

“基于多源知识融合的软件构造智能化方法与支撑环境”研发工具集 企业应用评测说明

本评测说明由北大博士王敏个人提供，仅代表个人观点，以供参考，各企业合作单位需根据自身情况与相应测试单位进行对接，完成项目应用评测。

评测要求：

- 评测报告需体现：企业单位在具备领域特性的软件项目上，对 SnowGraph 工具的使用情况，即利用 SnowGraph 工具进行特定领域软件项目知识图谱的自动构造，并对此应用过程进行评测；
- 评测报告需体现：企业单位构造的特定领域软件项目知识图谱的数据规模，即在上述自动构造的知识图谱的基础上，能够进行知识图谱的数据概览，并统计评测各类型数据的数据规模；
- 评测报告需体现：企业单位在上述特定领域软件项目知识图谱的基础上，能够结合领域特性，进行领域相关的智能问答服务，即能够提供领域相关的若干典型问答示例，并对问答应用效果进行评测；

下述三部分分别针对上述三点评测要求进行测试指南，其中包含如何构建软件项目知识图谱（即软件项目知识图谱的自动构造子系统），如何进行软件项目知识图谱的数据规模统计（即软件项目知识图谱可视化工具），如何进行软件项目知识图谱的智能化问答服务（即软件项目知识图谱相关制品查询与推荐工具）
企业可结合自身特定领域的软件项目，进行 SnowGraph 的应用评测。

以下部分可作为与第三方测试单位进行测试方案等有关内容确定的参考，企业也可根据自身应用情况进行扩展，能够体现评测要求即可。

一、 软件知识图谱自动构造子系统

构建与部署数据与环境准备

1. 编译构建

本系统的依赖环境包括：

Java 1.8+（用于知识图谱的自动构建与后端服务器的运行）

Node.js（用于前端服务器的运行）

Maven 3.2+（optional，用于从源代码开始编译构建）

Python 3（optional，用于 word 文档数据的预处理）

两种方式获得编译构建的 jar 包：

第一、使用 maven 从源代码开始进行编译构建：mvn package
源代码地址：

<https://github.com/linzeqipku/intellide-graph>

第二、直接下载已经编译好的 jar 包

jar 包下载地址：

<https://github.com/linzeqipku/intellide-graph/releases>

2. 数据准备（企业单位提供特定领域的项目源数据）

Java（或 C#等其他编程语言）源代码数据： 将整个项目的源代码统一解压到一个文件夹即可。

Git 版本库数据： 给出该项目的.git 文件夹即可

html 文档数据： 统一放在同一个文件夹中即可，对于 docx 格式的文档，使用附件提供的 python 脚本进行转化为 html 格式。

pptx 演示文稿数据： 统一放在同一个文件夹即可

自动构建知识图谱

1. 编写 yaml 配置文件

在任意目录中新建一个 yaml 文件，在该文件中配置：(1) 知识图谱的输出文件夹路径；(2) 需要允许哪些知识抽取模块；(3) 这些知识抽取模块所输入的源数据的路径。配置文件的示例如下：

```
graphDir: E:/graph.db-chinese # 知识图谱的输出文件夹路径，如果需要中文支持，该路径需要由“-chinese”来结尾(注：项目名可自由命名，比如 E:/X-chinese)

# 依次执行如下数据解析插件
cn.edu.pku.sei.intellide.graph.extraction.java.JavaExtractor: E:/data/src
cn.edu.pku.sei.intellide.graph.extraction.git.GitExtractor: E:/data/.git
cn.edu.pku.sei.intellide.graph.extraction.html.HtmlExtractor: E:/data/html
cn.edu.pku.sei.intellide.graph.extraction.pptx.PptxExtractor: E:/data/pptx

# 依次执行如下知识关联与挖掘插件
cn.edu.pku.sei.intellide.graph.extraction.tokenization.TokenExtractor:
cn.edu.pku.sei.intellide.graph.extraction.code_mention.CodeMentionExtractor:
or:
cn.edu.pku.sei.intellide.graph.extraction.doc_link.DocLinkExtractor:
```

2. 运行如下命令，自动生成知识图谱

```
java -jar intellide-graph.jar -gen yaml_config_path
```

intellide-graph.jar 是下载的或编译好的 jar 包名称
yaml_config_path 是第一步编写好的 yaml 文件的路径
运行完毕之后，可以在配置文件中所指定的输出文件夹路径中生成 neo4j 图数据库格式的知识图谱。

启动 web 服务

1. 知识图谱准备

将所有需要运行的知识图谱文件夹放到同一的一个文件目录下，例如：

```
E:/graphs/X
```

在本地任意目录下新建一个 json 文件，描述这些知识图谱，例如：

```
[
  {"name": "X", "description": "This is a domain-specific program"}
]
```

编辑 intellide-graph.jar 中的
BOOT-INF/classes/application.properties 文件，例如：

```
server.port=8004

graphDir= E:/graphs/
dataDir= E:/tmp/ # 临时文件存储路径
infoDir= E:/graphs/graphs.json #知识图谱描述文件
```

2. 运行如下命令，启动后端服务器

```
java -Xms1024m -Xmx4096m -XX:MaxPermSize=2048m -
XX:MaxNewSize=2048m -jar intellide-graph.jar -exec
```

3. 启动前端服务器

前端项目地址：

<https://github.com/woooking/snowview> (intellide-graph branch)

注意：选择 intellide-graph 的分支，不是 master 的分支

在 src/config.ts 中配置后端服务器的 URL

编译项目：npm install

启动前端服务器：npm start

注意：安装 node.js 需要版本 10 以上

4. 浏览器访问

<http://localhost:3000/>

注意：使用 google、Firefox 等现代浏览器。

测试方案：

测试对象：软件项目知识图谱自动构建系统

测试用例：企业领域特定的软件项目（企业提供给测试公司）

具体包括：

项目源代码

项目版本控制数据
项目文档数据

测试过程：按照上述过程构建部署知识图谱的安装环境，输入测试用例提供的数据库路径，即配置文件，运行自动构建命令，生成知识图谱图数据库，启动 web 服务，验证生成知识图谱。

测试结果：生成知识图谱图数据库文件；web 端可访问知识图谱。证明 SnowGraph 能够在特定领域的软件项目下进行自动化的知识图谱构造。

二、 软件知识图谱可视化工具

测试方案：

测试对象：软件知识图谱可视化工具

测试用例：企业领域特定的项目知识图谱

本用例在第一个工具的实现基础上提供，即已完成企业内部领域特定项目的知识图谱的自动构建以及 web 应用的搭建

测试过程：点击访问 web 应用端的浏览器，点击“USE IT”，进入项目列表页面，选中领域特定的软件项目，进入“图谱概览”界面，得到知识图谱的数据属性。

测试结果：可得到特定领域项目知识图谱的图谱数据概览，以及统计出给类型软件项目数据的具体规模数据。

三、 软件知识图谱相关制品查询与推荐工具

测试方案：

测试对象：软件知识图谱相关制品查询与推荐工具

测试用例：企业提供领域特定的软件项目知识图谱

本用例在第一个工具的实现基础上提供，即已完成特定领域的软件项目知识图谱的自动构建以及 web 应用的搭建。

具体输入查询用例：

用例一： 列举类 XXX 的方法（XXX 是企业源项目的一个 java 类名）

用例二： 提问关于领域特定的问题（例如：神州数码的市民服务项目：查询并添加公积金还款信息），此问题为领域开发的相关问题，必须与软件项目的业务逻辑或具体代码相关；

用例三： 谁修改了 类 XXX （XXX 是企业源项目的一个 java 类名）

测试过程：点击访问 web 应用端的浏览器，点击“USE IT”，进入项目列表页面，选中企业软件项目，进入“智能问答”界面，输入上述用例查询，回车，界面出现相应的知识图谱子图及相关制品的查询结果和推荐结果

测试结果：可得到特定领域软件项目知识图谱的查询用例的子图推荐结果，能够进行企业内部特定领域软件项目的相关业务开发问题的智能问答。