Question.A:

First, I define the two vecs using different vectorizer:

Then, I define the method returning the 11 most similar result of df[‘title’][2](the most similar one is always the file itself)

I got:

8273 Anti-inflammatory medicine can have a benefici...

3483 Taking immunosuppressives, anti-cancer drugs m...

8773 Meatballs might wreck the anti-cancer perks of...

6551 Anti-inflammatory therapy cuts risk of lung ca...

3342 Researchers dispute warning that drugs for hig...

871 Non-steroidal anti-inflammatory drugs linked t...

5708 Colorectal cancer prevention: A proven benefit...

7366 OSA in older adults: Often present, seldom inv...

8313 Anti-inflammatory drugs ineffective for preven...

7923 Hazelnuts improve older adults' micronutrient ...

7631 The link between cognitive function and sexual...

Name: title, dtype: object

For the countvect

8273 Anti-inflammatory medicine can have a benefici...

3483 Taking immunosuppressives, anti-cancer drugs m...

8773 Meatballs might wreck the anti-cancer perks of...

6551 Anti-inflammatory therapy cuts risk of lung ca...

3342 Researchers dispute warning that drugs for hig...

871 Non-steroidal anti-inflammatory drugs linked t...

5708 Colorectal cancer prevention: A proven benefit...

7366 OSA in older adults: Often present, seldom inv...

8313 Anti-inflammatory drugs ineffective for preven...

7923 Hazelnuts improve older adults' micronutrient ...

7631 The link between cognitive function and sexual...

Name: title, dtype: object

For the tfidfvect

Realistically speaking, at least for the given result, I don’t see a critical difference between using countvectorizer and tfidfvectorizer for cos\_sim

Question.B:

I will use tfidfvects

Then, I use an elbow method to determine how many clusters is optimal for this research, it is 8 in my opinion(most efficient)

Chart, line chart

Description automatically generated

I use the method of turning tf\_idf into an array, and then, I use the optimized trained KM(8 clusters) to predict tf\_idf’s data. Then, using these predictions, I return the top 15 features for each clusters.

[ features score

0 study 0.023177

1 health 0.017455

2 children 0.014267

3 brain 0.010922

4 care 0.010117

5 women 0.009508

6 use 0.009504

7 finds 0.009211

8 people 0.008901

9 adults 0.008322

10 shows 0.007424

11 research 0.007246

12 researchers 0.007233

13 help 0.007014

14 weight 0.006933,

features score

0 risk 0.143678

1 heart 0.052920

2 disease 0.050444

3 linked 0.037786

4 higher 0.029926

5 increased 0.029092

6 increase 0.027197

7 death 0.025883

8 diabetes 0.025256

9 patients 0.022769

10 associated 0.020292

11 high 0.019434

12 stroke 0.019245

13 women 0.018905

14 cardiovascular 0.017728,

features score

0 cancer 0.165966

1 patients 0.030158

2 study 0.024393

3 prostate 0.023069

4 treatment 0.022837

5 screening 0.019782

6 breast 0.019758

7 new 0.019717

8 lung 0.019605

9 survivors 0.017823

10 ovarian 0.016213

11 survival 0.014110

12 therapy 0.012907

13 finds 0.011535

14 pancreatic 0.011466,

features score

0 disease 0.048516

1 linked 0.045094

2 heart 0.042520

3 risk 0.041951

4 patients 0.022519

5 study 0.016420

6 kidney 0.015303

7 blood 0.014543

8 higher 0.013620

9 diabetes 0.013584

10 high 0.013370

11 children 0.011741

12 associated 0.011550

13 failure 0.011057

14 health 0.010694,

features score

0 cancer 0.213849

1 breast 0.126707

2 patients 0.084303

3 treatment 0.055267

4 survival 0.052704

5 prostate 0.051007

6 new 0.025098

7 study 0.024880

8 therapy 0.023041

9 lung 0.022166

10 women 0.021679

11 advanced 0.020148

12 test 0.020055

13 early 0.019337

14 chemotherapy 0.018547,

features score

0 risk 0.202174

1 heart 0.147744

2 linked 0.113234

3 disease 0.112654

4 increased 0.090886

5 higher 0.059871

6 death 0.048119

7 attack 0.039790

8 lower 0.037212

9 stroke 0.036167

10 failure 0.032452

11 diabetes 0.032444

12 increase 0.026452

13 high 0.024635

14 associated 0.023677,

features score

0 patients 0.064527

1 new 0.056376

2 treatment 0.047993

3 study 0.033573

4 finds 0.018682

5 shows 0.016007

6 drug 0.014779

7 therapy 0.013945

8 effective 0.012305

9 researchers 0.011223

10 test 0.010838

11 surgery 0.010697

12 improves 0.010404

13 care 0.010339

14 help 0.010013,

features score

0 cancer 0.193660

1 risk 0.171594

2 breast 0.072868

3 linked 0.041897

4 prostate 0.041068

5 increased 0.039816

6 increase 0.025253

7 higher 0.025048

8 women 0.024554

9 lower 0.024089

10 death 0.023907

11 patients 0.022851

12 colorectal 0.021738

13 associated 0.018242

14 study 0.017117]

Cluster 1:peoples Cluster 2:heart attack Cluster 3: Cancer treatment Cluster 4: diabetes

Cluster 5:breast cancer Cluster 6: age related diseases Cluster 7: medical help Cluster 8: male/female cancer risk

Question D:

I import and define the lda, and then, I define a method that returns the 10 most top words in each feature

First, I just change the no\_topics to manually decide the best topics number. It is 7 that makes the most sense.

Topic 0: covid19

physicians medical money depressed aid coronavirus errors psychosis physician bypass

Topic 1: fda approval

trials accelerate needs clinical compares vessel degeneration enables revealed antidepressant

Topic 2: racial problem

racial contribute disparities making ethnic hemorrhage play driving unhealthy safer

Topic 3: China

day epilepsy ocd think determines chinese mood pancreas lose acids

Topic 4: cancer and brain damage treatment

care patients study health cancer finds damage treatment brain stem

Topic 5: factors of health problem

medications live american south migraine inequalities healthier aged night prognosis

Topic 6: male/female decease risk difference

risk study cancer patients new disease linked heart health women

Question D:

I manually looked into the csv file and find out the cut point for each years:

113 358 762 1290 1668 2156 3289 4544 5767 7018 7967 9081 and but didn’t work out..