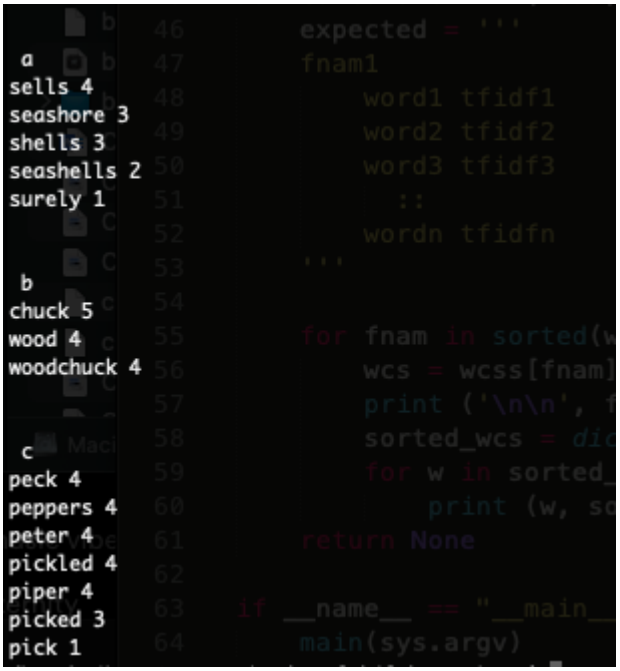
1. Cloning the repo to a local directory, run `python mapper.py < input.txt | sort | python reducer.py` get the following output



2. My modified code for reducer.py

TO CALCULATE TF:

```
def calculateTF(wordset,counter):
    termfreq_diz = dict.fromkeys(wordset,0)
    sum_doc = sum(counter.values())
    for w in counter:
        termfreq_diz[w]=counter[w]/float(sum_doc)
    return termfreq_diz
```

TO CALCULATE IDF:

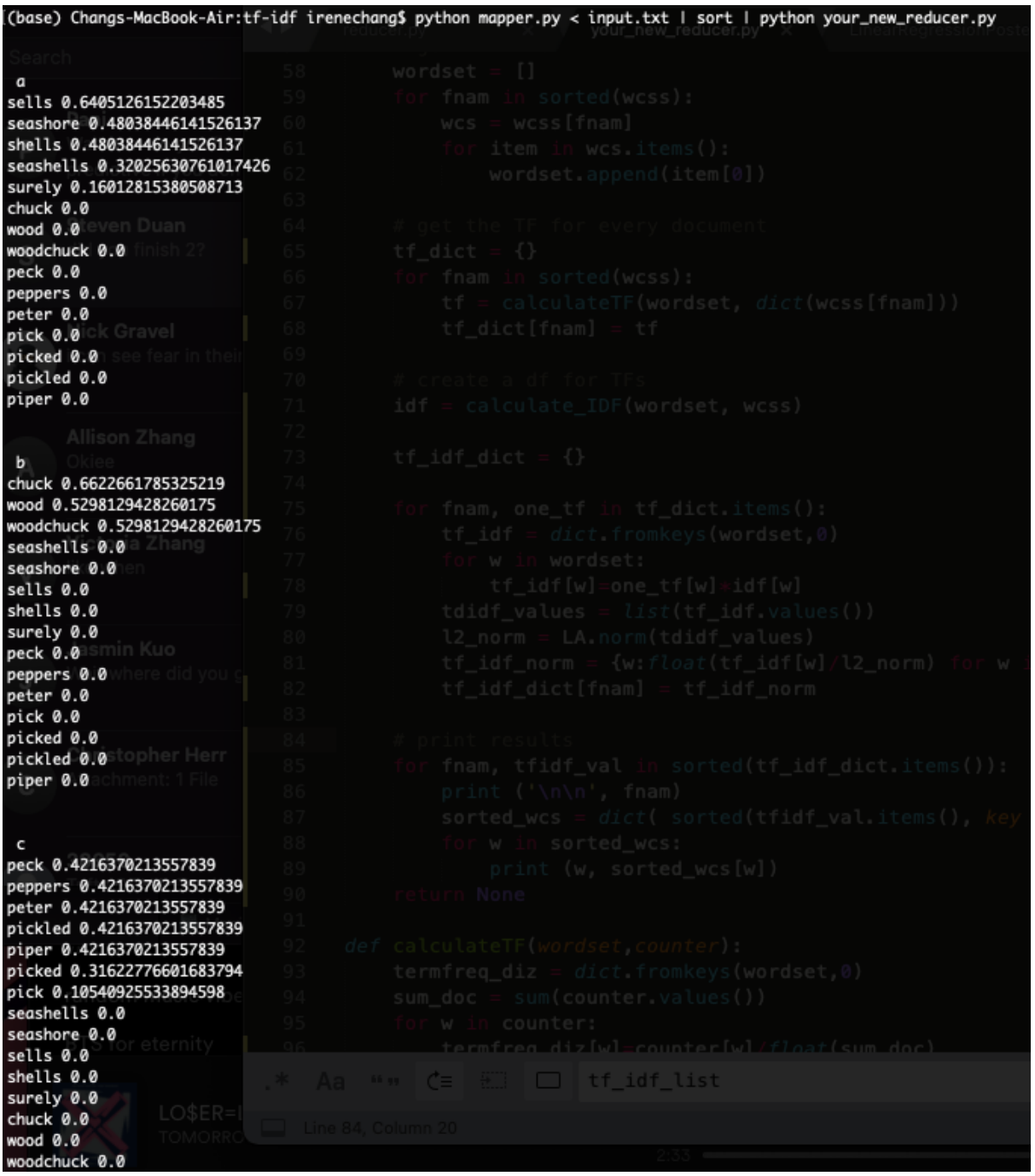
```
def calculate_IDF(wordset,wcss):
    N = len(sorted(wcss))
    counter = dict.fromkeys(wordset,0)
    for fnam in sorted(wcss):
        wcs = wcss[fnam]
        for w in wcs:
            counter[w] += 1
    idf_diz = {}
    for fnam in sorted(wcss):
        wcs = wcss[fnam]
        for w in wcs:
            idf_diz[w] = np.log((1+N)/(1+counter[w]))+1
    return idf_diz
```

TO CALCULATE IF-TDF:

```
def calculateTFIDF(wcss):
    # get all the words
    wordset = []
    for fnam in sorted(wcss):
```

```
        wcs = wcss[fnam]
        for item in wcs.items():
            wordset.append(item[0])

    # get the TF for every document
    tf_dict = {}
    for fnam in sorted(wcss):
        tf = calculateTF(wordset, dict(wcss[fnam]))
        tf_dict[fnam] = tf

    # calculate IDFs
    idf = calculate_IDF(wordset, wcss)

    # calculate TF-IDFs
    tf_idf_dict = {}
    for fnam, one_tf in tf_dict.items():
        tf_idf = dict.fromkeys(wordset,0)
        for w in wordset:
           tf_idf[w]=one_tf[w]*idf[w]
        tdidf_values = list(tf_idf.values())
        l2_norm = LA.norm(tdidf_values)
        tf_idf_norm = {w:float(tf_idf[w]/l2_norm) for w in wordset}
        tf_idf_dict[fnam] = tf_idf_norm

# print results
    for fnam, tfidf_val in sorted(tf_idf_dict.items()):
        print ('\n\n', fnam)
        sorted_wcs = dict(sorted(tfidf_val.items(), key = lambda
                         item: item[1], reverse = True))
        for w in sorted_wcs:
            print (w, sorted_wcs[w])
    return None
```

The final result matches with that on the notebook:



3. RUNNING THE CODE FROM STEP 1:

- Create a new cluster, go to the ssh terminal, use ‘Upload Files' to upload the scripts for mapper.py and reducer.py
- Following the instruction in the spec:
    - `hadoop fs -mkdir /user/inputs/`
    - `hadoop fs -mkdir /user/inputs/abc`
    - `gsutil cp gs://jsingh-bigdata-public/abc.zip .`
    - `unzip abc.zip -d abc`
    - `hadoop fs -put -f abc/* /user/inputs/abc`
    - `hadoop jar /usr/lib/hadoop/hadoop-streaming.jar -files mapper.py,reducer.py -mapper mapper.py -reducer reducer.py -numReduceTasks 1 -input /user/inputs/abc -output /user/j_singh/count_abc`

```
2022-03-11 20:58:40,207 INFO conf.Configuration: resource-types.xml not found
2022-03-11 20:58:40,207 INFO resource.ResourceUtils: Unable to find 'resource-types.xml'.
2022-03-11 20:58:40,642 INFO impl.YarnClientImpl: Submitted application application_1647031925587_0001
2022-03-11 20:58:40,712 INFO mapreduce.Job: The url to track the job: http://cluster-d630-m:8088/proxy/application_1647031925587_0001/
2022-03-11 20:58:40,715 INFO mapreduce.Job: Running job: job_1647031925587_0001
2022-03-11 20:58:53,876 INFO mapreduce.Job: Job job_1647031925587_0001 running in uber mode : false
2022-03-11 20:58:53,877 INFO mapreduce.Job:  map 0% reduce 0%
2022-03-11 20:59:02,962 INFO mapreduce.Job:  map 9% reduce 0%
2022-03-11 20:59:03,978 INFO mapreduce.Job:  map 14% reduce 0%
2022-03-11 20:59:09,021 INFO mapreduce.Job:  map 32% reduce 0%
2022-03-11 20:59:12,039 INFO mapreduce.Job:  map 45% reduce 0%
2022-03-11 20:59:19,082 INFO mapreduce.Job:  map 59% reduce 0%
2022-03-11 20:59:20,088 INFO mapreduce.Job:  map 77% reduce 0%
2022-03-11 20:59:26,121 INFO mapreduce.Job:  map 91% reduce 0%
2022-03-11 20:59:27,127 INFO mapreduce.Job:  map 100% reduce 0%
2022-03-11 20:59:33,157 INFO mapreduce.Job:  map 100% reduce 100%
2022-03-11 20:59:35,175 INFO mapreduce.Job: Job job_1647031925587_0001 completed successfully
2022-03-11 20:59:35,264 INFO mapreduce.Job: Counters: 54
        File System Counters
                FILE: Number of bytes read=951
                FILE: Number of bytes written=5710769
                FILE: Number of read operations=0
                FILE: Number of large read operations=0
                FILE: Number of write operations=0
                HDFS: Number of bytes read=4332
                HDFS: Number of bytes written=396
                HDFS: Number of read operations=71
                HDFS: Number of large read operations=0
                HDFS: Number of write operations=3
                HDFS: Number of bytes read erasure-coded=0
        Job Counters
                Launched map tasks=22
                Launched reduce tasks=1
                Data-local map tasks=22
                Total time spent by all maps in occupied slots (ms)=569736900
                Total time spent by all reduces in occupied slots (ms)=11244828
                Total time spent by all map tasks (ms)=180525
                Total time spent by all reduce tasks (ms)=3563
                Total vcore-milliseconds taken by all map tasks=180525
                Total vcore-milliseconds taken by all reduce tasks=3563
                Total megabyte-milliseconds taken by all map tasks=569736900
                Total megabyte-milliseconds taken by all reduce tasks=11244828
        Map-Reduce Framework
                Map input records=3
                Map output records=15
```

- Check the output by cat-ing the result file:
    - `hadoop fs -ls /user/j_singh/count_abc`
    - `hadoop fs -cat /user/j_singh/count_abc/part-00000`

```
binh_chang@cluster-d630-m:~$ hadoop fs -cat /user/j_singh/count_abc/part-00000
('\n\n', 'hdfs://cluster-d630-m/user/inputs/abc/a.txt')
('seashells', 2)
('sells', 4)
('surely', 1)
('seashore', 3)
('shells', 3)
('\n\n', 'hdfs://cluster-d630-m/user/inputs/abc/b.txt')
('wood', 4)
('woodchuck', 4)
('chuck', 5)
('\n\n', 'hdfs://cluster-d630-m/user/inputs/abc/c.txt')
('peter', 4)
('piper', 4)
('pickled', 4)
('picked', 3)
('pick', 1)
('peppers', 4)
('peck', 4)
```

Which matches with the output from step 1

RUNNING THE CODE FROM STEP 2:

- Upload the new reducer script to the terminal
- Run the command: `hadoop jar /usr/lib/hadoop/hadoop-streaming.jar -files mapper.py,your_new_reducer.py -mapper mapper.py -reducer your_new_reducer.py -numReduceTasks 1 -input /user/inputs/abc -output /user/j_singh/tfidf_abc`
- Check the output like above:

4. Repeating the same process and run the mapper and reducer on the presidential speech data, then do the following steps to get the result file to local directory: (output from running mapreduce:)

```
binh_chang@cluster-d630-m:~$ hadoop fs -head /user/j_singh/tfidf_prez_speech/part-00000
('\n\n', 'hdfs://cluster-d630-m/user/inputs/prez_speech/1981.txt')
('limited', 0.02912693640979673)
('todays', 0.022655881927760066)
('unhonored', 0.0)
('dissolution', 0.034263301312147)
('child', 0.0)
('dynamic', 0.0)
('sleep', 0.0)
('oldest', 0.0)
('saved', 0.0)
('belleau', 0.034263301312147)
('aggression', 0.0)
('tomorrows', 0.0)
('votes', 0.0)
('crises', 0.0)
('disability', 0.0)
('lord', 0.0)
('pride', 0.0)
('worth', 0.0)
('risk', 0.0)
('compassion', 0.025482627515077905)
('rise', 0.0)
('lurk', 0.0)
('misunderstanding', 0.068526602624294)
('softened', 0.0)
('govern', 0.02912693640979673)
('affect', 0.0)
('courageous', 0.0)
('encounter', 0.0)
('skills', 0.0)
('companies', 0.0)
('solution', 0.025482627515077905)
('convenience', 0.034263301312147)
('honor', 0.0)
('math', 0.0)
('reinvent', 0.0)
('heading', 0.034263301312147)
('triumph', 0.0)
('whirlwind', 0.0)
('enjoy', 0.0)
('charter', 0.0)
('civility', 0.0)
('force', 0.0)
('leaders', 0.0)
('rebuilding', 0.0)
```

- `hadoop fs -get /user/j_singh/tfidf_prez_speech/part-00000 .`
- `gsutil cp part-00000 gs://cs119-hw5`
- Download the file to my local directory
- Use a short python script to read in and display the table:

	1981.txt	1985.txt	1989.txt	1993.txt	1997.txt	2001.txt	2005.txt	2009.txt	2013.txt	2017.txt
0	('government', 0.21535)	('people', 0.17882)	('word', 0.17144)	('america', 0.24287)	('century', 0.36366)	('story', 0.28617)	('freedom', 0.34107)	('nation', 0.16078)	('complete', 0.16185)	('america', 0.304)
1	('heroes', 0.12741)	('human', 0.16155)	('breeze', 0.17144)	('change', 0.20804)	('nation', 0.1628)	('civility', 0.17101)	('liberty', 0.20464)	('common', 0.11672)	('requires', 0.16185)	('protected', 0.18394)
2	('special', 0.11651)	('freedom', 0.15914)	('dont', 0.15301)	('people', 0.1943)	('land', 0.15088)	('country', 0.15582)	('america', 0.14947)	('carried', 0.10872)	('people', 0.15486)	('american', 0.176)
3	('people', 0.11401)	('government', 0.15647)	('things', 0.14726)	('season', 0.17517)	('promise', 0.15036)	('citizens', 0.14226)	('americas', 0.13138)	('america', 0.10718)	('time', 0.14078)	('people', 0.16)
4	('americans', 0.11401)	('weapons', 0.13489)	('hand', 0.14726)	('today', 0.16191)	('time', 0.15027)	('nation', 0.12645)	('tyranny', 0.12528)	('generation', 0.09727)	('journey', 0.12265)	('country', 0.15772)
5	('freedom', 0.111)	('nuclear', 0.13489)	('friends', 0.14726)	('americans', 0.16191)	('people', 0.13775)	('america', 0.12645)	('human', 0.12003)	('crisis', 0.09585)	('creed', 0.11306)	('dreams', 0.14308)
6	('man', 0.11036)	('increase', 0.12091)	('fact', 0.14574)	('renewal', 0.14891)	('america', 0.13775)	('common', 0.11475)	('nation', 0.1121)	('people', 0.09379)	('equal', 0.11137)	('countries', 0.12983)
7	('maintaining', 0.10279)	('governments', 0.11241)	('door', 0.13716)	('idea', 0.13003)	('government', 0.12523)	('duty', 0.11308)	('country', 0.10914)	('today', 0.09379)	('happiness', 0.10071)	('obama', 0.12983)

8	('fall', 0.10279)	('federal', 0.11241)	('nation', 0.12677)	('time', 0.11334)	('american', 0.1127)	('affirm', 0.10903)	('excuse', 0.10107)	('spirit', 0.08832)	('knowing', 0.09711)	('wealth', 0.10279)
9	('productivity', 0.10279)	('reduce', 0.11241)	('great', 0.12677)	('raised', 0.11169)	('20th', 0.10161)	('character', 0.10155)	('americans', 0.09964)	('father', 0.08085)	('country', 0.09252)	('jobs', 0.10279)
10	('weapon', 0.10279)	('time', 0.11176)	('good', 0.12177)	('sake', 0.11169)	('human', 0.10057)	('purpose', 0.10155)	('history', 0.0955)	('day', 0.08043)	('freedom', 0.09252)	('borders', 0.09656)
11	('burden', 0.10279)	('history', 0.11017)	('free', 0.1141)	('spring', 0.11169)	('worlds', 0.10057)	('ideals', 0.10155)	('justice', 0.08973)	('work', 0.08039)	('enduring', 0.08496)	('foreign', 0.09656)
12	('intention', 0.10279)	('peace', 0.10735)	('hearts', 0.11336)	('posterity', 0.11169)	('work', 0.10018)	('compassion', 0.09539)	('day', 0.08973)	('time', 0.08039)	('citizens', 0.08447)	('capital', 0.09656)
13	('price', 0.10193)	('national', 0.09737)	('day', 0.10655)	('service', 0.10402)	('citizens', 0.10018)	('commitment', 0.09539)	('choice', 0.08911)	('greater', 0.07782)	('nation', 0.08447)	('nation', 0.096)
14	('federal', 0.10193)	('song', 0.09069)	('loyal', 0.10287)	('capital', 0.09771)	('children', 0.09021)	('promise', 0.09489)	('free', 0.08719)	('charter', 0.07248)	('america', 0.08447)	('great', 0.096)
15	('called', 0.10173)	('senator', 0.08993)	('expression', 0.10287)	('work', 0.09715)	('fellow', 0.08766)	('public', 0.0918)	('time', 0.08719)	('faced', 0.07248)	('american', 0.08447)	('nations', 0.08762)
16	('time', 0.10134)	('tax', 0.08993)	('blowing', 0.10287)	('millions', 0.09404)	('21st', 0.08638)	('nations', 0.08656)	('ideal', 0.08592)	('icy', 0.07248)	('generation', 0.08177)	('righteous', 0.08655)
17	('national', 0.09197)	('god', 0.08941)	('strong', 0.10029)	('generation', 0.09404)	('strong', 0.08255)	('freedom', 0.08656)	('goal', 0.08592)	('virtue', 0.07248)	('years', 0.0771)	('breath', 0.08655)
18	('dreams', 0.09062)	('america', 0.08941)	('work', 0.08874)	('serving', 0.08759)	('lives', 0.0823)	('whirlwind', 0.0855)	('institutions', 0.08592)	('calls', 0.07248)	('liberty', 0.0771)	('stops', 0.08655)
19	('economic', 0.09062)	('progress', 0.08841)	('people', 0.08874)	('compete', 0.08759)	('class', 0.07557)	('rides', 0.0855)	('defended', 0.08592)	('lower', 0.07248)	('lessons', 0.07616)	('industry', 0.08655)