DB-GPT：利用私有大型语言模型增强数据库交互能力

薛思桥[◇]，蒋才高[◇]，施文辉[◇]，程方银[♥]，陈克庭[◇]，杨红军[◇]，张志平[♡]，何建山[◇]，张红阳[♦]，魏刚林[◇]，王钊、
Fan Zhou[◇]，Danrui Qi[♣]，Hong Yi, Shaodong Liu[♠]，Faqiang Chen[◇,*]
[◇]蚂蚁金服集团、[♡]阿里巴巴集团、[♥]京东集团、[♠]美团网
[♦]中国西南财经大学
[♣]加拿大西蒙弗雷泽大学
{siqiao.xsq,caigao.jcg,faqiang.cfq}@antgroup.com

摘要

最近在大型语言模型（LLMs）方面取得的突破将使许多软件领域发生转变。数据库技术与 LLM 的关系尤为重要，因为高效、直观的数据库交互至关重要。在本文中，我们将介绍 DB-GPT，这是一个革命性的、可投入生产的项目，它将 LLM 与传统数据库系统整合在一起，以增强用户体验和可访问性。DB-GPT 可理解自然语言查询，提供上下文感知响应，并高精度地生成复杂的 SQL 查询，使其成为从新手到专家用户不可或缺的工具。DB-GPT 的核心创新在于其私有 LLM 技术，该技术在特定领域的语料库上进行了微调，以维护用户隐私并确保数据安全，同时提供最先进的 LLM 的优势。我们详细介绍了 DB-GPT 的架构，其中包括一个新颖的检索增强生成（RAG）知识系统、一个根据用户反馈持续改进性能的自适应学习机制，以及一个具有强大数据驱动代理的面向服务的多模型框架（SMMF）。我们的大量实验和用户研究证实，DB-GPT 代表了数据库交互模式的转变，提供了一种更加自然、高效和安全的数据存储方式。论文最后讨论了 DB-GPT 框架对未来人与数据库交互的影响，并概述了该领域进一步改进和应用的潜在途径。项目代码见 https://github.com/eosphoros-ai/ DB-GPT。请根据 https://github.com/eosphoros-ai/DB-GPT#install 上的说明安装 DB-GPT，亲身体验 DB-GPT，并在 https://www.youtube.com/watch?v=KYs4nTDzEhk 上观看 10 分钟的简明视频。

1 导言

大型语言模型（LLMs），如 ChatGPT（Brown 等人，2020 年）和 GPT-4（OpenAI，2023 年），已经展示了它们在进行类人交流和理解复杂查询方面的卓越能力，并带来了将 LLMs 纳入各个领域的趋势（Anil 等人，2023 年；Gunasekar 等人，2023 年）。外部工具进一步增强了这些模型，使它们能够搜索相关的在线信息（Nakano 等，2021；Xue 等，2023c），利用工具（Schick 等，2023），并创建更复杂的应用程序（Chase，2022；Wang 等，2023；Chu 等，2023）、

*通讯作者：

预印本。正在审查。

	兰克链（大通，2022 年）	LlmaIndex (Liu, 2022)	PrivateGPT (Martínez et al., 2023)	聊天数据库（Hu 等人，2023 年）	DB-GPT
多 LM 集成	✓	✓	✗	✓	✓
文本到 SQL 的微调	✗	✓	✗	✗	✓
多代理战略	✓	✓	✗	✗	✓
数据隐私和安全	✓	✗	✓	✗	✓
多源知识	✓	✓	✗	✗	✓
双语查询	✗	✗	✗	✓	✓
生成数据分析	✗	✗	✗	✗	✓

表 1：各种竞争方法在不同方面的比较摘要。

2023).在数据库领域，传统系统通常要求用户具备高度的技术敏锐性，并熟悉特定领域的结构化查询语言（SQL）以进行数据访问和操作，而 LLM 则为自然语言界面铺平了道路，使用户能够通过自然语言查询进行表达，从而实现更自然、更直观的数据库交互。

然而，如何利用 LLMs 增强数据库操作能力，从而构建功能强大的终端用户应用程序，仍然是一个未决问题。现有的大多数研究（Chase，2022；Zhou 等人，2023；Hu 等人，2023）都采用了一种直接的方法，即通过简短提示或上下文学习（Wei 等人，2022），直接为常用的 LLM（如 GPT-4）提供交互说明。这种方法的优点是不太可能过度拟合训练数据，而且易于适应新数据，缺点是与使用中值大小 LLM 的微调替代方法相比，性能可能不够理想（Sun 等人，2023 年）。此外，为了进一步促进与数据库的智能交互，许多著作（Chase，2022；Liu，2022；Richards，2022）已将 LLM 驱动的自动推理和决策过程（又称代理）纳入数据库应用。然而，知识代理通常是针对特定任务的，而不是与任务无关的，这限制了其大规模使用。同时，以 LLM 为中心的数据库交互的隐私敏感设置虽然很重要，但一直未得到充分研究。以前的研究（Martínez 等人，2023 年；H2O.ai，2023 年）大多是通用型的，并非专为数据库操作而设计。

在这项工作中，我们介绍了 DB-GPT，这是一个用于 LLM 增强型应用的智能化生产就绪项目，可利用私有化技术摄取、构建和访问数据。DB-GPT 不仅利用了 LLM 固有的自然语言理解和生成能力，还通过代理和插件机制不断优化数据驱动引擎。竞争者比较汇总见表 1。概括而言，DB-GPT 具有以下显著优点：

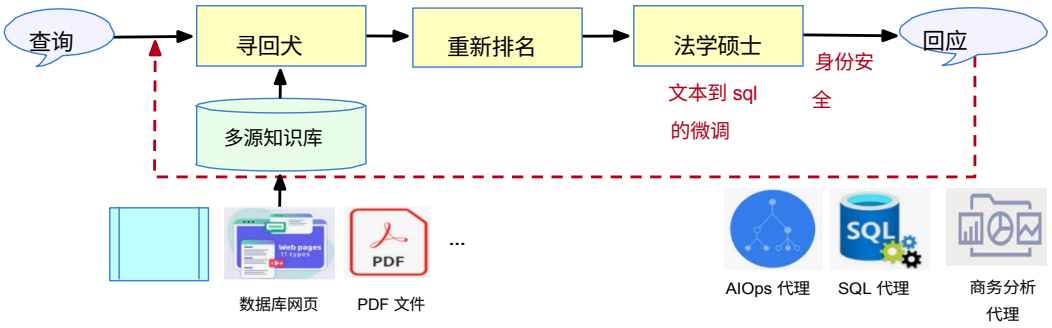

图 1：DB-GPT 的架构

- **隐私和安全保护。**DB-GPT 允许用户在个人设备或本地服务器上部署，甚至在没有互联网连接的情况下运行。数据在任何时候都不会离开执行环境，完全消除了数据泄露的风险。此外，在数据处理模块中还应用了代理去标识化技术（Wang 等人，2016），该技术作为一个中介，可以遮蔽数据集中的个人标识符，从而降低未经授权访问和利用私人信息的风险。

- **多源知识库问答优化。**与知识库问答（KBQA）的经典著作（Lan 等人，2022 年）相比，DB-GPT 建立了一个管道，可将多源非结构化数据（PDF、网页、图像等）摄取到中介表述中，将其存储到结构化知识库中，检索最相关的部分，并生成给定查询的综合自然语言回复。该管道经过效率优化，生成灵活，并可接受双语查询。
- **文本到 SQL 的微调。**为了进一步增强生成能力，DB-GPT 针对文本到 SQL 任务微调了几种常用的 LLM（例如，Llama-2（Touvron 等人，2023 年）、GLM（Zeng 等人，2022 年））。DB-GPT 大大降低了不具备 SQL 专业知识的用户与数据交互时的障碍。据我们所知，在相关著作中，只有LlamaIndex（Liu，2022）集成了这种微调替代方案，但它并没有针对双语查询进行优化。
- **知识代理和插件集成。**代理 "是一种自动推理和决策引擎。作为一个可投入生产的项目，DB-GPT 支持开发和应用具有高级数据分析功能的知识代理，这些自动决策有助于数据上的交互式用例。它还提供了各种查询和检索服务插件，可用作与数据交互的工具。

我们在各种基准任务（如文本到 SQL 和 KBQA）上对 DB-GPT 进行了严格评估。此外，我们还进行了案例研究和调查，以评估可用性和偏好。DB-GPT 在大多数方面都优于竞争对手。

2 系统设计

DB-GPT 的整体流程如图 1 所示。我们的 DB-GPT 系统以一般的检索-增强生成（RAG）框架（Chase，2022；Liu，2022；Xue 等人，2023c）为基础，集成了我们新颖的训练和推理技术，大大提高了系统的整体性能和效率。在本节中，我们将介绍每个阶段的设计，包括模型架构以及训练和推理范式。

2.1 用于质量保证的多源 RAG

LLM 通常是在大量开源数据或其他方的专有数据基础上进行训练的，而 RAG（Lewis 等人，2020 年）则是一种利用额外且通常是私有数据来增强 LLM 知识的技术。如图 2 所示，我们的 RAG 管道包括三个阶段：知识构建、知识检索和自适应上下文学习（ICL）（Dong 等人，2022 年）策略。

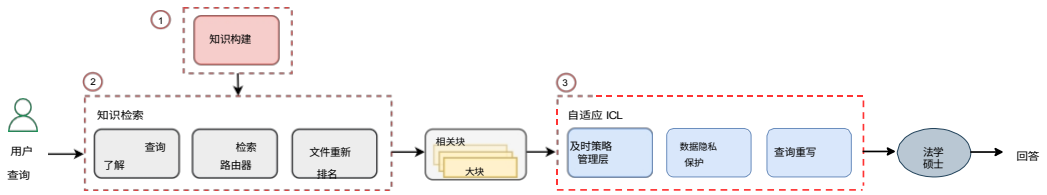

图 2：DB-GPT 中详细的 RAG 架构

知识构建。 我们的知识库 K 是来自不同来源的文档集合 $\mathbf{d}^{loc}, \ldots, \mathbf{d}^{loc}$，其中文档的数量为 N。根据 Chase（2022 年），我们将 K 分成每个文档 $\mathbf{d}_n$ 分成多个段落 $\mathbf{p}^{loc}_{n,1}, \ldots, \mathbf{p}^{loc}_{n,M_n}$，其中 M_n （以及下文中的 m）表示第 n 个文档的段落索引，并将每个段落嵌入一个多维度的通过神经编码器 f_{emb} 嵌入 $\mathbf{e}^{loc}_{key}$。值得注意的是，除了现有的如图 3 所示，DB-GPT 是基于向量的知识表示法，它还采用了倒排索引和图索引技术，以方便准确地查找与上下文相关的数据。

知识检索。 如图 4 所示，当一个语言查询 $\mathbf{x}$ 出现时，它将通过另一个神经编码器 f_{query} 嵌入到一个向量 $\mathbf{q}$ 中，然后我们检索出前 K 个相关段落

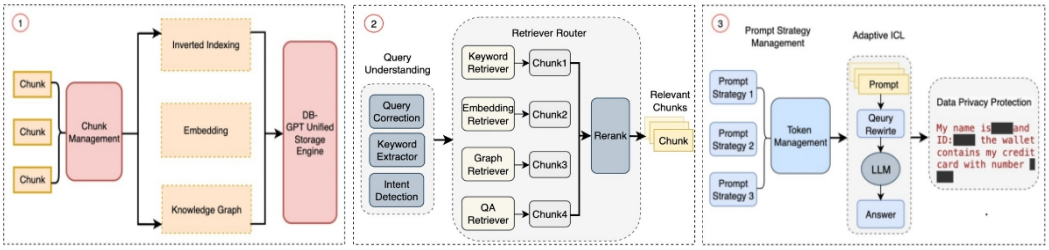

图 3： 的知识流水线-图 4： 的知识流水线-图 5： 的适应流水线-图 6： 的适应流水线
边缘构造 边缘检索 有效的 ICL 和响应生成

其中 K 是一个超参数。DG-GPT 支持多种检索器模型，例如，EmbeddingRetriever，它根据它们的余弦相似度进行检索，即 $\mathbf{q}^\top \mathbf{e}/\square\|\mathbf{q}\|\|\mathbf{e}\|$，KeywordRetriever，它匹配关键词而不是整个句子。在以下段落中，我们默认使用 EmbeddingRetriever。

学习嵌入和搜索。根据 Xue 等人（2023c）的研究，我们确信，由于训练了编码器 f_{key} 和 f_{query}，相似度越高，表示段落越相关。

其优化方法将在第 3.2 节中说明。直观地说，对于实际相关的查询-段落对，我们希望点积 $\mathbf{q}^\top$ e 相对较大。我们的编码器使用多语言-E5 基础模型架构（Wang 等人，2022a），因为我们支持账单语言编码文档。

自适应 ICL 和 LLM 生成。 在这一阶段，我们的系统执行 ICL（Dong 等人，2022 年）来生成响应：它根据 K 个搜索结果与查询的余弦相似度对其进行排序，然后将前 J 个（其中 $J \leq K$）结果插入预定义提示模板的上下文部分，最后由 LLM 生成响应。ICL 是一种用于提高 LLM 处理上下文信息性能的技术，方法是在训练或推理阶段加入额外的上下文。整个过程如图 5 所示。ICL 可增强语言模型对上下文的理解，提高推理和推断技能，并为其量身定制解决问题的能力。由于 ICL 的性能对特定设置非常敏感，包括提示模板、上下文示例的选择和示例的顺序等（赵等人，2021 年），因此在我们的 DB-GPT 系统中，我们提供了多种策略来制定提示模板（见清单 1 中的一个示例）。此外，我们还采用了隐私保护措施来掩盖个人信息。

```
背景信息：
{context_retro_1}
⋮
{context_retro_k}

 请根据所提供的信息，对用户的问题作出简洁而专业的答复。如果查询中有多个问题，请回答所有问题。如果用户的问题包含 "最近 "或 "最新 "等关键词，表示最近的时间范围，请注意当前日期与信息日期之间的对应关系。如果无法确定明确的答案，请回复 "根据所提供的信息无法回答该问题"。必须使用与问题相同的语言回答！

问题是：{问题}.
```

清单 1：LLM 的提示模板。

2.2 部署与推理：面向服务的多模型框架

模型即服务（MaaS）是一种基于云的人工智能方法，它为开发人员和企业提供了访问预构建、预训练机器学习模型的途径。在 DB-GPT 中，为了简化模型适配、提高效率并优化模型部署的性能，我们

介绍了面向服务的多模型框架（SMMF），它为多 LLM 的部署和推理提供了一个快速、易用的平台。

SMMF 由两个主要部分组成，即模型推理层和模型部署层。具体来说，模型推理层是为适应各种 LLM 推理平台而设计的，包括 vLLM（Kwon 等人，2023 年）、HuggingFace Transformers (HF)（？）、Text Generation Inference (TGI)（Huggingface，2021 年）和 TensorRT（英伟达，2021 年）。模型部署层是底层推理层和上层模型服务功能之间的中介。

部署层。在模型部署框架层的范围内，可以确定一套不可或缺的元素。由 API 服务器和模型处理程序组成的二人组负责为应用层提供强大的模型服务功能。模型控制器处于中心位置，负责管理元数据，同时也是广泛部署架构的枢纽。此外，模型工作者也非常重要，它与推理设备和基础设置建立了直接联系，从而确保了已实施模型的良好性能。

2.3 多代理战略

DB-GPT 支持多种角色与数据交互，如数据分析师、软件工程师和数据库架构师，提供数据库操作的整个流程以及精心策划的标准操作程序（SOP）。受 MetaGPT（Hong 等人，2023 年）的启发，DB-GPT 为各个代理分配了不同的角色，利用他们的优势和专长来解决棘手的任务。它通过协调机制协调不同 LLM 代理之间的协作，使它们能够沟通、共享信息和集体推理。DB-GPT 以文本到 SQL 微调 LLM 为基础，能够开发和应用具有高级数据库交互能力的代理。此外，与 LlamaIndex 不同的是，LlamaIndex 的组件为特定用例提供了更明确、更受约束的行为，而 DB-GPT 则赋予了代理更强的一般推理能力，同时减少了约束。

2.4 DB 插件

LLM 无疑是强大的，但它们可能并不擅长每项任务。LLM 不需要直接回答问题，而是可以通过整合插件（也称为工具）执行多个步骤来收集相关信息。[1].与通用插件不同（Schick et al、
2023），DB-GPT 的插件主要植根于数据库交互模式。这种设计有助于通过自然语言查询数据库，简化用户查询表达式，同时加强 LLM 的查询理解和执行能力。数据库交互模式由两部分组成：模式分析器和查询执行器，前者负责将模式解译为 LLM 可理解的结构化表示形式，后者负责根据 LLM 的自然语言回答在数据库中执行 SQL 查询。此外，DB-GPT 还集成了第三方服务，如 WebGPT（Nakano 等人，2021 年）中提出的网络搜索，无需离开聊天即可在另一个平台上执行任务。有了这些插件，DB-GPT 就能处理一些端到端数据分析问题，并具有很强的生成能力（我们称之为*生成式数据分析*）。请参阅示例。

3 模型和培训

3.1 文本到 SQL 的微调

尽管 LLMs，例如 CodeX（陈等人，2021 年）和 ChatGPT（刘等人，2023 年），已经在文本到 SQL 的 ICL 方面取得了成功的结果，但它们与具有中值大小 LLMs 的微调替代方案（孙等人，2023 年）相比仍有差距。因此，有必要根据特定领域的文本到 SQL 数据调整 LLM，使 LLM 更好地理解提示格式，从而进一步改进结果。

[1]在本文中，我们交替使用这两个术语。

模型架构。我们从预训练的 Qwen（Bai 等人，2023 年）开始，该模型已使用大量中英文语料进行了预训练。

数据集和训练。在我们的 DB-GPT 中，我们设计了一个特殊模块 DB-GPT-Hub，它封装了预处理记录（通过第 2.4 节中介绍的工具）、模型加载和微调的流水线。我们在 Spider（Yu 等人，2018 年）训练拆分的 Qwen 上进行微调，输入包括数据库描述和自然问题（见清单 2），输出是目标 SQL。

```
{"说明": "concert_singer 包含 stadium、singer、concert、singer_in_concert 等表。表 stadium 的列有 stadium_id、地点、名称、容量、最高、最低、平均。表 singer 的列有 singer_id、name、country、song_name、song_release_year、age、is_male。表 concert 有 concert_id、concert_name、theme、stadium_id、year 等列。表 singer_in_concert 的列有 concert_id、singer_id。演唱会的年份是体育场位置的外键。singer_in_concert 的 stadium_id 是歌手姓名的外键。concert_in_concert 中的 singer_id 是 concert_name 的外键、
"输入": "我们有多少名歌手？", "响应": "select count(*)
from singer"}
```

清单 2：文本到 SQL 微调的输入格式。

有关文本到 SQL 的微调 LLM 的架构和评估详情，请参阅我们即将发表的另一篇论文。

3.2 RAG 编码器

模型架构。由于我们支持双语应用，密钥和查询编码器 f_{key} 和 f_{query} 被初始化为多语言-E5-基础（ME5）模型架构（Wang 等人，2022a）。

它们的优化涉及最大化一个明确定义的目标：

$$\ell = \mathbf{q}^\top \mathbf{e} - \log \sum_{i=0}^{I} \exp\left(\mathbf{q}^\top \mathbf{e}\right), \tag{1}$$

其中，$\mathbf{e}_0$ 是已知包含查询相关信息的段落的嵌入，其他 I 个嵌入 $\mathbf{e}_1, \ldots, \mathbf{e}_I$ 属于否定段落集（参见第 3.2 节了解如何使用否定段落集）。

它们被选中）。通过优化公式 (1)，对于实际相关的查询-段落对来说，点积 $\mathbf{q}^\top \mathrm{e}$ 会变得相对较大。

数据集和训练。 根据 Xue 等人（2023c）的研究，我们使用查询-段落对来训练关键字编码器 f_{key} 和查询编码器 f_{query}。这些查询-段落对是从 DatabaseQA 中收集的（见第 4.2 节）：我们抽取 1,000 个查询-回复对作为正向对，并从整个段落库中随机抽取 5 个负向回复作为正向对。最后，我们收集 1000 对询问-回答，用于训练和评估。然后将所选的对分为 700 组训练对、100 组开发对和 200 组测试对。

我们将查询-回复配对传递给模型，为每个配对得出一个标量分数，并利用交叉熵损失使正向配对的分数最大化，同时使负向配对的分数最小化。

3.3 实施和部署细节。

知识库和 WebUI。关于 RAG 的实现，我们参考了 https://github.com/langchain-ai/langchain（Chase, 2022）GitHub 公共仓库中的代码，并采用 MIT 许可。至于 WebUI 的实现，我们自行开发，并以 MIT 许可发布在 https://github.com/eosphoros-ai/DB-GPT-Web 的 GitHub 代码库中。

部署详情。出于演示和测试目的，除非另有说明，我们的系统部署在阿里巴巴云上的一台服务器上，该服务器拥有 30G 内存、8 个逻辑内核（英特尔至强（Ice Lake）白金 8369B）和英伟达 A100 80G 张量核 GPU。

4 实验

我们介绍了为评估 DB-GPT 系统性能而设计的实验，包括文本到 SQL 响应的生成质量（第 3.1 节）和我们提出的 RAG 机制的质量保证性能（第 2.1 节）以及 SMMF 的效率性能（**?**）我们还提供了生成式数据分析的定性结果（第 2.4 节）。

4.1 文本到 SQL 评估

数据集。我们在 Spider（Yu 等人，2018 年）数据集上评估文本到 SQL 方法。Spider 是一个大型跨领域文本到 SQL 数据集，其中包含 8659 个训练实例和 1034 个开发实例，这些实例分布在 200 个数据库中。每个实例由特定数据库的自然语言问题及其相应的 SQL 查询组成。本文使用 Spider-dev 开发分库进行评估，因为测试分库尚未发布。根据问题的复杂程度，每个实例被分为不同的类别（简单、中等、困难和额外）。详见附录 B.1。

衡量标准。我们遵循先前的研究（Liu 等人，2023 年），使用执行准确率（EX）作为衡量标准。EX 将预测 SQL 查询的执行输出与基本真实 SQL 查询在某些数据库实例上的执行输出进行比较。EX 越高越好。

基础 LLM。我们将 DB-GPT 选用的 Qwen 与同样支持双语文本的 Baichuan（Baichuan，2023 年）进行了比较。

模型	指标（EX）				
	简单	中型	硬质	额外	总体情况
Qwen-7b-chat	0.395	0.256	0.138	0.042	0.235
Qwen-7b-chat-sft	**0.911**	**0.675**	**0.575**	**0.343**	**0.662**
Qwen-14b-chat	0.871	0.632	0.368	0.181	0.573
Qwen-14b-chat-sft	**0.919**	**0.744**	**0.598**	**0.367**	**0.701**
百川2-7b-聊天	0.577	0.352	0.201	0.066	0.335
Baichuan2-7b-chat-sft	**0.891**	**0.637**	**0.489**	**0.331**	**0.624**
百川2-13b-聊天	0.581	0.413	0.264	0.187	0.392
Baichuan2-13b-chat-sft	**0.895**	**0.675**	**0.580**	**0.343**	**0.659**

表 2：对 Spider-dev 数据集的评估

主要结果。表 2 显示了我们的 DB- GPT 系统中文本到 SQL 微调管道的有效性：与 EX 测得的原始 LLM 相比，微调后的 Qwen 和 Baichuan 版本都有显著改善。

4.2 RAG 评估

继（Lewis 等人，2020 年）之后，我们在一系列开放领域的质量保证任务中对 RAG 进行了实验。

数据集。我们构建了两个质量保证数据集：数据库质量保证（DatabaseQA）和金融质量保证（FinancialQA）。对于数据库质量保证，我们从三个具有代表性的数据库系统中收集了 1000 份 PDF 格式的公开教程：OceanBase（Group，2021 年）、MySQL（MySQL）和 MongoDB（MongoDB）。在金融质量保证方面，我们从研究机构发布的文档中抽取了 1000 个样本。对于每个数据集，我们构建了 100 个问题进行测试，其中问题由专家标注难点。有关数据集的详细信息，请参见附录 B.2。

衡量标准。我们请三位专家对每个回答进行评分，评分范围为 0 - 5 分，得分越高的回答越好，并取他们的平均分作为最终得分。

基础 LLM。我们使用四种 LLM：Qwen、Baichuan 和两个商用 LLM：ChatGLM- Turbo（Zeng 等人，2022 年）和 ChatGPT3.5（Brown 等人，2020 年）分别作为基础模型。对于 ChatGLM-Turbo 和 ChatGPT3.5，我们直接调用 API 来运行任务。

主要结果。如表 3 和表 4 所示，各数据集之间没有一致的优胜者：ChatGPT-3.5 在 DatabaseQA 数据集上胜出，而 ChatGLM2-7b 在 FinancialQA 数据集上表现最佳。由于 DB-GPT 集成了大多数流行的开源和商业 LLM，用户可以根据自己的 RAG 任务选择最合适的 LLM。

	ModelMetrics（平均分）			
	简单	中型	硬质	总体情况
Qwen-7b-chat	0.487	0.488	0.485	0.487
百川2-7b-聊天	0.470	0.468	0.466	0.468
聊天GLM-Turbo	0.460	0.459	0.464	0.461
CHATGPT-3.5	**0.663**	**0.644**	**0.628**	**0.645**

表 3：数据库 QA 数据集的 RAG 评估。

	ModelMetrics（平均分）			
	简单	中型	硬质	总体情况
Qwen-7b-chat	0.829	0.824	0.819	0.824
百川2-7b-聊天	0.897	0.893	0.895	0.895
聊天GLM-Turbo	**0.910**	**0.905**	**0.900**	**0.905**
CHATGPT-3.5	0.903	0.899	0.898	0.900

表 4：FinancialQA 数据集的 RAG 评估。

4.3 SMMF 评估

如第 2.2 节所述，我们的 DB-GPT 集成了 vLLM 作为主要推理框架。

数据集。测试在一台服务器上进行，该服务器配备 629G 内存、1TB 硬盘、40 个逻辑内核（英特尔至强处理器（Skylake，IBRS））、2992.953MHz 频率的 CPU 和 40G GPU 内存的英伟达 A100- PCIE GPU。CPU 频率为 2992.953MHz，英伟达 A100- PCIE GPU 带有 40G GPU 内存。在所有实验中，我们使用相同的 8 个令牌提示作为输入，同时将输出长度设置为 256 个令牌。

衡量标准。 我们使用以下三个指标：

- 第一个标记延迟（FTL）：以毫秒为单位，表示从 DB-GPT 模型部署框架收到请求到推理框架解码第一个标记所花费的时间。
- 推理延迟（IL）：以秒为单位，表示从 DB- GPT 模型部署框架收到请求到推理框架解码完整响应所花费的时间。
- 吞吐量：在所有请求中，DB-GPT 模型部署框架每秒处理的令牌总数。

基础 LLM。与第 3.1 节相同，我们使用 "曲文 "和 "百川 "作为实验的基础 LLM。

主要结果。从表 5 和表 6 中可以看出，结果表明，使用 vLLM 框架进行模型推理可以显著提高模型的吞吐量，同时大幅减少第一个标记延迟和整体推理延迟。此外，随着并发用户数量的增加，利用 vLLM 框架进行推理所带来的性能提升也尤为明显。因此，DB-GPT 选择集成 vLLM 作为 SMMF 的默认推理框架。

模型	# CCR	平台	衡量标准		
			超光速（MS）	IL(S)	吞吐量（托肯斯）
QWEN-7B-CHAT	4	VLLM	**22.5**	**4.0**	**258.9**
QWEN-7B-CHAT	4	高频	765.7	97.6	10.7
QWEN-7B-CHAT	16	VLLM	**23.1**	**4.1**	**258.7**
QWEN-7B-CHAT	16	高频	1152.0	138.9	9.2
QWEN-7B-CHAT	32	VLLM	**23.3**	**4.2**	**289.2**
QWEN-7B-CHAT	32	高频	1059.2	127.1	10.1

表 5：以 Qwen 为基础 LLM 的 SMMF 评估。

模型	# CCR	平台	衡量标准		
			超光速（MS）	IL (S)	吞吐量（托肯斯）
百川-7B-聊天	4	VLLM	**54.7**	**5.2**	**201.7**
百川-7B-聊天	4	高频	688.5	70.8	14.7
百川-7B-聊天	16	VLLM	**156.2**	**7.1**	**588.2**
百川-7B-聊天	16	高频	2911.7	985.4	4.2
百川-7B-聊天	32	VLLM	**380.0**	**9.6**	**870.2**
百川-7B-聊天	32	高频	6786.6	1630.7	5.1

表 6：以百川为基础 LLM 的 SMMF 评估。

5 相关工作

数据库的 LLM。LLM 的出现彻底改变了各种应用领域。近年来，研究人员探索了 LLM 在数据库方面的潜力，旨在彻底改变用户与数据库交互和查询的方式。最相关的著作是 LangChain（Chase，2022 年）和 LlmaIndex（Liu，2022 年）。我们的 DB-GPT 主要在双语查询和生成数据分析集成方面与它们有所不同。在其他相关著作中，PrivateGPT（Martínez 等人，2023 年）侧重于基于 LLM 的数据库应用的安全性和隐私设置，而 ChatDB（Hu 等人，2023 年）则主要通过符号记忆框架解决基于 LLM 的 SQL 生成和推理问题。表 1 总结了我们的 DB-GPT 系统与其他竞争方法的比较。

LLM 代理。代理接收用户输入或查询，并在执行查询时做出内部决策，以返回正确的结果。LLM （Yao 等人，2023 年）和 LLM 工具（Richards，2022 年；Hong 等人，2023 年）的最新发展普及了代理的概念。与 Langchain 和 LlamaIndex 中使用的代理框架相比，DB-GPT 在微调模型强大推理能力的支持下，实现了约束更少、任务更不可知的代理。

知识库问答与检索增强生成。知识库问题解答（KBQA）在利用知识库中存储的大量知识并使其为用户所用方面发挥着至关重要的作用（Lan 等人，2022 年；Cao 等人，2022 年）。LLMs 以最小的示范实现了显著的泛化（Shi 等人，2023 年），这暗示了它在 KBQA 中的潜力。我们的 DB-GPT 系统与基于 LLM 的数据库应用的研究成果一致，其中包括

通过整合外部数据存储（如 PDF、网页、谷歌文档等）来增强语言模型。与 LllmaIndex 相一致，我们的 DB-GPT 系统实现了一个强大的索引结构，将文档归类为节点，以实现高效的信息检索。将检索到的外部知识与预先训练的参数知识相结合的语言生成过程被称为检索增强生成（RAG）（Lewis 等人，2020 年），它已被广泛应用于知识密集型任务和数据库应用中。除了 RAG 的标准流水线外，DB-GPT 还提供了各种类型的双语文本分割、嵌入、排序方法，比竞争对手更加灵活。

基于 LLM 的应用程序的部署平台。基于 LLM 的应用的部署平台主要分为两类：分布式系统和云平台。分布式系统将 LLM 分布在多个节点上，通过网络协调和负载平衡提高性能和可靠性。相反，云平台将 LLM 托管在云服务器上，通过友好的用户界面或应用程序接口提供简单、方便的管理。在这种情况下，FastChat（Zheng 等人，2023 年）脱颖而出，能够在任何云平台上部署 LLM。它提供了一个用于简化模型和任务管理的 Web UI，并支持与 OpenAI 兼容的 RESTful API 以实现无缝集成。此外，SkyPilot（skypilot org，2022 年）提供了多种计费策略和优化技术，以提高成本效率和 GPU 利用率。云平台已成为 LLM 部署的首选和灵活之选，消除了对底层架构和细节的担忧。DB-GPT 支持这两种类型的平台，允许用户在个人设备或本地服务器上部署，即使在没有互联网连接的情况下也能运行，从而确保了数据的安全性和隐私性。

文本到 SQL 微调文本到 SQL 的目的是将从自然语言文本生成数据库 SQL 查询的过程自动化。这是一项长期存在的挑战，对于在不需要 SQL 专业知识的情况下提高数据库的可访问性至关重要。GPT-4（OpenAI，2023年）、PALM（Chowd-hery等人，2022年）和Llama-2（Touvron等人，2023年）等LLM在这项任务中通过少量提示或上下文学习取得了显著成果，其性能还可以通过微调进一步提高（Sun等人，2023年）。与 LangChain、PrivateGPT 和 ChatDB 等几个竞争对手相比，我们的 DB-GPT 对文本到 SQL 的常用 LLM 进行了微调。通过自动生成查询，DB-GPT 可以开发具有高级数据分析功能的会话代理。

6 结论

我们介绍了一个开源的数据库智能对话系统，该系统在解决各种任务方面的卓越能力证明，它优于现有的最佳解决方案。我们的系统方法为数据库 LLM 的研究做出了贡献。此外，我们的训练和推理策略可能有助于在一般领域开发基于检索的对话系统，使我们能够开启更广泛的实际应用。

参考资料

Anil, R., Dai, A. M., Firat, O., Johnson, M., Lepikhin, D., Passos, A., Shakeri, S., Taropa, E., Bailey, P., Chen, Z., et al. Palm 2 technical report. *ArXiv preprint arXiv:2305.10403*, 2023.

Bai, J., Bai, S., Chu, Y., Cui, Z., Dang, K., Deng, X., Fan, Y., Ge, W., Han, Y., Huang, F., Hui, B., Ji、L., Li, M., Lin, J., Lin, R., Liu, D., Liu, G., Lu, C., Lu, K., Ma, J., Men, R., Ren, X., Ren, X., Tan、C.、Tan、S.、Tu、J.、Wang、P.、Wang、S.、Wang、W.、Wu、S.、Xu、B.、Xu、J.、Yang、A.、Yang、H.、Yang、

J., Yang, S., Yao, Y., Yu, B., Yuan, H., Yuan, Z., Zhang, J., Zhang, X., Zhang, Y., Zhang, Z., Zhou, C., Zhou, J., Zhou, X., and Zhu, T. Qwen technical report. *arXiv preprint arXiv:2309.16609*, 2023.

Baichuan.百川 2：开放式大规模语言模型》。*arXiv preprint arXiv:2309.10305*, 2023.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. Language models are few-shot learners.*神经信息处理系统进展（NeurIPS）*，2020 年。

Cao, S., Shi, J., Pan, L., Nie, L., Xiang, Y., Hou, L., Li, J., He, B., and Zhang, H. KQA pro：用于知识库上复杂问题解答的显式组成程序数据集。在

Muresan, S., Nakov, P., and Villavicencio, A. (eds.), *Proceedings of the 60th* Annual *Meeting of the Association for Computational Linguistics (Volume 1: Long* Papers*)*, pp.计算语言学协会。

Chase, H. LangChain, 2022.

Chen, M., Tworek, J., Jun, H., Yuan, Q., de Oliveira Pinto, H. P., Kaplan, J., Edwards, H., Burda, Y., Joseph, N., Brockman, G., Ray, A., Puri, R., Krueger, G., Petrov, M., Khlaaf, H., Sastry, G.、Mishkin，P.、Chan，B.、Gray，S.、Ryder，N.、Pavlov，M.、Power，A.、Kaiser，L.、Bavarian，M.、Winter，C.、Tillet，P.、Such，F.P.、Cummings，D.、Plappert，M.、Chantzis，F.、Barnes，E.、Herbert-Voss，A、Guss，W.H.，Nichol，A.，Paino，A.，Tezak，N.，Tang，J.，Babuschkin，I.，Balaji，S.，Jain，S.，Saunders，W.，Hesse，C.，Carr，A.N.，Leike，J.，Achiam，J.，Misra，V.，Morikawa，E.，Radford，A. Knight，M、Knight, M., Brundage, M., Murati, M., Mayer, K., Welinder, P., McGrew, B., Amodei, D., McCandlish, S., Sutskever, I., and Zaremba, W. Evaluating large language models trained on code, 2021.

Chowdhery, A., Narang, S., Devlin, J., Bosma, M., Mishra, G., Roberts, A., Barham, P., Chung, H. W., Sutton, C., Gehrmann, S., Schuh, P., Shi, K.、Tsvyashchenko，S.、Maynez，J.、Rao，A.、Barnes，P.、Tay，Y.、Shazeer，N.、Prabhakaran，V.、Reif，E.、Du，N.、Hutchinson，B.、Pope，R.、Bradbury，J、Austin, J., Isard, M., Gur-Ari, G., Yin, P., Duke, T., Levskaya, A., Ghemawat, S., Dev, S., Michalewski, H., Garcia, X., Misra, V., Robinson, K.、Fedus, L., Zhou, D., Ippolito, D., Luan, D., Lim, H., Zoph, B., Spiridonov, A., Sepassi, R., Dohan, D., Agrawal, S., Omernick, M., Dai, A. M...、
Pillai, T. S., Pellat, M., Lewkowycz, A., Moreira, E., Child, R., Polozov, O., Lee, K., Zhou, Z.、Wang, X., Saeta, B., Diaz, M., Firat, O., Catasta, M., Wei, J., Meier-Hellstern, K., Eck, D., Dean, J., Petrov, S., and Fiedel, N. Palm：用路径扩展语言建模，2022.

Chu，Z.，Hao，H.，Ouyang，X.，Wang，S.，Wang，Y.，Shen，Y.，Gu，J.，Cui，Q.，Li，L.，Xue，S.，et al. Leveraging large language models for pre-trained recommender systems. *ArXiv preprint arXiv:2308.10837*, 2023.

Dong, Q., Li, L., Dai, D., Zheng, C., Wu, Z., Chang, B., Sun, X., Xu, J., and Sui, Z. A survey on in-context learning.2022.

集团，A. 海洋基地，2021 年。

Gunasekar, S., Zhang, Y., Aneja, J., Mendes, C. C. T., Del Giorno, A., Gopi, S., Javaheripi, M., Kauffmann, P., de Rosa, G., Saarikivi, O., et al. Textbooks are all you need. *arXiv preprint arXiv:2306.11644*, 2023.

H2O.ai.H2OGPT, 2023 年 5 月。

Hong, S., Zheng, X., Chen, J., Cheng, Y., Wang, J., Zhang, C., Wang, Z., Yau, S. K. S., Lin, Z., Zhou, L., Ran, C., Xiao, L., and Wu, C. Metagpt：多代理协作框架元编程》，2023 年。

Hu, C., Fu, J., Du, C., Luo, S., Zhao, J., and Zhao, H. Chatdb：用数据库作为符号存储器来增强llms》，2023 年。

拥抱脸文本生成推理》，2021 年 5 月。

Jiang, G., Jiang, C., Xue, S., Zhang, J. Y., Zhou, J., Lian, D., and Wei, Y. Towards anytime fine-tuning

：超网络提示下的连续预训练语言模型。*自然语言处理实证方法 2023 年会议论文集（EMNLP）*，2023 年。

Jin, M., Wen, Q., Liang, Y., Zhang, C., Xue, S., Wang, X., Zhang, J., Wang, Y., Chen, H., Li, X., Pan, S., Tseng, V. S., Zheng, Y., Chen, L., and Xiong, H. Large models for time series and spatio-temporal data：A survey and outlook, 2023.

Kwon, W., Li, Z., Zhuang, S., Sheng, Y., Zheng, L., Yu, C. H., Gonzalez, J. E., Zhang, H., and Stoica, I. Efficient memory management for large language model serving with pagedattention.*ACM SIGOPS 第 29 届操作系统原理研讨会论文集*》，2023 年。

Lan, Y., He, G., Jiang, J., Jiang, J., Zhao, W. X., and Wen, J.-R.复杂知识库问题解答：A survey, 2022.

Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Kuttler, H., Lewis, M., tau Yih, W., Rocktäschel, T., Riedel, S., and Kiela, D. Retrieval-augmented generation for knowledge-intensive nlp tasks.*ArXiv*, abs/2005.11401, 2020.

Liu, A., Hu, X., Wen, L., and Yu, P. S. A comprehensive evaluation of chatgpt's zero-shot text-to-sql capability, 2023.

Liu, J. LlamaIndex, 11 2022.

Martínez, I., Gallego Vico, D., and Orgaz, P. PrivateGPT, May

2023.MongoDB.MongoDB.

MySQL。MySQL。

Nakano, R., Hilton, J., Balaji, S., Wu, J., Ouyang, L., Kim, C., Hesse, C., Jain, S., Kosaraju, V., Saunders, W., et al. Webgpt：*ArXiv preprint arXiv:2112.09332*, 2021.

英伟达™（NVIDIA®）。TensorRT，2021 年 5 月。

OpenAI.GPT-4技术报告。*ArXiv预印本arXiv:2303.08774*，2023。

Pan, C., Zhou, F., Hu, X., Zhu, X., Ning, W., Zhuang, Z., Xue, S., Zhang, J., and Hu, Y. Deep optimal timing strategies for time series.In *ICDM*, 2023.

Qu, C., Tan, X., Xue, S., Shi, X., Zhang, J., and Mei, H. Bellman meets hawkes：基于模型的时点过程强化学习。2023 年 *AAAI 人工智能大会论文集*》。

Richards, T. B. Autogpt, 2022.

Schick, T., Dwivedi-Yu, J., Dessì, R., Raileanu, R., Lomeli, M., Zettlemoyer, L., Cancedda, N., and Scialom, T. Toolformer：*ArXiv preprint arXiv:2302.04761*, 2023.

Shi, X., Xue, S., Wang, K., Zhou, F., Zhang, J. Y., Zhou, J., Tan, C., and Mei, H. Language models can improve event prediction by few-shot abductive reasoning.*神经信息处理系统进展*》，2023 年。

Skypilot org.Skypilot, 2022.

Sun, R., Arik, S. O., Nakhost, H., Dai, H., Sinha, R., Yin, P., and Pfister, T. Sql-palm：改进的文本到 sql 大语言模型适应性，2023 年。

Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., Bikel, D., Blecher, L., Ferrer, C. C., Chen, M., Cucurull, G., Esiobu, D., Fernandes, J., Fu, J., Fu, W., Fuller, B., Gao, C., Goswami, V., Goyal, N., Hartshorn, A.、Fernandes，J.、Fu，J.、Fu，W.、Fuller，B.、Gao，C.、Goswami，V.、Goyal，N.、Hartshorn，A.、Hosseini，S.、Hou，R.、Inan，H.、Kardas，M.、Kerkez，V.、Khabsa，M.、Kloumann，I.、Korenev，A.、Koura，P.S.、Lachaux，M.-A.，Lavril，T.，Lee，J.，Liskovich，D.，Lu，Y.，Mao，Y.，Martinet，X.，Mihaylov，T.，Mishra，P.，Molybog，I.，Nie，Y.，Poulton，A.、Reizenstein, J., Rungta, R., Saladi, K., Schelten, A., Silva, R., Smith, E. M., Subramanian, R., Tan, X. E., Tang, B., Taylor, R., Williams, A., Kuan、

J.X., Xu, P., Yan, Z., Zarov, I., Zhang, Y., Fan, A., Kambadur, M., Narang, S., Rodriguez, A., Stojnic, R., Edunov, S., and Scialom, T. Llama 2: Open foundation and fine-tuned chat models, 2023.

Wang, H., He, D., and Tang, S. Identity-based proxy-oriented data uploading and remote data integrity checking in public cloud.*IEEE Transactions on Information Forensics and Security*, 11 (6):1165-1176, 2016.

Wang, L., Yang, N., Huang, X., Jiao, B., Yang, L., Jiang, D., Majumder, R., and Wei, F. Text embeddings by weakly-supervised contrastive pre-training. *ArXiv preprint arXiv:2212.03533*, 2022a.

Wang，Y.，Chu，Z.，Ouyang，X.，Wang，S.，Hao，H.，Shen，Y.，Gu，J.，Xue，S.，Zhang，J.Y.，Cui，Q.，et al. Enhancing recommender systems with large language model reasoning graphs. *ArXiv preprint arXiv:2308.10835*, 2023.

Wang, Z., Zhang, Z., Lee, C.-Y., Zhang, H., Sun, R., Ren, X., Su, G., Perot, V., Dy, J., and Pfister、T.学会提示，持续学习。*IEEE/CVF 计算机视觉与模式识别会议论文集*》，第 139-149 页，2022b.

Wei, J., Tay, Y., Bommasani, R., Raffel, C., Zoph, B., Borgeaud, S., Yogatama, D., Bosma, M., Zhou, D., Metzler, D., Chi, E. H., Hashimoto, T., Vinyals, O., Liang, P., Dean, J., and Fedus, W. Emergent abilities of large language models, 2022.

Xue, S., Shi, X., Hao, H., Ma, L., Zhang, J., Wang, S., and Wang, S. A graph regularized point process model for event propagation sequence.In *IJCNN*, pp.

Xue, S., Qu, C., Shi, X., Liao, C., Zhu, S., Tan, X., Ma, L., Wang, S., Wang, S., Hu, Y., Lei, L.、Zheng, Y., Li, J., and Zhang, J. A meta reinforcement learning approach for predictive autoscaling in the cloud.In Zhang, A. and Rangwala, H. (eds.), *KDD '22: The 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, August 14 - 18, 2022,* pp.ACM, 2022a.

Xue, S., Shi, X., Zhang, Y. J., and Mei, H. Hypro：用于事件序列长视距预测的混合归一化概率模型。*神经信息处理系统进展*》，2022b.

Xue, S., Shi, X., Chu, Z., Wang, Y., Zhou, F., Hao, H., Jiang, C., Pan, C., Xu, Y., Zhang, J. Y., Wen, Q., Zhou, J., and Mei, H. Easytpp：Easytpp: Towards open benchmarking the temporal point processes.2023a.

Xue, S., Wang, Y., Chu, Z., Shi, X., Jiang, C., Hao, H., Jiang, G., Feng, X., Zhang, J., and Zhou, J. Prompt-augmented temporal point process for streaming event sequence.In *NeurIPS*, 2023b.

Xue, S., Zhou, F., Xu, Y., Zhao, H., Xie, S., Jiang, C., Zhang, J., Zhou, J., Xiu, D., and Mei, H. Weaverbird：利用大型语言模型、知识库和搜索引擎增强金融决策能力，2023c.

Yao, S., Zhao, J., Yu, D., Du, N., Shafran, I., Narasimhan, K., and Cao, Y. ReAct：语言模型中推理与行为的协同。*国际学习表征会议（ICLR）*，2023 年。

Yu, T., Zhang, R., Yang, K., Yasunaga, M., Wang, D., Li, Z., Ma, J., Li, I., Yao, Q., Roman, S., Zhang, Z., and Radev, D. Spider：用于复杂跨域语义解析和文本到 sql 任务的大规模人类标注数据集。*2018 年自然语言处理经验方法会议论文集*》，比利时布鲁塞尔，2018 年。计算语言学协会。

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

Zhao, T. Z., Wallace, E., Feng, S., Klein, D., and Singh, S. Calibrate before use：*ArXiv preprint arXiv:2102.09690*, 2021.

Zheng, L., Chiang, W.-L., Sheng, Y., Zhuang, S., Wu, Z., Zhuang, Y., Lin, Z., Li, Z., Li, D., Xing、E.P., Zhang, H., Gonzalez, J. E., and Stoica, I. Judging llm-as-a-judge with mt-bench and chatbot arena, 2023.

Zhou, X., Li, G., and Liu, Z. Llm as dba, 2023.

附录

A 正在进行和未来的工作

我们目前正在探索一些扩展功能，以便在我们的系统中处理更复杂的对话和分析案例。我们对处理以下问题特别感兴趣

- 更强大的代理。用户可能希望我们的系统不仅能进行分析，还能提供更强大的能力，例如基于历史数据和预测决策能力的经典时间序列预测（Jin 等人，2023 年；Xue 等人，2021 年、2022 年 b、2023 年 a）（Xue 等人，2022 年 a；Qu 等人，2023 年；Pan 等人，2023 年）。
- 整合更多模型训练技术。除了预训练，语言模型的持续学习技术也是研究界关注的焦点，如持续预训练（蒋等人，2023）、及时学习（王等人，2022b；薛等人，2023b）。这些方法的整合将极大地促进这些领域的研究。
- 更方便用户的呈现方式。用户可能希望我们的系统能以表格和图表等更丰富的形式呈现答案。我们启动了一个新项目 DB-GPT-Vis[2]为由 LLM 支持的聊天框提供灵活多样的可视化组件。

B 实验详情

B.1 文本到 SQL 评估详情

数据集详情。 表 7 显示了数据集的分布情况。

数据集	# 问题				
	简单	中型	硬质	额外	总体情况
蜘蛛开发	248	446	174	166	1034

表 7：文本到 SQL 数据集详情。

B.2 RAG 评估详情

数据集详情。我们为数据库领域（DatabaseQA）和金融领域（FinancialQA）各收集了约 100 个问题。此外，我们还根据专家提出的难点对问题进行了注释。表 8 显示了两个数据集的统计数据。

数据集	# 问题			
	简单	中型	硬质	OVERRAL
数据库质量保证	37	35	16	88
财务质量保证	50	13	11	74

表 8：RAG 数据集详情。

B.3 SMMF 评估详情

更多结果我们提供了以 Vicuna 为基础 LLM 的结果，如表 9 所示。结果与表 5 和表 6 一致。

[2]https://github.com/eosphoros-ai/GPT-Vis

模型	# CCR	平台	衡量标准		
			超光速（MS）	IL (S)	吞吐量（托肯斯）
VICUNA-7B	4	VLLM	**23**	**5**	**217**
VICUNA-7B	4	高频	815	67	15
VICUNA-7B	16	VLLM	**36**	**7**	**646**
VICUNA-7B	16	高频	5128	914	5
VICUNA-7B	32	VLLM	**53**	**8**	**1000**
VICUNA-7B	32	高频	11251	1453	6

表 9：以 Vicuna 为基础 LLM 的 SMMF 评估。

图 6：DB-GPT 的主界面：配置和聊天框。

C 软件界面

DB-GP 系统的主界面如图 6 和图 7 所示。

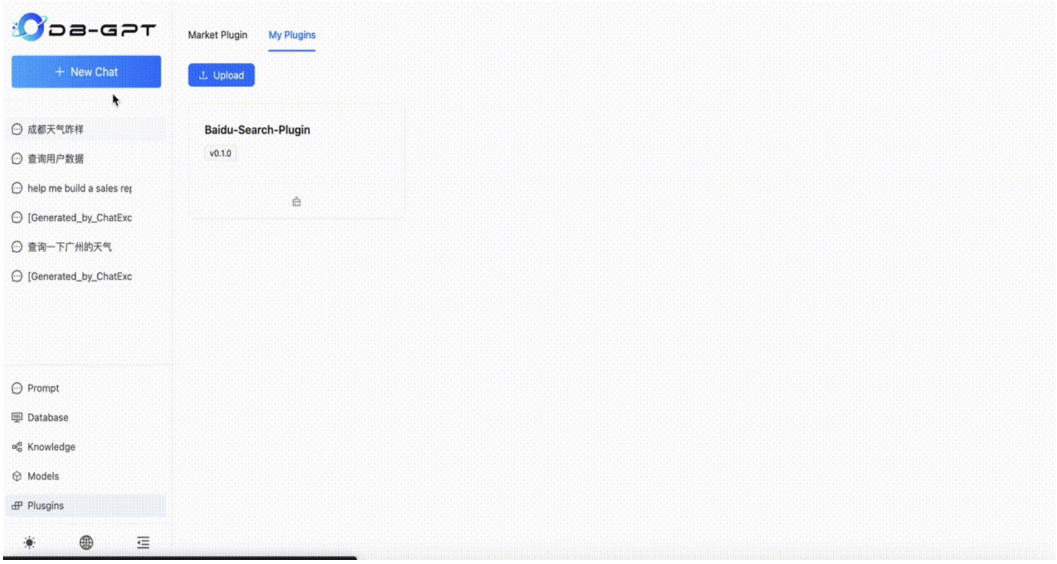

图 7：DB-GPT 的 "插件 "选项卡：用户可选择为质量检测任务加载代理插件（如网络 搜索代理）。