### 开源项目健康监控与指标设计

#### 团队介绍

开源项目健康监控与指标设计团队来自湖北大学计算机与信息工程学院OS-HUBU社区，团队队长是高纪翔，成员是尹诗兰和赵曼蔓。

#### 组员分工

高纪翔：前端设计和实现，赵曼蔓：数据分析和清洗，尹诗兰：数据接口开发，所有人共同完成指标的设计和数据库的设计。

#### 背景及意义

在当今数字化浪潮汹涌澎湃的时代背景下，开源软件已然崛起成为软件领域中一股不容小觑的强劲力量。开源所倡导的协作与共享理念，成功开辟了软件开发与创新的全新模式以全球规模首屈一指的开源软件平台 GitHub 为例，截止2024年一月，估算该平台有超过4000万个开源项目，已然成为软件源代码托管及发现的核心枢纽平台。

在开源项目蓬勃发展的生态体系中，健康程度堪称核心关键指标。如本项目所构建的体系，管理者可借助项目响应能力，从议题解决到变更请求响应各环节，精准把控团队协作节奏；根据新贡献者数量波动，洞察项目对外部人才的磁吸效应；凭借巴士系数，评估项目对关键人员依赖风险；借助项目发展趋势值，洞悉其动态走向，进而分析项目问题的根源，从而更好的管理开源项目。

鉴于此，如何构建一套精准且客观的开源项目健康评估体系，成为开源软件领域亟待攻克的关键课题，这也凸显出建立开源项目健康评估体系的必要性与迫切性，其旨在全面、系统地考量开源项目多方面的关键要素，构建开源项目健康度监控与指标设计这一过程中，数据经过严格的清洗并通过映射函数进行标准化处理，确保所有指标都能真实、准确地反映项目的实际健康状况。综合考虑项目响应速度、新贡献者数量等核心指标，全方位洞察项目健康程度，为资源调配、策略优化奠定基础，以保障开源项目能够持续、稳定且健康地发展。

#### 总体目标

依据开源大数据分析，结合主客观评估方法，提出一种开源项目健康评估体系，帮助客观公正地评估开源项目的健康程度。我们采用多个指标来判断开源项目的健康程度，这多个指标再综合成一个指标直观判断。

多项指标来定义一个开源项目的健康程度对于一个可能拥有上千开源项目的公司来说管理会有很大困难，这样就需要一个总指标来定义一个开源项目的健康程度，这样一个公司里的每个开源项目都会有一个健康指标，监测人可以根据该指标对这个项目做出一定的决策或者是管理。

我们获取oppendigger中的开源数据，对这些数据进行分析，之后将开源项目健康评估体系分为四个一级指标，每个一级指标又分为多个二级指标，二级指标是开源数据当中的具体标签，通过critic 赋权法与映射函数将二级指标转换为一级指标，之后通过熵值赋权法将二级指标结合为一级指标。

我们将把一级指标和二级指标在开源项目健康监控的大屏中进行展示，依据每个项目的健康程度总指标进行排名，当管理者点击排名中的某一项开源项目时，旁边的每个板块可以分别展示该项目的二级指标数据，方便管理者快速找出该项目的不健康原因。

#### 数据处理

##### 5.1 数据收集

数据收集是数据处理的首要步骤，也是至关重要的环节。为了满足项目需求，我们通过获取oppendigger中的开源数据，对这些数据进行分析。

##### 5.2 格式转换

将json转csv：Json数据在网络传输和数据交换中广泛使用，由于其自我描述性好和易于读写的特点，许多Web API都选择使用Json格式输出数据。然而，对于数据分析来说，CSV格式更具优势，因为CSV可以直接导入到大多数数据分析软件和数据库中，且文件占用空间较小，读取速度快。

我们对300个仓库收集到的不同的metrics的json文件的数据进行重新整合，输出对应的csv文件。每个csv文件对应了一个metric。

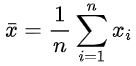
实验具体过程如下：先导入所需的模块，并定义输入和输出文件路径。在获取输入文件夹中所有仓库文件夹的路径后，遍历每个仓库文件夹，并查找Json文件。使用Python的Pandas库中的read\_json函数读取Json数据，然后使用to\_csv函数将数据框（DataFrame）转换为CSV格式。

##### 5.3 数据清洗

在对数据分析时，我们发现，某项二级指标样本会存在极端数据或者缺失数据，这些数据远大于或者小于其他数据，我们把它称为异常数据。这些数据在进行统计赋权和映射时，会对结果的正确性和客观性产生一定的影响，因此我们要对数据采用数据清洗。由于采用的映射方法是线性映射,那么这些极端数据会影响映射的准确性,因此采用箱线图方法(IQR)来识别异常值,清洗异常数据。

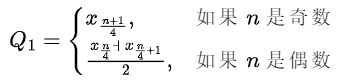
详细步骤：

1. 计算数据的平均值

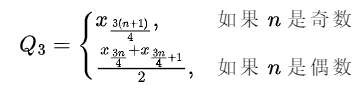
设原始数据集合为 ，数据个数为n，则平均值的计算公式为：

1. 计算四分位数Q1和Q3

如果n为数据个数，先将数据从小到大排序。计算Q1（下四分位数）:



计算Q3（上四分位数）:

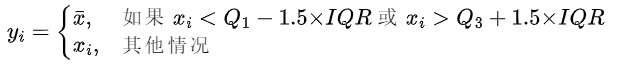


1. 计算四分位距IQR

公式为IQR=Q3-Q1。

1. 处理异常值

将异常值替换为均值。



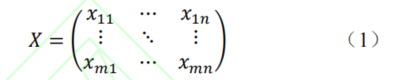
##### 5.4 熵值赋权法

熵值法是一种在多指标决策分析中确定指标权重的客观方法。它是基于信息熵的概念，通过计算指标的信息熵来衡量指标数据所包含的信息量，进而确定各指标在综合评价中的权重。信息论中，熵描述系统混乱或不确定性程度。随机变量取值与分布影响熵大小，变异程度大的指标熵值小，利用差异系数突出信息量大的指标，经归一化得权重，让其在评价开源项目健康时起更重要作用。

其详细步骤如下：

（1）获取数据清洗后的数据 *X*

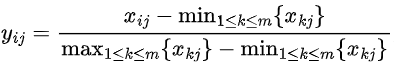
假设现有一组数据，有 *m* 个待评价对象，*n* 个评价指标，构成原始数据矩阵X，如公式（1）所示:



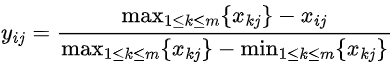
（2）对数据进行标准化

数据标准化的主要目的就是消除数据量纲的影响，使所有的数据能用统一的标准去衡量。

对于正向指标，指标值越大越好：



对于逆向指标，指标值越大越好：



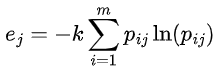
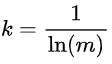
经过标准化后得到矩阵



1. 计算第j个指标下第i个样本的比重

 ，其中i=1,2，...,m；j=1,2，...,n。

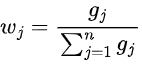
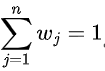
1. 计算第j个指标的熵值

根据信息熵的定义，，其中 ，当pij=0时，规定pijln（pij）=0

1. 计算第j个指标的差异系数

差异系数gj=1-ej，它反映了指标的数据差异程度。熵值越小，差异系数越大，说明该指标越重要。

1. 计算指标权重wj

最后，指标权重，且。

熵值赋权法利用了数据的离散程度，完全依靠数据自身的客观属性，并非数字越大就说明越重要，以达到科学评价的目的。

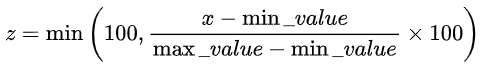
结合以上特性，熵值赋权法适合对开源项目进行健康评估。例如，通过熵值赋权法计算出项目响应能力指标的权重较高，说明在评估开源项目健康状况时，项目响应能力是一个相对更重要的因素。

本项目主要通过熵值赋权法，帮助有效确定开源项目健康程度指标的权重，来保证评估结果的客观性和准确性。

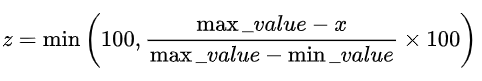
##### 5.5 线性映射

我们采用线性映射,通过映射函数将这些不同量纲的清洗后的数据映射到一个统一的区间（0 - 100），可以方便后续的综合分析。找出替换后数据中的最小值min\_value和最大值max\_value，将清洗前的数据带入公式。

对于正向指标 ：



对于负向指标：

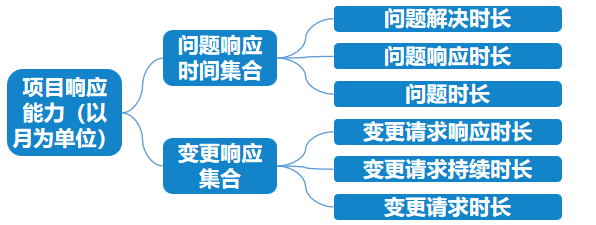


将数据清洗和线性映射配合使用，把数据标准化与归一化的同时，又保证了转换后的数据的科学性与合理性。这样可以在同一尺度上对开源项目的多个健康指标进行综合评估，可以更直观地看出指标的相对变化情况，同时也适应不同项目规模，对处理不同规模的开源项目在各项指标上的原始数据的巨大差异上更有利。

#### 各项指标设计

##### 6.1项目响应能力Hproject\_response

项目反应能力反应开发者活跃度、项目管理及团队沟通情况。健康开源项目需较短响应时间，能提高用户满意度，利于项目声誉提升，吸引新用户和开发者；反之，若长时间无响应，易被诟病，会使项目逐渐失去人气。项目响应能力Hproject\_response体现在问题响应时间集合和变更响应集合，问题响应集合包括议题解决时长、议题响应时间、议题年龄。变更响应集合包括变更请求响应时间、变更请求持续时间和变更请求年龄。



经过计算获得项目响应能力指标的二级指标权重

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 权重 | 问题解决时长 | 问题响应时长 | 问题时长 | 变更请求响应时长 | 变更请求持续时长 | 变更请求时长 |
| w | 0.1438 | 0.1636 | 0.2979 | 0.0423 | 0.0695 | 0.2829 |

##### 6.2新贡献者Hnew\_participant

新贡献者的增长或减少是衡量项目健康状态的重要指标。要深入理解新社区成员的行为和面临的挑战，识别他们的身份至关重要。一个健康发展的项目应该能够持续吸引新的贡献者。通过衡量首次为项目做出贡献的新成员数量，我们可以洞察项目的发展动态和参与度水平。

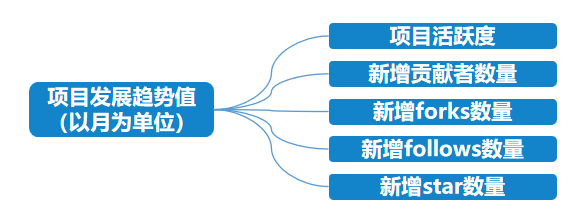
监测新贡献者的流入有助于揭示参与模式和识别潜在的参与障碍，从而为社区支持、新成员入职和外联活动提供宝贵的机会。如果一个项目能够持续吸引新的贡献者，这通常表明该项目具有健康的吸引力和参与度。相反，新贡献者数量的减少可能暗示项目在参与度或可访问性方面面临潜在的挑战。因此，对新贡献者的持续跟踪对于评估项目的健康状况和社区的活跃度至关重要。

##### 6.3巴士系数Hbus\_factor

巴士系数（Bus Factor）是衡量开源项目健康度的一个重要指标，通过确定贡献占总贡献量的50%的最小贡献者数量来评估项目对少数贡献者的依赖程度。在开源项目中，一个较低的巴士系数意味着项目的健康度较低，因为项目可能过度依赖少数关键贡献者，一旦这些贡献者离开，项目可能会陷入停滞或面临重大困难。相反，一个较高的巴士系数表明项目的健康度较高，因为项目的知识、技能和工作分配更为分散，项目受个别人员变动的影响相对较小。

##### 6.4项目发展趋势值HProject\_development\_trend

项目发展趋势指特定时间内的方向与变化，对项目成败意义重大。趋势值稳定增长是项目健康标志，意味着按计划推进、创造价值。把握趋势利于管理者提前制定策略，优化资源配置，提升项目竞争力与适应性。项目发展趋势值Hproject\_development\_trend体现在项目活跃度，新增贡献者数量，新增forks数量，新增follows数量，新增star数量。

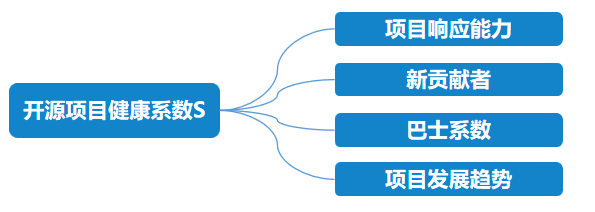


经过计算获得项目发展趋势指标的二级指标权重

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 权重 | 项目活跃度 | 新增贡献者数量 | 新增forks数量 | 新增follows数量 | 新增star数量 |
| w | 0.3920 | 0.0874 | 0.0632 | 0.3072 | 0.1502 |

##### 6.5开源项目健康系数

基于开源大数据与客观评估，构建开源项目健康评估体系，帮助管理者管理多个开源项目。从 oppendigger 获取数据，将项目响应能力，新贡献者，巴士系数，项目发展趋势这几个一级指标经熵值赋权法和映射函数转换结合成综合指标，综合评判开源项目健康程度。



经过计算获得开源健康系数的一级指标权重

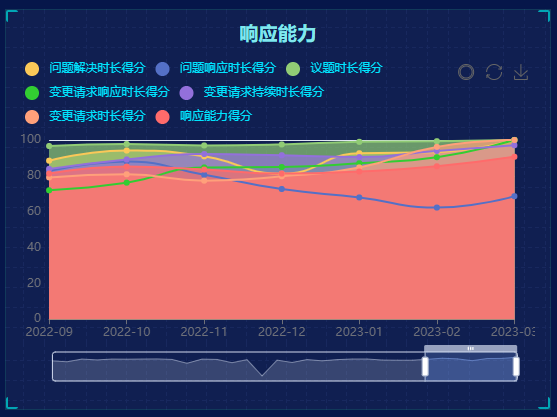
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 权重 | 项目响应能力 | 新贡献者 | 巴士系数 | 项目发展趋势 |
| w | 0.2241 | 0.2333 | 0.2747 | 0.2679 |

##### 项目展示

##### 7.1整体大屏



##### 7.2项目响应能力界面图



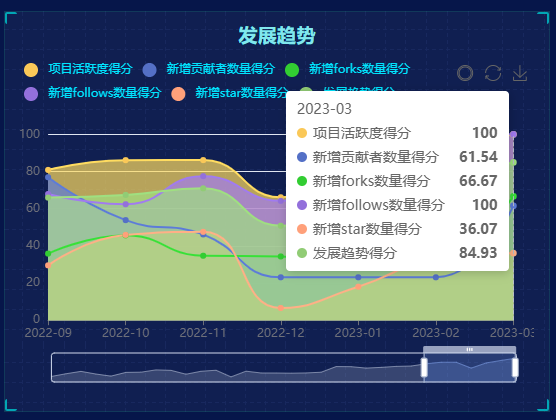
##### 7.3巴士系数界面图



##### 7.4新贡献者界面图



##### 7.5项目发展趋势界面图



##### 7.6排行榜与各一级指标得分界面图

最新一月项目健康度综合评估得分界面图（截止到2023年3月）：

##### 51FA2C5C9B91CEA9EB0E4E9C58CA93D5

最新一月项目一级指标得分界面图：



#### 开源实践

为了能够更好的参与到开源的实践以及贡献中去，本项目在GitHub上创建了。

#### 总结与展望

本文构建了一套开源项目健康评估体系，其目的在于全方位考量项目的关键要素，从而确保项目能够持续、稳定地发展。该体系运用熵值赋权法来明确指标权重，以此保证评估的客观与精准性。在数据处理方面，映射函数借助箱线图方法剔除异常数据，接着利用线性映射将数据转换至0 - 100的区间，实现合理评价。凭借对一级和二级指标的综合运用，此体系可在监控大屏上展示项目的健康程度，帮助管理者迅速找出不健康因素，进而做出高效的管理决策。

在应用方面，该健康度评估体系有望深度融入项目管理流程，成为项目启动、推进与优化的核心依据。在不同规模和领域的开源项目中广泛应用，助力项目团队精准定位问题、合理分配资源、及时调整策略，从而显著提升项目成功率与效率。

开源项目的健康度监控与指标设计是一个不断发展的领域，其目标是通过精准的数据监控和实时分析，为开源项目提供一个全面的健康评估系统，进一步推动开源项目的高质量、可持续发展，为全球开源事业注入强劲动力并引领行业评估标准的革新。

展望未来，开发者依靠本项目将能够精准判断项目是否处于积极维护状态，一旦察觉项目中某项指标异常，便可即时开展针对性的强化测试工作，为项目质量保驾护航。企业则可借助健康度监控提前洞悉潜在风险，在考量是否引入开源项目之际，依据其健康状况周全评估长期支持的可行性，从而做出更为明智的战略抉择。大型开源项目管理人员亦能依据各模块健康指标实现开发人员的科学调配，让人力资源得以高效利用，推动项目稳步前行。而投资者或赞助商可通过这一体系明确甄别哪些项目具备投资价值，在开源项目的广阔天地中精准布局，收获丰厚回报，各方携手共进，共同促进开源项目生态的繁荣昌盛与持续创新。