

OpenDigger CLI 作品介绍

一 项目背景

近年来，GitHub 等代码托管平台已成为程序员、开发者之间交流和合作的重要平台。对于项目质量的评估，在平台上一一直是一个十分重要的话题。其中，项目的 GitHub 指标成为评估项目的重要因素之一。而 OpenDigger 作为开源生态数据分析的核心项目，负责与开源测量学相关的一切数据、指标、模型、算法等落库与实现，是一个具有共识性的实施标准工具集。目前，OpenDigger 提供的这些指标以 HTTPS URL 的形式提供 JSON 数据文件的查询结果，用户想查询某一个仓库在某一个指标的对应数据时，可以通过修改 URL 的方式请求得到。

现在，随着人们对命令行交互的需求越来越高，我们可以考虑使用命令行查看 OpenDigger 的指标结果。命令行交互的优势在于，它不依赖于图形用户界面（GUI），无需鼠标和其他外设，可以通过键盘快捷键和命令行输入与程序进行交互。这使得命令行交互成为一种高效、快捷的交互方式，特别在需要快速访问和操作多个文件、目录和系统层面时，在效率上有着优势。对很多开发者而言，命令行交互的方式相比修改 url 更加快捷，方便他们基于此做更多开发工作。

本项目为 OpenSODA 挑战赛参赛项目。旨在为 **OpenDigger** 提供的各类统计型和网络型的指标提供统一的 Terminal User Interface (TUI)。

本项目的目标在于，通过命令行交互方式，使 OpenDigger 提供一种方便、快捷的指标查询方式，用户可以利用 OpenDigger 提供的命令接口，查询 GitHub 上的指定仓库在各种指标上的表现，并在命令行界面上直接得到查询结果和相关信息或者得到一份下载文件。通过命令行交互的方式，用户不必登录任何平台或第三方网站，直接在控制台上查询和处理 GitHub 项目指标信息，这使得 OpenDigger 可以更快速、方便地提供 GitHub 项目指标查询服务，为项目评估和开发带来更高效的方式。

项目特性：

- * 🚀 基于 ChatGPT 的 **自然语言化查询** 功能
- * 🚀 针对不同类型指标的 **SQL 和 Cypher** 查询操作
- * 🚀 丰富的 **交互式 UI** 展示功能
- * 🚀 基于 Rust 开发，**支持单文件分发，无需安装依赖**

二 用户需求分析

(1) 用户类型

- 开发人员：有相关的计算机基础知识和编程能力，希望通过 OpenDigger 发掘 Github 上项目的指标信息。熟悉命令行界面，能够快速找到合适的交互方式。
- 项目管理人员：本身参与了自己项目的管理，希望通过 OpenDigger 找到自己参与的项目的指标信息，进行相关数据分析，得到项目优化方向。大部分有命令行使用经验。
- 普通用户：希望能够使用 OpenDigger 得到新的指标信息，用于非计算机相关的其他领域。工程经验可能不足，需要专业门槛较低的命令行使用方案。

(2) 用户需求

通过命令行工具，用户能够得到相应的查询结果，包括：

1. 查询**特定仓库**在**特定指标**上的数据
2. 查询**特定仓库**在**某一类指标**上的数据
3. 查询**特定仓库**在**特定自然月**上在**特定指标**上的数据
4. 查询**特定仓库**在**特定自然月**上在**某一类指标**上的数据
5. 查询**特定仓库**在**特定自然月份**上的**整体报告**
6. 查询**特定仓库**在**特定自然月份**上的在**某一类指标**上的报告
7. 查询**特定用户**在**特定指标**上的数据
8. 查询**特定用户**在**特定自然月**上在**特定指标**上的数据
9. 查询**特定用户**在**特定自然月份**上的**整体报告**

注意命令行交互虽然简洁，但有一定的专业性。我们的目标是让所有用户（包括计算机相关专业和非计算机相关专业的用户）能够轻松地使用产品，并获得愉悦的使用体验。

同时查询得到的整体报告应当有一定的可视化设计，以图表、图像等形式表达数据和信息，具有良好的可读性和表现力，节省用户的时间和精力，使用户可以更深入地探究数据并进行进一步分析。

三 产品功能设计

该命令行工具的功能设计包括五个模块：

功能 1：原始数据下载

下载特定仓库（或者特定用户）在特定指标（或者某一类指标）上的数据

命令格式如下：

```
$ opendigger-cli download <owner> <metric> [-o <output_file_name>]
```

其中 **owner** 的格式如下：

当 **owner** 为一个仓库时，例如：

```
$ opendigger-cli download "x-lab2017/open-digger" openrank -o openrank.csv
```

当 **owner** 为一个用户时，例如：

```
$ opendigger-cli download "frank-zsy" openrank -o openrank.csv
```

功能 2：关系型数据查询

通过 SQL 语句查询特定仓库（或者特定用户）在特定指标（或者某一类指标）上的数据

查询结果被格式化为表格数据，可以直接在命令行中展示/交互或者导出为 CSV 文件。对于某些特定形式的数据，提供折线图等可视化方式。

```
$ opendigger-cli sql <SQL Query> [-o <output_file_name>]
```

例如:

1. 列出所有可用的关系表格

```
$ opendigger-cli sql "SHOW TABLES"
```

2. 列出 `Issues` 表格中的所有列

```
$ opendigger-cli sql "SHOW COLUMNS FROM Issues"
```

3. 查询 `X-lab2017/open-digger` 仓库在 `Openrank` 指标上的数据

```
$ opendigger-cli sql "SELECT * FROM openrank WHERE repo = 'X-lab2017/open-digger'" -o openrank.csv
```

功能 2：数据查询报告

针对特定仓库或者特定用户，根据用户需求，给出一份在特定自然月上的整体报告，报告包含各类指标，并且以图表的形式展现

```
$ opendigger-cli report <owner> <month> [-o <output_file_name>] [--ui]
```

例如:

```
$ opendigger-cli report "X-lab2017/open-digger" "2020-01" -o report.pdf
```

功能 3：图数据查询

注意指标中有一部分本身就是图状数据，例如开发者之间的网络、仓库之间的网络，对于这部分数据，我们考虑使用 Cypher 查询图形数据

```
$ opendigger-cli cypher <Cypher Query> [-o <output_file_name>] [--ui]
```

例如:

```
$ opendigger-cli cypher "MATCH (X-lab2017/open-digger)"
```

功能 4: AI 交互式查询

我们关注普通用户的需求，对于不了解 SQL/Cypher 的用户，我们考虑在命令行工具中结合 ChatGPT 相关功能。用户可以描述自己的需求，由我们帮助用户生成 SQL/Cypher 查询语句。

(需配置 `OPENAI_API_KEY` 环境变量)

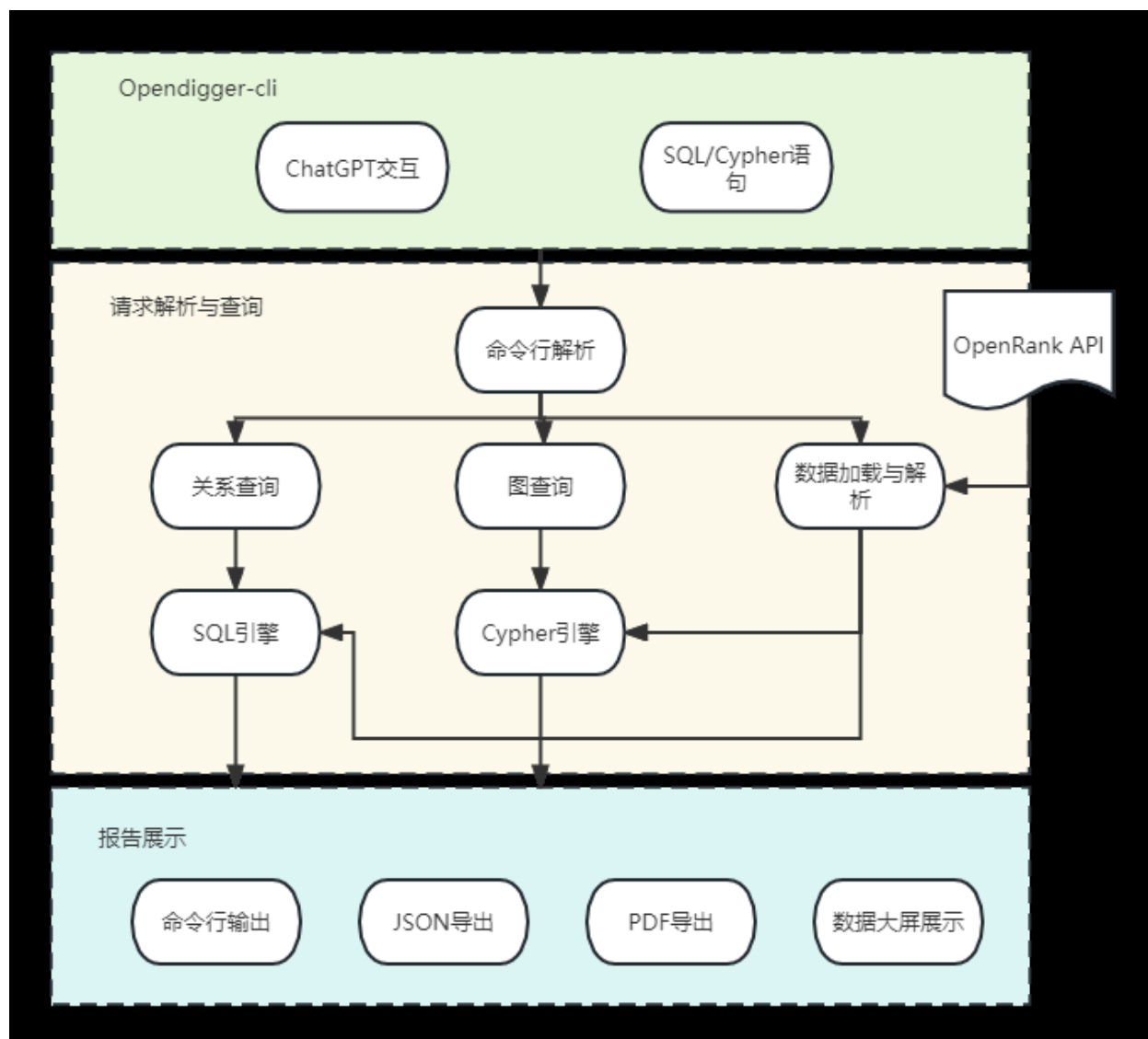
```
$ opendigger-cli chat <description> [-o <output_file_name>] [--ui]
```

例如：

```
$ opendigger-cli chat "仓库 'x-lab2017/open-digger', 'vuejs/core' stars 数量的变化，查询结果保留仓库名称"
```

四 产品架构、技术方案和项目构建

1. 产品架构



2.技术方案

1. 开发语言：Rust

使用 Rust 作为开发语言。选择 Rust 的理由是，Rust 是一种高性能、可靠、安全的系统编程语言，语言支持出色的并发和内存管理能力。同时，支持跨平台的交叉编译，编译产物为单二进制程序，方便分发。

2. ChatGPT 支持

支持 ChatGPT 生成查询语句。使用该技术是为了方便对 SQL/Cypher 语言不了解的用户使用该 CLI 工具，降低使用成本。

3. TUI 交互

TUI 界面以文本形式呈现，相比于 GUI 更轻量级，易于修改和维护。

4. DevOps 自动构建

采用 Github Actions 支持包括编译代码、运行测试、构建和部署应用程序，自动化构建可以显著提高应用程序的构建速度和开发效率。

3.项目构建

本项目在 Github Release 提供二进制文件 [ [链接](#)]。

若期望手动构建，可采用，需要 Nightly 版本的 Rust。

```
cargo install --git https://github.com/DarkHighness/opendigger-cli
```

请保证 `~/.cargo/bin` 在 `PATH` 环境变量中。

或手动克隆仓库并构建：

```
git clone https://github.com/DarkHighness/opendigger-cli
cd opendigger-cli
cargo build --release
```

构建完成后，可在 `target/release` 目录下找到可执行文件。

五 项目排期

- ✓ 2023 年 5 月 1 日前：开会，分工，完成该项目的方案设计和前期准备
- ✓ 2023 年 5 月 5 日前：在代码中完成基本数据下载
- ✓ 2023 年 5 月 10 日前：完成基本数据查询
- ✓ 2023 年 5 月 15 日前：完成 ChatGPT 支持查询语句生成
- ✓ 2023 年 5 月 22 日前：完成复赛方案提交和 Demo 演示

- ✓ 2023 年 6 月 5 日前：完成图数据查询
- ✓ 2023 年 6 月 15 日前：完成查询报告的可视化
- ✓ 2023 年 6 月 25 日前：完成项目测试和调整完善
- ✓ 2023 年 7 月 10 日前：完成产品说明书撰写，介绍如何安装和二次开发
- ✓ 2023 年 7 月 25 日前：完成决赛提交需要的完整内容，使用在线视频进行项目演示

六 产品 Demo

视频演示B站链接：

[OpenSODA-T2-决赛演示视频-SOLE回收站](#)

命令演示图文

基本帮助命令：

```
$ opendigger-cli help
```

```
> opendigger-cli help
Usage: opendigger-cli <COMMAND>

Commands:
  download  Download data from the API
  sql       Query data with sql
  chat      ChatGPT yes!
  report    Generate a report
  help      Print this message or the help of the given subcommand(s)

Options:
  -h, --help    Print help
  -V, --version  Print version
```

下载 [X-lab2017/open-digger](#) 仓库在 [Openrank](#) 指标上的数据：

```
$ opendigger-cli download "X-lab2017/open-digger" "openrank" -o openrank.json
```

```
> opendigger-cli download "X-lab2017/open-digger" "openrank" -o openrank.json
> head openrank.json
{"2020-08":3.46,"2020-09":3.7,"2020-10":4.28,"2020-11":5.2,"2020-12":7.82,"2021-01":7.29,"2021-02":4.45,"2021-03":2.95,"2021-04":3.36,"2021-05":3.1,"2021-06":4.78,"2021-07":3.59,"2021-08":2.56,"2021-09":2.18,"2021-10":2,"2021-11":1.88,"2021-12":2.3,"2022-01":3.52,"2022-02":3.55,"2022-03":4.49,"2022-04":2.69,"2022-05":2.88,"2022-06":5.59,"2022-07":6.47,"2022-08":6.01,"2022-09":7.16,"2022-10":9.32,"2022-11":11.82,"2022-12":15.16,"2023-01":15.38,"2023-02":32.22,"2023-03":16.17,"2023-04":13.55,"2023-05":13.42,"2023-06":10.05,"2023-07":10.56,"2023-08":9.83,"2023-09":8.59,"2021-10-row":1.85}
```

查询所有可用的关系表：

```
$ opendigger-cli sql "SHOW TABLES"
```

```

Table
Tables
  name
> Activity
  Attention
  Openrank
  ActiveDatesAndTimes
  Stars
  TechnicalFork
  Participants
  NewContributors
  NewContributorsDetail

Press <Up> or <Down> to navigate, <q> to quit

```

查询 `Issues` 表中的所有列：

```
$ opendigger-cli sql "SHOW COLUMNS FROM Issues"
```

```

Table
Columns
  Column                                Type
> name                                TEXT
  month                                TEXT
  new                                  INT
  closed                              INT
  comments                             INT

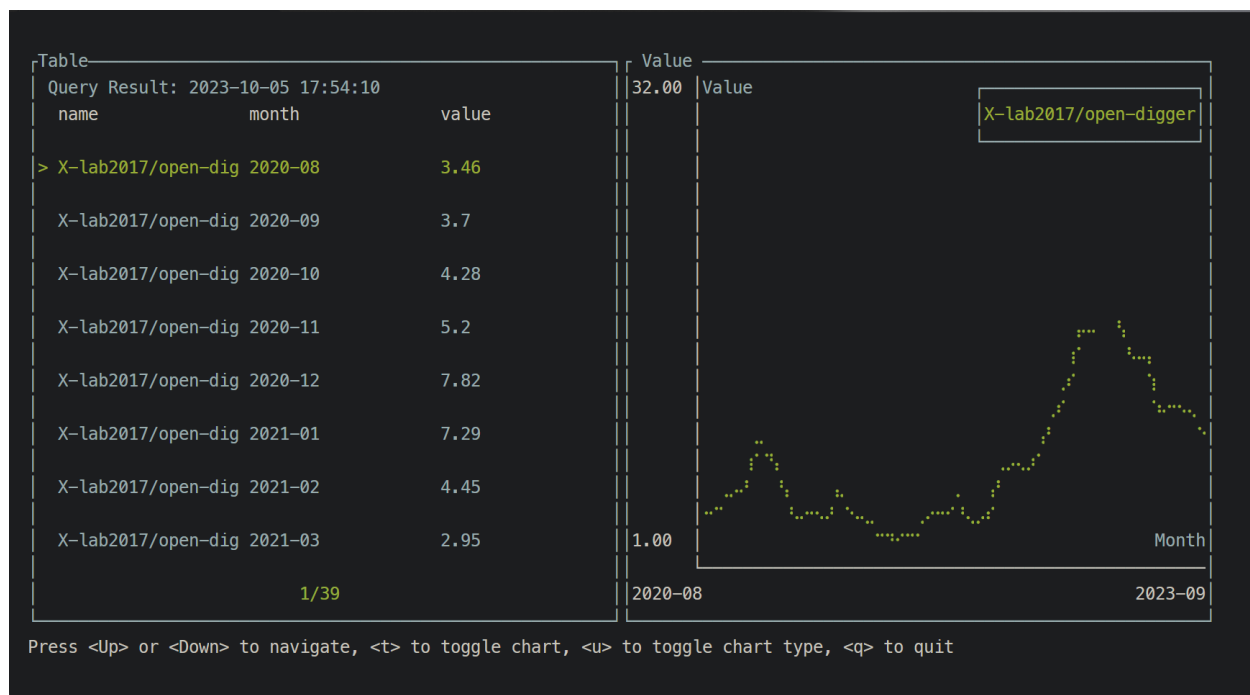
1/5

Press <Up> or <Down> to navigate, <q> to quit

```

查询 `X-lab2017/open-digger` 仓库在 `Openrank` 指标上的数据：

```
$ opendigger-cli sql "SELECT * FROM Openrank WHERE name = 'X-lab2017/open-digger'"
```



查询 `X-lab2017/open-digger` 仓库在 `Openrank` 指标上的数据，并将结果写入文件：

```
$ opendigger-cli sql "SELECT * FROM Openrank WHERE name = 'X-lab2017/open-digger'" -o openrank.csv > /dev/null
```

```
> opendigger-cli sql "SELECT * FROM Openrank WHERE name = 'X-lab2017/open-digger' ORDER BY value DESC" -o openrank.csv > /dev/null
> head openrank.csv
name, month, value
X-lab2017/open-digger, 2023-02, 32.22
X-lab2017/open-digger, 2023-03, 16.17
X-lab2017/open-digger, 2023-01, 15.38
X-lab2017/open-digger, 2022-12, 15.16
X-lab2017/open-digger, 2023-04, 13.55
X-lab2017/open-digger, 2023-05, 13.42
X-lab2017/open-digger, 2022-11, 11.82
X-lab2017/open-digger, 2023-07, 10.56
X-lab2017/open-digger, 2023-06, 10.05
```

查询 `X-lab2017/open-digger` 和 `vuejs/core` 仓库在 `Stars` 指标上的数据, 根据 `month` 进行连接：

```
$ opendigger-cli sql "SELECT * FROM Stars a LEFT JOIN Stars b ON a.month = b.month WHERE a.name = 'X-lab2017/open-digger' AND b.name = 'vuejs/core'"
```


Table					
Query Result: 2023-10-05 17:59:16					
name	month	value	name	month	value
> X-lab2017/open-digger	2020-08	10	vuejs/core	2020-08	781
X-lab2017/open-digger	2020-09	13	vuejs/core	2020-09	2938
X-lab2017/open-digger	2020-10	10	vuejs/core	2020-10	736
X-lab2017/open-digger	2020-11	4	vuejs/core	2020-11	557
X-lab2017/open-digger	2020-12	4	vuejs/core	2020-12	711
X-lab2017/open-digger	2021-01	1	vuejs/core	2021-01	625
X-lab2017/open-digger	2021-02	7	vuejs/core	2021-02	649
X-lab2017/open-digger	2021-03	3	vuejs/core	2021-03	595
X-lab2017/open-digger	2021-04	6	vuejs/core	2021-04	647
X-lab2017/open-digger	2021-05	4	vuejs/core	2021-05	603
X-lab2017/open-digger	2021-06	5	vuejs/core	2021-06	673
X-lab2017/open-digger	2021-07	4	vuejs/core	2021-07	679
X-lab2017/open-digger	2021-08	1	vuejs/core	2021-08	1361

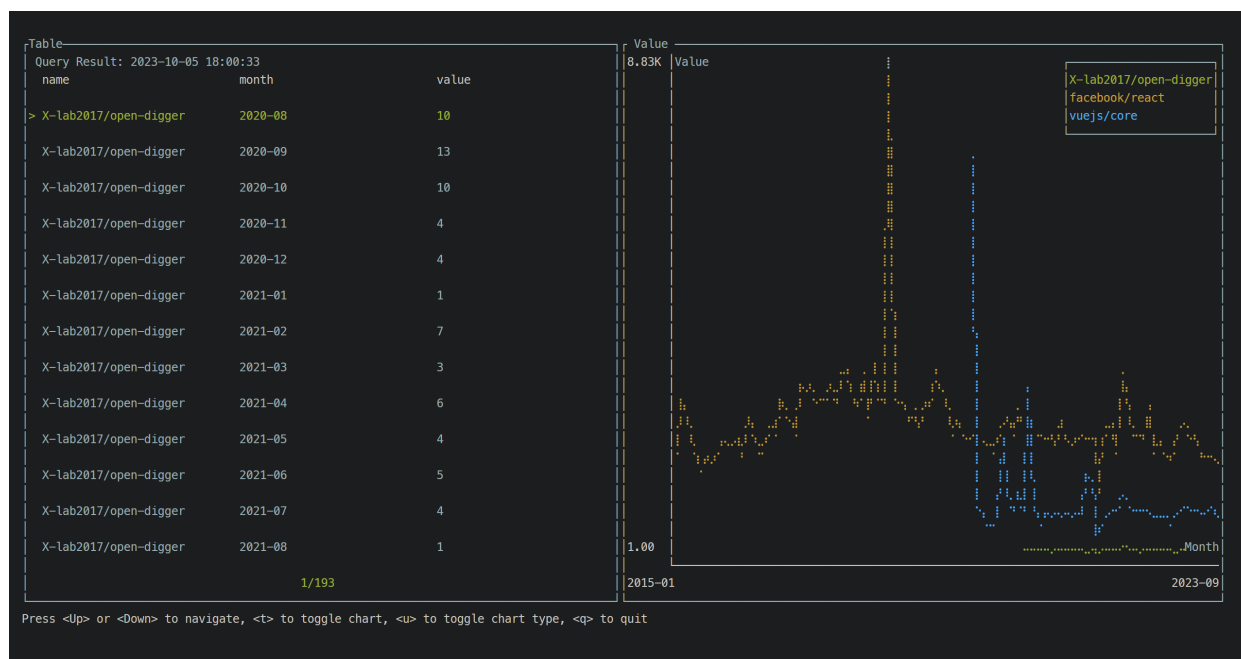
1/38

Press <Up> or <Down> to navigate, <q> to quit

查询 X-lab2017/open-digger , facebook/react' 和 vuejs/core 仓库在 Stars` 指标上的数据, 交互式 UI 展示:

```
$ opendigger-cli sql "SELECT * FROM Stars WHERE name = 'X-lab2017/open-digger' OR name = 'facebook/react' OR name = 'vuejs/core'" --ui
```

(支持切换图表类型: Treading (历史数据的前缀和) 和 Value (原始数据))



图查询: 查询用户'will-ww'在repo='X-lab2017/open-digger'上的'Developer network'指标数据, 使用TUI展示数据值。

```
$ opendigger-cli cypher "MATCH (n:Node {value: 'will-ww'}) WHERE n.owner = 'X-lab2017/open-digger' AND n.metric = 'Developer network' RETURN n"
```

```
> opendigger-cli cypher "MATCH (n:Node {value: 'will-ww'}) WHERE n.owner = 'X-lab2017/open-digger' AND n.metric = 'Developer network' RETURN n"
```

name
3.15

图查询：查询用户'Zzzzzhuzhiwei'在repo='X-lab2017/open-digger'上的'Developer network'上的所有有合作关系或者是有相关性的用户。

```
$ opendigger-cli cypher "MATCH (n:Node {value: 'Zzzzzhuzhiwei'})-[]-(neighbor) where n.owner='X-lab2017/open-digger' and n.metric='Developer network' RETURN neighbor"
```

```
> opendigger-cli cypher "MATCH (n:Node {value: 'will-ww'}) WHERE n.owner = 'X-lab2017/open-digger' AND n.metric = 'Developer network' RETURN n"
```

name
3.15

图查询：查询用户'Zzzzzhuzhiwei'在repo='X-lab2017/open-digger'上的'Developer network'上的合作最紧密的用户。

```
$ opendigger-cli cypher "MATCH (n:Node {value: 'Zzzzzhuzhiwei'})-[]->(neighbor) WHERE n.owner='X-lab2017/open-digger' and n.metric='Developer network' WITH neighbor ORDER BY neighbor.metric DESC LIMIT 1 RETURN neighbor"
```

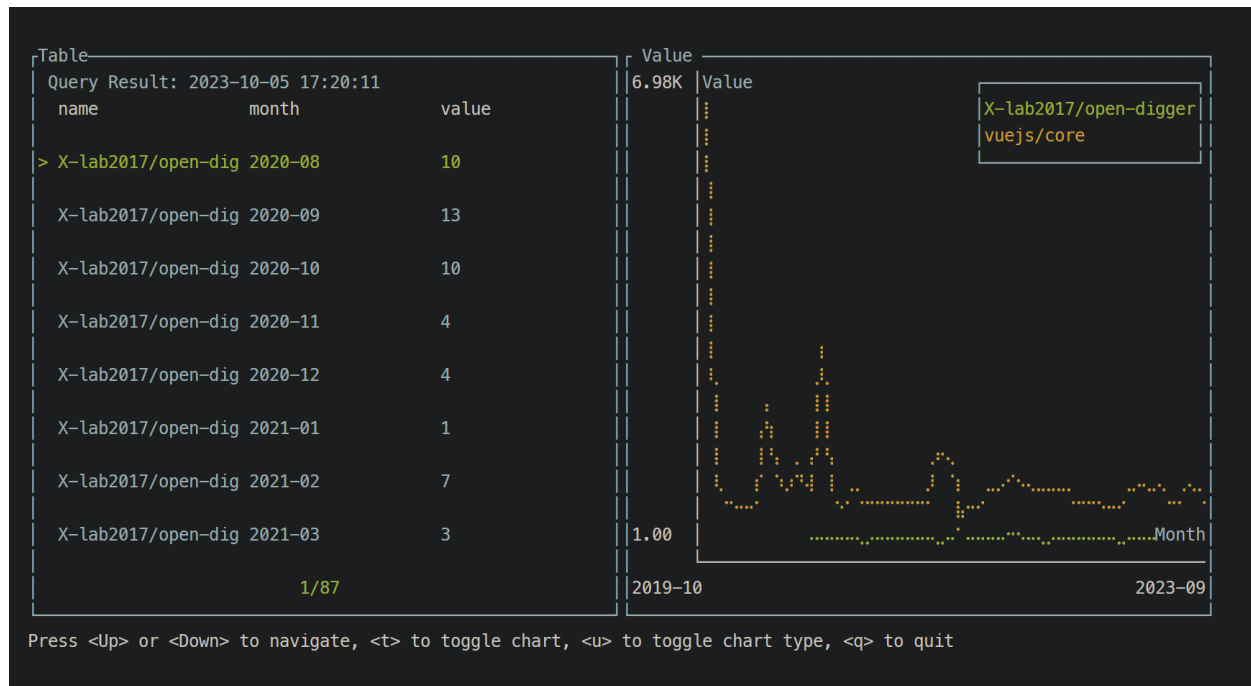
```
> opendigger-cli cypher "MATCH (n:Node {value: 'Zzzzzhuzhiwei'})-[]-(neighbor) where n.owner='X-lab2017/open-digger' and n.metric='Developer network' RETURN neighbor"
```

name
Lllllolita
yanchaomei
tisonkun
dwxa
WillemJiang
snyk-bot
tc2000731
xiaoya-yaya
stevending1st
bifenglin

ChatGPT 查询：

```
$ opendigger-cli chat "仓库 'x-lab2017/open-digger', 'vuejs/core' stars 数量的变化, 包含仓库名称" --ui
```

(支持切换图表类型: **Treading** (历史数据的前缀和) 和 **Value** (原始数据))



整体报告：查询仓库'x-lab2017/open-digger'的整体报告。

```
$ opendigger-cli report "x-lab2017/open-digger"
```

