Arch Linux-OpenEuler包名映射工具

2022.12.17

一、软件包名称映射工具总体介绍：

1. 本工具最终的输出是一个两个平台的包名映射表，该映射表为后续自动构建工作服务，一方面用于查找待构建软件包在目标平台的名称，也可以需要将还没有对应包的软件包新的映射关系加入到表中。
2. 本工具最初始的输入为两个平台的软件包列表，列表需要先行获取，不在本工具功能范围内。
3. 如图1所示，本工具将以上目标分为两个步骤：（1）通过【包名映射算法】（包括文本预处理和相似性计算）为每一个待构建软件包查找完全匹配的软件包，或者最为相似的Top K候选软件包；（2）通过【人工标注系统】，最终确定唯一匹配的或者没有匹配。

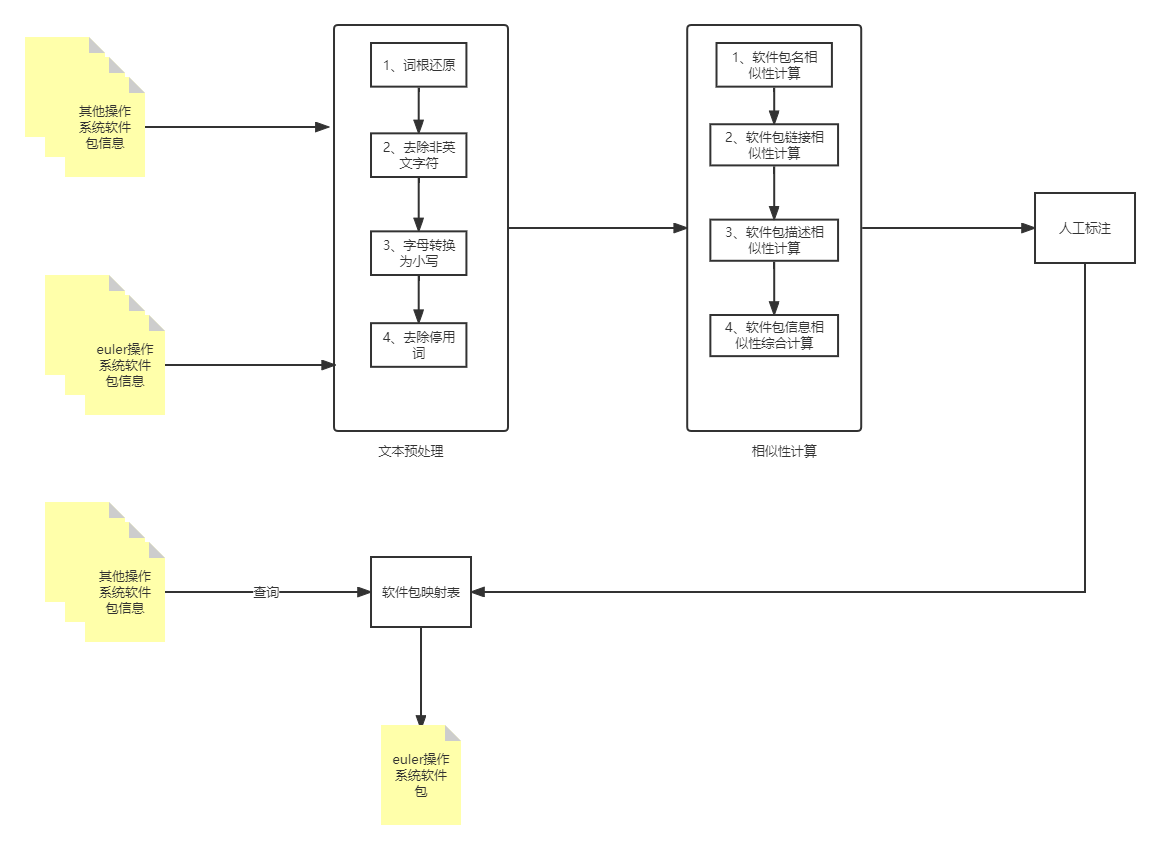


图1.工具工作流程图

1. 使用方法：按照要求准备好两个平台的软件包JSON文件，执行文本预处理、相似性计算，获得初步映射表；将结果导入标注系统，标注后确定最终映射表；后续通过该映射表的查找、更新API使用。

本文档后续内容将详细介绍【包名映射算法】、【工具具体实现】和【使用方法】。

二、包名映射算法

1. 文本预处理

（1）将一些常见的英文缩写，转换为原有格式：eg. it's - > it is, can't -> can not

（2）去除所有除英文和数字外的其他所有字符，去除多余的空白字符

（3）将所有字母转换为小写

（4）根据停用词表移除停用词

2. 相似性计算

（1）分别对软件包的名称、软件包的url和软件包的名称计算文本相似性得分，最终相似性得分为各个相似性得分的加权得分；

（2）采用Dice相似性系数衡量两个字符串之间的相似性。Dice相似性系数定义如下：

Dice（s1,s2）=2\*comm(s1,s2)/(leng(s1)+leng(s2))

其中，comm (s1,s2)是s1、s2 中相同字符的个数leng(s1)，leng(s2)是字符串s1、s2 的长度；

（3）软件包P和Euler操作系统上软件包Peuler之间相似性得分计算如下：

Score(P, Peuler）= W1\*dice(P\_name, Peuler\_name) + W2\*dice(P\_url, Peuler\_url) + W3\*dice( P\_description, Peuler\_description)

其中W1、W2、W3表示为不同内容匹配分配的权重，在测试时可以自行调节。

3. 根据相似性计算结果，为每一个待构建软件包确定候选匹配软件包，作为人工标注的输入。

1. 具体实现

全部代码仓库位于：<https://gitee.com/casserolefish/pkgbuild2-spec.git>

单独的包名映射工具仓库位于：<https://gitee.com/invictusjz/mapping_tools.git>

1. 相似软件包筛选工具和映射表查询、更新API的代码为：

仓库下mapping\_package文件夹中的内容，见图2。

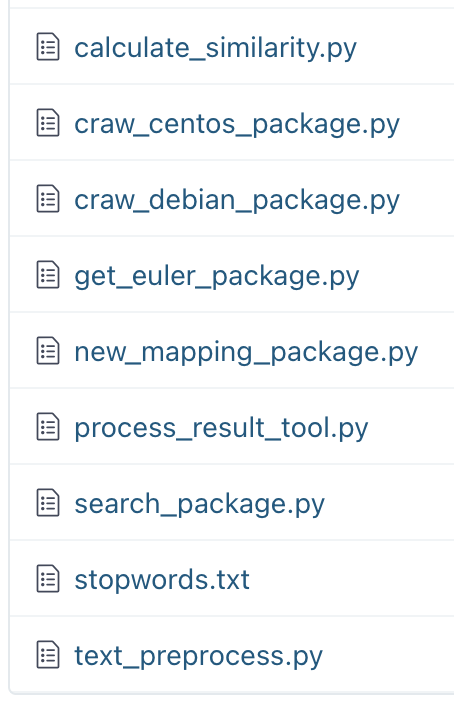


图2. mapping\_package文件夹下python文件

1. 函数之间的调用关系如图3所示

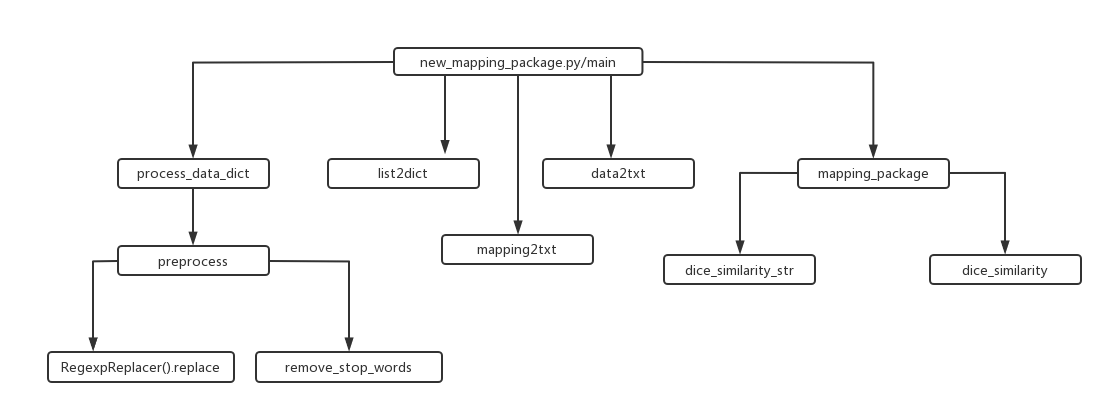


图3. mapping\_package文件夹下主要调用关系图

3. 具体函数解释

|  |  |
| --- | --- |
| calculate\_similarity.py | |
| 函数名称 | 功能简介 |
| cosine\_similarity | 计算两个向量的余弦相似度  输入：两个句子向量  输出：向量间的余弦相似度 |
| dot\_similarity | 计算两个向量的点积相似度  输入：两个句子向量  输出：向量间的点积相似度 |
| dice\_similarity | 计算两个句子间的Dice相似性系数  输入：两个句子  输出：两个句子间的Dice相似性系数 |
| dice\_similarity\_str | 计算两个字符串间的Dice相似性系数  输入：两个字符串  输出：两个字符串间的Dice相似性系数 |
| jaccard\_similarity | 计算两个句子间的Jaccard相似性系数  输入：两个句子  输出：两个句子间的Jaccard相似性系数 |

|  |  |
| --- | --- |
| new\_mapping\_package.py | |
| 函数名称 | 功能简介 |
| preprocess | 文本预处理过程：1、词根还原2、去除非英文字符3、字母转小写4、去除停用词  输入：待处理文本  输出：处理后文本 |
| process\_data\_dict | 将euler数据规范化  输入：爬取的数据字典  输出：筛选后的数据列表，包含package\_name、package\_version、package\_url、package\_description和package\_processed\_description。 |
| mapping\_package | 计算每两个软件包的相似性分数  输入：其他平台全部软件包的信息，euler全部软件包的信息  输出：其他平台软件包信息与euler软件包信息相似性分数数组 |
| list2dict | 将列表转换为字典  输入：列表形式存储的软件包信息  输出：字典形式存储的软件包信息 |
| data2txt | 将数据json文件转换为txt文件，用于标注  输入：软件包数据json文件  输出：软件包数据txt文件 |
| mapping2txt | 将映射json文件转换为txt文件，用于标注  输入：映射结果json文件  输出：映射结果txt文件 |

|  |  |
| --- | --- |
| process\_result\_tool.py | |
| 函数名称 | 功能简介 |
| process\_mapping\_result | 生成包名映射表  输入：sql文件路径，euler数据，其他平台数据  输出：包名映射表 |

|  |  |
| --- | --- |
| search\_package.py | |
| 函数名称 | 功能简介 |
| search\_package | 在包名映射表中寻找其他平台软件包对应的euler软件包  输入：其他平台软件包名称  输出：euler 软件包相关信息 |

|  |  |
| --- | --- |
| text\_preprocess.py | |
| 函数名称 | 功能简介 |
| RegexpReplacer.replace | 将缩略词还原为原有形式  输入：缩略词  输出：缩略词的原有形式 |

1. 人工标注系统的代码在仓库下MarkWeb4.0文件夹中

系统用到四张表描述如下：

aur\_archlinux表:描述其他平台软件包的一些必要信息

|  |  |  |
| --- | --- | --- |
| 名 | 类型 | 长度 |
| aur\_id | int | 0 |
| package\_name | varchar | 255 |
| package\_version | varchar | 255 |
| package\_url | varchar | 255 |
| package\_description | text | 0 |
| package\_processed\_description | text | 0 |

euler\_data表:描述euler软件包的一些必要信息

|  |  |  |
| --- | --- | --- |
| 名 | 类型 | 长度 |
| euler\_id | int | 0 |
| package\_name | varchar | 255 |
| package\_version | varchar | 255 |
| package\_url | varchar | 255 |
| package\_description | text | 0 |
| package\_processed\_description | text | 0 |

label\_result表:由相似性算法得出的与某个euler包相似度最高的包名对，三种关系用于人工标注。

|  |  |  |
| --- | --- | --- |
| 名 | 类型 | 长度 |
| id | int | 0 |
| euler\_id | int | 0 |
| aur\_id | int | 0 |
| strong\_correlation | tinyint | 0 |
| weak\_correlation | tinyint | 0 |
| no\_correlation | tinyint | 0 |

user表:用户表和当前进度

|  |  |  |
| --- | --- | --- |
| 名 | 类型 | 长度 |
| username | varchar | 255 |
| password | varchar | 255 |
| start\_id | int | 0 |
| end\_id | int | 0 |
| current\_id | int | 0 |

系统标注的标准如图4所示：

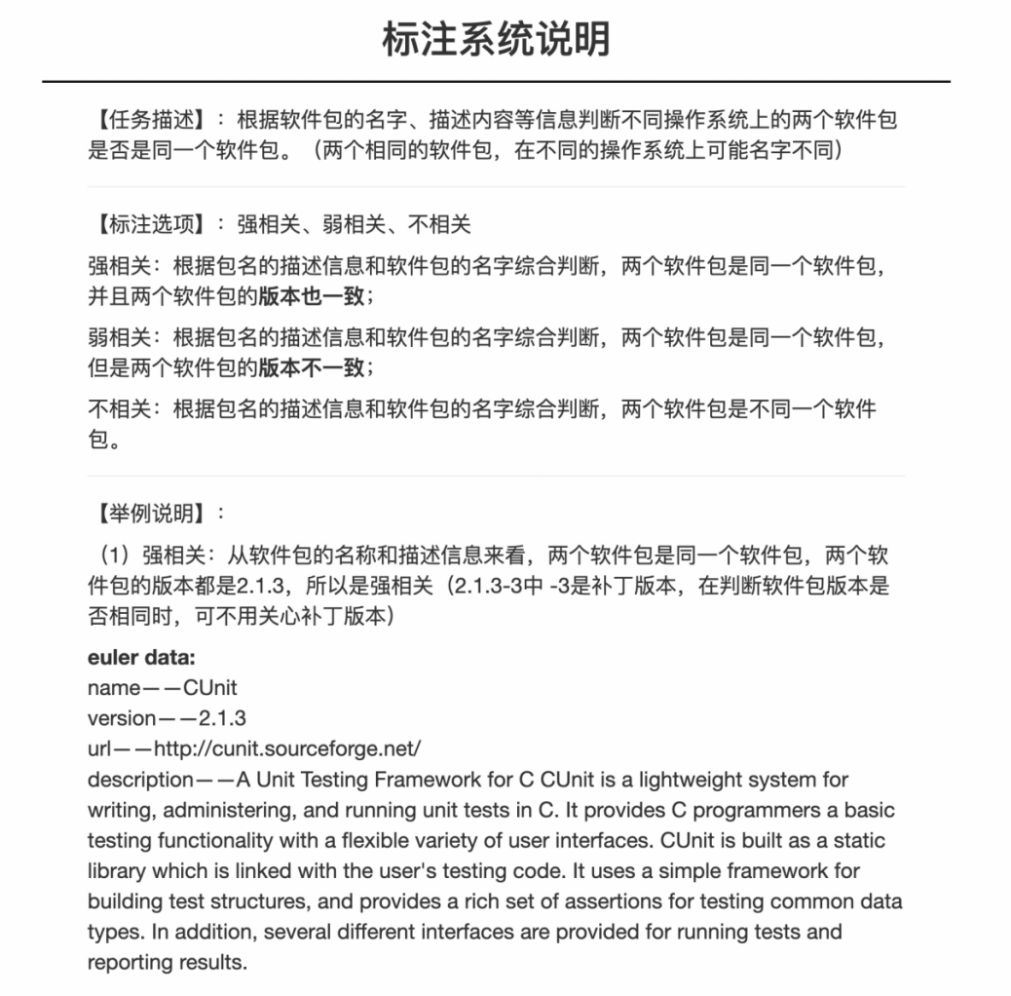


图4. 标注系统说明

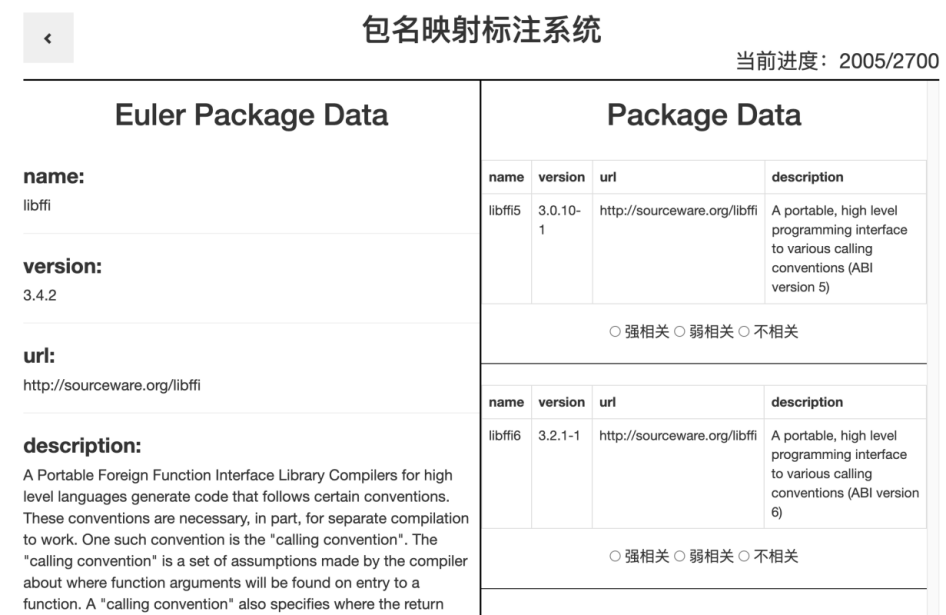


图5. 系统标注页面

1. 使用方法

python环境使用时需要安装tqdm nltk wget python-rpm-spec requests numpy pandas这些包

1. 按照数据样例准备输入数据

构建映射表用到的数据需要提前准备，euler软件包信息的获取方式位于mapping\_package包下的get\_euler\_package.py中，生成文件data/euler\_data/euler\_package.json

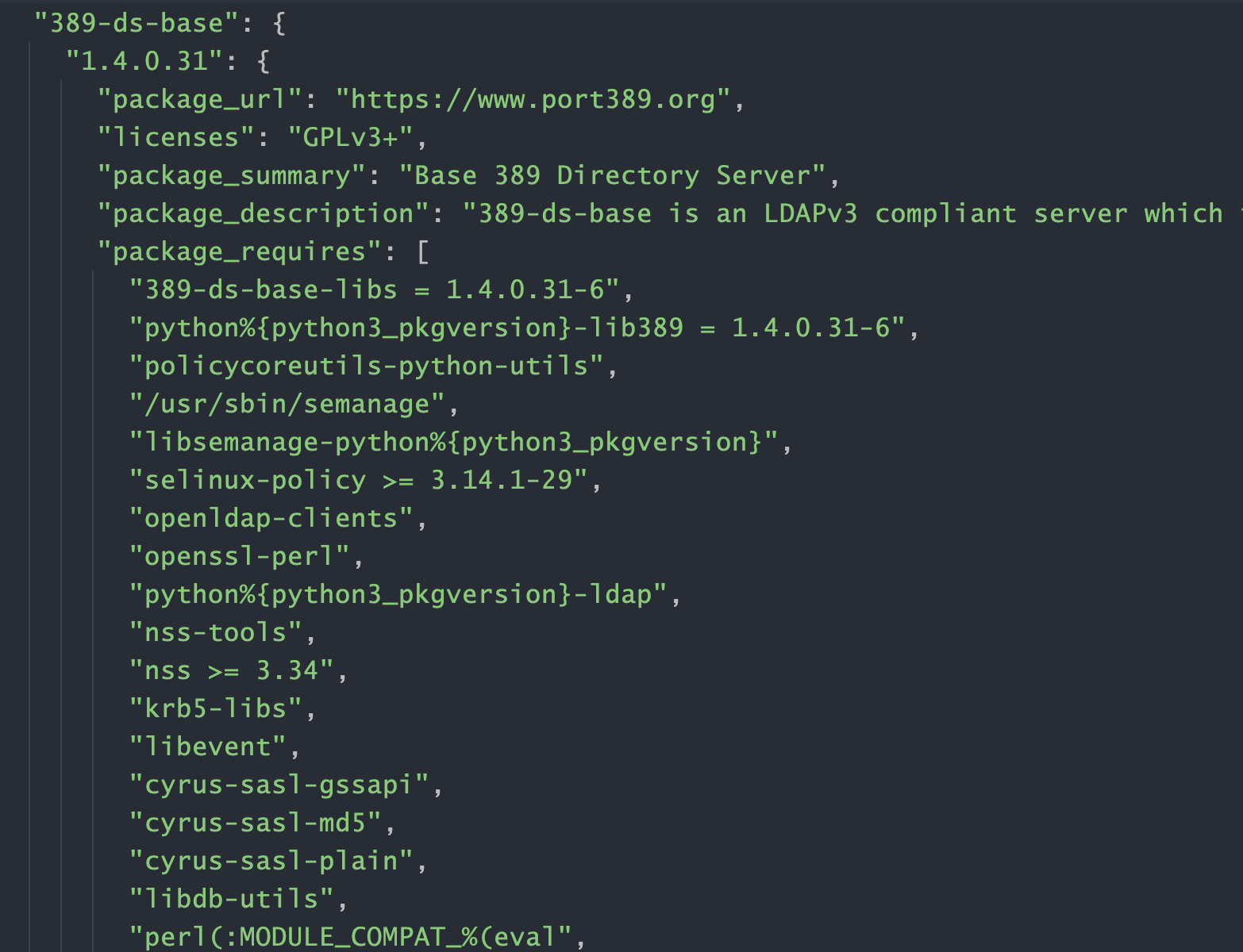


图6.euler\_package.json数据样例

运行new\_mapping\_package.py时会将euler\_package.json处理成映射所需要的格式euler\_data.json,该json文件已存储在data/centos\_euler/euler\_data.json和data/debian\_euler/euler\_data.json中，其余平台软件包也要同样处理成该格式进行映射。

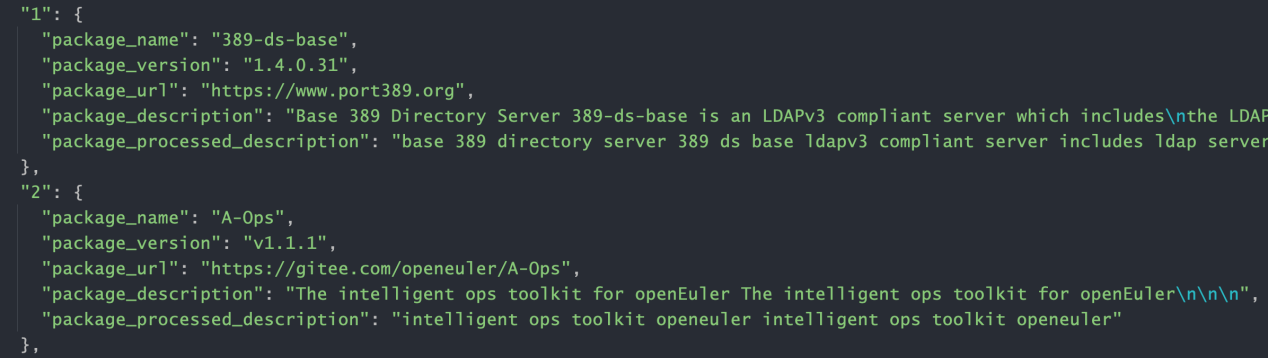


图7.euler\_data.json数据样例

centos软件包信息的获取方式位于mapping\_package包下的craw\_centos\_package.py中，已生成文件data/centos\_euler/centos\_data.json

debian软件包信息的获取方式位于mapping\_package包下的craw\_debian\_package.py中，已生成文件data/debian\_euler/debian\_data.json

1. 初始化“包名映射表”
   1. 获得相似软件包映射表

执行python new\_mapping\_package.py，得到软件包之间的相似性关系，选取排名前三的数据得到一个相似软件包映射表mapping\_package\_DICE.json,由人工标注确定最终映射表。如下图所示,一个euler软件包id对应三个最相似的软件包id。

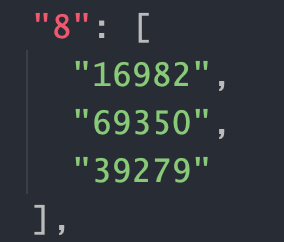


图8.mapping\_package\_DICE.json数据样例

* 1. 将json文件转为txt，导入标注系统数据库标注得到软件包映射表

执行new\_mapping\_package.py时使用data2txt函数将data/centos\_euler/euler\_data.json转化为txt文件data/centos\_euler/euler\_data.txt；data/centos\_euler/centos\_data.json转化为txt文件data/centos\_euler/centos\_data.txt，使用mapping2txt函数将data/centos\_euler/mapping\_package\_DICE.json转化为txt文件data/centos\_euler/mapping\_package\_DICE.txt，放入MarkWeb4.0下的src/com/Test进行导入。

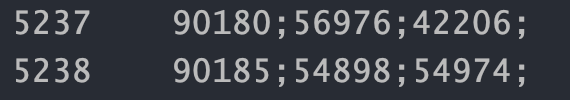


图9.mapping\_package\_DICE.txt文件

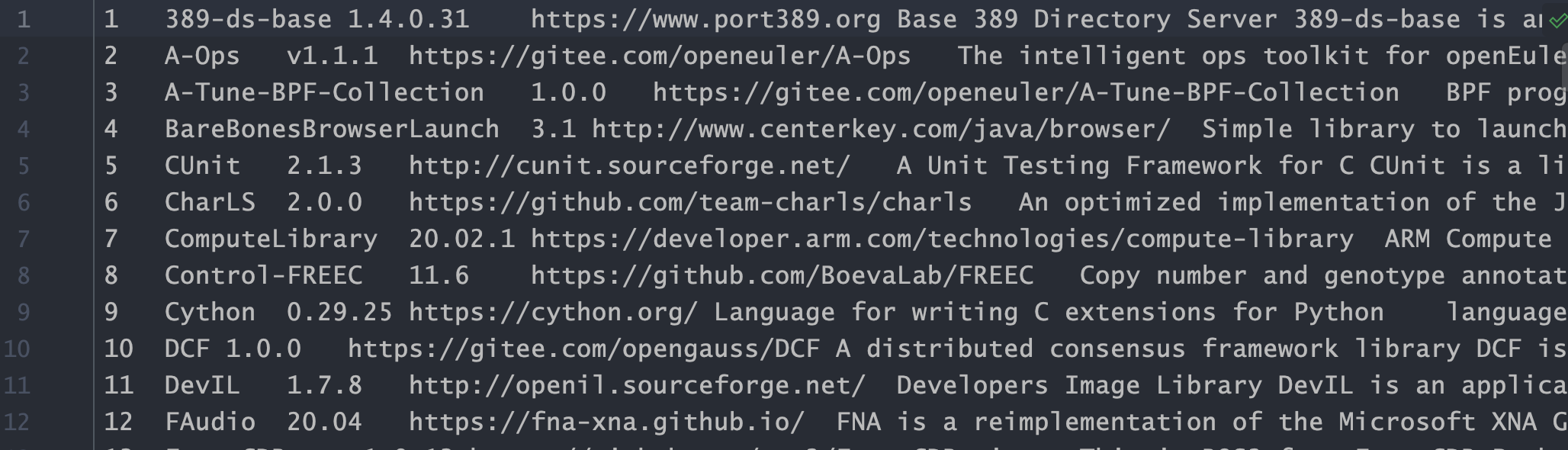
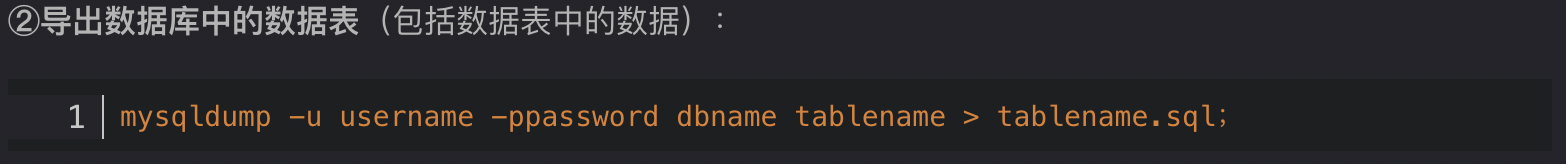


图10. 软件包信息txt文件

标记系统详细操作步骤见单独的标注系统运行说明

人工标注确认后,导出label\_result表中数据得到label\_result.sql



例如：mysqldump -u root -p12345678 mydb label\_result > ./Desktop/label\_result.sql

mydb为数据库名、label\_result为数据库中的表名、./Desktop/label\_result.sql

为文件存放路径

运行mapping\_package下process\_result\_tool.py文件得到包名映射表mapping\_package\_table\_centos2euler.json

1. 操作映射表

查找软件包：进入mapping\_package文件夹下，使用python search\_package.py --package\_name \*\* \*\*为其它平台要查找的软件包名称。