葛进_周文格_开源课程期末作业

葛进 51215903084

周文格 51215903071

任务分工

葛进：Task 2 项目的历史轨迹分析, Task 3 结合期中分析的归档项目，对比分析活跃/归档项目

周文格：Task 1 项目的基本背景和发展历程介绍

Task 1 项目的基本背景和发展历程介绍

技术类型

Atom是由GitHub开发的自由及开放源代码的**文字与代码编辑器**，支持macOS、Windows和Linux操作系统，支持Node.js所写的插件，并内置由Github提供的Git版本控制系统。多数的延伸包皆为开放源代码许可，并由社群建置与维护。Atom基于使用Chromium和Node.js的跨平台应用框架**Electron**（最初名为Atom Shell），并使用CoffeeScript和Less撰写。Atom也可当作IDE使用。被它的开发者称为"21 世纪的"高自定义性"文本编辑器（*hackable text editor for the 21st Century*）"。自2014年5月6日起，Atom的核心程序、包管理器以及Atom基于Chromium的桌面程序框架皆使用MIT许可协议发布。 在2022年6月8号，GitHub正式宣布在2022年12月15日关闭Atom，并存档其存储库。

版本发布历史

- v0.26.0 Iron 04 Dec 2013
- … …
- v1.0.0 Atom 1.0 25 Jun 2015
- … …
- v1.1.0 30 Oct 2015
- … …
- v1.60.0 08 Mar 2022
- v1.61.0-beta0 (pre-release) 08 Mar 2022
- -(Atom and all repositories under Atom will be archived on December 15, 2022.)-

主要贡献者的构成

- kevinsawicki 8655 commits 来自 Sammamish, WA
- benogle 2071 commits
- maxbrunsfeld 1703 commits
- 50Wliu 941 commits

- ... ...

CI/CD的使用

- Continuous Integration（持续集成），假设一个应用程序，其代码存储在 GitLab 的 Git 仓库中。开发人员每天都要多次推送代码更改。对于每次向仓库的推送，你都可以创建一组脚本来自动构建和测试你的应用程序，从而减少了向应用程序引入错误的机会。这种做法称为持续集成，对于提交给应用程序（甚至是开发分支）的每项更改，它都会自动连续进行构建和测试，以确保所引入的更改通过你为应用程序建立的所有测试，准则和代码合规性标准。
- Continuous Delivery（持续交付），持续交付是超越持续集成的更进一步的操作。应用程序不仅会在推送到代码库的每次代码更改时进行构建和测试，而且，尽管部署是手动触发的，但作为一个附加步骤，它也可以连续部署。此方法可确保自动检查代码，但需要人工干预才能从策略上手动触发以必输此次变更。

本项目中没有使用到这个方法。

其他有价值的信息

为什么Atom要被停止维护了?

以下引用自Sunsetting Atom：

> As new cloud-based tools have emerged and evolved over the years, Atom community involvement has declined significantly. As a result, we've decided to sunset Atom so we can focus on enhancing the developer experience in the cloud with GitHub Codespaces.

归结原因是随着新的基于云的工具出现和发展，Atom社区的参与度显著下降；Microsoft官方决定专注于VS Code和GitHub Codespaces的开发。

Atom因为是基于Electron开发的编辑器，在流畅程度上与Sublime等竞品差距较大；而用户可定制化程度高的特点又被VS Code大致覆盖了；在Codespaces和VS Code都在发展的情况下，Atom已经失去了存在的必要性。

Task 2 项目的历史轨迹分析

具体见`final.ipynb`。

Task 3 结合期中分析的归档项目，对比分析活跃/归档项目

我们小组期中分析的归档项目是Kotlin/anko。

项目基础数据的变化趋势

`anko`：

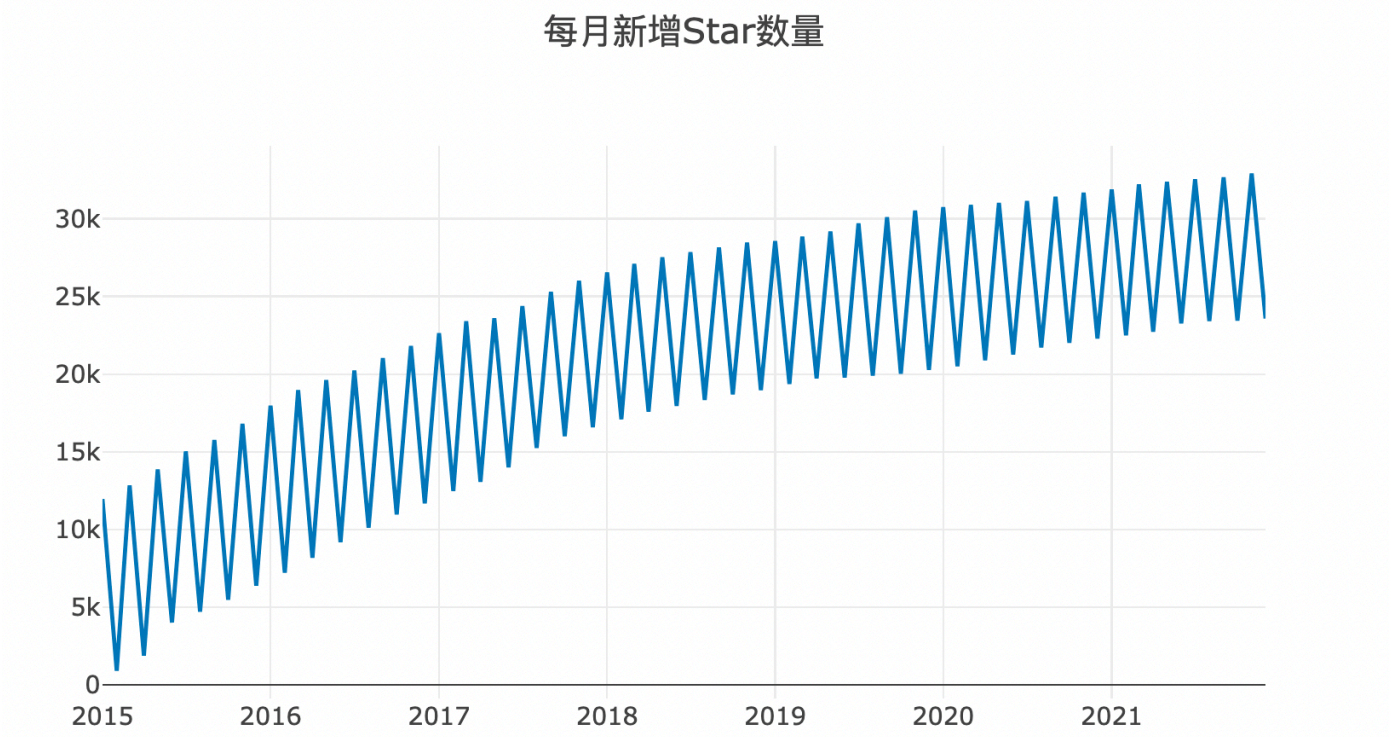
每月新增Star数量

每月新增Fork数量

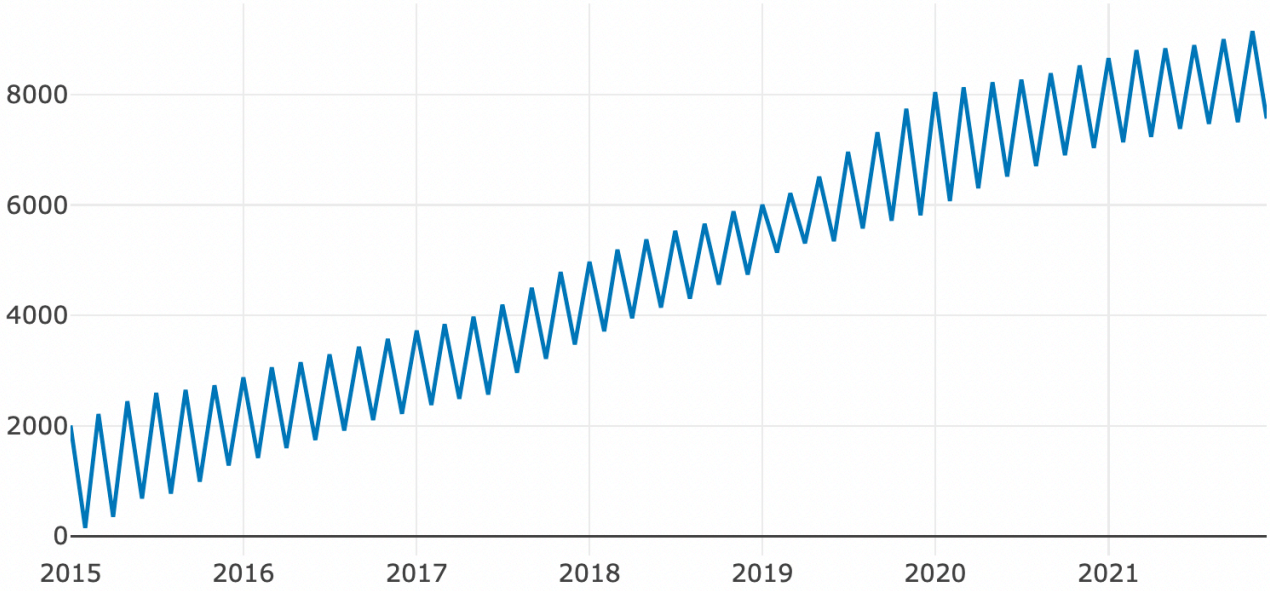
每月打开Issue数量


每月关闭Issue数量
20
15
10
5
0
Jul 2015
Jan 2016
Jul 2016
Jan 2017
Jul 2017
Jan 2018
Jul 2018

`atom`:


每月新增Star数量
30k
25k
20k
15k
10k
5k
0
2015
2016
2017
2018
2019
2020
2021

每月新增Fork数量



每月打开Issue数量

从以上分析图不难看出，活跃项目在每月Issue数量上明显超过已经被打归档项目。

开发者数量变化趋势

`anko`:

`atom`:

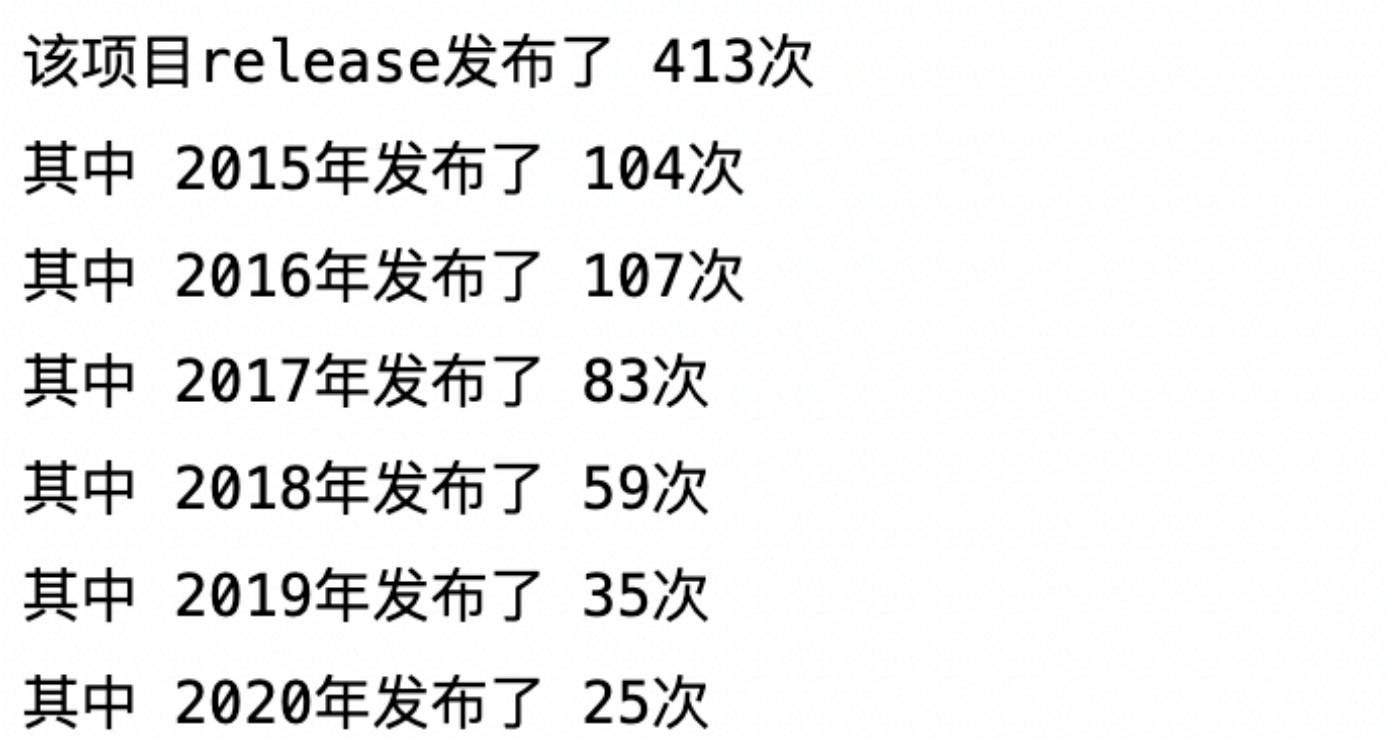
可以看出虽然二者的活跃开发者都是在下降趋势中，但是`atom`的活跃开发者绝对数量比`anko`要多出一倍多。

项目的release次数

`anko`:

```
该项目release发布了 31次
其中 2015年发布了 16次
其中 2016年发布了 4次
其中 2017年发布了 8次
其中 2018年发布了 3次
```

`atom`:



可以看出`atom`作为一个比`anko`规模大很多的开源项目，release次数比`anko`多了一个数量级。