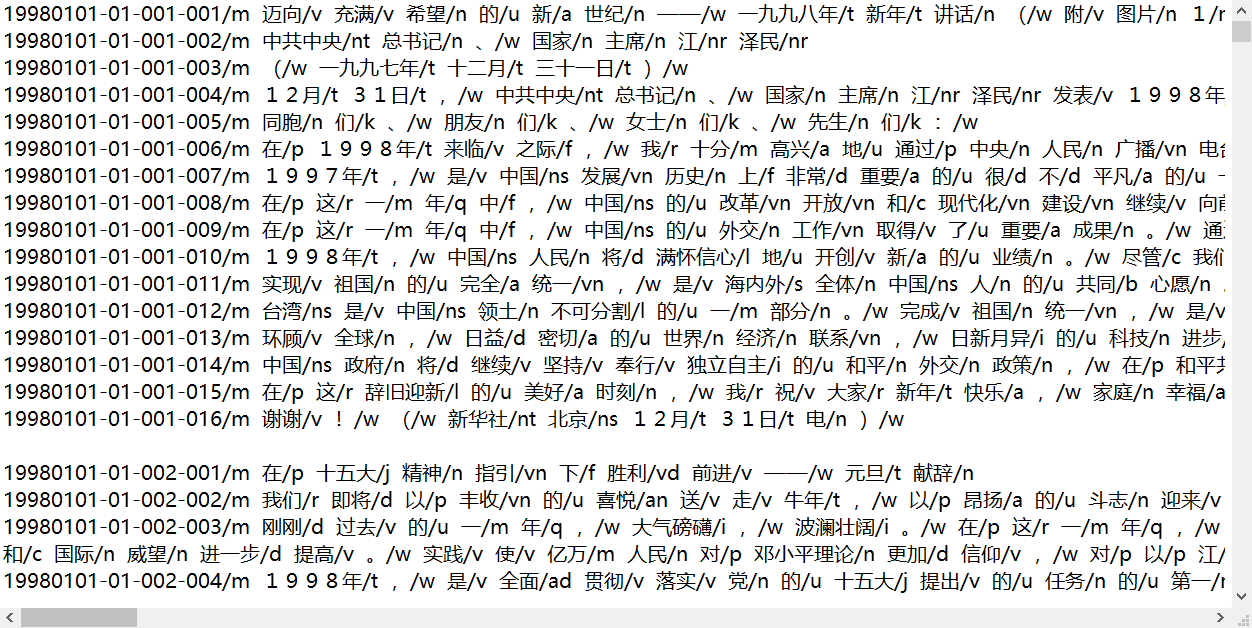
**VSM-LSI分析**

1. **工作概要**

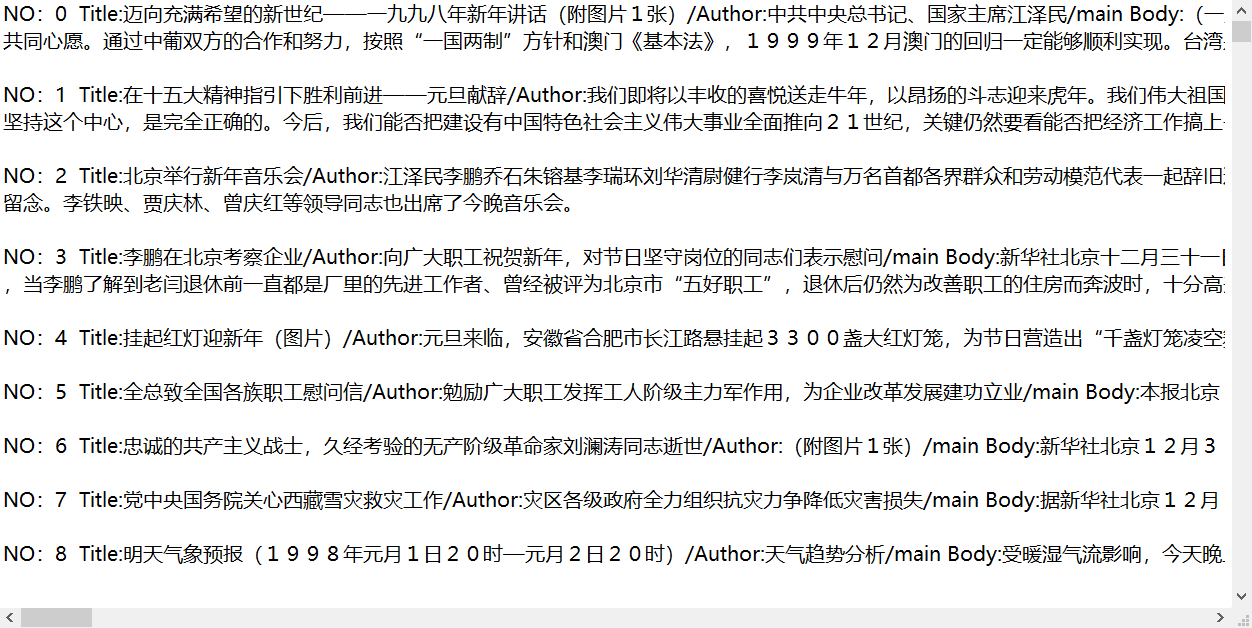
第二次作业是对VSM和LSI的简单使用，并体会它们的用处和不同，感谢前面几位同学的作业，从而了解到可以使用gensim这个开源库，并在此基础上进行了VSM-LSI的相似度计算和比较

1. **语料预处理**

首先，由于老师给的文档不是连续的，如下图，一篇文档分成了若干部分，并以空格隔开，所以首要任务就是把原来是一个文档的数据放到一起。



即如第一个同学分享的txt：



不过这里没必要弄成如上的样子，不然还得划分，上面的可以用来对结果比对时分析，下面的代码将相同文章的词进行了合并，得到词袋：

|  |
| --- |
| **""" 函数说明：语料预处理 输入：要打开的文件名 输出： documents - 处理过后的词空间列表 """ def** load\_data(filename =**'199801\_clear.txt'**):  *# 文档分词列表* documents = []  *#文档分词* doc = []  *# 打开语料库文件，逐行读取语料，将同一篇文章的不同章节合并，并过滤掉无意义的词和符号* fr\_open = open(filename, **'r'**, encoding=**'GBK'**)   **for** line **in** fr\_open:  **if** line.strip(): *# 跳过空行,这里strip（）无参数是去掉首尾空格* words = line.split()  *# 过滤掉无意义的词和符号* **for** i **in** range(len(words)):  word = words[i]  **if 'w' not in** word **and 'y' not in** word **and 'u' not in** word \  **and 'c' not in** word **and 'm' not in** word: doc.append(word)  **else**: *#出现空行，即将开始下一篇文档读入，所以把上一篇文档添加进documents，并将doc置为初始* documents.append(doc)  doc = []  **return** documents |

1. **TF-IDF权重**

再对文本进行重组后就可以统计它的词频了，并根据词袋计算它们的tfidf值，这里使

用gensim的corpora进行词频统计和计算权重，代码注释已经很详细了

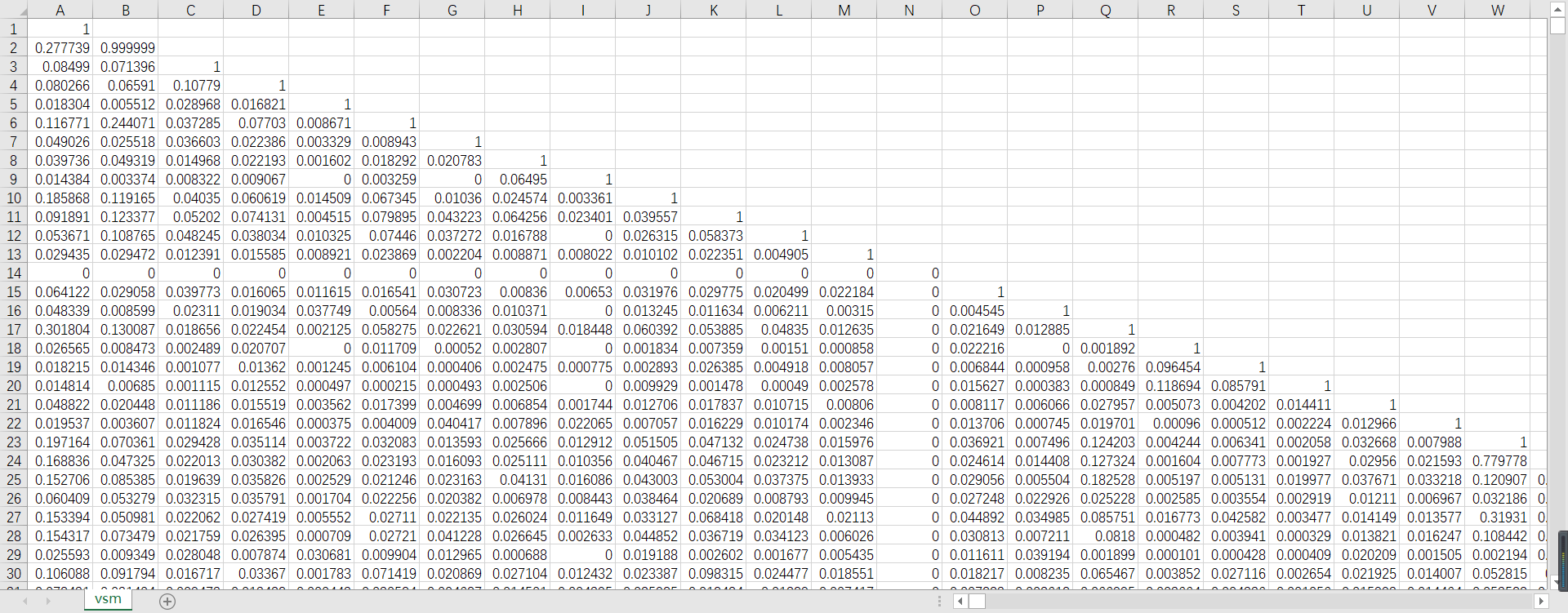
|  |
| --- |
| **""" 函数说明：统计并计算tf\_idf权重值 输入：documents - 处理过后的词空间列表 输出：corpus\_tfidf - tfidf的词向量空间 """ def** tf\_idf(documents):  *# 将documents放入一个字典,这个字典默认按照一定的顺序排列documents* dictionary = corpora.Dictionary(documents)  *#转换成词频，会去掉每个文本的重复词，并自动生成词向量空间* corpus = [dictionary.doc2bow(text) **for** text **in** documents]  *# 将corpus作为初始化的参数，得到tfidf关于corpus的模型* tfidf = models.TfidfModel(corpus)  *# 通过tfidf【】将词频转换成tfidf值* corpus\_tfidf = tfidf[corpus]  **return** corpus\_tfidf |

1. **VSM及结果分析**

将上一步得到的词向量空间计算它们的相似度，这里不能直接用similarities.MatrixSimilarity（）函数，会出现内存不够，应该先将其持久化到磁盘，再计算相似度，相当于扩大了虚拟内存。然后再将相似度输出到csv文件里面，因为相似度矩阵是一个对称阵，所以我输出了一个下三角型矩阵（节约存储空间），最后可以查找某一个文本和其他文本的相似度，我这里返回前3个最大相似度的文本的序号

|  |
| --- |
| **""" 函数说明：进行LSI分析，并计算各个文档之间的相似度 输入：corpus\_tfidf - tfidf的词向量空间 输出：CSV格式的VSM下三角相似度矩阵 """ def** VSM(corpus\_tfidf):  *# 必须先把corpus\_tfidf写入磁盘，因为矩阵太大，直接在内存计算相似度会导致内存不够* corpora.MmCorpus.serialize(**'./vsm\_test.mm'**, corpus\_tfidf)  corpus\_tfidf = corpora.MmCorpus(**'./vsm\_test.mm'**)  *# 用文档向量初始化一个相似度计算的对象  # index = similarities.MatrixSimilarity(corpus\_tfidf)* index = similarities.SparseMatrixSimilarity(corpus\_tfidf)  *#输出下三角矩阵* **with** open(**'./vsm.csv'**, **'w'**) **as** fr:  i = 1  **for** sims **in** index[corpus\_tfidf]:  a = sims[:i]  fr.write(**','**.join(map(str, a)) + **'\n'**)  i += 1  num = 1 *#查找第num篇文档的相似度，可自行修改* query = sorted(enumerate(index[corpus\_tfidf[num]]), key=**lambda** x: x[1], reverse=**True**)  *#返回最相似的前三个文档的序号* print(query[:4]) |

输出的下三角形矩阵：



可以再excel里面查找相似度较高的文章，这里我搜寻了大于0.8的值，如101行BH列的值为0.823，即第101篇和第60篇的相似度为0.823，它们是：

|  |
| --- |
| NO：59 Title:居民生活如常事业蒸蒸日上香港特区政府发表回归半年大事录/Author:据新华社香港１月１日电香港特区政府在１９９７年的最后一天，发表了《香港特别行政区———首六个月大事录》，认为香港回归后的情况“粉碎了一些对７月１日后香港变得灰暗的预言”。/main Body:大事录详细记录了香港回归以来政治、经济、法治、人权、教育、基本建设等方面的重大事项。包括：香港成为中国的一个特别行政区后，在“一国两制”方针政策下运作；基本法保障了香港资本主义制度及原有生活方式五十年不变；发行自己的货币；保持原有税制、独立关税区地位和公务员制度等。这份特区政府的报告长达２０页。文中说，香港顺利回归祖国，香港居民生活如常，正如基本法所保证的一样。报告说，回归两个月后，香港成功举行了世界银行和国际货币基金组织理事会年会，数千名来自各国和地区的财长、央行行长及官员和２０００多传媒人士，在香港得到了第一手的特区运作观感，香港回归后的情况赢得了人们的信心。报告认为，虽然亚洲出现了金融风波，但香港经济仍然跨步向前。报告重申了特区政府维持联系汇率制度的决心，指出香港拥有多种有利因素，包括：比邻近地区通胀率低、香港银行业采取审慎贷款及维持高资本充足比率的政策、租金和楼价下调、联系汇率可稳定汇率等。因此，“假若情况好转，本港的复原速度亦会较快”。报告还记录了行政长官董建华去年９至１１月的外访活动和１０月发表首份施政报告。  NO：100 Title:谱写脱贫致富新篇章——吕梁扶贫开发与计划生育相结合工作纪实/Author:赖林嵩周美林范双民张振明/main Body:山西省吕梁地区是个贫困大区。全区１３个县市中有９个是贫困县，其中国家重点扶持的贫困县有７个。１９７１年建区时，总人口２２６．６万，出生率高达３３．１０‰，死亡率８．４‰，人口自然增长率２４．７０‰。１９７６年开始推行计划生育，无计划生育势头有所遏制。近几年来，全区上下结合实施“八七扶贫攻坚计划”，按“经济要上去，人口要下来”的方针，计划生育工作连年取得好成绩。１９９６年，全区总人口控制在３２４万人，出生率下降到１４．５５‰，自然增长率９．４３‰。贫困人口由１９９２年的１０５万人减少到１９９６年的８０．４万人。吕梁地区的计划生育工作坚持服从、服务于经济建设这个中心，服从、服务于扶贫攻坚这个历史性任务，努力实现计划生育工作的“两个转变”，走出了一条扶贫开发与计划生育相结合的希望之路、成功之路。党政重视，步入人口发展的良性循环轨道吕梁地区坚持不懈、持之以恒地严格控制人口增长。他们采取的主要措施有：———明确一把手负总责。１１年来，一直把计划生育工作作为“一把手工程”来抓，地委、行署每年召开计划生育工作专题会议都在三次以上，每次会议都要切实解决几个实质性问题。每年还要定期召开一把手“交账会议”，既交经济账，又交计划生育账。———加强基层基础工作不松劲。全区各县市都建立了计划生育服务站；２３６个乡镇全部建立了计划生育服务所；４６２６个村配齐了计划生育服务员，人员、经费、待遇得到落实。———确保经费投入逐年增加。地区计划生育经费已由１９９０年前每年２０万元逐年增加到１９９７年的３００万元。县市财政用于计划生育的经费由１９９０年的１１７．６万元（人均０．６０元），增长到１９９６年的５９０万元（人均１．８５元）。穷则思变，走扶贫开发与计生相结合之路吕梁的特点是贫困人口占全省的１／３左右。吕梁的干部群众认识到，扶贫开发与计划生育相结合，是解决贫困地区人口与发展问题的必由之路。１９９０年，地委、行署在总结经验的基础上正式出台《关于扶贫开发与计划生育工作相结合的意见》，在全区实施。结合学习外地经验，吕梁地区扶贫开发与计划生育相结合工作开始形成党政牵头、部门配合、计生协调、群众参与的格局。柳林县下三交镇坪上村原是个有名的老大难村，党支部、村委会班子瘫痪，多年来人均收入徘徊在一二百元之间。１９８６年全村２４２名已婚妇女，有半数多超计划生育。这一年新党支部书记刘步飞上任，他带头只生了一个女孩，拿出自己放电影补助费捐赠给计划生育贫困户，垫支７０００元兴修水利、引水浇地，还带领群众创办了“少生快富甜蜜公司” |

看标题NO：59Title:居民生活如常事业蒸蒸日上香港特区政府发表回归半年大事录和NO：100 Title:谱写脱贫致富新篇章，知两个都是与人民生活越来越好相关。还是比较相关。

1. **LSI及结果分析**

前面的数据处理过程都一样，就是得到tfidf权重值。LSI主要是对矩阵用

SVD进行了一个降维，根据SVD，任何m\*n的矩阵M，都存在以下SVD分解：

C:\Users\xiang\AppData\Local\Temp\1542978443(1).png

通过几何意义对其证明如下：

假设存在M\*N矩阵A，事实上，A矩阵将n维空间中的向量映射到k（k<=m）维空间中，k=Rank(A)。现在的目标就是：在n维空间中找一组正交基，使得经过A变换后还是正交的。假设已经找到这样一组正交基：



则A矩阵将这组基映射为：

https://img-blog.csdn.net/20150123160626263

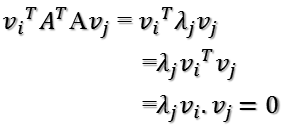
如果要使他们两两正交，即

https://img-blog.csdn.net/20150123160744762

根据假设，存在

https://img-blog.csdn.net/20150123160916671

所以如果正交基v选择为的特征向量的话，由于是对称阵，v之间两两正交，那么



这样就找到了正交基使其映射后还是正交基了，现在，将映射后的正交基单位化：

因为

https://img-blog.csdn.net/20150123161911890

所以有

https://img-blog.csdn.net/20150123162005218

所以取单位向量

https://img-blog.csdn.net/20150123162032674

由此可得

https://img-blog.csdn.net/20150123162324773

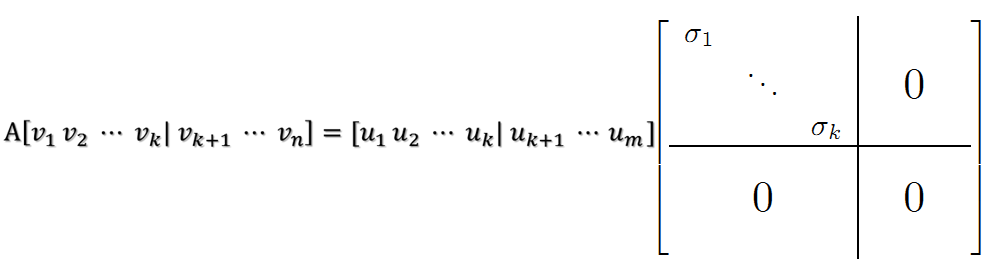
当k < i <= m时，对u1，u2，...，uk进行扩展u(k+1),...,um，使得u1，u2，...，um为m维空间中的一组正交基，即

https://img-blog.csdn.net/20150123162811221

同样的，对v1，v2，...，vk进行扩展v(k+1),...,vn（这n-k个向量存在于A的零空间中，即Ax=0的解空间的基），使得v1，v2，...，vn为n维空间中的一组正交基，即

https://img-blog.csdn.net/20150123202328388

则可得到



继而可以得到A矩阵的奇异值分解：

https://img-blog.csdn.net/20150123170018218

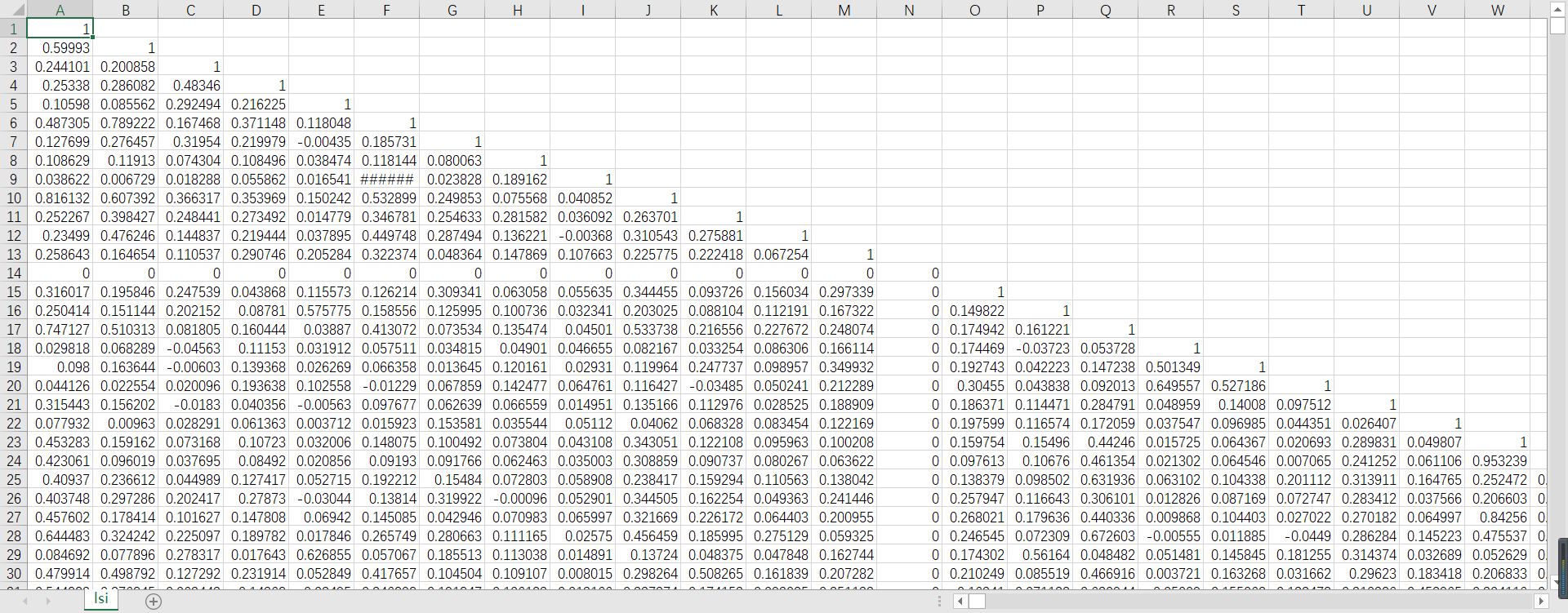
Σ是一个对角矩阵，其中的元素值就是奇异值，并且按照从大到小的顺序排列。在很多情况下，前10%甚至更少的奇异值的平方和就占了全部奇异值平方和的90%以上了，因此可以用前k个奇异值来近似描述矩阵：

C:\Users\xiang\AppData\Local\Temp\1542978451(1).png

使用models.LsiModel()函数构造LSI模型并计算相似度：

|  |
| --- |
| **""" 函数说明：进行LSI分析，并计算各个文档之间的相似度 输入：corpus\_tfidf - tfidf的词向量空间 输出：CSV格式的LSI下三角相似度矩阵 """ def** LSI(corpus\_tfidf):  *# 构造LSI模型并将待检索的query和文本转化为LSI主题向量  # 转换之前的corpus是BOW向量* lsi\_model = models.LsiModel(corpus\_tfidf,num\_topics=80)  corpus\_lsi = lsi\_model[corpus\_tfidf]  *# 转换成潜语义文档向量列表并持久化* corpora.MmCorpus.serialize(**'./lsi\_test.mm'**, corpus\_lsi)  corpus\_lsi = corpora.MmCorpus(**'./lsi\_test.mm'**)  *# 构建文档相似度矩阵索引用于查询，再使用文档列表本身进行相似度查询（默认使用 Cosine）* index = similarities.MatrixSimilarity(corpus\_lsi)  *# 输出下三角矩阵* **with** open(**'./lsi.csv'**, **'w'**) **as** fr:  i = 1  **for** sims **in** index[corpus\_lsi]:  a = sims[:i]  fr.write(**','**.join(map(str, a)) + **'\n'**)  i += 1 |

在降维后，再来计算它们的相似度，可以得到：



下面是从中选取的两篇相似度较高的文档的分析：

|  |
| --- |
| NO：0 Title:迈向充满希望的新世纪——一九九八年新年讲话（附图片１张）/Author:中共中央总书记、国家主席江泽民/main Body:（一九九七年十二月三十一日）１２月３１日，中共中央总书记、国家主席江泽民发表１９９８年新年讲话《迈向充满希望的新世纪》。（新华社记者兰红光摄）同胞们、朋友们、女士们、先生们：在１９９８年来临之际，我十分高兴地通过中央人民广播电台、中国国际广播电台和中央电视台，向全国各族人民，向香港特别行政区同胞、澳门和台湾同胞、海外侨胞，向世界各国的朋友们，致以诚挚的问候和良好的祝愿！１９９７年，是中国发展历史上非常重要的很不平凡的一年。中国人民决心继承邓小平同志的遗志，继续把建设有中国特色社会主义事业推向前进。中国政府顺利恢复对香港行使主权，并按照“一国两制”、“港人治港”、高度自治的方针保持香港的繁荣稳定。中国共产党成功地召开了第十五次全国代表大会，高举邓小平理论伟大旗帜，总结百年历史，展望新的世纪，制定了中国跨世纪发展的行动纲领。在这一年中，中国的改革开放和现代化建设继续向前迈进。国民经济保持了“高增长、低通胀”的良好发展态势。农业生产再次获得好的收成，企业改革继续深化，人民生活进一步改善。对外经济技术合作与交流不断扩大。民主法制建设、精神文明建设和其他各项事业都有新的进展。我们十分关注最近一个时期一些国家和地区发生的金融风波，我们相信通过这些国家和地区的努力以及有关的国际合作，情况会逐步得到缓解。总的来说，中国改革和发展的全局继续保持了稳定。在这一年中，中国的外交工作取得了重要成果。通过高层互访，中国与美国、俄罗斯、法国、日本等大国确定了双方关系未来发展的目标和指导方针。中国与周边国家和广大发展中国家的友好合作进一步加强。中国积极参与亚太经合组织的活动，  NO：47 Title:北国有寒趣南疆闻花香各地群众喜庆元旦/Author:据新华社北京１月１日电１９９８年元旦之际，从寒峭的北国大地，到如春的海南宝岛；从美丽的东海之滨，到莽莽的西部边陲，全国各地各族人民纷纷喜庆佳节，决心在新的一年里，高举邓小平理论的伟大旗帜，在党的十五大精神指引下胜利前进。/main Body:元旦的北京，节日气氛格外浓厚。清晨，来自祖国各地及港澳台地区的４万多人，冒着零下７摄氏度的严寒，自发地聚集到了祖国的心脏———天安门广场，参加了庄严的元旦天安门广场升国旗仪式。在长安街沿线及大街小巷的高大建筑物上，国旗、彩旗飘扬。京城各大公园、商场及各种娱乐场所也都充满了欢乐气氛。巨大的氢气红灯笼和各种颜色的彩球飘荡在冬日京城的天空中。今天，哈尔滨市的最低气温达到零下２５摄氏度，但市民们欢庆新年的热情却没有丝毫减弱。在冰雪覆盖的松花江上，刚刚开放的江上游乐园成了人们游玩的好去处。一座座冰雕构成了另一个崭新的世界，人们穿梭其间，尽情地享受冰雪带来的无穷乐趣。与寒冷的哈尔滨形成鲜明对照，南国春来早，花城今更艳。’９８广州风情贺年缤纷大巡游今天在广州隆重举行，为广州新年第一天增添了喜庆的节日气氛，同时也拉开了’９８华夏城乡游（广州）活动的序幕。来自香港特区１０多个团体的４００多位香港同胞首次来广州参加此次贺年活动。海口市阳光明媚。 |

这两篇文档一个是喜庆新年，一个喜庆元旦，具有一定的相似度。

1. **VSM和LSI比较**

首先看运算时间，这里因为前面的数据预处理和，tfidf权重计算都一样，所以就没加入，只考虑了后面VSM和LSI处理的计算时间

|  |
| --- |
| **""" 函数说明：进行测试 """ def** test():  *#加载数据并生成tfidf词向量空间* documents = load\_data(**'199801\_clear.txt'**)  corpus\_tfidf = tf\_idf(documents)  *#对词向量空间进行VSM的相应处理并计时* VSMstart\_time = time.time()  VSM(corpus\_tfidf)  VSMend\_time = time.time()  print(**'VSM耗时：'** + str(VSMend\_time - VSMstart\_time) + **'s'**)  *# 对词向量空间进行LSI的相应处理并计时* LSIstart\_time = time.time()  LSI(corpus\_tfidf)  LSIend\_time = time.time()  print(**'LSI耗时：'** + str(LSIend\_time - LSIstart\_time) + **'s'**) |

测试结果：

C:\Users\xiang\AppData\Local\Temp\1542979165(1).png

C:\Users\xiang\AppData\Local\Temp\1542979173(1).png

可以看出LSI大概比VSM多用10S，基本上是矩阵进行SVD分解，和奇异值计算造成的。但原本的向量空间太过于稀疏，有很多的0，进行降维后压缩了矩阵，也去除了噪音，使得结果更加精确。

并且VSM无法处理文本中隐含的潜语言和一次多意，而LSI在压缩矩阵时，减轻了此问题，但SVD主动舍去了一部分不重要的特征，所以在降维的同时也要承担数据损失的风险。

1. **参考**

<https://www.cnblogs.com/lzllovesyl/p/5243370.html> SVD推导

<https://blog.csdn.net/u011630575/article/details/80159100> SVD推导

<https://blog.csdn.net/zhongkejingwang/article/details/43053513> gensim