Ospaf: 大数据背景下的开源项目成熟度分析工具

1.概述

软件成熟度评估的最终目标是帮助软件的可持续发展，并为用户应用提供必要的技术参考。开放源代码软件成熟度评估也不例外。我们通过软件的成熟度评估，形成全面的涉及技术、应用、法律等层面的评价报告，帮助那些正在或潜在的开源软件使用者准确的了解软件的技术特性和应用特性，从而为他们选择适合自身需求的开源软件提供参考。同时报告中涉及的大量评测数据，为开源软件的开发者提供帮助，促进他们有效的改善软件在技术方面和使用方面的质量，使软件不断成熟和可持续的发展。

开放源代码软件由于开发模式和运作模式的独特性，其软件带有鲜明的特点。开放源码软件成熟度评估的方法需要我们在实践中不断的探索。我们将提出一些基本的原则和方法，并建立我们自己的评估体系和计算模型。

2.项目分析

2.1 项目简介

Ospaf（open source project analyze framework）项目的发起是作为 CSDN 举办的 summer code 的项目之一。由 SUSE Linux 组织指导，北邮在读硕士李博同学编写完成的。Ospaf 工具的主要功能是可以通过采集开源项目的相关数据，通过机器学习的算法建立开源项目成熟度评估模型，从而实现对于开源项目的评估。

2.2 数据来源

目前世界上最火的项目托管网站是 github，ospaf 采用 github 作为入手点。通过读取 Ghtorrent 和 github-api 采集数据。

同时 ohloh 也对开源项目进行分析并开放数据，ospaf 的部分数据采集自 ohloh。

2.3 数据挖掘

2.3.1 特征值的提取

Ospaf 项目的特征主要包括三个方面，分别是原始特征、衍生特征、抽象特征。

原始特征包含一些 github-api 提供的参数，例如项目的

star 数量和 fork 数量等。

衍生特征包括对原始特征进行处理产生的特征，比如提取任意相邻两个月的 star 数量的增长数做比值，可以得到 star 的增长率作为特征。

抽象特征分为以下几种类型。第一种，通过提取项目 commit 语句中的高频词汇（包含 revert、update 等）作为特征。第二种，计算开源项目 contributor 中 star-contributor 的比重。第三种，分析邮件列表等数据。

2.3.2 模型的建立

Ospaf 项目模型的建立主要是通过机器学习算法来实现。

第一步，去除噪音

将数据库中的数据按照高斯去噪法，将噪声数据去除。

第二步，归一化处理

因为建立模型用到了回归算法，为了减小不同量纲特征对结果的影响，对所有特征进行归一化处理。

第三步，聚类产生目标序列

将公认的比较成熟的开源项目的数据导入训练集并聚类分析，生成目标序列。

第四步，利用逻辑回归生成数学模型

利用逻辑回归算法，对训练集进行训练，生成最终的数学模型。

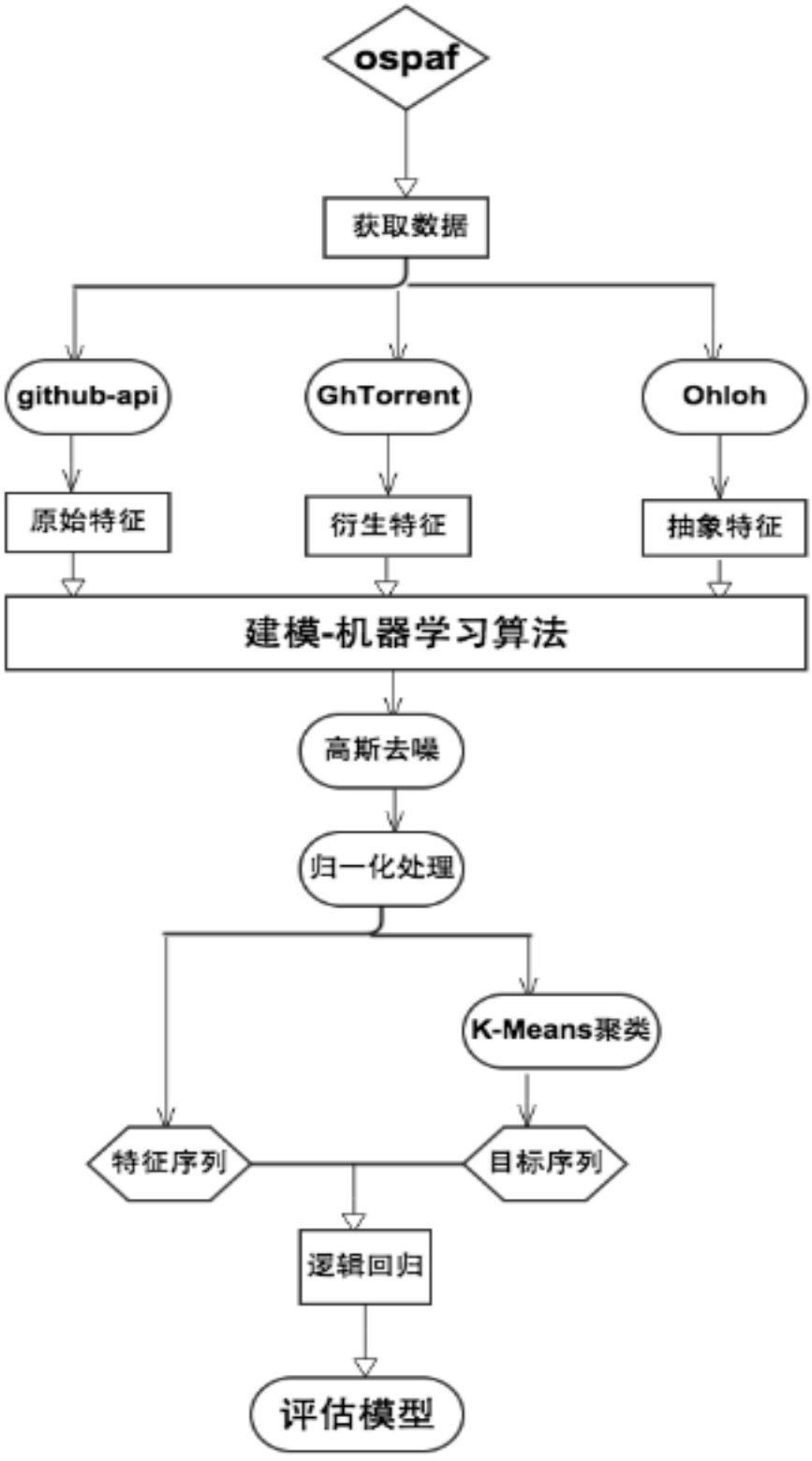
2.4 项目创新点

（1）以大数据为背景，利用机器学习算法进行开源项目评估的建模。目前，对开源项目的评估一般都是基于 KQI 指标的用户评价，例如 Ospaf 这种利用 KPI 指标进行数学建模评估的案例仍不多见。

（2）特征的多样性，之前对一个开源项目的评价可能只是简单地利用 star 数或是用户打分的方式。Ospaf 在这些特征的基础上，更添加了一些抽象特征，比如跟时间序列有关的一些增长率特征，用户 commits 语句中提取的特征

（3）可以根据用户的需求进行评测。因为各个特征都是独立的，所以可以通过改变特征的权重来对项目进行评测。比如用户 A 需要用户关注度高的项目，那么就可以相应的提高用户关注度方面的特征的权重。

3.项目流程图



Ospaf 流程图

联系方式

项目地址：https://code.csdn.net/gshengod/ospaf-1

相关博客：http://blog.csdn.net/buptgshengod

email：jimenbian@gmail.com