



2023 CCF国际AIOps挑战赛决赛
暨 “大模型时代的AIOps” 研讨会

基于大语言模型的数据库异常诊断系统

*D-Bot: Database Diagnosis System using
Large Language Models*

周煊赫 清华大学计算机系

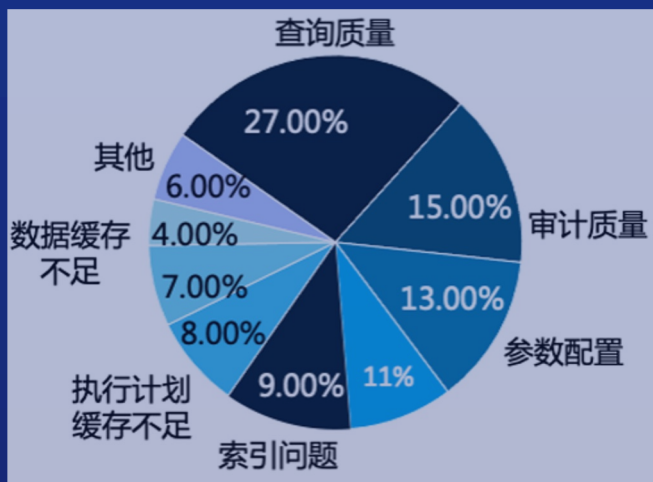
主办单位：中国计算机学会（CCF）、清华大学、中国建设银行股份有限公司、南开大学

承办单位：中国计算机学会互联网专委会、清华大学计算机科学与技术系、中国建设银行股份有限公司运营数据中心、南开大学软件学院、北京必示科技有限公司

赞助单位：华为技术有限公司、国网宁夏电力有限公司电力科学研究院、软通动力信息技术（集团）股份有限公司

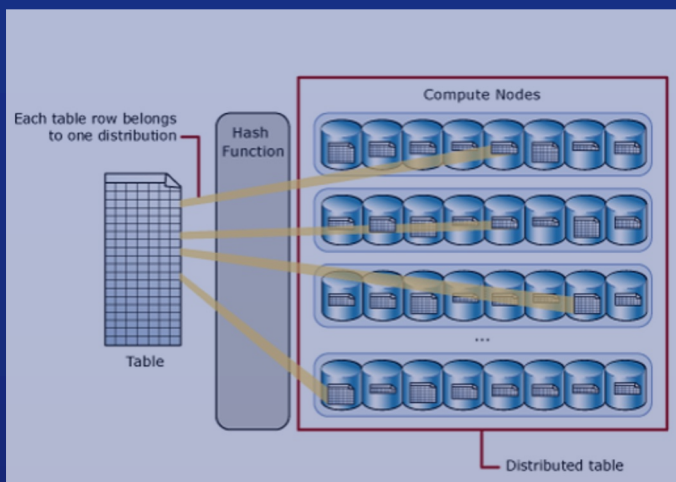
背景：数据库稳定性至关重要

调查数据显示，金融在数据库选型中，最看重的三个因素 **稳定性**、安全性、性能，分别占比80.4%、72.6%、59.6%



系统运维文档 **多**

异常种类繁多，运维调优手册等文档多达1w多页



云上运维压力 **大**

云数据库实例数巨大，运维工程师（DBA）短缺



复杂问题诊断 **难**

紧急问题多层连环，DBA难以及时响应

愿景：数据库智能运维

- **目标：**基于大模型的“智能运维助手”，学习人类运维经验、使用工具、诊断根因
- **好处：**节省人力；提升运维工具使用能力；提升运维知识泛化能力

典型异常&根因

查询执行慢

复杂查询结构

...

🔥 系统资源占满

负载并发冲突

大表数据插入

...

数据库宕机

长运行查询

死锁

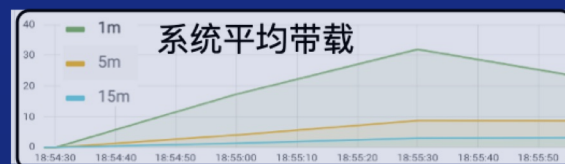
...

数据库崩溃

磁盘空间满

...

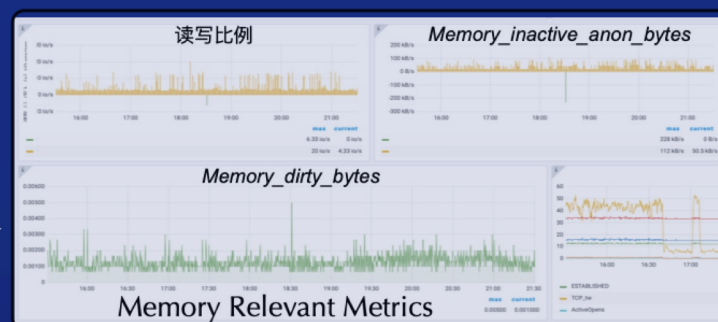
数据库宏观指标分析



DBA: 刚发生Critic级别Alert 🔥

Load-1min: 175% 超过阈值

数据库细粒度指标分析



DBA: 分析统计信息，发现大表插操作

```
"name": "large_data_insert",
"content": "Identify excessive inserted tuples in a table or query operations.",
"metrics": ["inserts", "query", "index_schema"],
"steps": "For each inserted table, if the count of inserted tuple is equal to or exceeds the {threshold}, it's flagged as a potential root cause."
```

分析结果

过度的内存使用可能
是对大表进行密集插入导致

需进一步研究划分
表数据是否能解决该异常

数据库会话分析

SQL	User	Calls
SELECT current_catalog, current...	r	624,630
SELECT CURRENT_CATALOG AS d...	r	370,849

SQL	User	Calls
INSERT INTO products VALUES	user1	624,630
INSERT INTO products VALUES	user1	370,849

DBA: 分析索引更新代价，推荐解决策略

```
"name": "large_data_insert_optimization",
"content": "Provide optimization guidance (e.g., table design and query refinement).",
"metrics": ["inserts", "live_tuples", "query", "index_schema"],
"steps": "Examine the insert operators that may cause high index update costs. And recommend table redesigns, such as index update or partitioning for manageable table"
```


愿景：数据库智能运维

- **目标：**基于大模型的“智能运维助手”，学习人类运维经验、使用工具、诊断根因
- **好处：**节省人力；提升运维工具使用能力；提升运维知识泛化能力

典型异常&根因

查询执行慢

复杂查询结构

🔥 系统资源占满

负载并发冲突

大表数据插入

数据库宕机

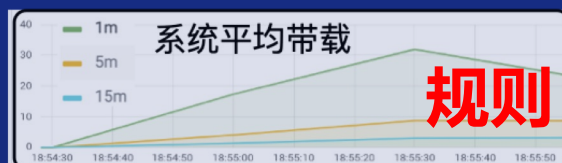
长运行查询

死锁

数据库崩溃

磁盘空间满

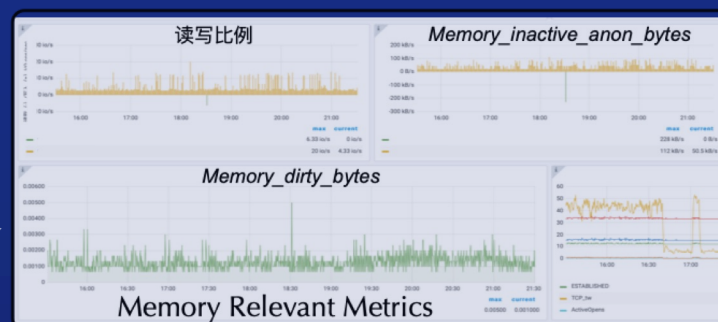
数据库宏观指标分析



DBA: 刚发生Critic级别Alert 🔥

Load-1min: 175% 超过阈值

数据库细粒度指标分析



DBA: 分析统计信息，发现大表插操作

```
"name": "large_data_insert",  
"content": "Identify excessive inserted tuples in a table or  
query operations.",  
"metrics": ["inserts", "query", "index_schema"],  
"steps": "For each inserted table, if the count of inserted  
tuple is equal to or exceeds the {threshold} value,  
flagged as a potential root cause."
```

知识

分析结果

过度的内存使用可能
是对大表进行密集插入导致

需进一步研究划分
表数据是否能解决该异常

数据库会话分析

SQL	User	Calls
SELECT current_catalog, current...	r	624,630
SELECT CURRENT_CATALOG AS d...	r	370,849

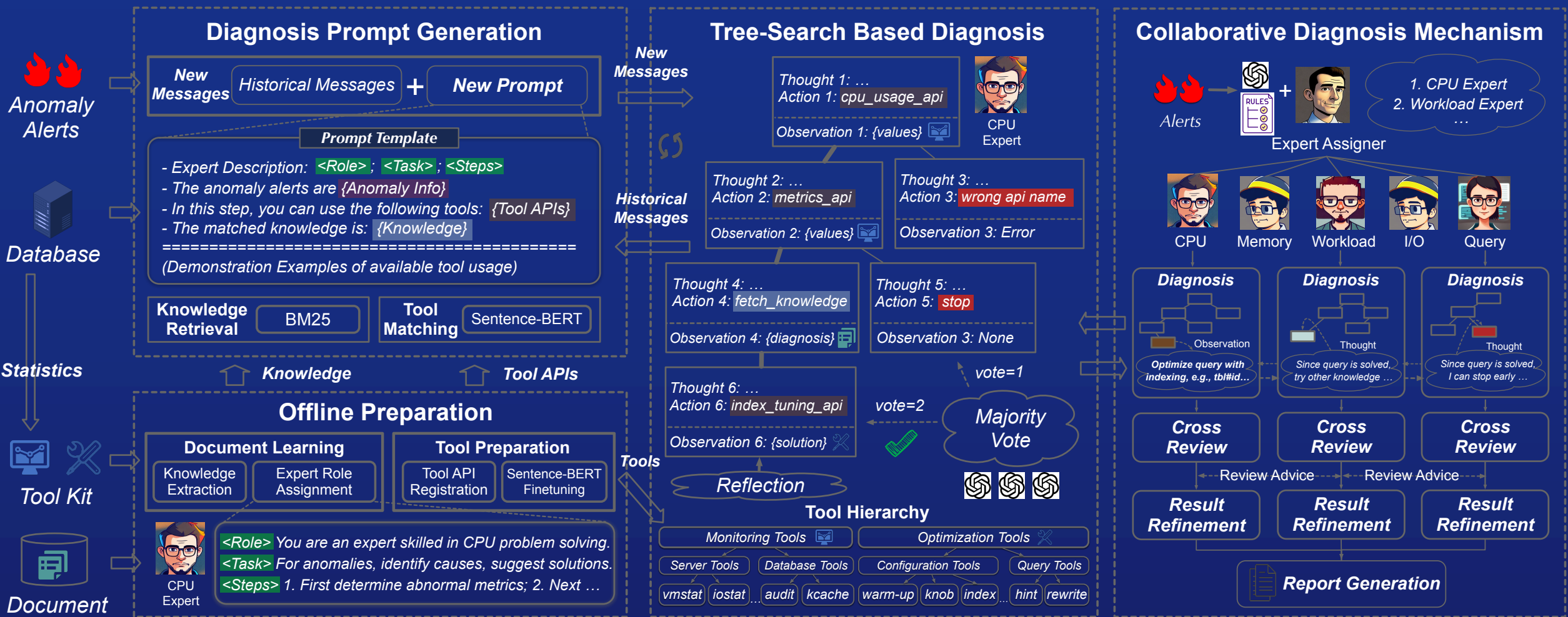
SQL	User	Calls
INSERT INTO products VALUES	user1	624,630
INSERT INTO products VALUES	user1	370,849

DBA: 分析索引更新代价，推荐解决策略

```
"name": "large_data_insert_optimization",  
"content": "Provide optimization guidance (e.g., table  
design and query refinement).",  
"metrics": ["inserts", "live_tuples", "query", "index_schema"],  
"steps": "Examine the insert operators that may cause high  
index update costs. And recommend table redesigns, such  
as index update or partitioning for manageable table"
```

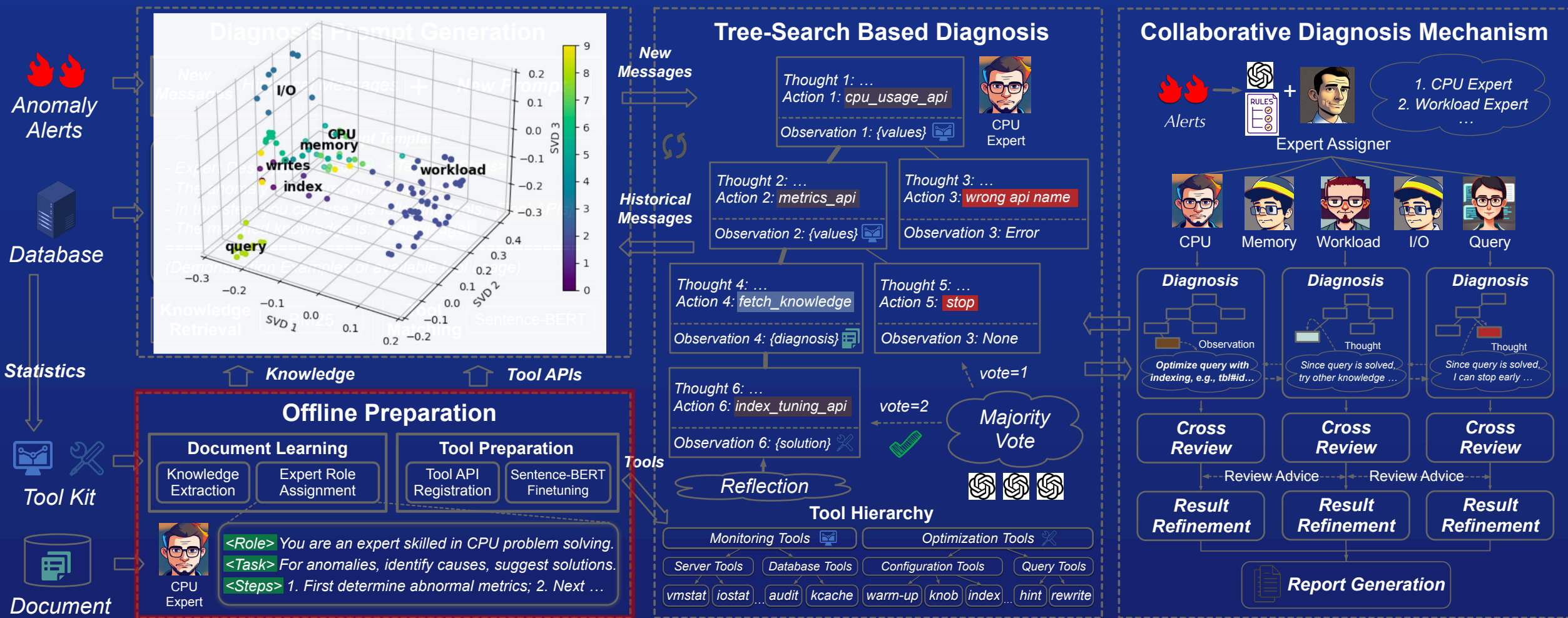
工具

D-BOT 技术架构



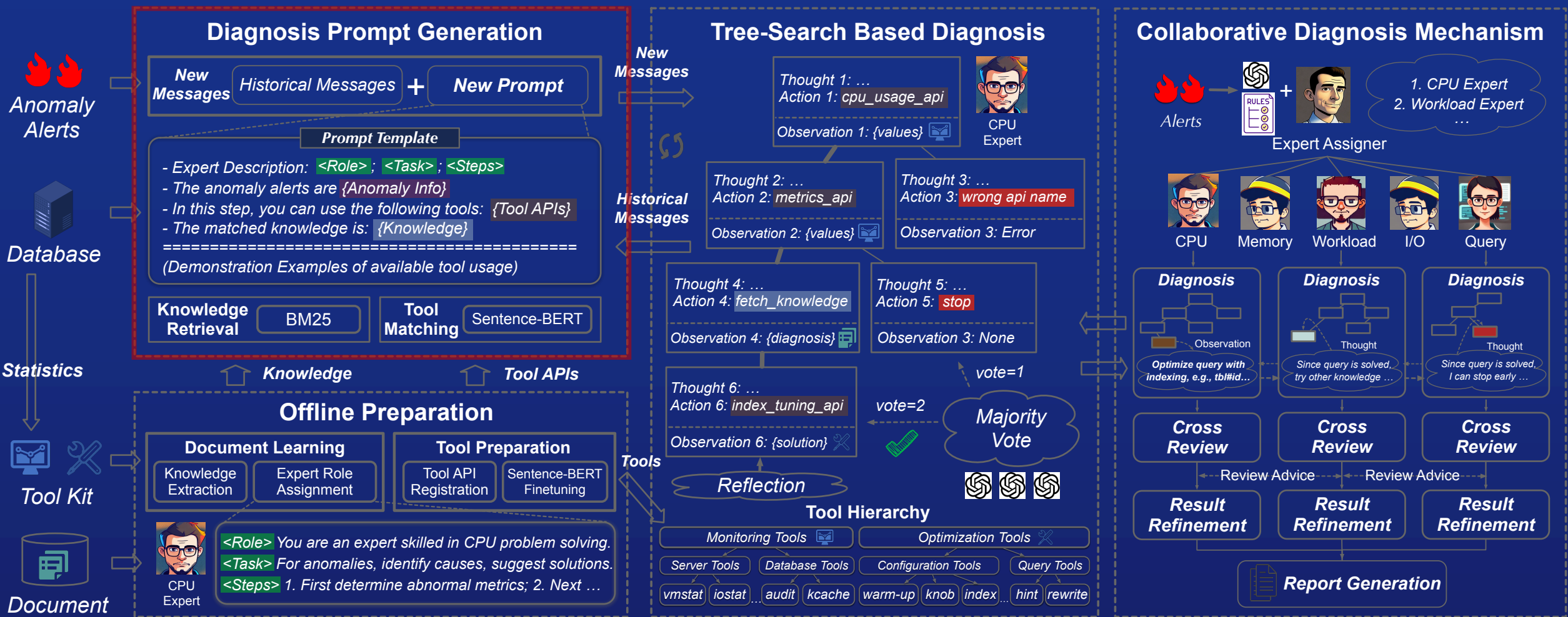
D-Bot: 离线知识和工具准备 + 面向诊断任务的提升工程 + 基于树搜索的根因分析 + 多专家协作诊断

D-BOT 技术架构



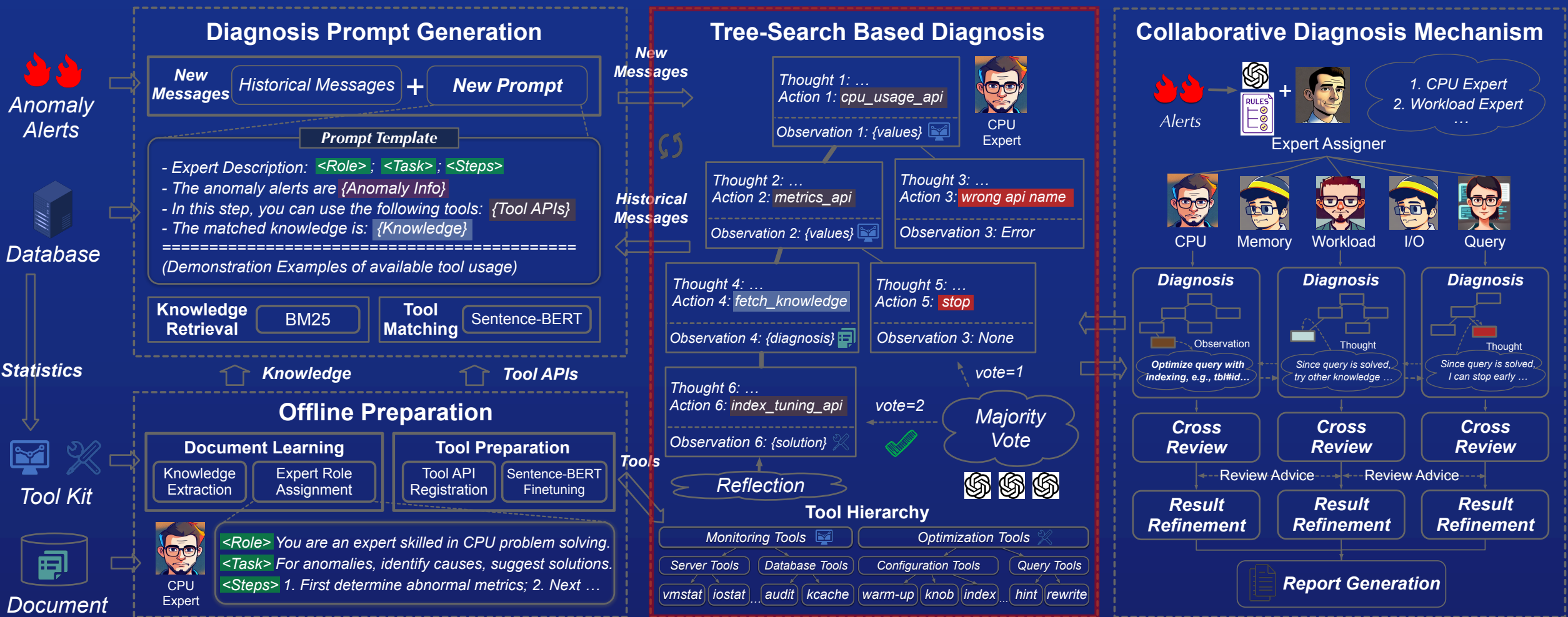
D-Bot: 离线知识和工具准备 + 面向诊断任务的提示工程 + 基于树搜索的根因分析 + 多专家协作诊断

D-BOT 技术架构



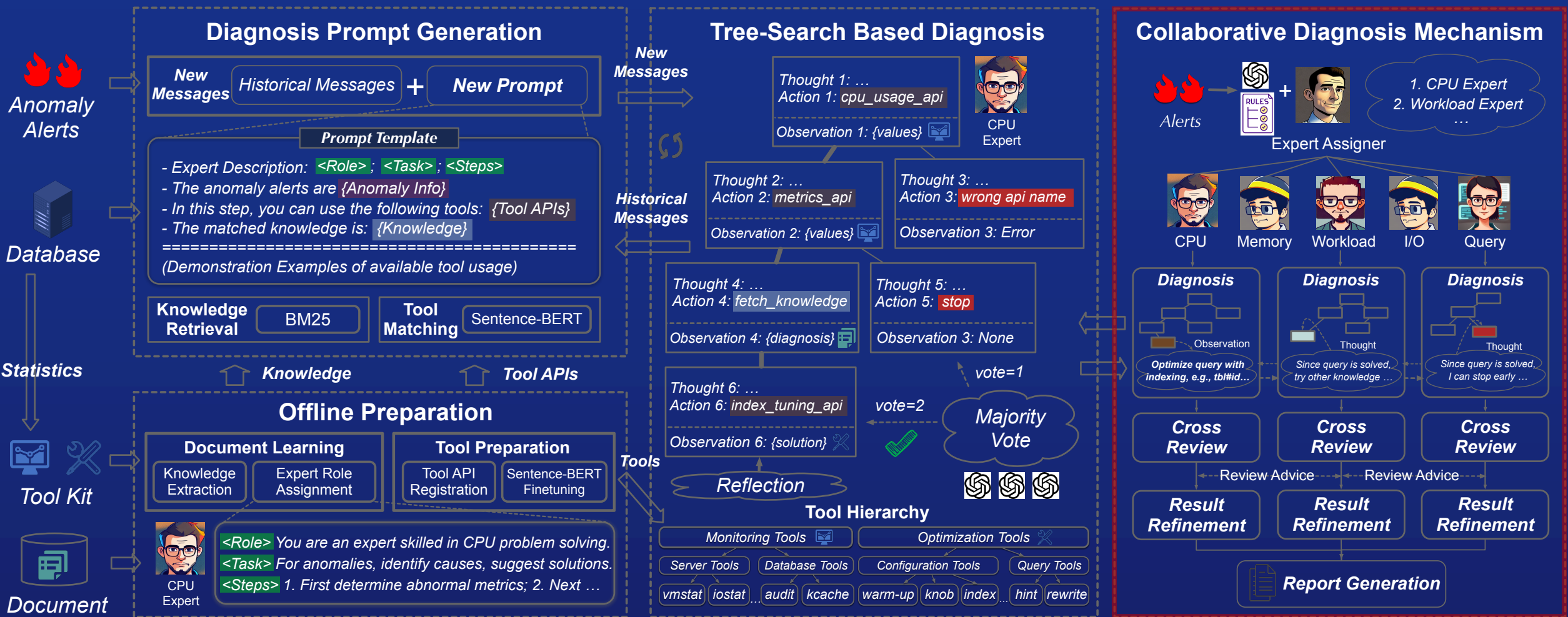
D-Bot: 离线知识和工具准备 + 面向诊断任务的提示工程 + 基于树搜索的根因分析 + 多专家协作诊断

D-BOT 技术架构



D-Bot: 离线知识和工具准备 + 面向诊断任务的提示工程 + 基于树搜索的根因分析 + 多专家协作诊断

D-BOT 技术架构



D-Bot: 离线知识和工具准备 + 面向诊断任务的提示工程 + 基于树搜索的单专家诊断 + 多专家协作诊断

Online Demo (<http://dbgpt.dbmind.cn>)



Anomaly Time: 🕒 Start Time - Start Time

Playback Animation 🔴

Reports are generated by [GPT4-0613](#)

[WARN🔥] NodeLoadHigh 2023-10-15 01:17:44

Playback

Report

node_entropy_available_bits (for cpu expert)

4,000
3,000
2,000
1,000
0

2023-10-14 23:00 2023-10-14 23:30 2023-10-15 00:00 2023-10-15 01:00

[WARN🔥] NodeLoadHigh 2023-10-15 01:14:15

Playback

Report

[WARN🔥] NodeLoadHigh 2023-10-14 23:29:48

Playback

Report

[WARN🔥] PostgresRestart 2023-10-15 15:57:53

Playback

Report

[CRIT🔥] PostgresDown 2023-10-15 15:49:58

Playback

Report

[WARN🔥] NodeFsSpaceFull 2023-10-15 15:36:32

Playback

Report

[WARN🔥] NodeLoadHigh 2023-10-15 15:34:21

Playback

Report

[WARN🔥] NodeLoadHigh 2023-10-15 15:08:58

Playback

Report

[WARN🔥] NodeOutOfMem 2023-10-15 14:54:43

Playback

Report

[WARN🔥] NodeLoadHigh 2023-10-15 14:49:13

Playback

Report

Analysis Report of Exceeding Maximum Standard Load on a Database Node

Anomaly Date	2023-10-15 01:17:44
Anomaly Description	<p>During the examination, it was noticed that from 09:17:49 to 09:23:49, on 15th October 2023, the standard load of the node 'ins' was excessively higher, about 159%, which exceeds its total capacity. Though the issue was resolved and the alert status suggested 'warn', it signifies a potential problem. It is recommended that this issue be addressed in a careful and precise manner to prevent future occurrences.</p>
Root Cause	<p>Refined Root Cause Analysis</p> <p>Root causes identified by CpuExpert</p> <ol style="list-style-type: none"> 1. Blocked Processes (node_procs_blocked) Blockages in processes are indicative of a contention for CPU resources. These halted operations are threads awaiting a turn for CPU execution. While a mix of waiting, running, and blocked processes is normal, a significant rise suggests an issue. In this scenario, the metrics show many processes vying for the CPU simultaneously (node_procs_blocked). As the scheduling policy of the Operating System determines the sequence and duration of processes, an imbalance can cause delays and inefficiencies. 2. High Entropy (node_entropy_available_bits) Entropy measures the level of system randomness, essential in cryptography, random number generation, and secure communications. A high entropy level, like observed in node_entropy_available_bits, suggests substantial random activity. The randomness could mean several things - inefficient random I/O operations, high context switching, or frequent cache misses. These conditions can cause extended wait times for processes, lead to idle CPUs, and ultimately high CPU usage. 3. Heavy Workload Query on 'table1' A query running frequently with a heavy workload can significantly increase CPU usage. A heavy workload query typically involves complex calculations, consumes considerable memory, and may require numerous I/O (Input/Output) operations. In this case, a heavy workload query running frequently on 'table1' appears to be straining the CPU. Such tasks could slow down other operations due to increased process waiting and contribute to a spike in entropy, leading to high CPU usage. <p>In conclusion, a mix of heavy processes, high entropy, and a heavy workload on 'table1' are driving the high CPU usage in our system.</p>

Online Demo (<http://dbgpt.dbmind.cn>)

Analysis Process Review

×

☒ Anomaly Analysis

☐ Round Table Discussion

☐ Presentation Of Report

1. After receiving an exception notification, the DBA analyzes the exception and assigns tasks to different colleagues. The colleagues who receive the task perform the analysis independently first.



RoleAssigner

17:30:45

During the examination, it was noticed that from 09:17:49 to 09:23:49, on 15th October 2023, the standard load of the node 'ins' was excessively higher, about 159%, which exceeds its total capacity. Though the issue was resolved and the alert status suggested 'warn', it signifies a potential problem. It is recommended that this issue be addressed in a careful and precise manner to prevent future occurrences.

17:30:49

Based on the task description, I decide to select the following experts to diagnose the problem:
CpuExpert
WorkloadExpert



CpuExpert

17:36:28

Thought: Now that I have obtained the start and end time of the anomaly, I need to check whether the CPU usage is abnormal during that time period.

▼ using Tools:

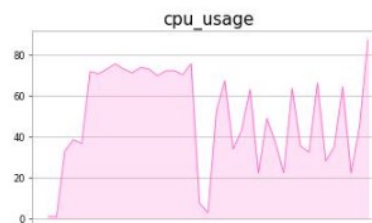
Command Name: whether_is_abnormal_metric

Arguments: {"start_time": "2023-10-15 09:17:49", "end_time": "2023-10-15 09:23:49", "metric_name": "cpu_usage"}

Command Status: Success

17:36:28

The metric cpu_usage is abnormal



17:36:28

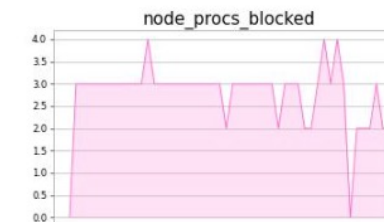
Thought: Given that the CPU usage is abnormal, I must diagnose the cause of the anomaly using the metrics, queries and knowledge gained from match_diagnose_knowledge.

▼ using Tools:

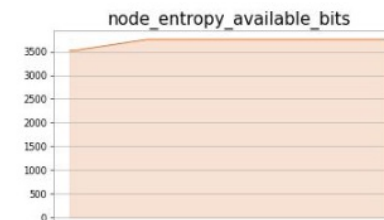


WorkloadExpert

1. node_procs_blocked contains abnormal patterns: the max value is 1.0, the min value is 0.0, the mean value is 0.58, the deviation value is 0.49, and the evenly_sampled_values are [0.0, 0.0, 0.0, 0.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0].



2. node_entropy_available_bits contains abnormal patterns: the max value is 3754.0, the min value is 3504.0, the mean value is 3704.29, the deviation value is 91.56, and the evenly_sampled_values are [3504.0, 3507.0, 3606.0, 3750.0, 3754.0, 3754.0, 3754.0, 3754.0, 3754.0, 3754.0, 3754.0, 3754.0].



The workload queries are:

```
{ "insert into 'table1' select generate_series(1,66),(SELECT substr(md5(random()::text), 1, 48)), now();"; 98}
```

17:36:28



2. After the colleagues receiving the task independently analyze the anomaly, they join the group for a roundtable discussion.



CpuExpert 17:38:00

Sure, I'll review the above diagnosis and provide some revisions:

1. Anomaly alert:

The alert notification suggested an unusually high CPU usage with the standard workload going above 100%.

2. Inspection of metrics:

Upon observing the metrics during this anomaly period, the 'cpu_usage' metric was confirmed to be abnormal. This signifies a potential issue with resource management.

3. Process and entropy analysis:

Closer examination uncovered abnormal patterns in 'node_procs_blocked' and 'node_entropy_available_bits' metrics. A high 'node_procs_blocked' implies that there might be certain processes which are experiencing delays due to high CPU usage. The 'node_entropy_available_bits' metric being high suggests that there is a lot of unpredictability and randomness in the activity during this time which could be adding to the CPU usage.

4. Workload review:

We also discovered a heavy workload query running with an unusually high frequency on 'table1' especially in the time of anomaly. This could be a significant reason behind the high CPU usage.

5. Diagnosis:

Therefore, the root cause of the high CPU usage can likely be traced back to process stalling and the frequently run heavy query on 'table1', compounded by a high level of random activity.

6. Recommendations:

As the root cause seems to stem from inefficient querying, the best course of action is to optimize the high frequency query. It could be done by:

- Checking the query to reduce any unnecessary load.
- Implementing efficient indexing on 'table1'
- Better structuring the query to reduce CPU strain.
- Evaluating and minimizing the frequency at which the query is run, if possible.

These steps could alleviate the high CPU usage and return the system performance to normal.

Please consider these recommendations and adjust your operations accordingly for an optimized system.



WorkloadExpert 17:38:57


Absolutely, here is the revised analysis:


1. Anomaly Alert


- The alert initiated due to an unusually high CPU utilization which effectively caused the standard workload to exceed 100%. It is essential to ascertain the reasons behind this abnormal usage of CPU resources.


2. Metrics Analysis























Analysis Process Review

✔ Anomaly Analysis

>

✔ Round Table Discussion

>

○ Presentation Of Report

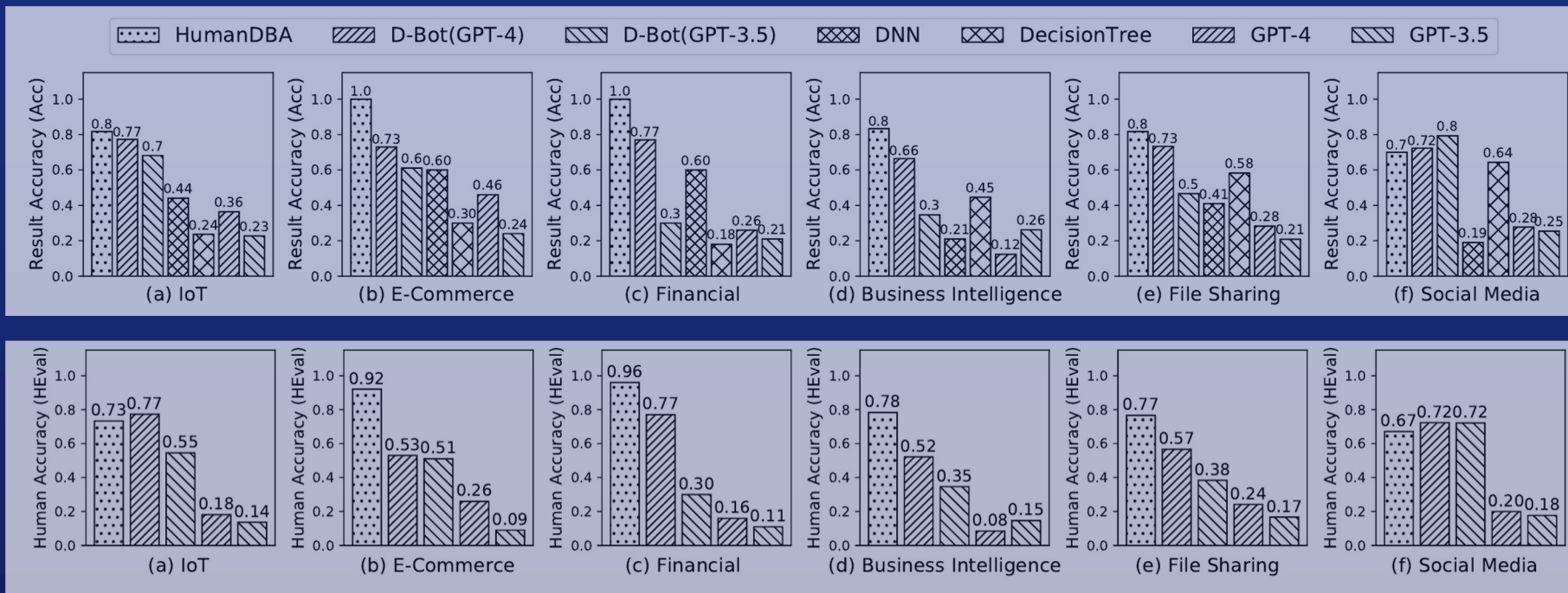
Analysis Report of Exceeding Maximum Standard Load on a Database Node

Anomaly Date	2023-10-15 01:17:44
Anomaly Description	During the examination, it was noticed that from 09:17:49 to 09:23:49, on 15th October 2023, the standard load of the node 'ins' was excessively higher, about 159%, which exceeds its total capacity. Though the issue was resolved and the alert status suggested 'warn', it signifies a potential problem. It is recommended that this issue be addressed in a careful and precise manner to prevent future occurrences.
Root Cause	<div>Refined Root Cause Analysis</div> <div>Root causes identified by CpuExpert</div> <div><div>1. Blocked Processes (node_procs_blocked)</div><div>Blockages in processes are indicative of a contention for CPU resources. These halted operations are threads awaiting a turn for CPU execution. While a mix of waiting, running, and blocked processes is normal, a significant rise suggests an issue. In this scenario, the metrics show many processes vying for the CPU simultaneously (node_procs_blocked). As the scheduling policy of the Operating System determines the sequence and duration of processes, an imbalance can cause delays and inefficiencies.</div></div> <div><div>2. High Entropy (node_entropy_available_bits)</div><div>Entropy measures the level of system randomness, essential in cryptography, random number generation, and secure communications. A high entropy level, like observed in node_entropy_available_bits, suggests substantial random activity. The randomness could mean several things - inefficient random I/O operations, high context switching, or frequent cache misses. These conditions can cause extended wait times for processes, lead to idle CPUs, and ultimately high CPU usage.</div></div> <div><div>3. Heavy Workload Query on 'table1'</div><div>A query running frequently with a heavy workload can significantly increase CPU usage. A heavy workload query typically involves complex calculations, consumes considerable memory, and may require numerous I/O (Input/Output) operations. In this case, a heavy workload query running frequently on 'table1' appears to be straining the CPU. Such tasks could slow down other operations due to increased process waiting and contribute to a spike in entropy, leading to high CPU usage.</div></div> <div>In conclusion, a mix of heavy processes, high entropy, and a heavy workload on 'table1' are driving the high CPU usage in our system.</div> <div>Root causes identified by WorkloadExpert</div> <div><div>1. High Frequency Workload Query</div><div>Investigation identifies a workload query running frequently in your database. This high frequency can cause performance bottlenecks by demanding high CPU usage, depleting resources, and causing anomalies.</div></div> <div><div>2. High CPU Usage</div><div>The high execution rate of the workload query results in increased CPU usage, which disrupts normal system functioning. This high CPU usage relates to the system anomalies observed.</div></div> <div><div>3. Increased 'node_procs_blocked' Metric</div><div>The 'node_procs_blocked' metric spikes during high CPU usage. This could mean that many processes are getting blocked due to the system overload.</div></div> <div><div>4. Decreased 'node_entropy_available_bits' Metric</div></div>

实验结果：总体表现（6个模拟场景的539个异常）

• 结果对比：D-Bot相对baselines取得较高的诊断准确率

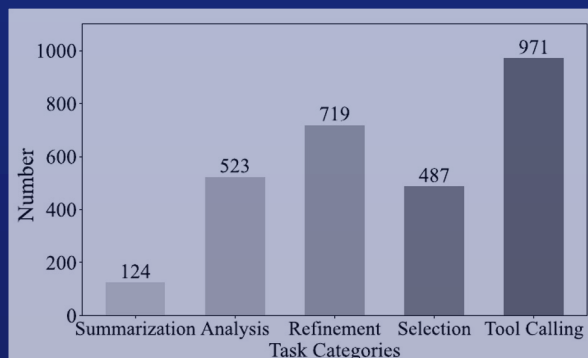
1. 推理能力更强，因此能够探索细粒度的系统运行指标/视图
2. 使用领域知识，避免“捏造根因”的情况
3. 学会使用工具，给出有实用价值的优化建议



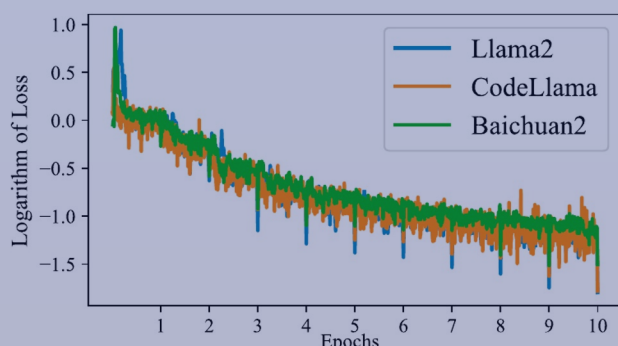
实验结果：本地微调（5个子任务的2819条微调样本）

- 结果对比：本地模型在多个测例下达到接近甚至超过GPT-4的诊断精度，但整体的诊断稳定性不如GPT-4：

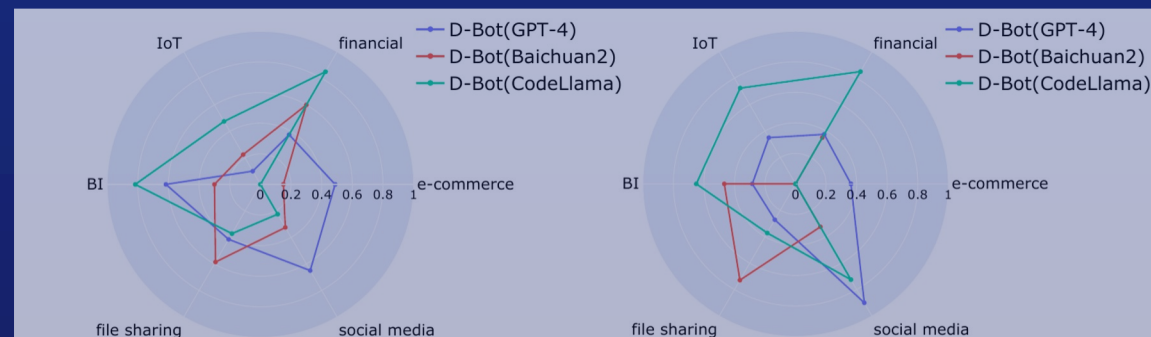
1. 训练Loss有明显下降，但是存在过拟合现象；
2. 本地模型的测试结果良好：基于微调样本学习到了工具调用、知识匹配、异常指标分析等的多方面能力；
3. 具体任务上的表现精度依赖微调样本的质量



(a) Subtasks in Finetuning Samples



(b) Training Loss for Localized LLMs



(c) Testing Results (Acc)

(d) Testing Results (HEval)



2023 CCF国际AIOps挑战赛决赛暨“大模型时代的AIOps”研讨会

THANKS

<https://github.com/TsinghuaDatabaseGroup/DB-GPT>

