软件需求工程LAB3

开源社区SmartIDE的需求变化分析

学校：南京大学

院系：计算机科学与技术系

姓名：许家铭、黄逸维、陈永健

学号：171860581、171860542、171860568

邮箱：[381026323@qq.com](mailto:381026323@qq.com)

1. 概述

本次实验是针对SmartIDE的需求进行识别，找出其中的需求变更，并且排出时间轴。通过对开源社区的Smart IDE抓取他们开发的文档和代码变更，从中分析获取他们所对应的需求，分析其是否有变更过程。本次实验，我们选择了使用Word2Vec来进行自然语言处理和分析，尽我们所能的完成了本次实验。

1. 实验背景

在日常开发的过程中，我们经常会有出现需求变更的过程，项目经理发现某个初始需求并不能够满足用户，或者该需求有更好的实现办法时，往往会进行需求的变更。而这些变更，集中体现在了代码的变更（即commit）与issue的评论中。

实验任务：

1. 明确提出需求 R 的文本，获取需求 R 的有关讨论文本

从issue的comment中抓取出需求文本，将所有有关issue的内容全部抓取下来作为原始数据。

1. 识别出实现需求 R 的代码

根据选择的需求，从commit中选出对应的fix该issue的commit

1. 如果需求有变更，识别出需求变更

从抓取的comment中找出其中有需求变化的comment，统计需求变化的条数，超过一定标准，我们则认定为其存在需求变化。将所有存在变化的comment标注出来，根据此去识别需求变更代码

1. 给出需求 R 的全生命周期的时间线。

将抓取的comment和commit排成一个时间轴。

1. 名词解释

Word2Vec

Python的一种词转词向量的第三方库，用于将进行过分词处理的词转化为计算机可以使用的向量的形式。接着通过训练进行模型建立。

1. 实验方法

首先，我们选取了比较方便识别的有关vscode的Git支持的issue，抓取label有Git的所有issue。

接着我使用eclipse的bug report作为训练的语料库，进行模型训练。

我们把抓取的issue作为原始数据逐条输入，对每一个标题计算其与git的相关性，具体的计算方法为取所有词与git的相关性的平均值，即取

该值即为title。

接着我们把每一条comment放入其中同样求每一句话的相关性平均值，即为point，将计算所得的point与title比较，如果差的绝对值超过m，并且存在超过n条，则将该issue记为存在需求变更的需求。再次遍历文件，找到相应的需求变化的comment，在时间轴上标注出来。

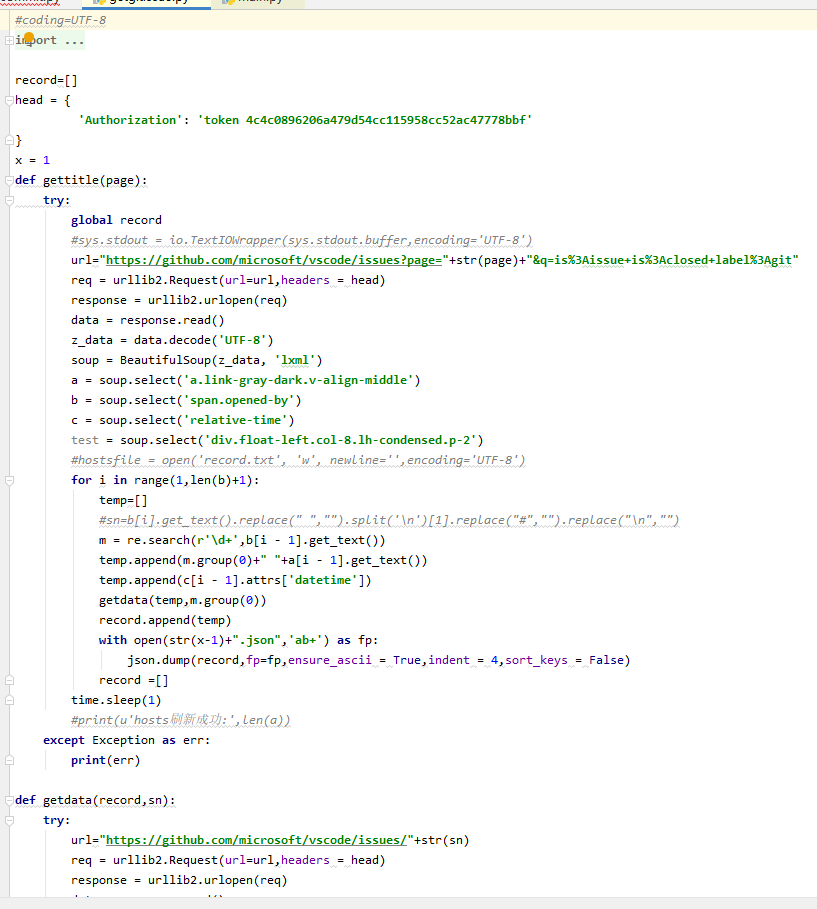
接着我们进行调参，调整m与n到一个相对合理的值。调整下来m为0.3，n为5.为了方便选取需求，我将变更数最大的issue找了出来并存储下来。

最后，根据识别出来的需求进行对应的commit抓取，并存放进入时间轴。

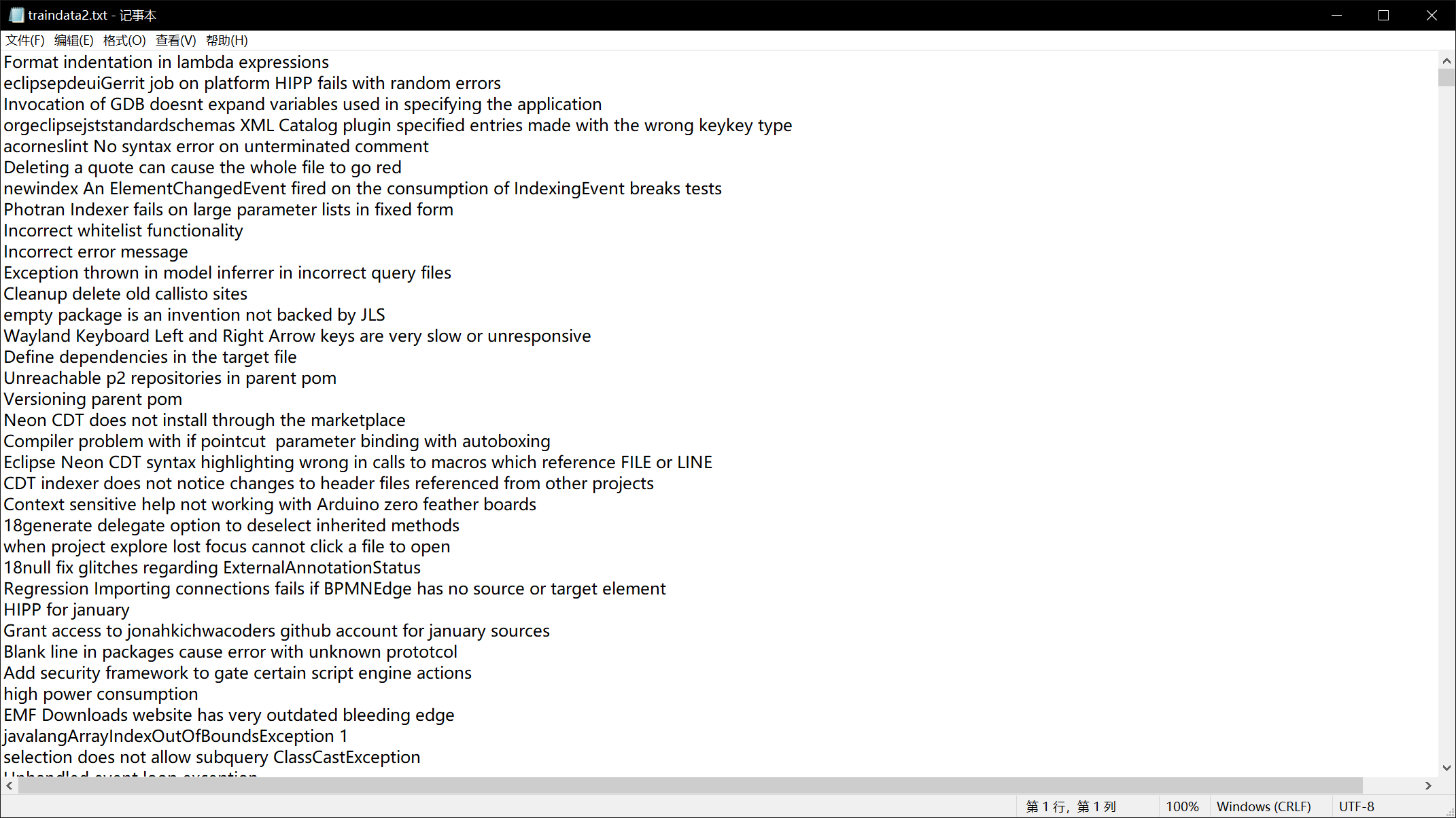
在最后的commit抓取过程中，我们本来打算使用issue编号抓取对应的commit，但遗憾的是vscode早期的commit并不规范，没有写明对应的issue，导致我们无法找到，于是我通过筛选找出了变化参数最大的那个open的issue，在此基础上，我找出了这个issue里面的关键词key，ssh，password，以此为查询依据进行排查，找出可能的所有commit并进行筛选，得出了最终结果。

1. 实验过程

Issue抓取：



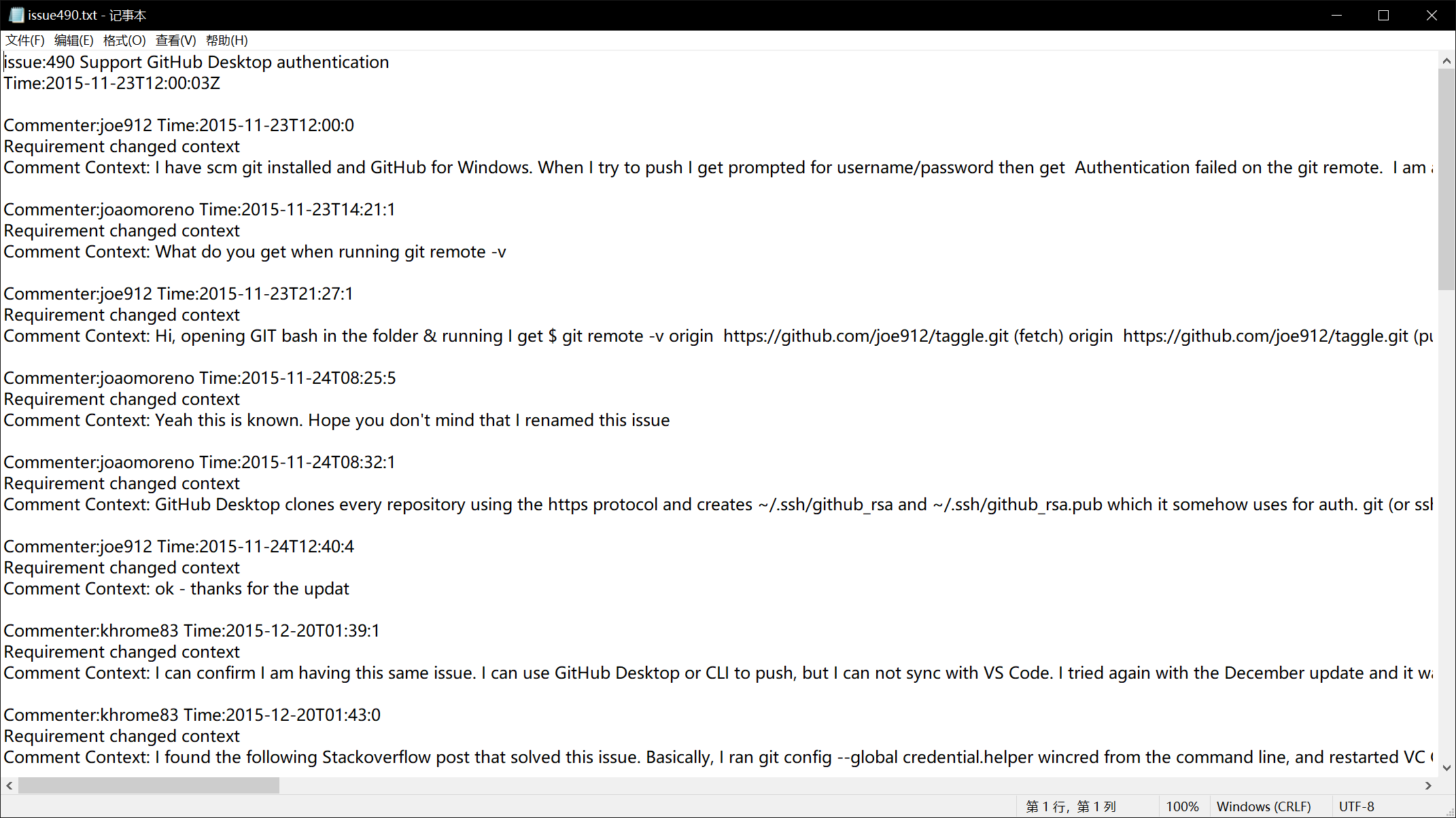
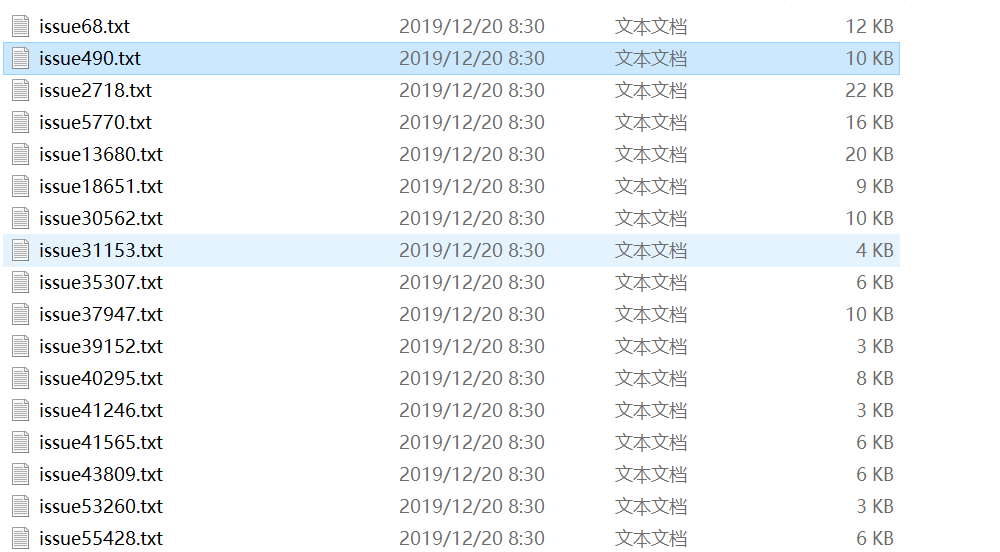
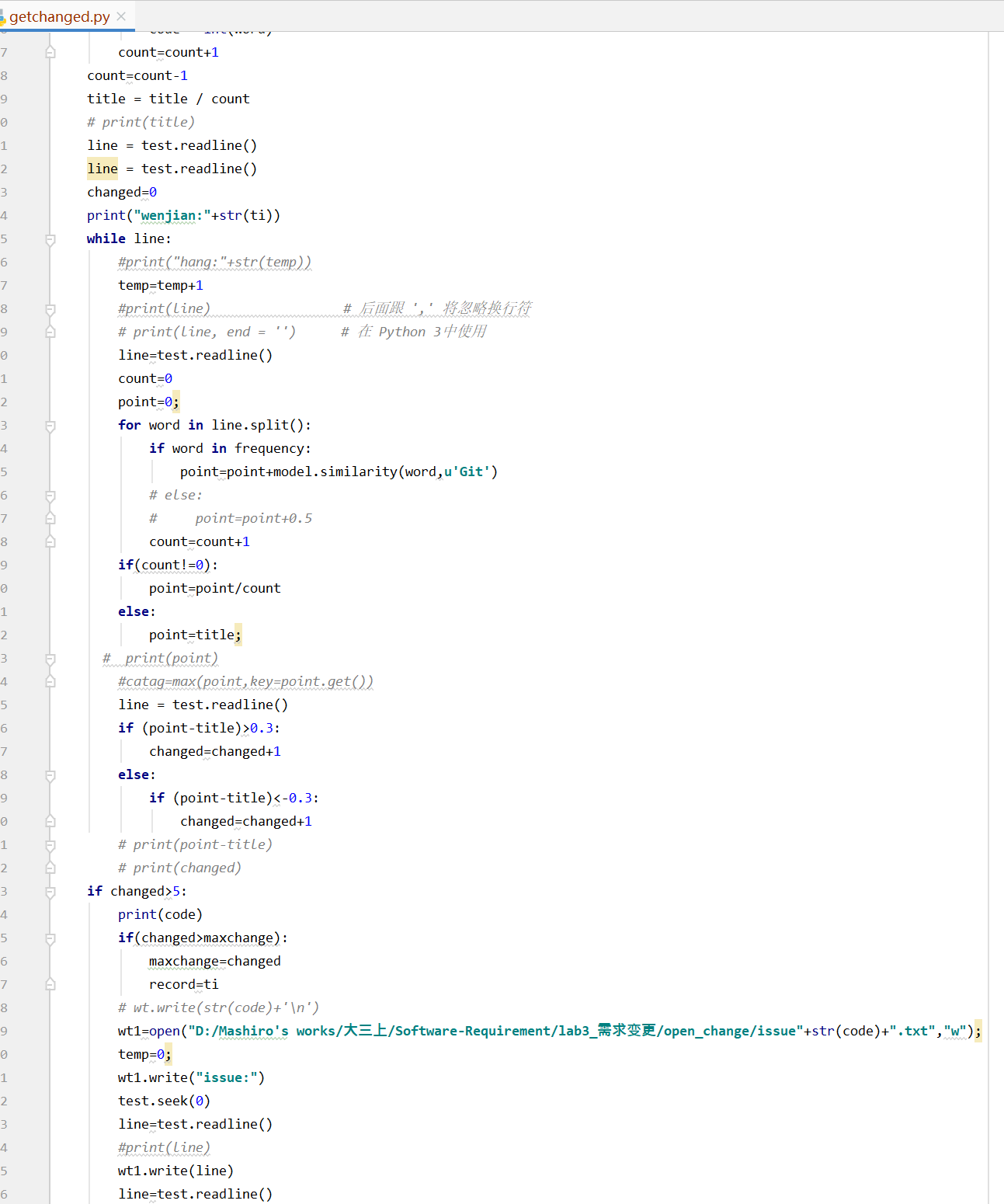
语料库建立：



原始数据处理：

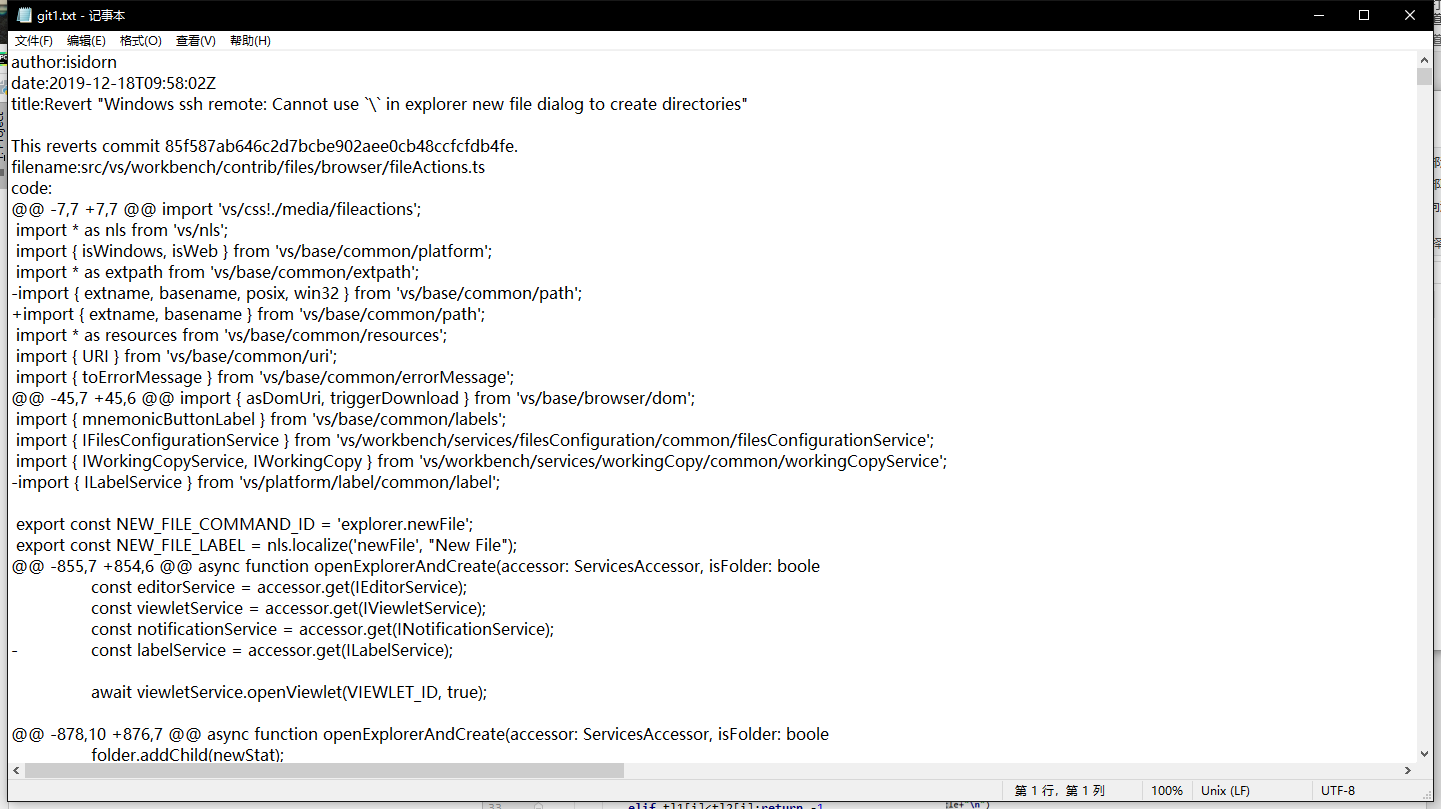


词转词向量、模型的建立、结果测试运行：



Commit抓取：





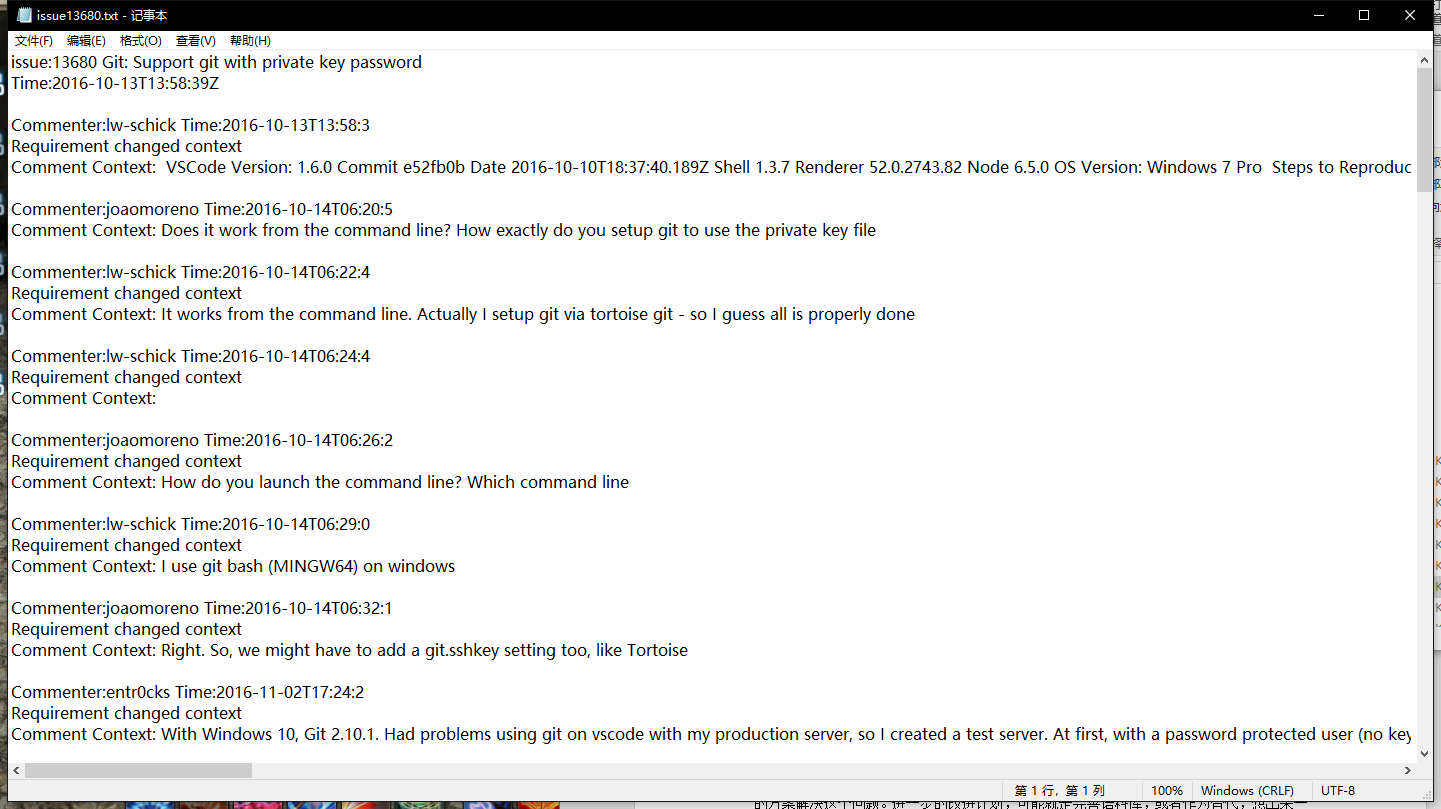
以上使用的所有程序均附在GitHub中

1. 实验结果

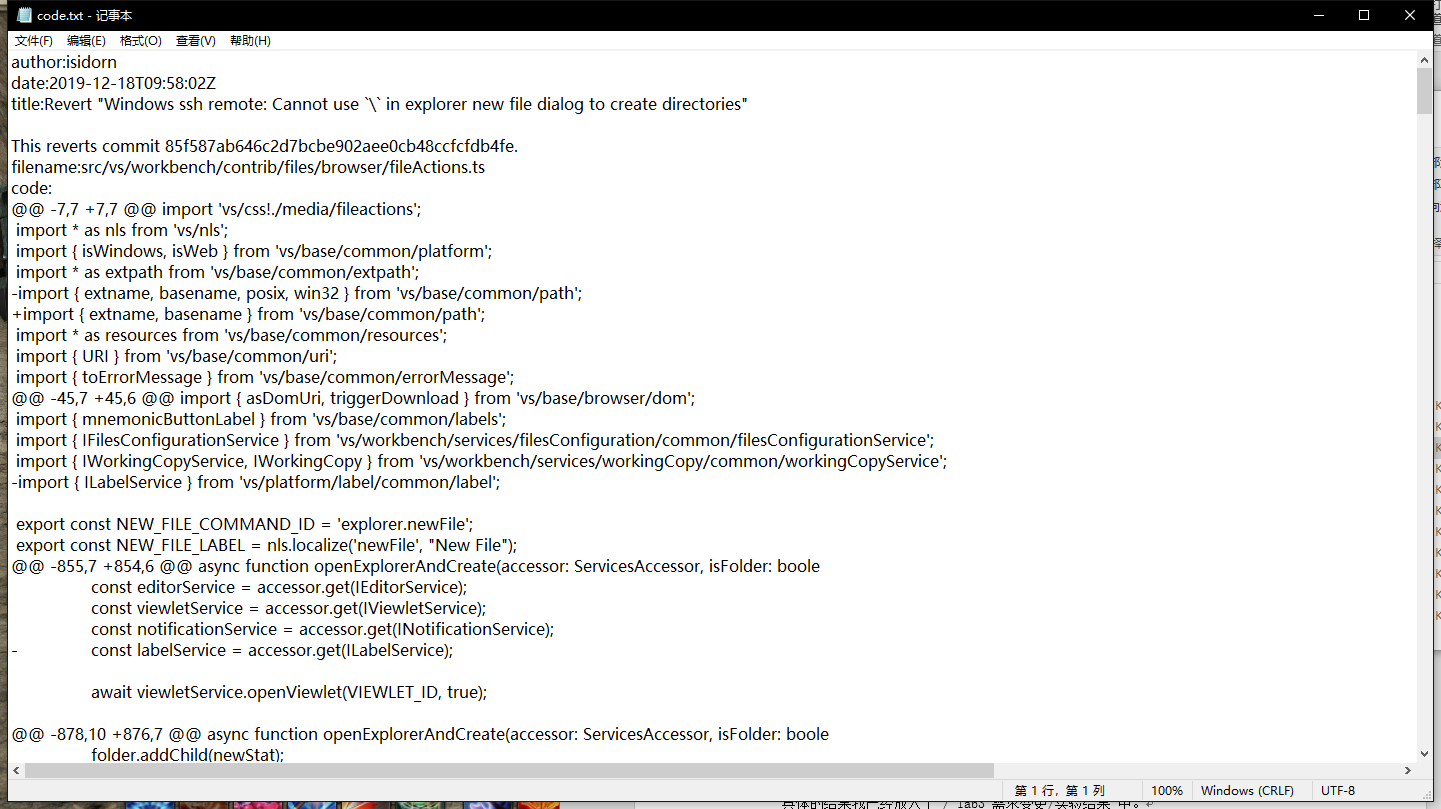
从结果来看，有一定准确性，能够识别出相应的需求变更代码和需求变更讨论文本。

结果示意图如下：

时间轴：



变化代码：



具体的结果我已经放入了~/ lab3\_需求变更/实验结果 中。

1. 结果分析

从我们测试所得的数据来看，我们分析出来的数据虽然有一定的精确度，但还是和人工分析有着一定的差距。

这些误差主要来源分别如下：

Word2Vec这个方法的第一个问题就在于，我没有对分词完的数据进行过滤。我没有来得及在做这个算法的时候，整理出一份过滤的名单，因此有很多无关紧要的词，比如be动词，介词等没有实际意义的词语也被我纳入了统计范围。如果需要改进，我可以通过过滤无实际意义的词语等方式，提高结果准确度。

第二个问题在于，我的语料库不够全，由于没有现成的语料库，我的词语模型不够全，很多的词语都没有，计算相关性的时候可能会不够准确，由此我们产生了第三个问题。

第三个问题在于，我的词语不在语料库中的词，处理不够完善，我没有想出一个比较好的方案解决这个问题。进一步的改进计划，可能就是完善语料库，或者作为替代，想出来一个处理不在语料库中的词的相关性解决方案。

第四个问题在于我的参数调节，我的调节参数标准比较主观，主要根据调出来的数量和我自己的理解来决定效果的好坏。如果要进行改良，可以整理出一种比较好的调参方式和标准。

1. 结论

本次实验，我们使用了Python的gensim库进行了自然语言处理，取得的结果还是有精确性的，但是还是存在一定误差。在后续实验中，我们可以添加白名单，优化语料库，完善不在语料库中的词的相关性计算等方式进行实验的细化。

在本次实验中，我们组的分数分配为每个人得分1/3。