海量数据处理第 2 次作业

–用 VSM 计算文档间相似度

张盛强 1801210730

zhangsq5829@pku.edu.cn

1. Abstraction

本文用 python 实现 VSM 来计算文档间的相似度，在包含大约 3000 篇文档的部分《人民日报》语料上测试我们的程序，总计耗时大约为 30 秒。根据对结果的简单分析，与每篇文档最相似的 3 篇文档基本可以认为与源文档主题一致。

详细代码可以在我的 Github 上找到：https://github.com/Shengqiang-Zhang/documents-similarity

2. Brief Introduction to VSM

VSM 的基本思想是将每篇文档表示为一个向量，向量中的每个值可以用每个词的 tf-idf 值计算得到。每篇文档之间的相似度就可以用每个文档向量间的距离 (在本文中，我们用余弦距离) 来近似度量。

3. Tricks to Improve the Efficiency

- 大量使用 hash 表来存储不需要在计算向量间余弦距离时实时计算的量，例如：
  - 计算 tf-idf 时需要的每篇文档的词频表，整个文档集合的反比文档频率表
  - 预计算每篇文档向量并保存
  - 预计算每篇文档向量的 L2 范数并保存
- 对每篇文档，计算出该文档的 tf-idf 后，按照 tf-idf 值从大到小将词排序，抽取出可以表示总 tf-idf 和值 80% 的词，并将这些词的 tf-idf 值用来表示该篇文档的向量，其余的词舍弃。
- 计算向量间余弦距离时，在计算向量间内积时并不采用将两篇文档对齐为相同维度再直接内积的方式，而是以长度小的向量为基准，将相同词的 tf-idf 值相乘，最后对所有的积求和。
- 在生成所有文档间相似度矩阵时，因为矩阵为对称矩阵，所以只计算下三角矩阵或上三角矩阵。

4. Brief Introduction to Codes

4.1. 语料预处理

数据预处理部分在 `dictionary_builder.py` 中，该部分代码将原始语料转化为一个文档列表，并统计每个词的词频。

值得注意的是我并没有使用停用词表，因为我认为后续的过滤每篇文档中 tf-idf 值较低的词可以近似取代这部分工作。

```
def build_dictionary(self):
    """
        build a dictionary contains all words except punctuations in datafile,
        and mark the word frequency of each word.
        build a doc_list contains all documents without date, POS tags
        and punctuations.
    """
    word_dict = dict()
    doc_list = []
    with open(self.datafile, "r", encoding="utf-8") as f:
        doc_id_list = []
        doc = []
        for line in f.readlines():
            line = line.strip().split("  ")[0:]
            if len(line) == 0:
                continue
            if line[0][:15] not in doc_id_list:
                doc_id_list.append(line[0][:15])
                if len(doc) > 0:
                    doc_list.append(doc)
                doc = []
            for word_pos in line[1:]:
                word = word_pos.split("/")[0]
                pos = word_pos.split("/")[1]
                if pos == "w":
                    continue
                doc.append(word)
                word_dict[word] = word_dict.get(word, 0) + 1
```

```
return word_dict, doc_list
```

4.2. 计算文档间相似度

此部分代码在 doc_similarity.py 中。简单起见，简要介绍一下几个重要部分。

- 对所有单个文档生成文档向量时，将文档中的词和 tf-idf 值用字典存储，方便后面计算向量内积时使用。然后舍弃掉 tf-idf 值较低的词。代码如下：

```
def build_doc_vector(self):
    """
    use dict structure to save the doc vector,
    and we only reserve words that take up 80% tf-idf weights
    """
    all_docs_vector = []
    for doc_idx, doc in enumerate(self._doc_list):
        doc_vector_dict = dict()
        for word in doc:
            word_tf_idf = self.cal_tf_idf(word, self.word_frequency_list[doc_idx])
            doc_vector_dict[word] = word_tf_idf
        sorted_doc_vector_dict = dict(sorted(doc_vector_dict.items(), key=lambda d: d[1],
                                             reverse=True))
        # reserve words which take up top 80% tf-idf weights
        sum_weights = 0
        total_weights = sum(sorted_doc_vector_dict.values()) * 0.8
        doc_vector_dict_reserved = dict()
        for key, value in sorted_doc_vector_dict.items():
            if sum_weights <= total_weights:
                doc_vector_dict_reserved.update({key: value})
                sum_weights += value
        all_docs_vector.append(doc_vector_dict_reserved)
    return all_docs_vector
```

- 计算向量间的余弦距离。以长度小的向量为基准，将对应词的 tf-idf 值相乘，然后对所有乘积求和。分式的分母部分使用前面已经保存的每个向量的 L2 范数，直接从字典中取出对应向量的 L2 范数做乘积即可。后面的异常处理部分是为了防止在数据集预处理时没有处理好，有些向量为空

导致结果出错。

```
def cosine_distance(self, doc1_id: int, doc2_id: int):
  """calculate cosine distance between two docs"""
  doc1_vec, doc2_vec = self.all_docs_vector[doc1_id], self.all_docs_vector[doc2_id]
  if len(doc1_vec) <= len(doc2_vec):
      base_vec = doc1_vec
      cmp_vec = doc2_vec
  else:
      base_vec = doc2_vec
      cmp_vec = doc1_vec
  inner_product = 0
  for word in base_vec:
      if word in cmp_vec:
          inner_product += base_vec[word] * cmp_vec[word]
  vec_norm_product = self.doc_vec_l2norm[doc1_id] * self.doc_vec_l2norm[doc2_id]
  distance = 0
  try:
      distance = inner_product / vec_norm_product
  except ZeroDivisionError as e:
      print("doc1_id = ", doc1_id, doc1_vec, self._doc_list[doc1_id])
      print("doc2_id = ", doc2_id, doc2_vec, self._doc_list[doc2_id])
      print(e)
  return distance
```

- 构建所有文档间相似度矩阵时，只计算下三角矩阵。在每处理完矩阵的 100 行后，输出程序的运行时间。

```
def cal_all_docs_similarity(self):
  time_start = time.time()
  doc_similarity_vec = np.zeros((self._num_doc, self._num_doc), dtype=np.float32)
  for i in range(self._num_doc):
      doc_similarity_vec[i, i] = 1
  for i in range(self._num_doc):
      if i % 100 == 0:
          print("processed %d, time %f" % (i, time.time() - time_start))
      for j in range(i):
          doc_similarity = self.cosine_distance(i, j)
```

```
            doc_similarity_vec[i][j] = doc_similarity_vec[j][i] = doc_similarity
    return doc_similarity_vec
```

5. Results and Analysis

5.1. 程序用时

程序每处理 100 行矩阵用时如下图所示。因为程序是计算下三角矩阵，所以越往下计算，每处理完 100 行矩阵所需时间越长。

```
processed 0, time 0.005011
processed 100, time 0.036979
processed 200, time 0.119049
processed 300, time 0.264957
processed 400, time 0.462399
processed 500, time 0.792211
processed 600, time 1.177529
processed 700, time 1.638250
processed 800, time 2.178959
processed 900, time 2.820206
processed 1000, time 3.456852
processed 1100, time 4.134087
processed 1200, time 4.969207
processed 1300, time 5.913406
processed 1400, time 6.891847
processed 1500, time 7.914900
processed 1600, time 9.037378
processed 1700, time 10.334132
processed 1800, time 11.421064
processed 1900, time 12.632036
processed 2000, time 14.035227
processed 2100, time 15.411969
processed 2200, time 16.964562
processed 2300, time 18.772117
processed 2400, time 20.456364
processed 2500, time 22.768183
processed 2600, time 24.745307
processed 2700, time 26.738226
processed 2800, time 28.479913
processed 2900, time 30.249134
processed 3000, time 32.389218
processed 3100, time 34.310197
[[ 1.          0.20328686  0.04269459 ...,  0.          0.          0.        ]
 [ 0.20328686  1.          0.02908987 ...,  0.          0.          0.        ]
 [ 0.04269459  0.02908987  1.         ...,  0.          0.          0.        ]
 ...,
 [ 0.          0.          0.         ...,  1.          0.          0.        ]
 [ 0.          0.          0.         ...,  0.          1.          0.        ]
 [ 0.          0.          0.         ...,  0.          0.          1.        ]]
```

5.2. 相似度结果分析

我们列举了对几个文档最相似的三篇文档的结果。

```
base doc:
['大连港', '年', '吞吐量', '超', '七千万', '吨', '本报', '大连', '十二月', '三十一日', '电', '记者', '张', '书政', '报道', '截至',
 '十二月', '二十八日', '十八时', '大连港', '年', '吞吐量', '突破', '七千万', '吨', '比', '上年', '同期', '增加', '六百万', '吨', '目前',
 '大连港', '已', '开通', '至', '威海', '烟台', '韩国', '仁川', '等', '六', '条', '国内外', '客货', '滚装', '航线', '开通', '至',
 '北美洲', '东南亚', '欧洲', '等', '八', '条', '集装箱', '班轮', '航线', '开通', '至', '哈尔滨', '延吉', '沈阳', '三', '条', '集装箱',
 '班列', '开通', '至', '广州', '杂货', '班轮', '航线']
the most similar doc index:  [  17 3000 2973]
they are:
['胜利', '海上', '油田', '产', '油', '创', '新高', '本报', '济南', '十二月', '三十一日', '电', '截至', '一九九七年', '十二月',
 '三十一日', '胜利', '海上', '油田', '全年', '共', '生产', '原油', '一百五十万零一百', '吨', '超', '计划', '一百', '吨', '与', '上年',
 '同期', '相比', '海上', '油田', '的', '年产', '能力', '增加', '了', '五十万', '吨', '八五', '以来', '胜利', '油田', '在', '渤海湾',
 '极', '浅', '海', '海域', '展开', '了', '大规模', '勘探', '开发', '会战', '一九九七年', '初', '胜利', '油田', '提出', '了', '奋战',
 '一', '年', '在', '海上', '建成', '一百五十万', '吨', '产能', '的', '奋斗', '目标', '全年', '完钻', '交井', '三十', '口', '投产',
 '新', '井', '三十六', '口', '杜', '中武', '孙', '传刚']
['节日', '港口', '生产', '忙', '宋', '学春', '张', '仁明', '除夕夜', '正', '当', '千家万户', '欢聚一堂', '之际', '青岛港', '三千',
 '多', '名', '干部', '职工', '仍然', '坚持', '节日', '生产', '用', '特殊', '方式', '迎接', '虎年', '的', '到来', '刚刚', '在', '牛年',
 '创出', '了', '年', '吞吐量', '六千九百一十六万', '吨', '集装箱', '吞吐量', '超', '百万', '箱', '货物', '通过', '能力', '由',
 '七千八百万', '吨', '增加', '到', '八千四百三十万', '吨', '的', '青岛港', '人', '正', '满怀信心', '地', '迎接', '虎年', '的', '到来',
 '向', '国际', '亿', '吨', '港', '迈进', '除夕', '十四时', '二十八分', '青岛港', '全体', '职工', '以', '自己', '独特', '的', '方式',
 '展开', '了', '港', '外', '过年', '港', '内', '大战', '的', '春节', '安全', '生产', '大会战', '坚持', '节日', '生产', '的', '集装箱',
 '公司', '职工', '王', '保俊', '自豪', '地', '说', '我', '很', '高兴', '能', '在', '除夕夜', '为', '港口', '生产', '出', '把', '力',
 '今晚', '我们', '的', '泊位', '无', '一', '闲置', '生产', '紧张', '而', '有序', '这样', '下去', '今年', '的', '集装箱', '目标', '定',
 '能', '实现', '在', '一', '号', '码头', '明达', '轮', '卸', '大豆', '的', '作业', '现场', '三', '台', '门机', '七', '台', '灌包机',
 '隆隆', '欢唱', '工人', '们', '一丝不苟', '地', '灌包', '缝包', '大港', '公司', '装卸', '四', '队', '队长', '逢', '新学', '说',
 '今晚', '我们', '队', '决心', '创出', '单舱', '作业', '五百四十', '吨', '的', '新', '纪录', '正', '在', '新', '港区', '慰问', '一',
 '线', '干部', '职工', '的', '全国', '优秀', '企业家', '青岛', '港务局', '局长', '常', '德传', '对', '记者', '说', '今年', '是',
 '我们', '把', '青岛港', '的', '改革', '开放', '和', '现代化', '建设', '全面', '推向', '二十一', '世纪', '的', '关键', '一', '年',
 '我们', '要', '高举', '邓小平理论', '伟大', '旗帜', '认真', '贯彻', '落实', '党', '的', '十五大', '精神', '中央', '经济', '工作',
 '会议', '精神', '使', '港口', '的', '两', '个', '文明', '建设', '得到', '协调', '发展', '共同', '进步']
['中远', '集装箱', '运输', '有限公司', '在', '沪', '成立', '吴', '邦国', '出席', '揭', '牌', '仪式', '新华社', '上海', '１月',
 '２７日', '电', '记者', '孙', '杰', '罗', '康雄', '中远', '集装箱', '运输', '有限公司', '今天', '上午', '在', '上海', '浦东', '新区',
 '正式', '成立', '国务院', '副', '总理', '吴', '邦国', '为', '公司', '成立', '揭', '牌', '中远', '集装箱', '运输', '有限公司', '是',
 '由', '中国', '远洋', '运输', '集团', '总公司', '简称', '中远', '集团', '的', '集装箱', '总部', '与', '上海', '远洋', '运输', '公司',
 '合并', '同时', '将', '中远', '集团', '所属', '的', '天津', '上海', '广州', '三', '家', '远洋', '公司', '的', '集装箱', '运输',
 '资源', '通过', '资产', '重组', '全部', '归并', '纳入', '而', '组建', '的', '目前', '中远', '集装箱', '运输', '有限公司', '现有',
 '总', '资产', '达', '２３０亿', '元', '拥有', '集装箱', '船舶', '１４０', '余', '艘', '标准箱', '位', '超过', '２１万', '箱', '整体',
 '实力', '跃居', '世界', '第五', '位', '这家', '公司', '共有', '班轮', '航线', '５０', '多', '条', '直接', '挂靠', '全球', '１００',
 '多', '个', '主要', '港口', '交通部', '副', '部长', '胡', '希捷', '在', '致词', '中', '说', '上海', '正在', '逐步', '发展', '成为',
```

上面的 4 篇文档都在说和港口和运输相关的事。

```
base doc:
['挂', '起', '红灯', '迎', '新年', '图片', '元旦', '来临', '安徽省', '合肥市', '长江路', '悬挂', '起', '３３００', '盏', '大',
 '红灯笼', '为', '节日', '营造', '出', '千', '盏', '灯笼', '凌空', '舞', '十', '里', '长街', '别样', '红', '的', '欢乐', '祥和',
 '气氛', '新华社', '记者', '戴', '浩', '摄', '传真', '照片']
the most similar doc index:  [1195 2599 2276]
they are:
['合肥', '大红', '灯笼', '当', '街', '舞', '本报', '记者', '刘', '杰', '元旦', '的', '前', '两', '天', '合肥', '市民', '一', '觉',
 '醒', '来', '发现', '长江', '中路', '两侧', '一下子', '挂', '满', '了', '喜迎', '新春', '的', '大红', '灯笼', '嗨', '神', '了', '一',
 '街', '两', '巷', '昨', '一夜间', '冒', '出', '了', '那么', '多', '的', '大红', '灯笼', '前', '不见', '头', '后', '不见', '尾', '的',
 '远远', '望', '去', '像', '两', '条', '红', '龙', '似的', '多', '喜人', '呀', '６４', '岁', '的', '陈', '玉英', '老太太', '在',
 '长江路', '纺织品', '商店', '门前', '看', '车', '３', '年', '了', '尽管', '看', '着', '合肥', '一', '年', '一个', '大', '变样', '但',
 '看到', '如此', '新景观', '仍然', '是', '惊奇', '得', '大呼小叫', '的', '为', '合肥', '人', '送', '来', '这', '份', '惊喜', '的',
 '是', '合肥', '晚报', '的', '全体', '同仁', '们', '总编辑', '仇', '旭东', '说', '合肥', '是', '全国', '闻名', '的', '文明', '省会',
 '城市', '元旦', '春节', '更', '要', '营造', '些', '热烈', '祥和', '的', '节日', '气氛', '所以', '我们', '出资', '１０万', '元', '让',
 '十', '里', '长江路', '挂', '上', '鲜艳夺目', '的', '大红', '灯笼', '号称', '安徽', '第一', '路', '的', '十', '里', '长江路', '早已',
 '是', '灯', '的', '海洋', '广告', '的', '世界', '如今', '再', '添', '上', '数千', '盏', '大红', '灯笼', '无疑', '是', '锦上添花',
 '但', '也', '有人', '对', '这么', '多', '的', '红灯笼', '挂', '在', '外边', '表示', '担心', '几', '天', '来', '更', '多', '的', '人',
 '在', '关注', '着', '合肥', '的', '大红', '灯笼', '到底', '能', '挂', '几', '天', '元月', '６日', '下午', '记者', '约请', '合肥',
 '晚报', '总编', '仇', '旭东', '一起', '坐', '车', '从', '长江路', '东头', '的', '九狮苑', '到', '大西门', '沿途', '徐行', '细细',
 '查看', '只见', '３３００', '盏', '大红', '灯笼', '无', '一', '损失', '一个个', '迎风', '舞动', '分外', '鲜艳', '与', '道路', '两侧',
 '各式各样', '的', '灯饰', '广告', '交相辉映', '令', '人', '留连忘返', '仇', '旭东', '又', '与', '记者', '漫步', '街头', '访问', '了',
 '许多', '人', '共同', '的', '感受', '是', '人', '的', '素质', '的', '提高', '是', '文明', '城市', '的', '根本', '大红', '灯笼', '作',
 '了', '最', '好', '的', '明证', '陈', '玉英', '老人', '说', '昨', '会', '丢', '呢', '人', '的', '觉悟', '都', '高', '了', '大伙',
 '同', '乐', '才', '是', '真', '乐', '呀', '市', '交警', '三', '大队', '二', '中队', '队长', '蔡', '涛', '说', '用不着', '管', '呢',
 '大红', '灯笼', '高高', '挂', '是', '合肥', '人', '的', '脸面', '共同', '的', '荣誉感', '凝聚', '了', '更', '多', '人', '的', '心',
 '千', '盏', '灯笼', '凌空', '舞', '十', '里', '长街', '分外', '红', '我们', '有', '信心', '让', '灯笼', '红', '到', '正月十五',
 '元宵节', '仇', '旭东', '如是说']
['装点', '京城', '万象新', '图片', '为', '迎', '新春', '佳节', '北京市', '在', '天安门', '广场', '国旗', '两侧', '安装', '了', '两',
 '盏', '总', '高', '１３', '米', '直径', '６', '米', '的', '大红', '灯笼', '这', '两', '盏', '目前', '国内', '最', '大', '的', '灯笼',
 '与', '广场', '东西', '两侧', '百', '米', '灯', '廊', '相映生辉', '本报', '记者', '张', '悦', '摄']
['万', '盏', '红灯', '迎', '新岁', '图片', '北京', '地坛', '春节', '文化', '庙会', '正', '加紧', '筹备', '由', '万', '盏', '红灯',
 '构成', '的', '万灯耀园', '将', '与', '百鸟朝凤', '祥虎送福', '火树金花', '组成', '迎新', '主题', '本报', '记者', '陶', '源明', '摄']
```

上面的 4 篇文档都在描述新春佳节的喜庆场面。

```
base doc:
['忠诚', '的', '共产主义', '战士', '久经考验', '的', '无产阶级', '革命家', '刘', '澜涛', '同志', '逝世', '附', '图片', '１', '张',
 '新华社', '北京', '１２月', '３１日', '电', '忠诚', '的', '共产主义', '战士', '久经考验', '的', '无产阶级', '革命家', '我党', '党务',
 '工作', '和', '统一战线', '工作', '的', '杰出', '领导人', '原', '中共中央', '顾问', '委员会', '常务', '委员会', '委员', '中国', '人民'
 '政治', '协商', '会议', '第四', '五', '六', '届', '全国', '委员会', '副', '主席', '刘', '澜涛', '同志', '因', '病', '医治', '无效',
 '于', '１９９７年', '１２月', '３１日', '１０时', '４４分', '在', '北京', '逝世', '终年', '８８', '岁', '根据', '刘', '澜涛', '同志',
 '生前', '遗愿', '和', '家属', '的', '意见', '刘', '澜涛', '同志', '的', '丧事', '从简', '不', '举行', '仪式', '不', '保留', '骨灰']
the most similar doc index:  [ 456  485 2187]
they are:
['忠诚', '的', '共产主义', '战士', '久经考验', '的', '无产阶级', '革命家', '刘', '澜涛', '同志', '遗体', '在', '京', '火化', '江',
 '泽民', '李', '鹏', '乔', '石', '朱', '镕基', '李', '瑞环', '刘', '华清', '胡', '锦涛', '尉', '健行', '李', '岚清', '荣', '毅仁',
 '等', '在', '刘', '澜涛', '同志', '生病', '住院', '期间', '和', '逝世', '后', '分别', '以', '不同', '方式', '对', '他', '表示',
 '亲切', '慰问', '和', '深切', '哀悼', '新华社', '北京', '１月', '６日', '电', '记者', '汪', '金福', '没有', '送别', '仪式', '没有',
 '保留', '骨灰', '刘', '澜涛', '同志', '的', '遗体', '今天', '在', '京', '火化', '他', '的', '亲属', '将', '骨灰', '撒', '在',
 '八宝山', '革命', '公墓', '的', '一', '棵', '常青树', '下', '以', '实现', '这', '位', '忠诚', '的', '共产主义', '战士', '久经考验',
 '的', '无产阶级', '革命家', '丧事', '从简', '的', '遗愿', '刘', '澜涛', '同志', '是', '我党', '党务', '工作', '和', '统一战线',
 '工作', '的', '杰出', '领导人', '原', '中共中央', '顾问', '委员会', '常务', '委员会', '委员', '中国', '人民', '政治', '协商', '会议',
 '第四', '五', '六', '届', '全国', '委员会', '副', '主席', '他', '因', '病', '于', '１９９７年', '１２月', '３１日', '在', '京',
 '逝世', '享年', '８８', '岁', '江', '泽民', '李', '鹏', '乔', '石', '朱', '镕基', '李', '瑞环', '刘', '华清', '胡', '锦涛', '尉',

['刘', '澜涛', '同志', '生平', '附', '图片', '５', '张', '忠诚', '的', '共产主义', '战士', '久经考验', '的', '无产阶级', '革命家',
 '我党', '党务', '工作', '和', '统一战线', '工作', '的', '杰出', '领导人', '原', '中共中央', '顾问', '委员会', '常务', '委员会', '委员',
 '中国', '人民', '政治', '协商', '会议', '第四', '五', '六', '届', '全国', '委员会', '副', '主席', '刘', '澜涛', '同志', '因', '病',
 '医治', '无效', '于', '１９９７年', '１２月', '３１日', '１０时', '４４分', '在', '北京', '逝世', '享年', '８８', '岁', '刘', '澜涛',
 '同志', '１９１０年', '１１月', '出生', '于', '陕西省', '米脂县', '一个', '贫苦', '家庭', '少年', '时期', '就', '接受', '进步', '思想',
 '追求', '革命', '真理', '１９２５年', '五卅', '运动', '爆发', '后', '他', '怀着', '救国救民', '的', '思想', '积极', '投身', '反帝',
 '爱国', '运动', '１９２６年', '加入', '中国', '共产主义', '青年团', '任', '米脂县', '团委', '委员', '宣传部长', '等', '职',
 '１９２８年', '５月', '参加', '组织', '领导', '了', '米脂', '学生', '运动', '抗议', '日本', '侵略者', '在', '济南', '制造', '的',
 '五三', '惨案', '并', '掀起', '反抗', '当地', '土豪劣绅', '的', '群众运动', '后', '任', '靖边县', '团委', '书记', '三', '边',

['张', '建良', '同志', '逝世', '新华社', '北京', '１月', '２０日', '电', '中国', '共产党', '优秀', '党员', '我党', '隐蔽', '战线',
 '上', '的', '杰出', '战士', '国家', '安全部', '离休', '干部', '张', '建良', '同志', '因', '病', '于', '１９９８年', '１月', '７日',
 '在', '北京', '逝世', '享年', '９６', '岁', '张', '建良', '原名', '华', '克之', '１９０２年', '１２月', '出生', '于', '江苏', '宝应',
 '早年', '就', '深受', '一些', '著名', '共产党人', '的', '影响', '从事', '进步', '活动', '他', '于', '１９３７年', '６月', '参加',
 '革命', '１９３９年', '加入', '中国', '共产党', '张', '建良', '同志', '受', '毛', '泽东', '主席', '朱', '德', '总司令', '亲自',
 '派遣', '在', '廖', '承志', '潘', '汉年', '同志', '领导', '下', '长期', '在', '上海', '香港', '从事', '我党', '秘密', '联络', '任务',
 '出生入死', '多次', '完成', '艰险', '使命', '做出', '了', '杰出', '贡献', '受到', '党中央', '的', '嘉奖', '和', '高度', '评价',
 '建国', '后', '张', '建良', '同志', '受', '潘', '汉年', '同志', '冤案', '株连', '长期', '蒙冤', '受屈', '党', '的', '十一', '届',
 '三中全会', '后', '党中央', '为', '他', '平反', '昭雪', '恢复', '名誉', '张', '建良', '同志', '坚信', '马列主义', '毛泽东思想', '和',
 '邓小平理论', '有', '坚定', '的', '共产主义', '信念', '他', '坚决', '拥护', '十一', '届', '三中全会', '以来', '党', '的', '路线',
 '方针', '和', '政策', '衷心', '拥护', '以', '江', '泽民', '同志', '为', '核心', '的', '第三', '代', '领导', '集体', '他', '顾全大局',
 '胸怀坦荡', '不', '计', '个人', '得失', '表现', '了', '一', '名', '优秀', '共产党员', '为', '党', '的', '事业', '埋头苦干', '对',
 '党', '忠诚', '的', '优秀', '品质', '是', '隐蔽', '战线', '干部', '的', '楷模']
```

上面的 4 篇文档都在描述某位优秀的共产党员逝世的消息。

从结果来看，计算出的最相似的 3 篇文档与源文档的主题基本一致，可以一定程度上说明我们的模型以及我们的程序是正确的。

6. Acknowledgements

感谢刘洋同学，牺牲休息时间在深夜与我一起讨论，给了我很多思路和启发。

感谢把作业分享出来的各位同学们，你们的作业也给了我一些启发。