprint arXiv:2312.05979*, 2023.

[177] C. Bhagavatula, J. D. Hwang, D. Downey, R. L. Bras, X. Lu, K. Sakaguchi, S. Swayamdipta, P. West, and Y. Choi, "I2d2: Inductive knowledge distillation with neurologic and self-imitation," *arXiv preprint arXiv:2212.09246*, 2022.

[178] Y. Fu, H. Peng, L. Ou, A. Sabharwal, and T. Khot, "Specializing smaller language models towards multi-step reasoning," *arXiv preprint arXiv:2301.12726*, 2023.

[179] J. S. Park, J. Hessel, K. R. Chandu, P. P. Liang, X. Lu, P. West, Y. Yu, Q. Huang, J. Gao, A. Farhadi *et al.*, "Localized symbolic knowledge distillation for visual commonsense models," *arXiv preprint arXiv:2312.04837*, 2023.

[180] 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.

[181] 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.

[182] 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.

[183] 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/

[184] 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/

[185] S. Arora, A. Narayan, M. F. Chen, L. Orr, N. Guha, K. Bhatia, I. Chami, F. Sala, and C. Ré, "Ask me anything: A simple strategy for prompting language models," *arXiv preprint arXiv:2210.02441*, 2022.

[186] P. Agrawal, C. Alberti, F. Huot, J. Maynez, J. Ma, S. Ruder, K. Ganchev, D. Das, and M. Lapata, "Qameleon: Multilingual qa with only 5 examples," *arXiv preprint arXiv:2211.08264*, 2022.

[187] C.-Y. Hsieh, C.-L. Li, C.-K. Yeh, H. Nakhost, Y. Fujii, A. Ratner, R. Krishna, C.-Y. Lee, and T. Pfister, "Distilling step-by-step! outperforming larger language models with less training data and smaller model sizes," *arXiv preprint arXiv:2305.02301*, 2023.

[188] 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.

[189] A. Mitra, L. Del Corro, S. Mahajan, A. Codas, C. Simoes, S. Agrawal, X. Chen, A. Razdaibiedina, E. Jones, K. Aggarwal *et al.*, "Orca 2: Teaching small language models how to reason," *arXiv preprint arXiv:2311.11045*, 2023.

[190] J. Renkhoff, K. Feng, M. Meier-Doernberg, A. Velasquez, and H. H. Song, "A survey on verification and validation, testing and evaluations of neurosymbolic artificial intelligence," *IEEE Transactions on Artificial Intelligence*, 2024.

[191] K. Acharya, W. Raza, C. Dourado, A. Velasquez, and H. H. Song, "Neurosymbolic reinforcement learning and planning: A survey," *IEEE Transactions on Artificial Intelligence*, 2023.

[192] A. Velasquez, "Transfer from imprecise and abstract models to autonomous technologies (tiamat)," *Defense Advanced Research Projects Agency (DARPA) Program Solicitation*, 2023.

[193] Q. Huang, M. Tao, C. Zhang, Z. An, C. Jiang, Z. Chen, Z. Wu, and Y. Feng, "Lawyer llama technical report," *arXiv preprint arXiv:2305.15062*, 2023.

[194] 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.

[195] H. Zhang, J. Chen, F. Jiang, F. Yu, Z. Chen, J. Li, G. Chen, X. Wu, Z. Zhang, Q. Xiao *et al.*, "Huatuogpt, towards taming language model to be a doctor," *arXiv preprint arXiv:2305.15075*, 2023.

[196] 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.

[197] 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, 2023, pp. 4435–4439.

[198] T. Xie, Y. Wan, W. Huang, Z. Yin, Y. Liu, S. Wang, Q. Linghu, C. Kit, C. Grazian, W. Zhang *et al.*, "Darwin series: Domain specific large language models for natural science," *arXiv preprint arXiv:2308.13565*, 2023.

[199] 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," *arXiv preprint arXiv:2401.07950*, 2024.

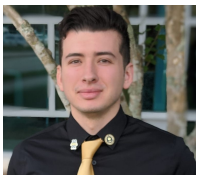
[200] W. Shao, M. Chen, Z. Zhang, P. Xu, L. Zhao, Z. Li, K. Zhang, P. Gao, Y. Qiao, and P. Luo, "Omniquant: Omnidirectionally calibrated quantization for large language models," *arXiv preprint arXiv:2308.13137*, 2023.

[201] Y. Shang, Z. Yuan, Q. Wu, and Z. Dong, "Pb-llm: Partially binarized large language models," *arXiv preprint arXiv:2310.00034*, 2023.

[202] C. Zhou, P. Liu, P. Xu, S. Iyer, J. Sun, Y. Mao, X. Ma, A. Efrat, P. Yu, L. Yu *et al.*, "Lima: Less is more for alignment," *Advances in Neural Information Processing Systems*, vol. 36, 2024.

Kamal Acharya (Graduate Student Member, IEEE) received his Engineering degree in Electronics and Communication Engineering from Tribhuvan University, Kathmandu, Nepal in 2011 and Masters degree in Information System Engineering from Purbanchal University, Kathmandu, Nepal in 2019. Currently, he is pursuing PhD. in the Information Systems from University of Maryland, Baltimore County (UMBC), Baltimore, MD.

He has been involved in teaching profession for about 7 years in the various universities of Nepal, Tribhuvan University and Purbanchal Univesity were among few of them. He is mainly associated with the courses like programming(C,C++,Python), Computer Networks and Computer Architecture. He is working as Graduate Research Assistant in UMBC. He is also serving as an reviewer for IEEE Transactions on Artificial Intelligence, IEEE Transactions on Intelligent Transportation Systems and IEEE SMC Magazine. His preferred areas of research are Natural Language Processing(NLP), Deep Learning and Reinforcement Learning.



Alvaro Velasquez is a program manager in the Innovation Information Office (I2O) of the Defense Advanced Research Projects Agency (DARPA), where he currently leads the Assured Neuro-Symbolic Learning and Reasoning (ANSR) program. Before that, Alvaro oversaw the machine intelligence portfolio of investments for the Information Directorate of the Air Force Research Laboratory (AFRL). Alvaro received his PhD in Computer Science from the University of Central Florida in 2018 and is a recipient of the AAAI Distinguished Paper Award, the National Science Foundation Graduate Research Fellowship Program (NSF GRFP) award, the University of Central Florida 30 Under 30 award, and best paper and patent awards from AFRL. He has co-authored over 80 papers and two patents and serves as Associate Editor of the IEEE Transactions on Artificial Intelligence.

Houbing Herbert Song (M'12–SM'14-F'23) received the Ph.D. degree in electrical engineering from the University of Virginia, Charlottesville, VA, in August 2012.

He is currently a Professor, the Founding Director of the NSF Center for Aviation Big Data Analytics (Planning), the Associate Director for Leadership of the DOT Transportation Cybersecurity Center for Advanced Research and Education (Tier 1 Center), and the Director of the Security and Optimization for Networked Globe Laboratory (SONG Lab, www.SONGLab.us), University of Maryland, Baltimore County (UMBC), Baltimore, MD. He is a Distinguished Visiting Fellow of the Scottish Informatics and Computer Science Alliance (SICSA). Prior to joining UMBC, he was a Tenured Associate Professor of Electrical Engineering and Computer Science at Embry-Riddle Aeronautical University, Daytona Beach, FL. He serves as an Associate Editor for IEEE Transactions on Artificial Intelligence (TAI) (2023-present), IEEE Internet of Things Journal (2020-present), IEEE Transactions on Intelligent Transportation Systems (2021-present), and IEEE Journal on Miniaturization for Air and Space Systems (J-MASS) (2020-present). He was an Associate Technical Editor for IEEE Communications Magazine (2017-2020). He is the editor of ten books, the author of more than 100 articles and the inventor of 2 patents. His research interests include AI/machine learning/big data analytics, cyber-physical systems/internet of things, and cybersecurity and privacy. His research has been sponsored by federal agencies (including National Science Foundation, National Aeronautics and Space Administration, US Department of Transportation, and Federal Aviation Administration, among others) and industry. His research has been featured on popular news media outlets, including IEEE Spectrum, IEEE GlobalSpec's Engineering360, IEEE Transmitter, insideBIGDATA, Association for Uncrewed Vehicle Systems International (AUVSI), Security Magazine, CXOTech Magazine, Fox News, U.S. News & World Report, The Washington Times, and New Atlas.

Dr. Song is an IEEE Fellow, an Asia-Pacific Artificial Intelligence Association (AAIA) Fellow, an ACM Distinguished Member, and a Full Member of Sigma Xi. Dr. Song has been a Highly Cited Researcher identified by Web of Science since 2021. He is an ACM Distinguished Speaker (2020-present), an IEEE Computer Society Distinguished Visitor (2024-present), an IEEE Communications Society (ComSoc) Distinguished Lecturer (2024-present), an IEEE Intelligent Transportation Systems Society (ITSS) Distinguished Lecturer (2024-present), an IEEE Vehicular Technology Society (VTS) Distinguished Lecturer (2023-present) and an IEEE Systems Council Distinguished Lecturer (2023-present). Dr. Song received Research.com Rising Star of Science Award in 2022, 2021 Harry Rowe Mimno Award bestowed by IEEE Aerospace and Electronic Systems Society, and 10+ Best Paper Awards from major international conferences, including IEEE CPSCom-2019, IEEE ICII 2019, IEEE/AIAA ICNS 2019, IEEE CBDCom 2020, WASA 2020, AIAA/IEEE DASC 2021, IEEE GLOBECOM 2021 and IEEE INFOCOM 2022. He has been an IEEE Impact Creator since 2023.