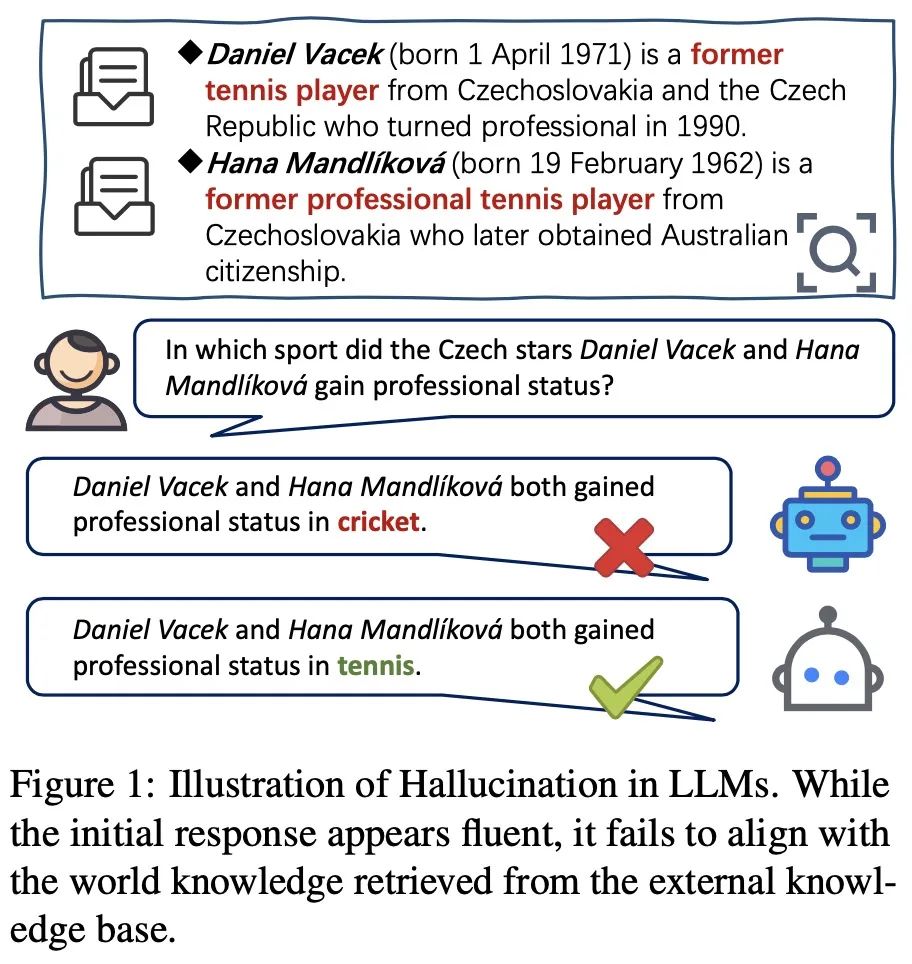
（介绍搜索引擎的部分）

We utilized these search engines, which can be categorized into four main types: academic search engines, relevant technical blogs and trending articles, and related technical communities. This slide lists a series but is not limited to these.

一．

1. 幻觉在大型语言模型中的表现



幻觉问题是大型语言模型（LLMs）面临的一个关键挑战，它指的是模型生成看似合理但事实上错误或无意义信息的现象。这一问题在LLMs广泛应用的背景下引起了安全和伦理方面的担忧，因为这些错误信息可能会误导用户，导致不可预测的后果。

The issue of hallucinations is a key challenge faced by large language models (LLMs), referring to the phenomenon where models generate seemingly plausible but in fact incorrect or meaningless information. This problem has raised concerns regarding safety and ethics in the context of the widespread application of LLMs, as such erroneous information could mislead users, leading to unpredictable outcomes.

1. 幻觉的成因通常归咎于**数据**、**训练**和**推理**阶段的问题，如数据质量差、信息错误、偏见、过时知识、模型架构和策略缺陷、注意力机制问题以及推理过程中的随机性等。

The causes of hallucinations are typically attributed to issues in the data, training, and inference stages, such as poor data quality, misinformation, bias, outdated knowledge, flaws in model architecture and strategies, problems with attention mechanisms, and randomness during the inference process.

数值混淆：当LLM处理与数字有关的文本，如日期或数值时，容易产生幻觉。

处理长文本：在需要解读长期依赖关系的任务中，例如文档摘要或长对话历史，模型可能会生成自相矛盾的内容。

逻辑推断障碍：若模型误解了源文本中的信息，它有可能产生不准确的结论。因此，模型的逻辑推理能力至关重要。

上下文与内置知识的冲突：模型在处理信息时，可能会过度依赖于预训练阶段获取的知识，而忽略实际上下文，导致输出不准确。

1. 根据幻觉的分类我们有内在幻觉，外在幻觉和忠实度幻觉**内在幻觉**发生在LLM输出与提供的输入相矛盾时，而**外在幻觉**则是LLM输出无法通过输入信息进行验证。此外，还有基于用户指令考虑的**忠实度幻觉**，包括指令性、上下文和逻辑不一致。

Based on the classification of hallucinations, we have intrinsic hallucinations, extrinsic hallucinations, and fidelity hallucinations. Intrinsic hallucinations occur when the output of an LLM contradicts the provided input, while extrinsic hallucinations refer to outputs that cannot be verified through input information. Furthermore, there are fidelity hallucinations considered based on user instructions, including directive, contextual, and logical inconsistencies.

（我们都知道以chatgpt3.5为例它的实用性收到了截止日期的阻碍，而New Bing的出现扭转了局面，它是一个一体化的搜索引擎，而经过对它大量的应用可以发现这个搜索引擎并不只是仅仅是机械的关键文本匹配，而是使响应与用户意图和上下文保持一致。而这样的方法也是导致忠实度幻觉的一个重要因素，它为了高度与你想要的资源相匹配往往会为你推荐虚假的理论和文章，在后面会据具体的例子）

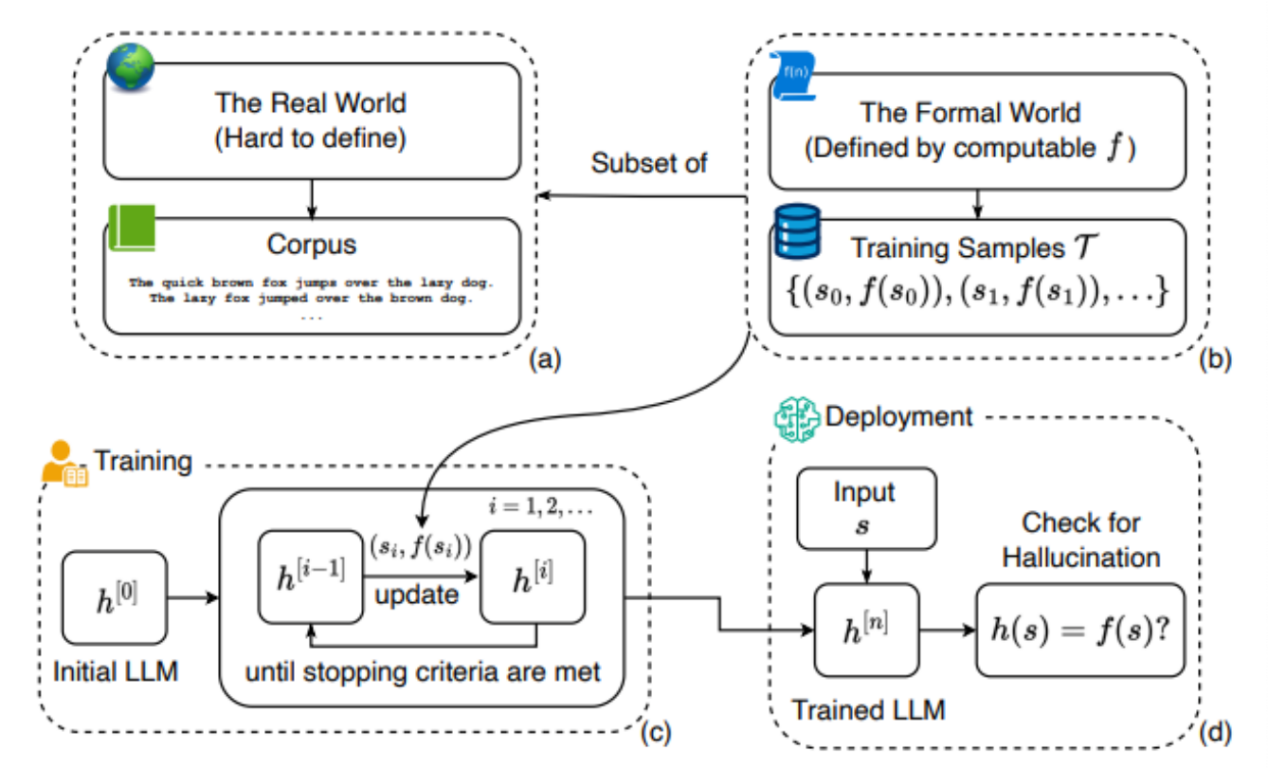
1. 构建形式化世界分析视觉视角

对于幻觉这个词还是比较抽象很难定量分析的词汇，在现实世界中，由于模型的输入与输出复杂，其实很难定义什么是幻觉，我们最常说的就是幻觉是输出不符合事实？那么什么又是事实，什么时候会出现不一致的情况，这都需要根据复杂的输入输出进一步讨论。

我们需要抛开了现实世界中“正确性”的复杂定义，转而在一个形式化的世界，用数学为幻觉下定义，由此我们可以在一个精确的讨论环境下探讨幻觉问题。从而确定幻觉的基准，如下图片描述了从模型、语料、训练部署过程中所有相关概念的形式化定义与关系示意图。

The term "hallucination" is quite abstract and challenging to quantify. In the real world, due to the complexity of models' inputs and outputs, it's tough to define what constitutes a hallucination. The most common understanding is that hallucinations are outputs that do not align with facts. But then, what constitutes a fact? When do inconsistencies arise? These questions require further discussion based on complex inputs and outputs.

We need to move beyond the complex definitions of "correctness" in the real world and instead define hallucinations mathematically in a formalized context. This approach allows us to discuss the issue of hallucinations in a precise environment. To establish a baseline for hallucinations, the following image describes a formal definition and the relationships among all relevant concepts during the model, corpus, training, and deployment processes.



上图中(a)展示了真实世界的语料库，是包含了（b）真实世界中的基本事实函数和训练样本的超集。（c）展示了定义了一个详细过程训练的LLM ，该过程使用训练样本并更新直到达到停止准则。最后，在(d)中，经过训练的LLM被部署，并针对未知字符串生成输出。通过将LLM的答案h(s)与基本事实值f(s)进行比较，之间的差异也就定义了幻觉问题。（用中文讲）

这篇论文也用形式化的方式解释了为什么LLM无法避免幻觉问题只能进行缓解：

LLM无法学习所有可计算的函数，因此总是会产生幻觉。既然幻觉无法被避免，那我们都有什么方法来缓解幻觉的产生，这也就是我们将介绍的第三部分

This paper also explains in a formalized manner why LLMs cannot avoid the issue of hallucinations but can only mitigate it:

LLMs are unable to learn all computable functions, hence they will always produce hallucinations. Given that hallucinations cannot be avoided, what methods do we have to mitigate their occurrence? This will be the focus of the third section we are about to introduce.

而基于幻觉的产生我们可以想到缓解幻觉的两大原则

Based on the occurrence of hallucinations, we can consider two main principles for mitigating them:

所以所有幻觉缓解的方法会基于以下两大原则：

幻觉缓解方法通常基于两个原则：(a) 提高语言模型能力，和 (b) 向语言模型提供更多关于真实世界的信息的真实函数f，通过使用训练样本或归纳偏差。

**而与其说这些方法会缓解大语言模型的幻觉问题，换句话来说他们是来提高大预言模型的能力和安全性**

Hallucination mitigation methods typically are based on two principles: (a) enhancing the capabilities of language models, and (b) providing language models with more accurate functions \(f\) about the real world, through the use of training samples or inductive biases. Rather than saying these methods mitigate hallucinations in large language models, it would be more accurate to say they aim to improve the capabilities and safety of large predictive models.

现在我们从理论上我们会有大致的这几个思路：

1.更大的模型和更多的训练数据

大家以前觉得，越大的语言模型因为有更多的参数和数据，就会有小模型没有的超强能力。所以人们本能地认为，模型做得越大，出错的几率就会越小。

增加模型的参数和数据，理论上这个模型就能处理更难的问题，但前提是对LLM可学习的任务而言。然而，如果**真实函数根本无法被LLM完全捕捉**,增加参数和数据则是徒劳的。

Now, theoretically, we have these general ideas:

Larger models and more training data

Previously, it was believed that larger language models, due to having more parameters and data, would possess capabilities beyond those of smaller models. Thus, people instinctively thought that making models larger would reduce the chances of errors.

Increasing the model's parameters and data, in theory, would enable the model to tackle more challenging problems, but this is only applicable to tasks that are learnable by LLMs. However, if the true function cannot be fully captured by LLMs, then adding more parameters and data would be futile.

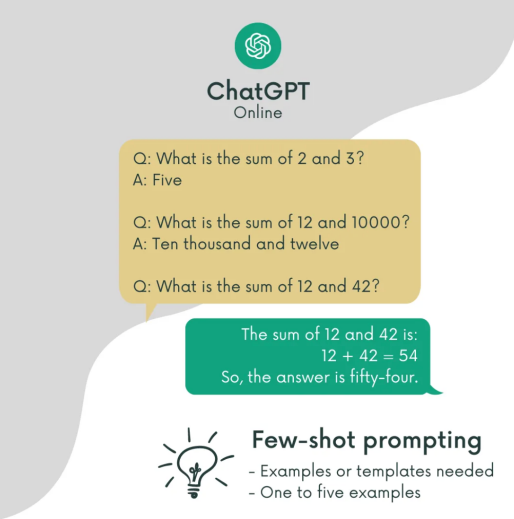
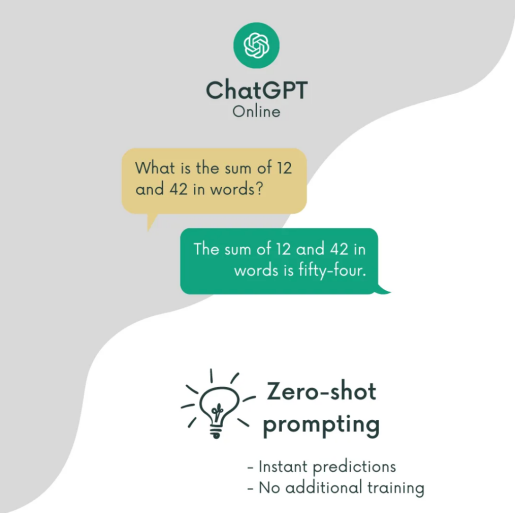
2.以思维链／反思／验证引导LLM

(a)Chain-of-Thought

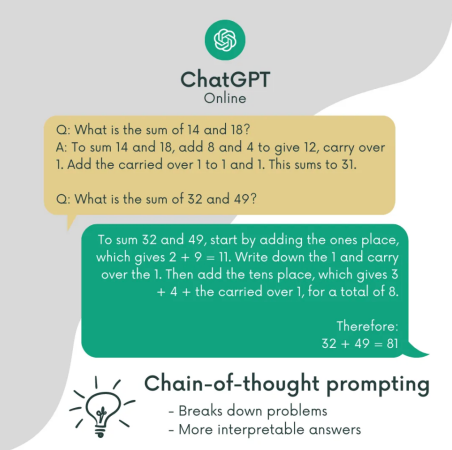
这种方法就是教计算机根据上下文来学习，像是给它看一些正确的例子或者相关知识。目的是让计算机在解决问题时，能像人类一样思考，找到简单有效的方式。比如算斐波那契数列，如果用递归方法，计算量会非常大，但如果用动态规划，就能快很多。给计算机一些提示，引导它学习**人类倾向的简单方法**，这样可以减少计算机产生错误的情况。不过，这个办法并不是万能的，它只对某些特定的问题有效。

This method involves teaching the computer to learn from context, like showing it correct examples or relevant knowledge. The goal is to enable the computer to think like a human when solving problems, finding simple and effective methods. For instance, calculating the Fibonacci sequence could require substantial computational effort if done recursively, but much less if approached through dynamic programming. By providing the computer with hints and guiding it to learn simple methods preferred by humans, we can reduce the likelihood of errors. However, this approach is not a panacea; it is only effective for certain specific problems.

结合这种方法举一个论文中的例子：



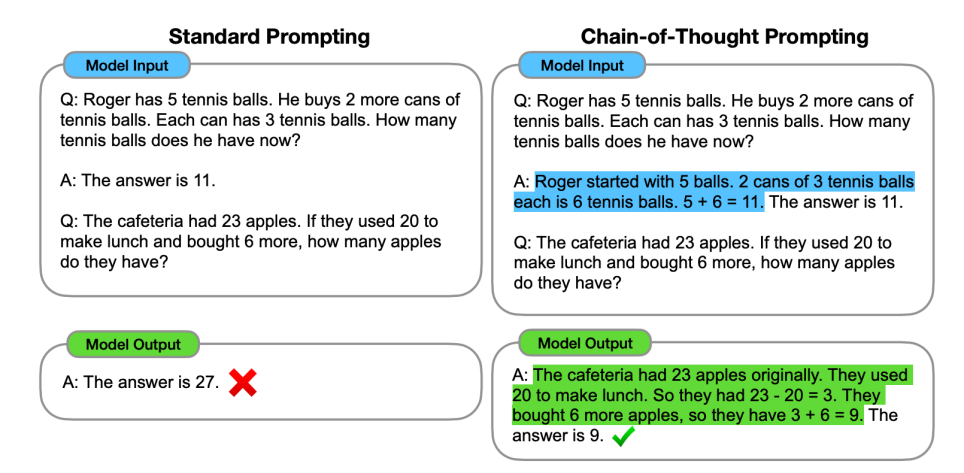
1. 无中间思考过程提示 (b)少量思考过程提示

(c)以思维链的方式进行提示

（b）反思 (Self-Reflection):

在反思策略中，模型被鼓励在生成答案后进行自我评估和批判性思考。模型会对其初步的答案或推理进行反思，探讨可能的错误，并考虑其他可能的解释或答案。这个过程可能涉及到检查答案的合理性、可能的逻辑漏洞，或是评估答案的信心水平。这可以通过训练模型来识别和解决自身推理中的不确定性和矛盾来实现。

In the reflection strategy, the model is encouraged to self-assess and engage in critical thinking after generating an answer. The model reflects on its initial response or reasoning, explores possible errors, and considers alternative explanations or answers. This process may involve examining the plausibility of the answer, potential logical flaws, or assessing the level of confidence in the answer. This can be achieved by training the model to recognize and address uncertainties and contradictions in its own reasoning.



（c)验证 (Verification):

验证策略是关于模型在给出答案之后进行事实检查或证据查找，以确保其响应的准确性。在这种策略中，模型可能会寻找额外的信息来支持其答案，或者重新考虑问题以验证其先前的推理。例如，模型可能会用外部数据或知识（如果可用）来验证其答案，或者在不同上下文中测试其答案的有效性。

对于一个完整的大语言模型生成链这三种策略都旨在通过不同的方式都意在促进模型的深度理解和精确推理。"思维链"侧重于展现解题过程，"反思"侧重于批判性评估和改进推理过程，而"验证"侧重于确保答案的准确性和可靠性。

The verification strategy involves the model performing fact-checking or evidence gathering after providing an answer, to ensure the accuracy of its response. In this strategy, the model may seek additional information to support its answer or reconsider the question to verify its previous reasoning. For example, the model could use external data or knowledge (if available) to validate its answers, or test the validity of its answers in different contexts.

For a comprehensive large language model generation chain, these three strategies aim to promote deep understanding and precise reasoning in different ways. "Thought chains" focus on displaying the problem-solving process, "reflection" emphasizes critical assessment and improvement of the reasoning process, and "verification" concentrates on ensuring the accuracy and reliability of answers.

LLM集成

这个方法是用好几个语言模型来一起解决同一个问题。这些模型互相讨论后，通过投票或者达成一致来决定答案。这样做的想法是，如果多个模型都同意某个答案，那么这个答案很可能是对的。整合多个模型比用单独一个模型更厉害，因为它们可以互相补充，减少出错的机会。但是，把模型合在一起的时候，它们还是像一个模型一样有局限性，也就是说，它们还是可能会出现错误。并且集成为一个模型 can be a time-consuming and challenging task due to their unique architectures, APIs, and compatibility requirements.现在主流的两个集成工具是LLM -client和LangChain。

This method involves using several language models to solve the same problem together. After discussing among themselves, they decide on an answer through voting or consensus. The idea is that if multiple models agree on an answer, then that answer is likely correct. Integrating multiple models can be more powerful than using a single model alone, as they can complement each other and reduce the chance of errors. However, when models are combined, they still have limitations, just as a single model does, meaning they can still make mistakes. Additionally, integrating them into a single model can be a time-consuming and challenging task due to their unique architectures, APIs, and compatibility requirements. The two mainstream integration tools currently are LLM-client and LangChain.

防护栏和篱笆

“防护栏”这个概念是指把一些原则用于语言模型，确保它们的输出符合人类的价值观、道德和法律要求。而“篱笆”则是指那些绝不应该完全由语言模型自动完成的关键任务清单。

这两者都作为安全约束条件，防止语言模型（以及其他人工智能模型）生成不良结果。可以通过正式编程的方式来实现护栏和围栏，从而明确影响语言模型的行为。因此，它潜在地是正式世界和某些现实问题的有用幻觉缓解因素。然而，在现实世界中的可扩展性仍然是一个开放问题。

The concept of "guardrails" refers to applying certain principles to language models to ensure their outputs align with human values, ethics, and legal requirements. "Fences," on the other hand, refer to a list of critical tasks that should never be fully automated by language models.

Both serve as safety constraints to prevent language models (and other artificial intelligence models) from generating harmful outcomes. Guardrails and fences can be implemented through formal programming methods to clearly influence the behavior of language models. Thus, they potentially serve as useful mechanisms for mitigating hallucinations in a formalized world and addressing certain real-world issues. However, their scalability in the real world remains an open question.

知识增强的语言模型

这种方法利用外部知识源（如知识图谱和数据库）和符号推理方法（如逻辑 辅助语言模型。通过改变信息的提取方式（通过来自知识数据库的检索）或通过逻辑推断来明确控制语言模型的工作流程。通过这种方式，语言模型除了通过训练数据外，还可以获得关于真实函数的额外信息，因此可能是形式世界中幻觉的有效缓解方式。

Knowledge-enhanced language models leverage external knowledge sources (such as knowledge graphs and databases) and symbolic reasoning methods (like logic) to assist language models. By altering the way information is extracted (through retrieval from knowledge databases) or by controlling the language model's workflow through logical inference, these models can gain additional information about the real function beyond what is available through their training data. Thus, this method could be an effective way to mitigate hallucinations in a formalized world.

**在这里着重讲一下知识图谱与LLM的结合**

先举一个不成熟的大语言模型的例子，如果向 LLM 提问：「爱因斯坦在什么时候发现了引力？」它可能会说：「爱因斯坦在 1687 年发现了引力。」但事实上，提出引力理论的人是牛顿。这种问题会严重损害 LLM 的可信度。（中文说）

LLM 是黑箱模型，缺乏可解释性，LLM 通过参数隐含地表示知识。因此，我们难以解释和验证 LLM 获得的知识。此外，LLM 是通过概率模型执行推理，而这是一个非决断性的过程。对于 LLM 用以得出预测结果和决策的具体模式和功能，人类难以直接获得详情和解释。尽管通过使用思维链（chain-of-thought），某些 LLM 具备解释自身预测结果的功能，但它们推理出的解释依然存在幻觉问题。这会严重影响 LLM 在事关重大的场景中的应用，比如医疗诊断和法律评判。在医疗诊断场景中，LLM 可能误诊并提供与医疗常识相悖的解释。这就引出了另一个问题：在一般语料库上训练的 LLM 由于缺乏特定领域的知识或新训练数据，可能无法很好地泛化到特定领域或新知识上。

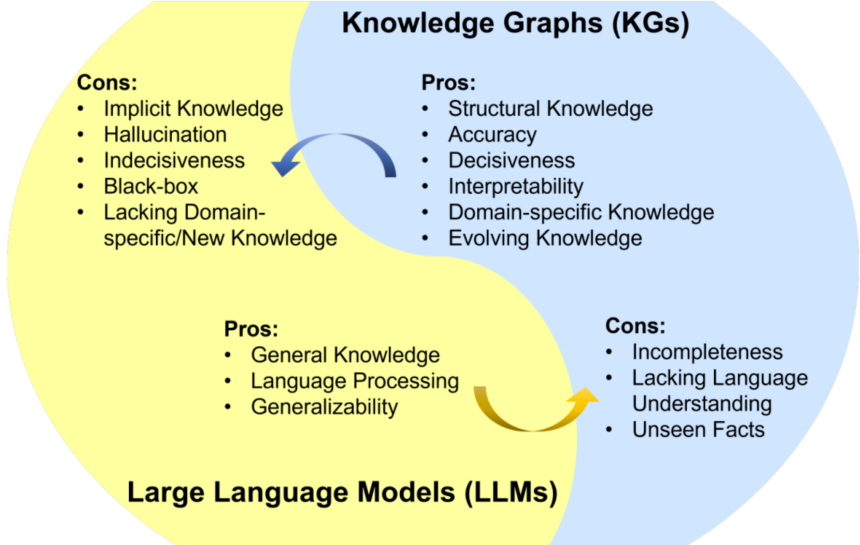
LLMs are black box models lacking interpretability, representing knowledge implicitly through parameters. As a result, it's challenging to explain and verify the knowledge obtained by LLMs. Moreover, LLMs perform reasoning through probabilistic models, which is an indeterminate process. The specific patterns and functions LLMs use to derive predictions and make decisions are difficult for humans to directly access and interpret. Although some LLMs can explain their predictions by using a chain-of-thought process, the explanations they generate can still suffer from hallucinations. This significantly impacts the application of LLMs in critical scenarios, such as medical diagnosis and legal judgments. In medical diagnostic scenarios, LLMs might misdiagnose and provide explanations contrary to medical common sense. This raises another issue: LLMs trained on general corpora may struggle to generalize well to specific domains or new knowledge due to a lack of domain-specific knowledge or new training data.

为了解决上述问题，一个潜在的解决方案是将知识图谱（KG）整合进 LLM 中。知识图谱能以三元组的形式存储巨量事实，即 (头实体、关系、尾实体)，因此知识图谱是一种结构化和决断性的知识表征形式，例如维基百科。知识图谱对多种应用而言都至关重要，因为其能提供准确、明确的知识。此外众所周知，它们还具有很棒的符号推理能力，这能生成可解释的结果。知识图谱还能随着新知识的持续输入而积极演进。此外，通过让专家来构建特定领域的知识图谱，就能具备提供精确可靠的特定领域知识的能力。

然而，知识图谱很难构建，并且由于真实世界知识图谱往往是不完备的，还会动态变化，因此当前的知识图谱方法难以应对。这些方法无法有效建模未见过的实体以及表征新知识。此外，知识图谱中丰富的文本信息往往会被忽视。不仅如此，知识图谱的现有方法往往是针对特定知识图谱或任务定制的，泛化能力不足。因此，有必要使用 LLM 来解决知识图谱面临的挑战。

To address the issues mentioned above, a potential solution is to integrate Knowledge Graphs (KGs) into LLMs. Knowledge Graphs can store vast amounts of facts in the form of triples, namely (head entity, relationship, tail entity), making them a structured and deterministic form of knowledge representation, like Wikipedia. Knowledge Graphs are crucial for various applications as they provide accurate, explicit knowledge. They are also known for their excellent symbolic reasoning capabilities, which can produce interpretable results. Furthermore, Knowledge Graphs can evolve positively with the continuous input of new knowledge. Additionally, by having experts build domain-specific Knowledge Graphs, it's possible to provide precise and reliable domain-specific knowledge.

However, building Knowledge Graphs is challenging, and current methods struggle due to the incompleteness and dynamic nature of real-world Knowledge Graphs. These methods cannot effectively model unseen entities or represent new knowledge. Moreover, the rich textual information in Knowledge Graphs is often overlooked. Not to mention, existing methods for Knowledge Graphs are usually tailored for specific graphs or tasks, lacking generalization capability. Therefore, it's necessary to employ LLMs to overcome the challenges faced by Knowledge Graphs.



LLM 的优点：一般知识、语言处理、泛化能力。LLM 的缺点：隐含知识、幻觉问题、无法决断问题、黑箱、缺乏特定领域的知识和新知识。知识图谱的优点：结构化的知识、准确度、决断能力、可解释性、特定领域的知识、知识演进。知识图谱的缺点：不完备性、缺乏语言理解、未见过的知识。（看图读英文）

大多数 LLM 都源自 Transformer 设计，其中包含编码器和解码器模块，并采用了自注意力机制。（根据我们日常使用chatgpt也会发现，不同的输入也分质量，好的prompt生成的回答也会更加的精准专业，不好的prompt只会返回给你百度百科那样的答案）

从而在2021年我们引入了新的领域，新职业prompt engineering

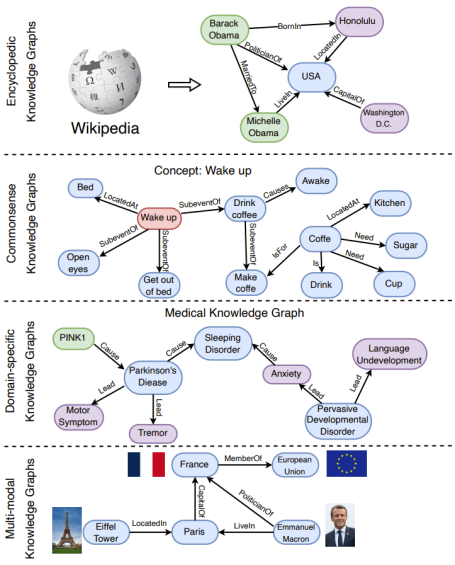
prompt 工程设计是一个全新领域，其关注的是创建和优化 prompt，从而让 LLM 能最有效地应对各种不同应用和研究领域。prompt 是 LLM 的自然语言输入序列，需要针对具体任务创建。prompt 可包含多个元素，即：指示、背景信息、输入文本。指示是告知模型执行某特定任务的短句。背景信息为输入文本或少样本学习提供相关的信息。输入文本是需要模型处理的文本。

Prompt engineering is an emerging field focused on creating and optimizing prompts to enable LLMs to effectively address a wide range of applications and research areas. A prompt is a sequence of natural language input tailored for a specific task that an LLM receives. It can consist of multiple elements: instructions, background information, and input text. Instructions are short sentences that inform the model about a specific task to perform. Background information provides context for the input text or for few-shot learning. Input text is the text that the model needs to process.

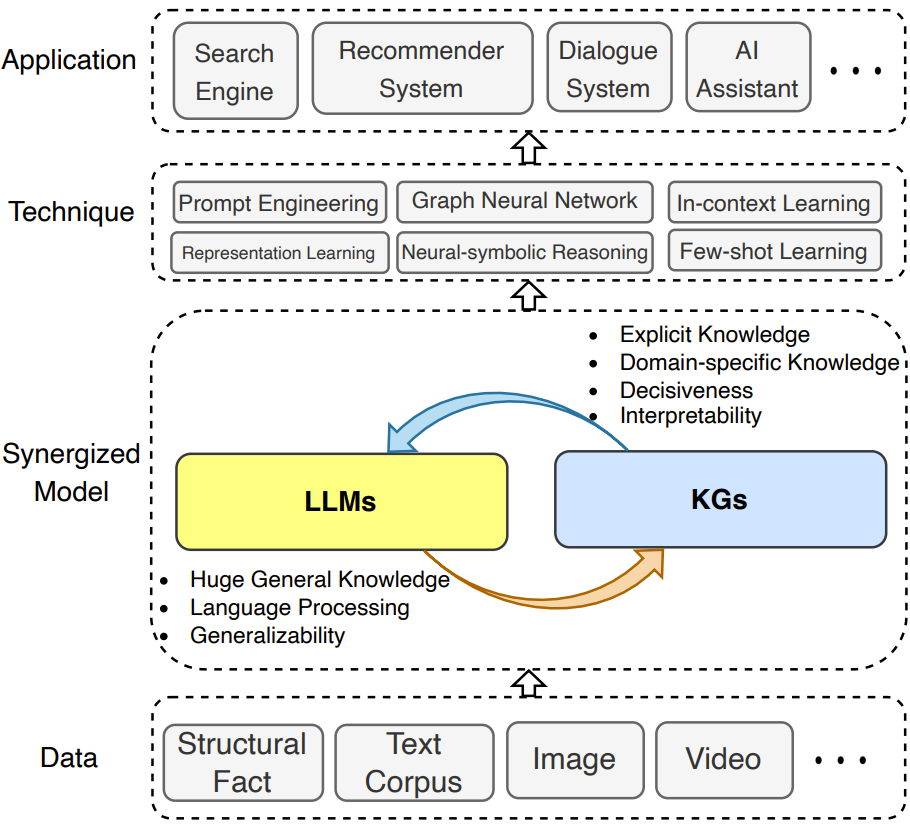
虽说这样但是如今上架了GPTs，会自动生成awesome prompt，也就是让机器自己去指令自己。

现有的知识图谱可分为四大类：百科知识型知识图谱、常识型知识图谱、特定领域型知识图谱、多模态知识图谱。

Existing knowledge graphs can be categorized into four main types: encyclopedic knowledge graphs, commonsense knowledge graphs, domain-specific knowledge graphs, and multimodal knowledge graphs.



如下：LLM 与知识图谱协同的一般框架，其中包含四层：数据、协同模型、技术、应用。



然而，这项技术它能否在真实世界的任务中可扩展使用仍然是一个开放问题。

这种**可扩展性问题**就联系到了如何对知识图谱与大语言模型进行嵌入，由于数据的过于庞大和更新速度之快，一定要选择一个高效的算法来继承

以下是几种方法：

1.直接嵌入

一种方法是直接将知识图谱或数据库的内容作为模型训练数据的一部分。这可以通过在训练材料中加入来自知识源的陈述或信息片段来实现，使模型能够学习这些知识并在适当的上下文中使用它们。

This scalability issue is related to how to embed knowledge graphs with large language models. Due to the vast amount of data and its rapid update rate, it is crucial to choose an efficient algorithm for integration.

Here are several methods:

1. Direct Embedding

One approach is to directly incorporate the contents of knowledge graphs or databases as part of the model's training data. This can be achieved by adding statements or information snippets from knowledge sources into the training materials, enabling the model to learn this knowledge and use it in the appropriate context.

2.检索-增强模型

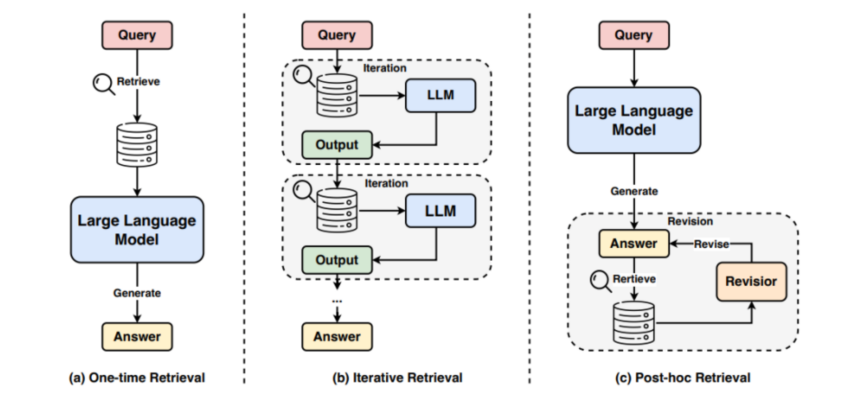
另一种方法是开发检索-增强模型，这种模型在生成响应之前，首先从外部知识库中检索相关信息。这要求模型具备一定的检索能力，能够根据给定的查询或上下文，找到并整合外部知识源中的相关信息。除了要具备检索能力意外，这种方法对于解决幻觉问题生成文本的过程中也起作用，通过实时检查每个句子以确保其准确性。如果模型生成了不准确的信息（即幻觉），此方法会介入，指出错误，并引导模型进行自我纠正。

如下图所示，检索的阶段和方式分为以下三种：

Retrieval-Enhanced Models

Another method involves developing retrieval-enhanced models, which retrieve relevant information from an external knowledge base before generating a response. This requires the model to have certain retrieval capabilities, enabling it to find and integrate pertinent information from external knowledge sources based on a given query or context. Besides having retrieval capabilities, this approach also plays a role in addressing the issue of hallucinations during text generation by conducting real-time checks on each sentence to ensure its accuracy. If the model produces inaccurate information (i.e., hallucinations), this method intervenes, points out the mistakes, and guides the model to self-correct.

The stages and methods of retrieval are divided into the following three types, as shown in the diagram below:



第一种在文本进入大预言模型前就进行检索，第二种检索之后进入大语言模型并进行多次的迭代，第三种则是先进入大语言模型输出结果后面对外部数据库进行检索一次次更新答案。

3.符号推理集成

通过将符号推理（如逻辑推理）集成到模型中，可以使模型能够利用外部知识进行更加复杂和准确的推理。这通常涉及到一些形式的规则引擎或逻辑框架，这些框架能够解释和应用从知识图谱中提取的关系和属性。

4.神经符号集成

近年来，也有研究探索神经网络和符号逻辑的集成，旨在结合两者的优点：神经网络在处理大规模数据和学习复杂模式方面的能力，以及符号逻辑在进行精确推理方面的优势。

举个例子：这是一篇医学和深度学习融合的文章

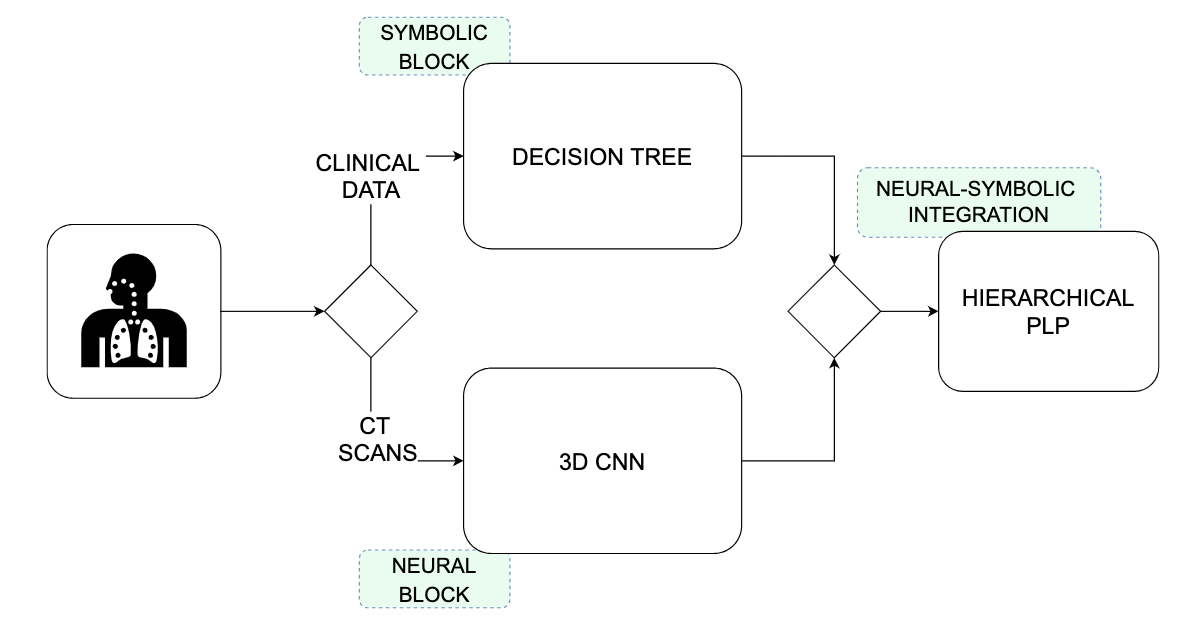
3. Symbolic Reasoning Integration

By integrating symbolic reasoning (such as logical reasoning) into the model, it can utilize external knowledge for more complex and accurate reasoning. This usually involves some form of rule engine or logical framework capable of interpreting and applying the relationships and properties extracted from knowledge graphs.

4. Neuro-Symbolic Integration

In recent years, research has also explored the integration of neural networks and symbolic logic, aiming to combine the strengths of both: the ability of neural networks to process large-scale data and learn complex patterns, and the advantage of symbolic logic in conducting precise reasoning.

Example: This refers to an article that merges medical science and deep learning.



Neural-Symbolic Integra-tion system: DT and 3D-CNNare integrated using HPP

这张图展示了一个融合了神经网络和符号逻辑（即神经符号集成）的决策系统框架，用于处理临床数据和CT扫描。

首先是神经块（Neural Block）的部分它使用3D CNN（三维卷积神经网络）用于处理三维的CT扫描图像数据。三维CNN能够从这些图像中学习空间特征，用于医学图像分析，比如识别肿瘤、异常等。

符号集成通常指的是将多个符号方法（如规则系统、逻辑推理）结合起来，用于增强系统的推理和决策能力。符号方法依赖于明确的知识表示和逻辑规则，而不是数据驱动的统计模型。典型的符号人工智能方法包括：

专家系统：使用规则和知识库进行推理和决策。

逻辑推理：基于一阶逻辑或其他逻辑系统进行推理。

规则系统：使用明确的规则集进行决策和推理。

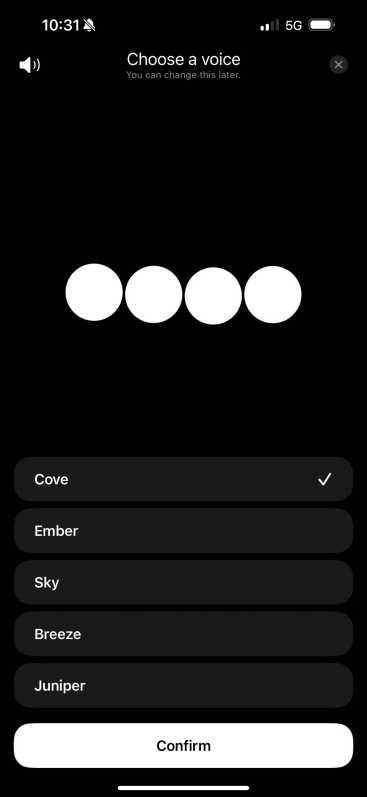
而上面的部分是基于决策树来处理数据也是符号块部分，决策树通常用于处理结构化的临床数据，它们可以基于规则来做出决策。在医疗数据分析中，决策树可以用来根据患者的症状、体征、实验室测试结果等信息来预测结果或分类。

最后是对这两个部分的集成也是通过一定的算法来进行信用分配层次化PLP（Hierarchical 这篇论文则是一种结合了概率逻辑编程的方法，它能够整合和推理不确定性知识整合来自3D CNN和决策树的信息，结合神经网络的学习能力和符号逻辑的推理能力，以便做出更为准确和可解释的医疗决策。

整体而言，这个系统设计反映了一个结合了深度学习（用于图像处理和特征提取）和符号推理（用于结构化数据分析和逻辑决策）的混合方法。神经符号集成的目标是利用两者的优势，提高模型的性能和解释能力。在医疗诊断的场景下，这种集成方式有助于提高决策的准确性，并且可能提供更好的解释性，这对于医疗专业人员理解模型推荐的诊断和治疗计划是非常重要的。

最后结语：

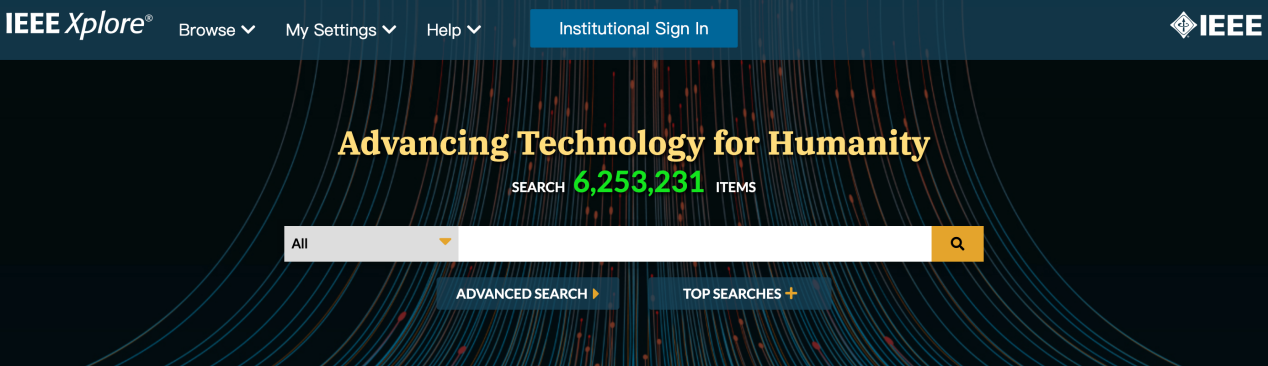
我们训练大语言模型最终的目的是让他像人一样思考，而这个特性可也以用一句话来总结：**泛化则是智能**。而一个成熟的大语言模型可以看作是一个单一的智能体，对于文本的生成来说它内部运用的是监督学习，投入大量的数据来训练。对于大语言模型更广阔的应用场景比如说对话功能内在采用的是强化学习，与人机交互领域和强化学习领域相结合来实现更好的智能。

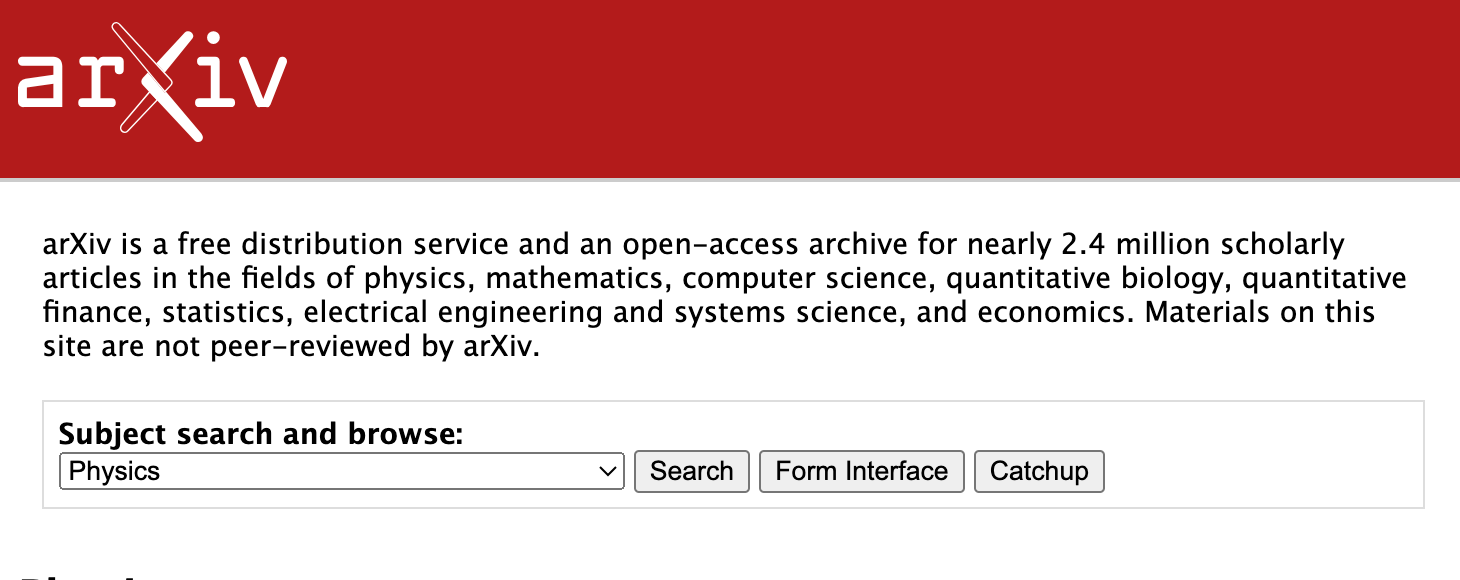


有关搜索引擎：

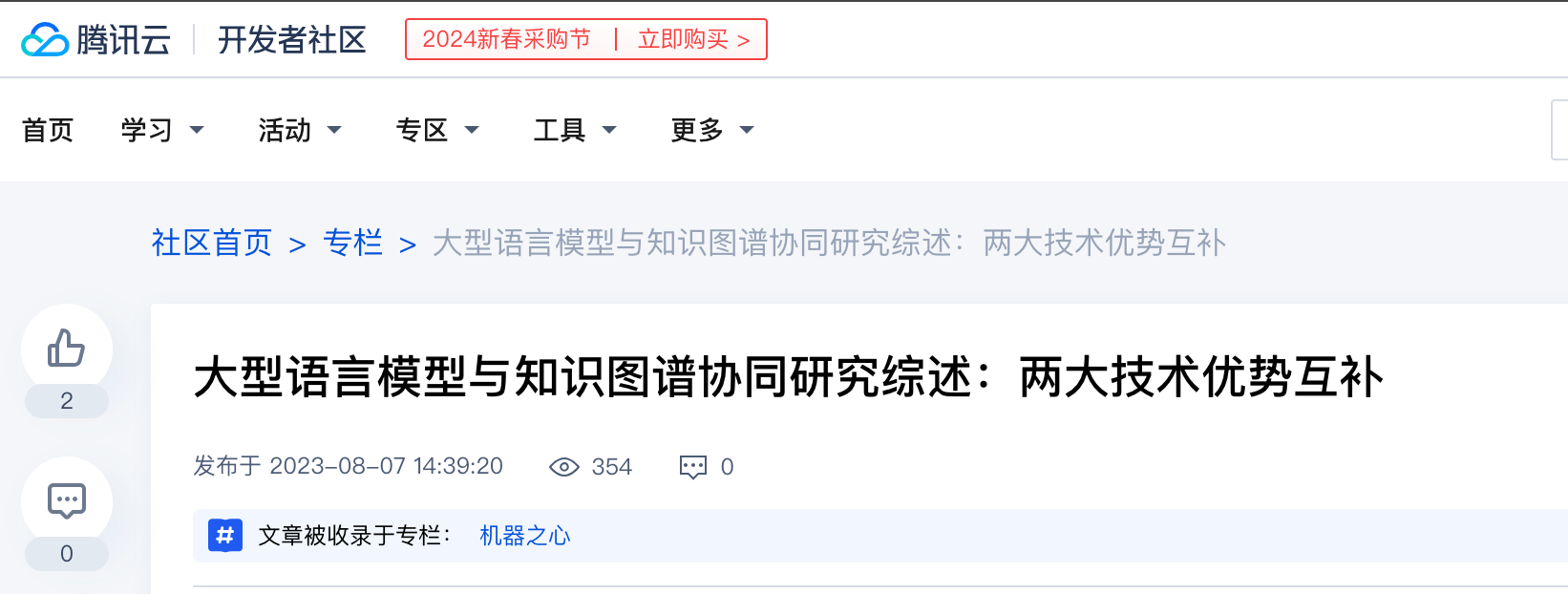
学术搜索引擎：Google Scholar、IEEE Xplore、ArXiv等学术搜索引擎查找与幻觉相关的论文和研究报告。







技术博客和新闻网站：机器之心，HACKERNOON





专业社群和在线论坛：Reddit在线社群

