

软件开发知识工程

本项目旨在为软件生产的不同环节提供知识服务。为软件的开发者和维护者提供必要的知识资源。

对于任意时刻的软件项目，该项目的未来，由已经存在的代码和将要被实现的代码组合而成。因此，代码和工程师是软件项目的基石。

我们将一个任意时刻的软件项目抽象为已完成的代码和未完成的代码的组合。对于未完成的代码，我们考虑如何实现。对于已完成的代码，我们考虑如何修改，删除和查询。

如果将软件项目看作是代码数据，我们考虑一下四个方面：

1. 增：复制已有的代码，或联系能写这个代码的人
2. 改：联系原来写这个代码的人
3. 删：联系原来写这个代码的人
4. 查

综上所述，我们考虑三类功能：

1. 如何合理的推荐可复制的代码片段
2. 将已完成或未完成的代码与其开发者进行绑定
3. 搜索代码

为实现上述三类功能，我们需要完成如下四类映射，作为基础设施：

1. `match (code - code)`
2. `map (code - developer - bug)`
3. `structural map (project - code)`
4. `map (code - function(功能))`

相应的，我们现阶段由以下四个课题组成：

1. 基于AST的代码匹配
2. 软件工程领域知识关联（代码，开发者，BUG）
3. 基于NLP的软件模块功能分析
4. 软件模块化结构分析

任务分配

基于AST的代码匹配

制定每行代码的唯一ID的生成规则。考虑多个项目，多个文件，同名文件。
给定函数或类的声明完整名字，对该声明对应的函数和类进行定位
构思软件中的模块种类，例如函数，类。构思模块间可能存在的多种关系，例如继承，函数调用。
实现查询接口：输入 任意行号 ，返回所在 模块行号

软件工程领域知识关联（代码，开发者，BUG）

分析Commit History，将任意一行代码与开发者进行关联
分析Issue，将开发者与Issue关联（或BugZilla）
实现查询接口：输入 模块行号 ，返回其开发者，和Issue信息

基于NLP的软件模块功能分析

使用静态分析工具提取其模块的名字（所考虑的模块类别由zhenghui制定），并组织为扁平的集合（考虑多个项目的情况，例如模块名称的重复）。
利用NLP，以三元组的形式重新组织模块名，将该模块名集合构建成图谱（将节点边与行号相关联）。
实现查询接口：输入关键字，返回与关键字相关的模块行号
实现查询接口：输入 模块行号集合 ，返回对应子图

软件模块化结构分析

根据zhenghui的设计，使用静态分析工具，提取软件模块的关系，并组织成结构图。
基于该图，考虑计算模块间结构上相似度的方法
实现查询接口：输入 模块行号 ，返回其结构上相近（proximity）的模块，以及其调用与被调用关系。