Team report

计 64 陶东来　学号:2016011322
计 62 张熙林　学号:2016050022

June 24, 2017

1 Reader

Reader 类是一个用来对文本预处理，并以 token 的形式将文本传递给 Detector 的类。由于自然语言与我们平时所用的程序设计语言判断抄袭的标准有着很多不同之处，如果我们直接将自然语言的分词方法套用到 c++ 等语言上，显然不会得到很好的效果。

一个比较直接的想法是对该文本进行语法层面的剖析。这也是一个事实上相对可行的方法，但是这就要求我们对于不同的语言设计不同的语法分析器。就算我们摸鱼，只试图实现对 c++ 的支持，其语法之复杂，令人望而生畏。

因此我暂时只实现了其极小的一部分，之后还会继续完善。

2 Charbuf

Charbuf 类是我为了实现 Reader 而添加的辅助类，主要目的是优雅地封装原本 ungetch() 和 getch() 的任务，以及实现对一些诸如预编译指令的去除。当然，宏展开这件事由"g++ -E" 实现。

3 Token

Token 类，顾名思义。

4 future improvement

1. 完善 Reader 类；
2. 实现对多种语言的支持。

5 Manual

Manual 类提供一个如何使用该程序的提示信息。

6 HashFactory

HashFactory 类中存在三个函数。
printInfo() 用来输出我们已有的 Hash 赋值方式。
ChooseHash() 用来通过传参的方式来获取进行那种 Hash 方式。
剩下的为具体 Hash 赋值的实现。

7 Detect

Detect 类为一个抽象类，其中含有一个 startDetect() 的纯虚函数，用来在积累中重新定义。之所以将 Detect 设为抽象类是因为，在以后的扩展中，可能会加入一些新的检查相似度的方法，这为以后的发展起了很大作用。

8 similarity

similarity 类是 Detect 类的一个派生类，其重新定义了 startDetect() 函数。它用来实现相似度检测的一个类，首先他会从一个 Reader 的函数接口获取一个文件被预处理之后的一个个单词，接下来会将这些单词赋予 Hash 值，另外在 similarity.h 中，提供了一个 Sig 结构，该结构用来存储一个文件的 Hash 值构成的特征信息。在将所有的文件都经过处理得到其 Sig 结构之后，我们调用 compare() 函数来获取其相似度。

9 使用及效果

```
E:\2017 Spr\OOP\Team project\project>main data/01/*.cpp
Please input a number of Ntoken to ensure how many words do you want to taken t
 use sherlock.(default value is 3)
3
Please input a number of the way to compute the value of hash.
1.Using multiply to ensure the value of hash.
2.Using FNV to ensure the value of hash.
3.Using SDBM to ensure the value of hash.
1
data/01/49_38a.cpp
data/01/639_536.cpp
data/01/91_a1.cpp
data/01/9l_3_2dEd.cpp
data/01/t61_31.cpp
data/01/49_38a.cpp and data/01/t61_31.cpp: 90%
data/01/91_a1.cpp and data/01/9l_3_2dEd.cpp: 90%
```

我们通过命令行参数传入所要进行检测的文件名，然后在提示信息下，我们输入一个数字代表一次性对几个连续的单词进行赋予特征值（Hash），然后第二个提示信息我们将输入一个代表 Hash 赋值方式的数字，目前为止，我们可以提供三种不同 Hash 赋值的方式，可以多次进行检测，增加准确度。接下来的几行则是输出所有待检测文件的文件名，最后会输出相似度。

因为是抄袭检测，所以我们设置了一个参数——20%，如果相似度低于 20