|  |  |  |
| --- | --- | --- |
| 技术交底书  （以下由知识产权部填写）  案号：  日期： | 申请人：赛优来博有限公司  邮编：518055，总机：18926010972 |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 交底书名称 | 一种大模型自我反思的安全知识凝练的方法  →一种大模型自我反思的安全知识凝练的方法与系统 | | | |
| 发明人 | 李挥 杨傲 王滨 刘傲凡 | | | |
|  |  |  | （以下由知识产权部填写） | |
| 交底书撰写人 | 杨傲 |  | IPR负责人 |  |
| 撰写人办公电话 |  |  | |
| 撰写人手机号码 （急案必须填写） | 18336398909 | 流程助理 |  |
| 撰写人 E-mail | jarvisya@stu.pku.edu.cn |  | |

|  |  |
| --- | --- |
| 撰写指导意见  （IPR填写） | *比如：对于可能存在单一性问题的多个发明希望在一篇申请文件中提交时，提醒代理人可以不解决单一性问题；对于有特别事项需要提醒代理人时，要求其在作业前和专利工程师电话联系等等* |

缩略语和关键术语定义

LLM Large Language Model，大型语言模型是一种基于深度学习技术的人工智能模型，具有规模庞大的参数量以及强大的自然语言处理能力。

Transformer 由论文《Attention is All You Need》提出的基于Self-Attention机制的机器学习模型。

Self-reflection 自我反思，是指大型语言模型具有的一种评估并改善自身生成的输出的能力，旨在提升模型可靠性。

Knowledge Condensation 知识凝练，大型语言模型所具备的能够将海量、繁杂的知识信息进行提炼、压缩与整合，形成更加结构化、高效且精准的知识表示的能力。

## 1．相关技术背景以及最接近的现有技术

### 1.1 背景技术：用于方便读者理解本发明的技术背景

当前大型语言模型（Large Language Models, LLMs），例如ChatGPT和Claude，在推理、编程和科学研究等各个领域都显示出巨大的潜力。LLMs在各种应用场景中得到广泛采用。然而，LLMs并不总是可靠的。它们会产生有毒或不安全的内容，并且易受到“幻觉”（Hallucination）的影响，从而导致错误或不安全的输出。在政府、企业等人工智能应用场景中，数据安全是至关重要的，尤其是采用大模型输出内容参与工作流程的情况下，如何保证大模型生成的文本、代码等内容不含有风险内容和安全漏洞，成为大模型应用的关键问题之一。

2024 年 6 月，一名在字节跳动商业化技术团队实习的高校博士田某，因对团队资源分配不满，利用 Hugging Face（HF）的漏洞，在公司的共享模型里写入破坏代码。导致模型的训练效果忽高忽低，AML 团队无法核查原因，影响了团队的模型训练任务。字节跳动起诉该实习生，请求法院判令其赔偿公司侵权损失 800 万元及合理支出 2 万元，并公开赔礼道歉。

2024年11月，发生了大模型“语料污染”第一真实案例，一名用户通过ChatGPT编程来构建一个辅助交易机器人，但是在ChatGPT生成的代码调用了一个恶意的API地址，且直接把私钥明文提供给恶意API处理，代码运行后用户钱包被盗取$2.5k的损失。可见当模型采用搜索方式进行输出时未对搜索结果进行审查，使用恶意的参考资料导致生成恶意内容。

但是大模型应用的发展，可以通过多种防护措施对大模型的输出和推理过程进行防护：

1、在用户输入侧需对用户的输入进行严格审查，以确保其合规性与安全性。当系统监测到用户输入中包含恶意内容时，应立即中止当前会话，避免潜在安全风险。

2、对大模型输出的内容进行持续的检测和审查，重点检测输出内容中是否存在违规内容、侵权内容和隐私泄露的问题，确保用户能获得安全可靠的合法信息。

3、模型侧加强管理：监控用户与大模型交互的全过程，及时发现并阻止恶意用户的Prompt注入攻击，避免引导大模型产生违规内容。针对涉及敏感问题的情况，不回答或是使用安全回复，防止大模型误导用户或传播不合规信息。

4、针对幻觉问题，采取外部知识库，例如RAG (Retrieval Augmented Generation) 即检索增强生成技术，一种结合了信息检索和文本生成的人工智能技术，从而在生成时确保准确性，或是采用人类专家参与审查的方式减少幻觉。

为了进一步提升大模型应用的安全性，通过在生成过程中添加安全知识，并采用反思机制(Self-reflection)进一步增强生成内容的可靠性和鲁棒性。另外由于当前缺少轻量化的通用性的能够用于生成过程的RAG安全知识库，而且使用外部知识库可能会引入不相关的信息或噪声，这会误导大模型，并且大模型从冗长信息中获取知识的过程也会引入额外的开销，此时过滤浓缩的知识内容尤为重要。通过采取大模型自生成的知识凝练方法对安全知识进行生成，使得在任何大模型上都能够实现自身思考构建的安全机制，而无需借助外部知识库。

### 1.2 与本发明相关的现有技术一

#### 1.2.1 现有技术一的技术方案

Ziwei Ji等人提出了一种交互式自我反思方法，该方法结合了知识获取和答案生成。通过反馈过程稳步提高了生成答案的事实性、一致性和内涵。利用LLMs的交互性和多任务处理能力，逐步产生更准确的答案。

这项技术方案聚焦在医疗领域，涉及不常见的专业概念和潜在的社会风险，使得幻觉带来的挑战尤为关键，因为不准确或误导性的信息会对患者护理造成严重后果。

系统设计了一个迭代的、内省的过程，它利用了 LLMs的多轮交互性和多任务能力。其中涉及到自我反思方法，首先为给定问题生成相关背景知识，然后进行事实性评估。一旦检测到差异，模型就会被敦促自我纠正，利用其固有的能力来完善知识。这个循环过程会重复下去，直到达到令人满意的事实水平。在回答阶段，采用类似的生成-分数-细化策略来确保生成的答案与背景知识之间的一致性。此外，在答案和问题之间进行蕴涵评估。如果生成的答案不符合标准，该过程返回到初始阶段，重复该循环。

具体来说，系统包括三个循环：事实知识获取循环、知识一致性回答循环和问题蕴涵回答循环。

(1)事实知识获取循环：首先，模型根据提供的问题生成背景知识。此步骤利用了LLMs理解上下文的固有能力。然后，使用评分器对生成的知识进行事实性评估。如果事实性分数低于评估阶段设定的阈值，则会要求模型进行自我反思，要求它“请完善知识以提高其真实性”。

这种 generate-score-refine 策略以交互方式重复，直到生成的知识达到令人满意的事实性水平。此迭代过程促进了系统与其生成的知识之间的动态和迭代交互。并确保模型逐步完善生成的背景知识，将其与既定事实相结合。

(2)知识一致性回答循环：模型继续根据提供的问题和生成知识来生成答案。如果生成的答案的一致性分数低于阈值，则系统会提示模型进行内省、自我纠正，要求它“请优化回复以提高其一致性。”，重复此 generate-score-fine 策略，直到生成的答案达到一致。此迭代过程可确保模型根据经过审查的背景知识逐步细化生成的答案，从而保持其完整性。

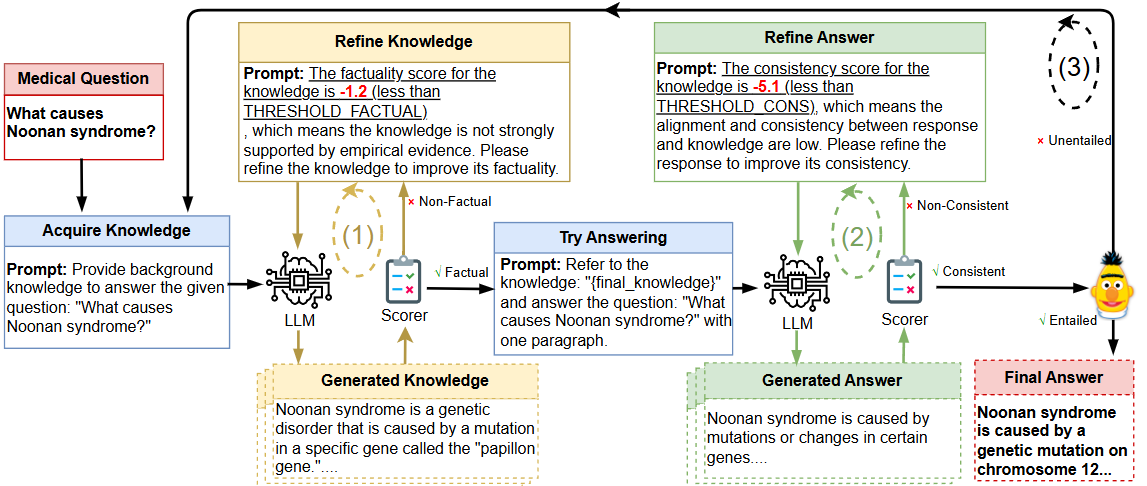
(3)问题蕴涵回答循环：通过相似性分数来评估生成的答案的蕴涵关系，判断这个答案与整个任务所涉及的背景知识、预期语义等是否存在合理的蕴含关系，以此确保答案的合理性以及确实是可被认可的、能够有效回应问题的（即具备可回答性）。如果不符合相应蕴含要求，就得重新循环整个流程来优化答案。

图1 技术一系统架构

[1] Ziwei Ji, Tiezheng Yu, Yan Xu, Nayeon Lee, Etsuko Ishii, and Pascale Fung. 2023. Towards Mitigating LLM Hallucination via Self Reflection. In Findings of the Association for Computational Linguistics: EMNLP 2023, pages 1827–1843, Singapore. Association for Computational Linguistics.

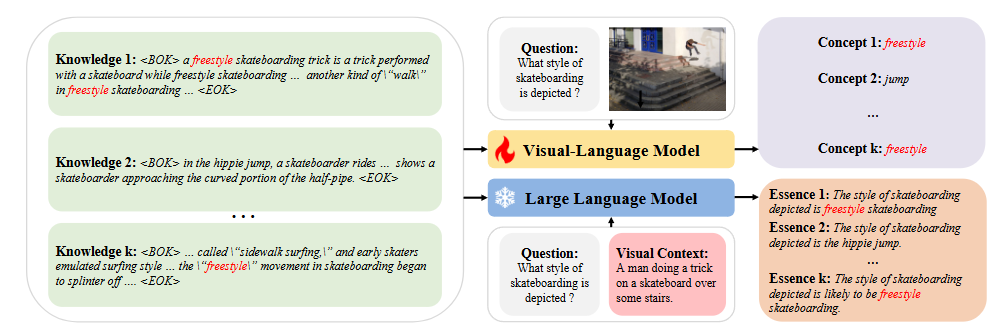
#### 1.2.2 现有技术一的缺点

本方案的主要工作是对医疗领域的生成式 QA场景进行减少幻觉的工作。该工作在安全领域或是模型不具有专业知识的情况下，尤其是在复杂或不明确的应用场景下，仍处于早期阶段，尚未准备好直接实际部署。且完全依赖于大模型自身的判断和评分，对模型自身能力要求高，无法从历史记录中收获知识以提升能力。

### 1.3 与本发明相关的现有技术二

#### 1.3.1 现有技术二的技术方案

Dongze Hao等人提出了两个协同模型：知识浓缩模型和知识推理模型。首先，利用视觉语言模型的多模态感知和推理能力，从检索到的冗长段落中提炼出简洁的知识概念，确保与视觉内容和问题的相关性。其次，利用大型语言模型的文本理解能力，将段落总结并浓缩为有助于回答问题的知识本质。然后将这两种类型的浓缩知识无缝集成到知识推理模型中，该模型明智地浏览合并的信息以得出结论性的答案。



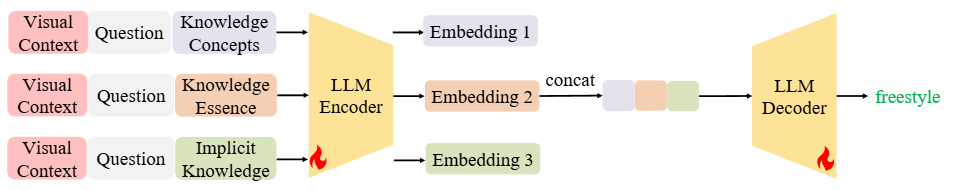


图2 知识浓缩模型和知识推理模型架构

工作聚焦于基于知识的视觉问答 （KB-VQA），要求模型利用外部知识来理解和回答基于视觉内容的问题。最近的研究从外部知识库中检索知识段落，然后使用它们来回答问题。但是，这些检索到的知识段落通常包含不相关或嘈杂的信息。例如将图像转换为视觉上下文，并将它们与问题和检索到的知识段落一起发送给LLM生成答案。由于检索到的知识段落包含许多嘈杂的信息，它们会误导模型预测错误的答案。

所提出的方法由两个模型组成：知识浓缩模型和知识推理模型。如图2所示，上方为知识浓缩模型的架构，下方为知识推理模型架构。

知识浓缩模型，采用 BLIP作为视觉语言模型，以及采用开源的大型语言模型Vicuna作为知识浓缩器，从检索到的知识中提取有用的信息。将图像首先输入图像编码器以提取视觉特征，再将视觉特征和文本嵌入输入到LLM以生成文本。将检索到的知识段落浓缩为知识概念和知识本质。

知识推理模型，在得到浓缩的知识概念和知识本质后，利用编码器-解码器架构对这些知识进行推理来预测答案。采用两种类型的知识推理方法来生成答案。对于串联知识模式，将视觉语境、问题、知识概念和本质等拼接成一句话，然后将句子输入到一系列编码器层中，共同编码这些文本信息。然后我们得到嵌入向量，传递给一系列解码器层以生成答案。对于串联嵌入模式，将视觉上下文和问题与不同类型的知识连接为不同的句子。这些句子被输入到一系列编码器层中，以分别编码不同的信息。然后我们将这些嵌入连接到一起并将它们传递给解码器。模型训练和推理同上。

[2] Hao, Dongze, Jian Jia, Longteng Guo, Qunbo Wang, Te Yang, Yan Li, Yanhua Cheng et al. "Knowledge Condensation and Reasoning for Knowledge-based VQA." arXiv preprint arXiv:2403.10037 (2024).

#### 1.3.2 现有技术二的缺点

技术二中的工作聚焦于基于知识的视觉问答（KB-VQA）场景，旨在通过浓缩保留更多的有效信息，提高模型的性能。但是仍然依赖于额外的知识库以及正确地获取知识，并且存在压缩模型将所有知识段落转换为无用的信息的情况，因为缺少了模型的深入思考和自我反思。

## 2．本发明技术方案的详细阐述（发明内容）

### 2.1 本发明所要解决的技术问题

尽管有用于大模型应用安全场景的措施，但值得注意的是，大多数措施聚焦于对特定模型的微调和安全对齐，或是输入输出侧的过滤机制。然而，本发明认为应当采取一种对各类大模型通用的，无需微调训练的，能够自我优化的安全机制，实现开箱即用的能力。通过大模型自身的任务中生成的信息，对其中的安全场景、安全漏洞和安全知识进行反思，对于反思生成的信息内容进行凝练总结，将其中安全编码、安全规范、安全知识和概念等凝练出来，形成大模型自生成的RAG知识库且可不断扩展，在之后的生成任务中采用其中的安全知识进行附加，从而提高大模型输出的安全性。提高了大模型生成的思考深度，且能够可扩展地将安全知识用于生成过程，同时不依赖于外部知识库。这种安全方案对于政府、企业等人工智能应用场景中格外关注数据安全的情况至关重要，对于确保大模型生成的文本、代码等内容不含有风险内容和安全漏洞具有实际意义。

### 2.2 本发明技术实现的具体实施

#### 2.2.1 大模型的自我反思机制实现

本发明对使用到的大模型自我反思机制进行了实现，将自我反思概念和RAG技术相结合，旨在提高LLMs在文本生成任务中的安全性，尤其是代码生成领域，而不需要对模型本身进行微调。该机制的工作流包括三个核心操作。首先，大模型根据特定的用户需求生成初始回复。随后LLMs会对该初始回复进行深度反思和迭代优化。一旦回复的安全程度达到了标准，将会从任务中凝练出安全规范的知识和概念，更新到自生成安全知识库中，以指导后续的相关任务。

第一个核心操作：初始回复生成。在该阶段，系统提供一个输入Prompt，生成模型将会生成初始输出，虽然这个初始输出通常满足输入中的基本要求，但仍存在一些安全问题或不符合规范，需要进一步改进。

第二个核心操作：反思和优化。在该阶段，系统使用大模型来自我反思和评判输出中是否存在任何潜在的缺陷及安全漏洞。如果输出没有缺陷，系统将正常显示输出结果。如果发现了缺陷或不符规范的情况，系统就会过渡到一个反思和迭代的流程。本流程所涉及的具体步骤如下：利用初始输出和任务的输入进行RAG查询，从而发现相关的安全知识，包括针对安全编码的规范、针对敏感问题的安全回复规范和历史记录中的标准回复等信息。当查询到适用的安全知识时，系统会将此信息与初始输入进行集成，从而进行参考。之后无论是否从RAG查询中获取到相关知识，都将进入到下一步迭代优化流程。

第三个核心操作：迭代优化。基于从初始回复中已识别的缺陷，LLMs将提出针对潜在改进方向的评估和措施，并进行迭代改进，不断重复这个过程，直到改进后的回复满足了所有指定的安全要求，此时改进后的回复和任务流程将会被投入安全知识凝练机制，从中凝练出安全规范的知识和概念，并采用统一的格式存储在安全知识库中。

通过反思机制，能够在生成任务中修复不安全的部分，优化后的回复不仅满足了初始的功能需求，而且显著提高了其安全性和规范性。并且每次迭代都建立在前一次迭代的结果基础上，结合来自之前输出的见解来动态更新知识库。这种不断发展的知识库使系统能够随着时间的推移，更有效地识别和解决潜在的安全缺陷。

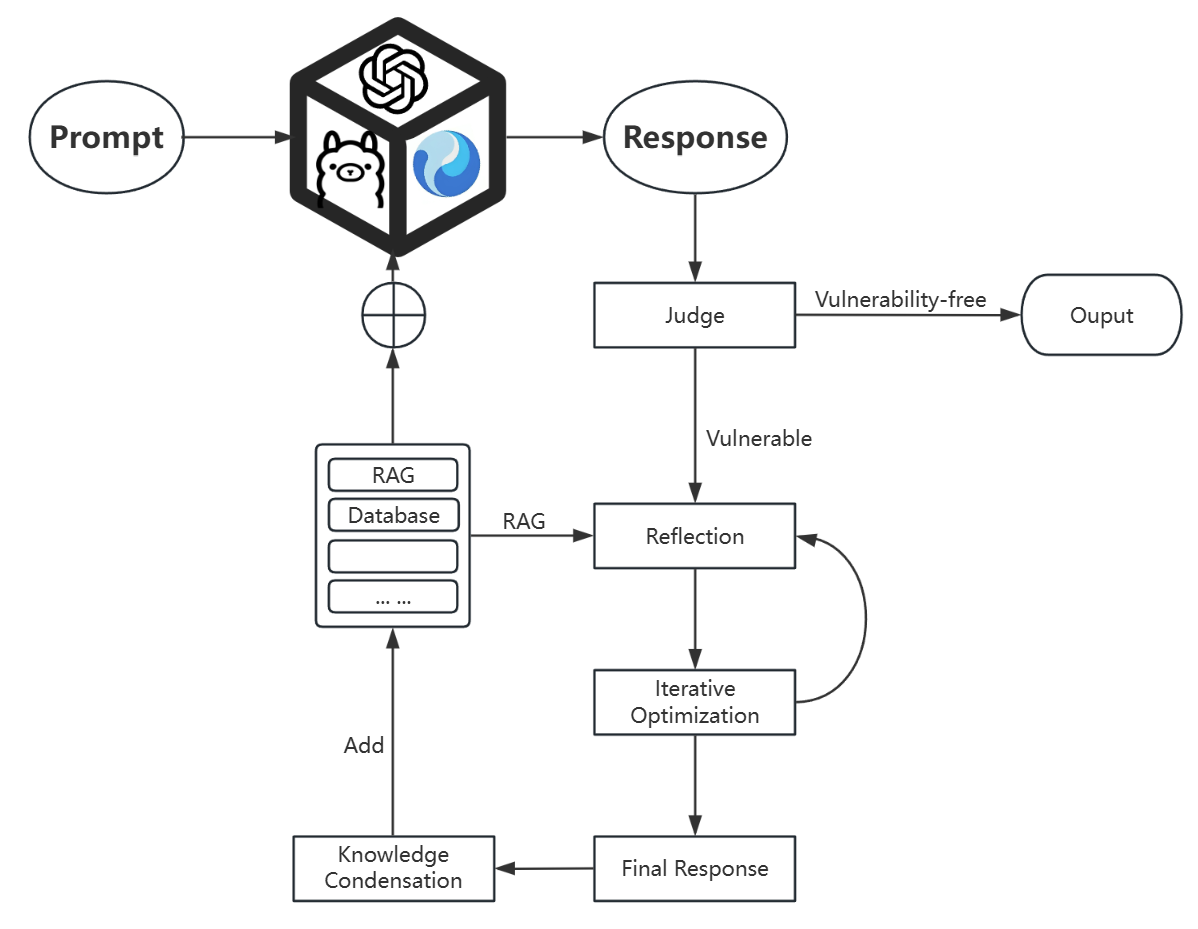


图3 大模型的自我反思机制流程

#### 2.2.2 基于大模型自我反思的安全知识凝练机制

迭代改进流程中，对于改进后的回复满足了所有指定的安全要求的情况，此时改进后的回复和任务流程将会被投入安全知识凝练机制，从中凝练出安全规范的知识和概念，并采用统一的格式存储在安全知识库中。

安全知识凝练机制将采用大模型进行递进式反思，结合COT(Chain of Thought)流程，进行安全表述层级，到安全知识层级，再到安全概念层级的的递进式凝练。

首先，对于最终的满足安全要求的回复，以及对应的上下文中的文本生成任务，首先将要求模型根据任务的背景，对于已经安全的回复和针对的问题，对比初始的具有缺陷的回复，指出这个修复过程中从哪些表述中实现了从存在缺陷到安全的转变，例如修复了哪些行的代码从而实现了安全代码生成，或是修改了哪些名词或句式的表述从而实现了符合安全规范的表达。这里完成了第一个层级，即安全表述层级。

其次，对于已经识别出的多条安全表述的转变，以及对这种转变的原因进行深入反思，对于思考过程的全部思考内容使用COT的形式展现出来，并将这些内容重新送到大模型，要求从中使用精简、本质化、无噪声的方式进行总结，从中提取安全知识，即安全表述的深层次原因，例如“代码输出前必须针对溢出漏洞进行审查，尤其注意使用strcpy()等函数的情况”，或是“对于strcpy()函数应当使用更安全的strcpy\_s进行代替”，抑或是“生成内容中涉及有毒物品的配置流程信息不予回答”。这些简介明了的安全知识将被从上下文思考过程中提炼出来，并被投入RAG安全知识库中。

最后，对于提取的安全知识，将会进一步精炼提取安全概念，此处的概念主要为安全知识涉及的核心点，例如strcpy()函数、溢出漏洞、边界审查，或是有毒物品等核心概念，对于这些核心概念将会另外记录一条概念及其对应的释义。便于后续RAG流程更快地检索到相关的核心概念，加快反思和检索流程速度。

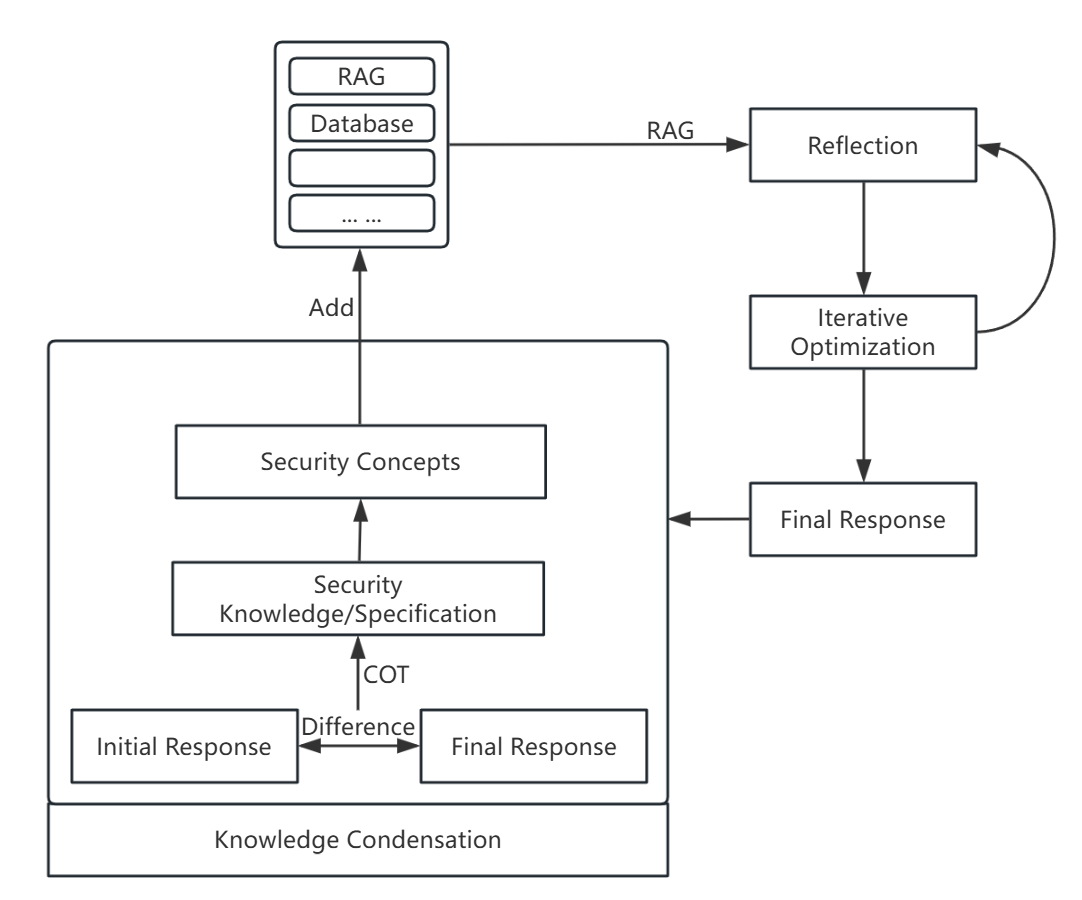


图4 基于大模型自我反思的安全知识凝练机制

## 2.3 本发明技术方案带来的有益效果

本发明技术方案提出了一种大模型自我反思的安全知识凝练的方法，能够给各类大模型带来通用的，无需微调训练的，能够自我优化的安全增益效果，实现开箱即用的能力。

通过自我反思机制提高了大模型生成的思考深度，且能够可扩展地将安全知识用于生成过程，对于确保大模型生成的文本、代码等内容不含有风险内容和安全漏洞具有实际意义。

通过大模型自身的任务中生成的信息，对其中的安全场景、安全漏洞和安全知识进行反思，对于反思生成的信息内容进行凝练总结，将其中安全编码、安全规范、安全知识和概念等凝练出来，形成大模型自生成的RAG知识库且可不断扩展，在之后的生成任务中采用其中的安全知识进行附加，从而提高大模型输出的安全性。同时不依赖于外部知识库。

这种大模型自我反思的安全知识凝练方法对于政府、企业等人工智能应用场景中格外关注数据安全的情况至关重要。

## 2.4 本发明的技术关键点和欲保护点是什么

1. 本方案提出了一种全新大模型自我反思的安全知识凝练的方法，通过大模型自身的任务中生成的信息进行反思，并对反思生成的信息内容进行凝练总结，用于后续生成任务提高大模型输出的安全性。
2. 本方案通过反思和凝练两种机制，实现了大模型自生成安全知识库，无需借助外部知识库，具有自我迭代优、可扩展等特点。

上述均为本发明技术关键点，同时也是欲保护点。