TiInsight：一种基于大型语言模型的基于SQL的自动化探索性数据分析系统

Jun-Peng Zhu
西北农林科技大学
PingCAP

Boyan Niu
PingCAP，中国

Peng Cai
华东师范大学，中国

Zheming Ni
PingCAP，中国

Kai Xu
PingCAP，中国

Jiajun Huang
PingCAP，中国

Shengbo Ma
PingCAP，中国

Bing Wang
PingCAP，中国

Xuan Zhou
华东师范大学，中国

Guanglei Bao
PingCAP，中国

Donghui Zhang
PingCAP，中国

Liu Tang
PingCAP，中国

Qi Liu
PingCAP，中国

arXiv:2601.09404v1 [cs.DB] 14 Jan 2026

摘要

基于 SQL 的探索性数据分析在数据分析社区中备受关注。大语言模型（LLM）的兴起推动了数据探索从人工转向自动化的范式变革。然而，现有方法通常缺乏跨领域分析能力，对大语言模型能力的探索也仍不充分。本文介绍了 TiInsight，这是一个基于 SQL 的自动化跨领域探索性数据分析系统。首先，TiInsight 提供了一个用户友好的图形用户界面，用户可以通过自然语言查询探索数据。其次，TiInsight 提供了一套强大的跨领域探索性数据分析流程：分层数据上下文（即 HDC）生成、问题澄清与分解、文本转SQL（即 TiSQL）以及数据可视化（即 TiChart）。第三，我们已在 PingCAP 的生产环境中实现并部署 TiInsight，并通过具有代表性的数据集展示了其能力。

CCS 概念

• **信息系统 → 信息系统应用**。

关键词

探索性数据分析、大语言模型

ACM 引用格式：

Jun-Peng Zhu、Boyan Niu、Peng Cai、Zheming Ni、Kai Xu、Jiajun Huang、Shengbo Ma、Bing Wang、Xuan Zhou、Guanglei Bao、DonghuiZhang、Liu Tang 和 Qi Liu。2018年。TiInsight：一种通过大语言模型实现的基于SQL的自动化探索性数据分析系统。载于请务必填写权利确认邮件中的正确会议标题（会议缩写 '*XX*）。ACM，美国纽约州纽约市，4页。https: //doi.org/XXXXXXX.XXXXXXX

允许免费复制本作品的全部或部分内容，以供个人或课堂使用，但不得为牟利或商业利益而复制或传播，并且副本首页须包含本声明及完整引文。作者以外的其他人拥有的本作品组成部分的版权必须予以尊重。注明出处后，允许摘录。若要以其他方式复制、重新出版、发布到服务器或分发至列表，则须事先获得明确许可和/或支付费用。请通过 permissions@acm.org 申请许可。会议缩写 '*XX*，纽约州伍德斯托克 © 2018 年版权归所有者 / 作者所有。出版权已授权给 ACM。ACM ISBN 978-1-4503-XXXX-X/2018/06
https://doi.org/XXXXXXX.XXXXXXX

1 引言

基于 SQL 的探索性数据分析 [5] 在数据分析社区中备受关注。此过程包括编写 SQL查询以实现特定的数据探索目标，并使用专用工具直观地呈现结果。用户会反复执行这些操作，直到获得满意的数据探索结果。此过程涉及两个影响数据探索效率和质量的关键难点：（1）SQL 熟练程度；（2）推荐合适的数据可视化图表。为解决这些难点，已有若干文本转SQL和表格到图表方法被提出。大语言模型（LLMs）的兴起为应对这些挑战带来了新的机遇。据我们所知，最先进（SOTA）的方法仍存在局限：

(1) 文本转SQL的跨领域泛化能力有限。基于大语言模型的文本转SQL方法已展现出优于传统方法的性能。然而，这些方法 [3] 通常需要对大语言模型进行微调，以适应不同的数据领域。这种方法需要带有准确标注的数据，且耗时较长。在实践中，微调对于跨现实数据领域的数据探索并不奏效。其他基于大语言模型的方法则采用多种提示技术来提高文本转SQL的准确率。然而，这些技术主要面向基准测试，在 [3] 现实场景中的准确率仍然不足。

（2）文本转SQL难以有效应对用户意图不明确的情况。在真实的探索性数据分析场景中，用户在处理陌生数据集时，往往难以清楚表达自己的探索目标。这会导致当前最先进的文本转SQL方法无法生成正确的SQL语句，甚至在许多情况下根本无法生成SQL语句。例如，在PingCAP的生产环境中，用户经常会询问："增长率是多少？"。这个问题可能缺少时间参数；更准确的表述应为："今年的增长率是多少？"。在最糟糕的情况下，用户在数据分析任务开始时可能难以清楚表达自己的意图。要准确回答这些问题，不仅需要大语言模型具备语义解析能力，还需要强大的数据分析能力，以便推断明确问题之外的用户意图。现有最先进的文本转SQL方法难以在这种情境下生成SQL，更遑论解决准确性问题。

（3）表格到图表推荐过程过于复杂。现有的表格到图表方法 [4] 通常依赖强化学习或深度神经网络，因此往往过于复杂。某些方法 [1]要求用户掌握特定的可视化查询语言，显著限制了其在端到端探索性数据分析系统中的适用性。许多用户研究发现，大多数图表都遵循特定的可视化规则。例如，饼图适用于展示分类数据的占比或分布。特别是，数据分析师积累了大量数据可视化的启发式规则，但目前缺乏一种简单直接的方法将这些规则用于可视化推荐。大语言模型的出现为基于规则的数据可视化带来了新的机遇。

为了解决这些局限性，我们提出了 TiInsight，这是一个基于SQL的端到端自动化跨领域探索性数据分析系统。TiInsight 使用户无需具备先前的专业知识即可进行端到端数据探索。在自然语言用户查询的驱动下，TiInsight 能够从自然语言输入无缝过渡到可视化探索结果。该系统具有以下独特且创新的特性。首先，TiInsight 提供直观的图形用户界面，支持用户通过自然语言输入与探索性数据集无缝交互。其次，TiInsight 具备 HDC 生成能力，可高效执行跨领域数据探索任务。第三项关键特性是在 HDC 的基础上集成了高效的文本到 SQL工具 TiSQL。此外，TiInsight 还集成了基于规则的数据可视化工具 TiChart。凭借这些能力，TiInsight 在广泛的跨领域数据分析场景中展现出自身价值。

通过本次演示，SIGMOD与会者可以体验 TiInsight 的实际运行。与会者指定数据集后，可以：(1) 使用 HDC组件，利用大语言模型的上下文摘要能力来描述用户的数据集；(2) 使用自然语言提出希望从数据集中获取的问题。随后，TiInsight 会使用问题澄清与分解组件来细化并分解用户的问题。在此过程中

在此过程中，访问者可以通过图形用户界面获取明确的问题；(3) 利用 TiSQL 自动生成相应的 SQL语句，并针对每个子问题在数据库中执行查询；最后 (4) 生成与结果相对应的可视化图表。本文通过两个具有代表性的现实场景展示 TiInsight。第一个是 Financial，这是 PingCAP 在概念验证（PoC）阶段面向目标行业业务的数据集。该数据集详细提供了有关美国联邦储备系统联邦基金利率的信息，以及关键经济指标的数据。第二个是 Bird [2]。系统可通过以下网址访问：https://www.tiinsight.chat。演示视频可通过以下网址观看：https://youtu.be/JzYFyYd-emI。

2 系统概述

TiInsight 提供自然语言界面，使最终用户能够开展在线探索性数据分析。图 1 展示了 TiInsight 的架构。TiInsight 的架构大致可分为以下五个组成部分：

1 **HDC 生成**。我们设计了 HDC 生成组件，它利用大语言模型在探索开始时在线高效生成数据库模式的分层数据上下文。

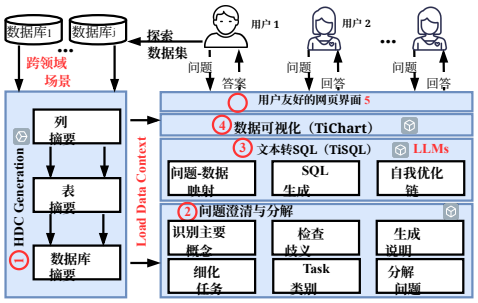


图 1：TiInsight 的整体架构。

大语言模型擅长总结海量数据集中的文本内容。它们能够从数据库模式中提取最重要的内容，以指导后续的探索过程。该组件使我们能够从数据库模式的不同层级理解用户提供的数据。

2 **问题澄清与分解**。我们利用大语言模型解读并澄清来自不同用户的问题。具体而言，该组件会消除用户意图中的歧义，并提供可能的解释。随后，TiInsight 判断是否应将用户的任务分解为一系列子任务。

3 **文本转SQL（TiSQL）**。我们使用 TiSQL 将用户问题转换为 SQL 语句。首先，TiSQL 需要将用户问题映射到特定的表和列，同时为 SQL条件语句提供取值。在这一过程中，我们提出了一种从粗粒度到细粒度的两阶段映射方法。随后，TiSQL 使用大语言模型生成 SQL语句。我们精心设计了 TiSQL 的提示词。不过，此步骤生成的 SQL语句仍然存在错误。因此，TiSQL 使用自我修正链来改进生成的 SQL。

4 **数据可视化（TiChart）**。在 TiInsight 中，我们利用 TiChart 尽可能充分地可视化用户的查询结果。数据可视化过程中的一项关键挑战源于用户任务的复杂性：任务可能会生成大量子任务，导致最终查询内容（例如属性和数据）存在不确定性。这大大增加了对结果数据进行可视化的复杂度。为应对这一挑战，TiChart 将基于规则的方法与大语言模型相结合，推荐合适的可视化类型。

5**用户友好的网页界面**。TiInsight 的用户界面提供了丰富的功能。用户可以灵活导入数据以进行探索。此外，直观的界面支持用户通过自然语言与数据交互。TiInsight 还包含一项书签功能，方便用户比较不同的数据探索结果。此外，TiInsight 支持通过网页用户界面在不同的大语言模型之间切换。

以下章节将简要介绍 TiInsight 的两个组件。更多技术细节和实验结果请参阅我们的研究论文 [6]。

3 分层数据上下文生成

分层数据上下文主要用于增强对数据库模式的理解，其中包括列摘要、表描述、表间关系以及完整的数据库摘要。这一设计源自 PingCAP 在概念验证阶段积累的经验。在 PingCAP 的数据分析业务中，

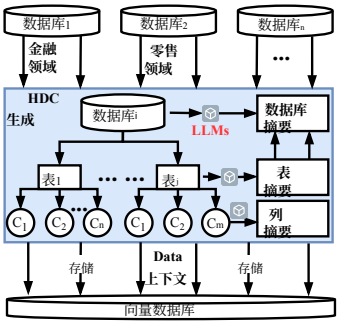


图 2：分层数据上下文概览。

数据分析师在接触陌生数据集时，首先会收集并整理相关的数据库模式信息，以便更好地理解数据库。我们借鉴了这一经验，并使用大语言模型将该过程实现了全自动化，如图 2 所示。**列摘要**。逐列进行摘要会由于反复调用大语言模型而导致较高的延迟。此外，为每一列单独使用大语言模型的效率较低，会造成大量令牌浪费并增加成本。为应对这些挑战，我们提出了一种并行分组方法，通过对表进行垂直分区来缓解这一问题。首先，HDC 通过垂直分区将列组织成固定的集合。这一限制主要源于这些大语言模型的上下文窗口限制。随后，在大语言模型启动列描述流程之前，会从线程池中为每个分组分配线程。模式为每一列提供的模式信息中可能包含注释。在完成上述流程后，我们会将列注释附加到每个列描述中。此外，对于每一列，我们会随机选择若干行（即本文中的三行）纳入列描述，以帮助大语言模型理解该列的值类型及其他相关信息。最后，大语言模型接收列内容，并动态构建思维链提示，以生成列摘要。

表描述。为了创建全面的表描述，我们需要收集更多信息，包括主键、关键属性、表类型、表实体和自然语言描述。具体而言，LLM 可能会识别出多个主键，因此我们会提示它选出最有可能的一个。该主键应能更好地体现表的重要性和唯一性。关键属性对于理解表的用途和内容至关重要。我们会提示 LLM 从本文所示的表中识别并列出最多五个关键属性。此外，我们将表类型归类为维度表、桥接表或事实表。这种分类对于 OLAP 分析尤为重要。最后，我们会识别该表所关注的主要实体。

表关系。我们提出了一种两阶段方法来识别表关系。为了确定特定表与其他表之间的参照完整性，在**第 1 阶段（即粗粒度搜索）**，我们使用指定的表描述，从向量数据库中检索相似_数量的表。在**第 2 阶段（即细粒度探索）**，我们将指定表以及第 1 阶段检索到的表输入大语言模型，通过思维链提示进行细粒度识别。

我们遍历每个表，以确定数据库中所有表之间的完整性关系。给定 n 个表，针对每个指定表探索所有表的时间复杂度为 $O(n^2)$。使用粗粒度搜索可降低这一复杂度，使时间复杂度达到 $O(n)$。

数据库摘要。为了便于在多个数据库中高效地进行数据探索，我们提出为每个数据库提取代表性**实体**，并基于这些实体生成**数据库摘要**。数据库由许多具有复杂关系的表组成。根据影响力最大化原则，与其他表关系数量更多的表在数据库中被认为更重要。因此，我们提出一种利用表间关系数量的实体提取方法。该方法从当前数据库中选出关系最多的前 N 个表。然后，利用这些表及其摘要构建提示，并让 LLM 推断其中的实体。最后，将推断出的实体保存到向量数据库中。我们使用带有思维链提示的 LLM，根据这些实体推断数据库摘要。

4 TiSQL：基于 HDC 的文本转SQL

我们提出了一种基于 MapReduce范式的模式过滤框架，用于筛选表和列。TiSQL 首先进行粗粒度搜索，根据澄清后的问题和余弦相似度，从向量数据库中检索排名前 N 的相关表。随后，TiSQL 执行细粒度过滤，排除无关的表和列。为避免关联的表和列摘要超出 LLM 的上下文限制，TiSQL 将这些表划分为更细致的相关表组。接下来，系统利用空闲线程，通过 MapReduce 处理相应的数据块。在映射阶段，系统根据当前组的表和列摘要以及澄清后的任务，动态构建提示。随后，结合思维链提示，使用 LLM 识别相关表和所需列。在归约阶段，我们对每个细粒度表组应用归约模式，合并检索到的表和列。TiSQL 同时利用 HDC 和 MapReduce 框架精简 LLM 提示，显著降低模式链接的复杂度。

通过上述流程生成的 SQL语句仍可能存在错误；因此，我们提出了一种自我修正链 .
从 LLM 生成 SQL语句后，首先将结果输入解释-修正流程，以识别潜在错误。*EXPLAIN* 语句是数据库中广泛使用的工具，无需执行 SQL语句即可反馈其中可能存在的问题。使用 *EXPLAIN* 语句细化 SQL语句后，某些错误可能只有在执行过程中才会显现；因此，我们将*EXECUTION* 语句集成到 TiSQL 修正链中。执行 SQL语句，并将错误信息中的反馈传回 LLM 进行修正。经过多轮自我修正链迭代后，即可生成最终的 SQL 输出。

5 演示场景

我们向参与者介绍了一款网络应用程序，并通过两个现实场景展示 TiInsight：金融数据集和鸟类数据集。

5.1 场景 1：图形用户界面

为了提升用户体验，TiInsight 提供了对话式图形用户界面，以促进用户-系统交互。当用户在浏览器中输入“https://www.tiinsight.chat/”时，页面会显示如图 3 所示的内容。界面包括 新建聊天 按钮（❶）、两个默认的 TiInsight 数据集（❷）、数据集探索界面（❸），以及左侧工具栏（❹），其中包含书签、API 文档和其他功能。在接下来的演示中，我们将使用 TiInsight 提供的两个默认数据集来展示系统的功能。首先，点击新建聊天 按钮并选择*Financial* 数据集，即可创建新的数据探索与分析会话。

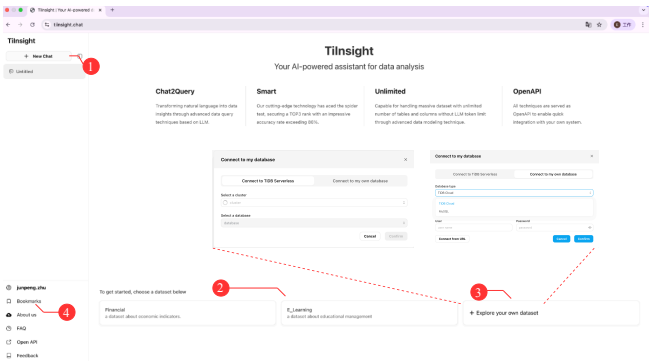


图 3：图形用户界面的示意图。

5.2 场景 2：金融数据集

我们使用自然语言问题“识别美联储加息的影响。”以及不完整的问题“油价变化？”作为演示示例。图 4 展示了用户界面的主要特征。

步骤 1：HDC 生成。首先，TiInsight 提供用户数据集的 HDC（❶），其中包括摘要、描述、关键词、表信息等内容。用户可以通过自然语言问题“快速了解此数据集”（❷）获取 HDC。

步骤 2：EDA 任务/目标。用户既可以在 对话框 （❸）中输入问题，也可以从 TiInsight 推荐的数据探索问题开始。对话框 功能支持用户在各种大语言模型之间无缝切换（❹）。接下来，TiInsight 利用当前数据集的 HDC 解读用户的问题，并细化探索任务。

步骤 3：EDA 结果。TiInsight 根据细化后的探索任务，TiSQL 工具生成相应的 SQL语句，同时 TiChart 选择最合适的图表类型，通过可视化呈现分析结果，如图 5-❶ 所示。TiChart 可能会为结果推荐多种可视化类型，用户可以根据需要在这些类型之间切换（❷）。对于每个探索结果，TiInsight 都提供“书签”按钮，用户可以将结果保存到书签中，以便进行比较（❸）。用户可以点击左侧工具栏中的书签 按钮，比较不同的 EDA 结果。此外，用户还可以点击分享 按钮（❹）来分享其分析结果。

5.3 场景3：鸟类数据集

在本次演示中，我们选择使用探索任务“按数量降序列出各测试结果及其数量”。

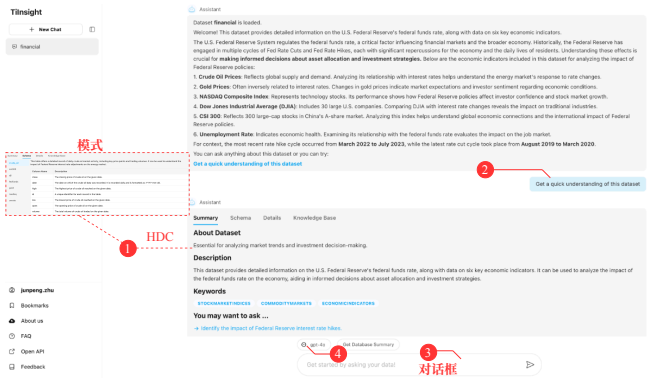


图4：金融数据集上的一项探索性数据分析任务示意图。



图5：金融数据集上的探索性数据分析结果示意图。

作为探索 TiInsight 功能的示例。若要更全面地了解其功能，请观看我们的视频了解更多详情。

6 结论

本文介绍了 TiInsight，这是一个基于 SQL 的自动化跨领域探索性数据分析系统。我们介绍了 TiInsight 的实现，并展示了其诸多新颖特性，包括用户友好界面、强大的分层数据上下文（即 HDC）生成、TiSQL、TiChart，以及高效的探索效率。

参考文献

[1] Doris Jung-Lin Lee、Dixin Tang 等。2021。Lux：为探索性数据框工作流提供始终在线的可视化推荐。*VLDB* 15，3（2021），727–738。[2] Jinyang Li、Binyuan Hui、Ge Qu、Jiaxi Yang、Binhua Li、Bowen Li、Bailin Wang、Bowen Qin、Ruiying Geng、Nan Huo 等。2024。大语言模型能否胜任数据库接口？面向大规模数据库落地文本转 SQL 的基准测试。*NIPS* 36（2024）。[3] Chao Zhang、Yuren Mao、Yijiang Fan、Yu Mi、Yunjun Gao、Lu Chen、Dongfang Lou 和 Jinshu Lin。2024。FinSQL：面向金融分析的模型无关、基于大语言模型的文本转 SQL 框架。*arXiv*预印本 *arXiv:2401.10506*（2024）。[4] Mengyu Zhou、Qingtao Li、Xinyi He、Yuejiang Li、Yibo Liu、Wei Ji、Shi Han、Yining Chen、Daxin Jiang 和 Dongmei Zhang。2021。Table2Charts：通过学习共享表表示来推荐图表。载于 *SIGKDD*。2389–2399。[5] Jun-Peng Zhu、Peng Cai、Boyan Niu、Zheming Ni、Kai Xu 等。2024。Chat2Query：基于大语言模型的零样本自动化探索性数据分析系统。载于*ICDE*。5429–5432。[6] Jun-Peng Zhu、Boyan Niu、Peng Cai、Zheming Ni、Jianwei Wan、Kai Xu 等。2025。借助大语言模型实现自动化跨领域探索性数据分析。*VLDB*基金会会刊18，12（2025年8月），5086–5099。https://doi.org/10.14778/3750601.3750629