

TalentRank 算法设计

TalentRank 算法用于衡量开发者在技术社区中的综合表现。通过个人和公司两个维度的权重系统，算法能够更全面地评估开发者的技术能力和影响力。该算法的用户评分包含 **个人影响力**、**个人贡献活跃度** 和 **参与项目的重要性** 三大主要因素。

1. 用户个人权重

用户个人权重衡量开发者的技术能力和社区影响力，从以下几个方面进行评估：

1.1 个人影响力

个人影响力指标评估开发者在社区中的知名度和受欢迎程度，主要包括：

- Star 数**：反映项目的受欢迎程度，可作为开发者影响力的一个基本指标。
- 粉丝数 (Followers)**：直接反映开发者的受欢迎程度和影响力。

个人影响力得分公式

影响力得分通过以下公式计算：

$$\text{influence_score} = (0.7 \times \text{repo_star}) + (0.3 \times \text{repo_fork}) \quad (1)$$

- repo_star** 的权重设置为 0.7，因为 Star 数可以直接反映出项目的受欢迎程度。
- repo_fork** 的权重设置为 0.3，表示 Fork 数对影响力得分的影响稍弱于 Star 数。

等级划分

根据 `influence_score` 的值，函数将项目的影响力分为五个等级：

- 5级**：`influence_score` ≥ 1000 （影响力最高）
- 4级**： $500 \leq \text{influence_score} < 1000$
- 3级**： $100 \leq \text{influence_score} < 500$
- 2级**： $10 \leq \text{influence_score} < 100$
- 1级**：`influence_score` < 10 （影响力最低）

1.2 个人贡献活跃度

个人贡献活跃度用于评估开发者在项目中的技术投入和活跃程度。评估指标包括以下几方面：

- 上传次数 (Push)**：通过 `PushEvent` 衡量开发者在项目中的代码提交频率。权重为 0.4。
- 拉取请求 (Pull Requests)**：通过 `PullRequestEvent` 反映贡献质量，考虑 PR 的数量和被接受率。被合并的 PR 得分较高，未合并的 PR 权重减半。权重为 0.3。
- 问题参与度 (Issues)**：通过 `IssuesEvent` 统计开发者在创建和回复 Issue 中的参与情况，权重为 0.2，反映开发者的协作和问题解决能力。
- 项目的关注 (WatchEvent) 和派生 (ForkEvent)**：用于衡量开发者在项目中的附加贡献，权重各为 0.05。

贡献度评价等级

根据上述活动事件的加权得分计算每个仓库的贡献分数，并分为三档贡献度评价：

- **高（3级）**：贡献分数 ≥ 1.5 ，表明开发者对项目有显著的贡献和活跃度。
- **中（2级）**： $0.5 \leq \text{贡献分数} < 1.5$ ，表明开发者对项目有一定的技术投入。
- **低（1级）**：贡献分数 < 0.5 ，表明开发者对项目的参与较少。

个人贡献得分公式

通过对项目的不同活动赋予权重，计算用户在项目中的总体贡献得分。公式如下：

$$\text{个人贡献得分} = (0.5 \times \text{项目影响力得分}) + (0.5 \times \text{贡献度得分}) \tag{2}$$

在这个公式中，我们认为项目的影响力和个人在项目中的实际贡献度（如提交次数、PR、Issue 等）同样重要，因此将两者的权重设置为 一半一半，即各占 50%。这样既能够反映项目的受欢迎程度，又能有效衡量开发者在项目中的技术投入。

1.3 参与项目的重要性

参与项目的重要性反映了开发者所参与项目的整体影响力和受欢迎程度，通过项目的 Star 和 Fork 数量来衡量。

2. 综合 TalentRank 计算

通过对 `total_stars`（总 Star 数）、`followers`（粉丝数）和 `contribution_score`（个人贡献分数）赋予不同权重，计算出最终的 TalentRank 得分。此分数能够综合反映开发者的个人影响力、技术贡献度和项目重要性。

TalentRank 得分公式

$$\text{TalentRank} = (w_{\text{total_stars}} \times \text{总 Star 数}) + (w_{\text{followers}} \times \text{粉丝数}) + (w_{\text{contribution_score}} \times \text{贡献分数}) \tag{3}$$

在此计算中，建议为每个分量设定不同权重，以确保 TalentRank 的合理分布。例如，如果个人贡献度比个人影响力和项目的重要性更重要，可以将 `contribution_score` 的权重设为更高。具体权重设置如下：

- **权重建议：**
 - $w_{\text{total_stars}} = 0.4$
 - $w_{\text{followers}} = 0.3$
 - $w_{\text{contribution_score}} = 0.3$