Amirkabir University of Technology
(Tehran Polytechnic)

Department of
Computer Engineering

Natural Language Processing

Homework 2

Najmeh Mohammadbagheri

99131009

گزارش تمرین

بخش اول

پیش‌پردازش داده‌ها:

برای انجام پیش‌پردازش‌های لازم از کتابخانه‌ی پارسی‌ور استفاده شده‌است. در گام اول تمام اسناد(اخبار) **نرمالایز** می‌شود. هدف از نرمالایز کردن، یکسان کردن حروف یکسان با شکل‌های نوشتاری متفاوت است. مثلا "ی فارسی" و "ی عربی" یا شکل‌های مختلف "ک". در نرمالایزر همچنین تمامی علام اضافه‌ی موجود در متن خذف می‌شوند و اعداد فارسی موجود در متن به شکل استاندارد اعداد انگلیسی تبدیل می‌شوند. علاوه‌بر استفاده از نرمالایزر لازم بود کلمات دو قسمتی همچون "بین‌المللی" که نیم‌فاصله دارند به فرمت یکسانی باشند که در گام بعد این کلمات خراب نشوند. به همین منظور پس از نرمالایز کردن، تمامی **نیم‌فاصله‌ها(\u200c) حذف** و دو قسمت کلمه بهم چسبانده می‌شوند. در گام بعد از **توکنایزر** استفاده می‌شود. در این مرحله تمامی توکن‌ها (کلمات) از همدیگر جدا می‌شوند و به صورت یک آرایه از کلمات در می‌آیند. گام بعد **حذف استاپ‌وردها** است. برای انجام این قسمت یک لیست نسبتا جامع از استاپ‌وردهای فارسی تهیه شده‌است و هر توکن در صورتی که عضوی از این لیست باشد حذف می‌شود. در گام بعد **ریشه‌یابی** توکن‌های باقی‌مانده انجام می‌شود. یعنی دو کلمه‌ی گفتند و بگو به یک کلمه‌ی گفت نگاشت می‌شود و تنها همین کلمه در ادامه استفاده می‌شود. پس از آنکه هر توکن به ریشه‌اش تغییر یافت لیست کلمات حاصل به عنوان سند پیش‌پردازش‌شده ذخیره می‌شود.

توجه: به دلیل زمان‌بر بودن فرایند پیش‌پردازش، یکبار این عملیات انجام و خروجی ذخیره می‌شود و در هر بار اجرای برنامه فقط از اسناد پیش‌پردازش‌شده‌ی ذخیره‌شده استفاده می‌شود.

بخش دوم

در ابتدا با استفاده از تمامی توکن‌های بخش قبل مدل skip-gram با پارامترهای زیر آموزش داده می‌شود و در ادامه از بردار کلمات این مدل استفاده می‌شود.

```
min_count=1,size=300,workers=4, window=3,sg = 1
```

الف) به منظور بازنمایی سند به صورت میانگین کلمات، بردار تمامی کلماتی که در متن وجود دارد باهم جمع می‌شوند و تمام مولفه‌های بردار سند تقسیم بر تعداد کل کلمات آن سند می‌شود.

ب) در این قسمت برای میانگین‌گیری وزن‌دار از tf-idf استفاده می‌شود. برای محسابه‌ی tf-idf نیز از کتابخانه‌ی genism کمک گرفته می‌شود. در این کتابخانه یک مدل tfidfmodel وجود دارد که دیکشنری پیکیره(ساخت دیکشنری پیکره را نیز می‌توان با یک تابع از همین ابزار انجام داد.) را دریافت می‌کند و به ازای هر کلمه در هر سند مقدار tf-idf آنرا خروجی می‌دهد. پس از آنکه وزن هر کلمه را گرفتیم در بردار مربوط به آن کلمه ضرب می‌کنیم و در نهایت تقسیم بر جمع تمام وزن‌ها می‌کنیم.

ج) از بردارهای از قبل آموزش داده‌شده برای این قسمت استفاده می‌شود و روند کار مانند حالت الف است.

د) مانند قسمت قبل از بردارهای آماده استفاده می‌شود و روند کار مانند حالت ب است.

ه) در این قسمت ابتدا به ازای تمام کلمات هر سند، مقدار tf آن محاسبه می‌شود و یک ماتریس کلمه-سند ساخته می‌شود. سپس این ماتریس به تابع svd داده می‌شود و ۳۰۰ مولفه‌ی اول آن به عنوان بردار هر سند درنظر گرفته می‌شود. (ساخت svd با کتابخانه‌ی sklearn و ساخت ماتریس کلمه-سند با کتابخانه‌ی genism انجام می‌شود.)

بخش سوم

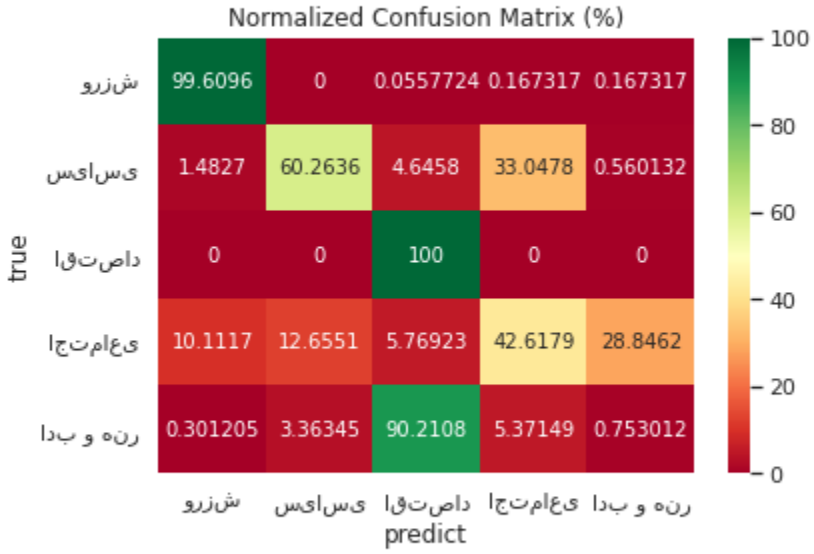
در این بخش در نتایج خوشه‌بندی مشاهده شد که دسته‌ی ادب و هنر هیچگاه برچسب نمی‌گیرد. به همین دلیل در یک حالت سعی شده‌است که دسته‌ی ادب و هنر به صورت اجباری تولید شود و در حالت دوم بر اساس بیشترین تکرار برچسب، دسته‌ها مشخص شده‌اند. برای حالت اول ماتریس درهم ریختگی رسم شده است و میتوان نتایج را مقایسه کرد. در ماتریس اول تعداد هر برچسب در هر خوشه مشخص شده‌است، اما در ماتریس دوم(شکل) بر اساس تعداد اسناد هر خوشه این میزان نرمال شده است.

در آزمایشات ذکر شده f1-score در حالت macro محاسبه شده‌است.

بازنمایی الف)

حالت اول:

```
[[1.786e+03 0.000e+00 1.000e+00 3.000e+00 3.000e+00]
 [4.500e+01 1.829e+03 1.410e+02 1.003e+03 1.700e+01]
 [0.000e+00 0.000e+00 1.670e+02 0.000e+00 0.000e+00]
 [1.630e+02 2.040e+02 9.300e+01 6.870e+02 4.650e+02]
 [6.000e+00 6.700e+01 1.797e+03 1.070e+02 1.500e+01]]
```




```
{0: 'ادب و هنر', 1: 'سیاسی', 2: 'اقتصاد', 3: 'اجتماعی', 4: 'ورزش'}
accuracy: 0.521456
f1_score: 0.442000
NMI: 0.572292
```

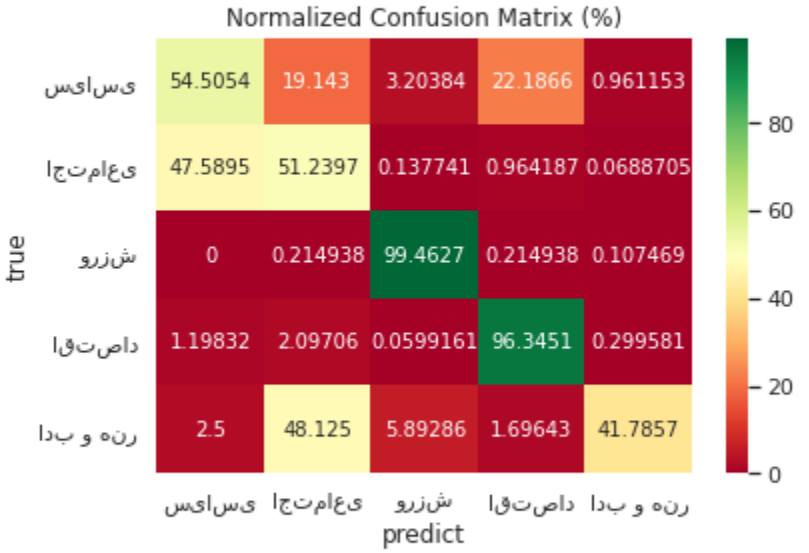
حالت دوم:

```
{0: 'اقتصاد' ,4: 'اجتماعی' ,3: 'اقتصاد' ,2: 'سیاسی' ,1: 'ورزش'}
accuracy: 0.728689
f1_score: 0.591626
NMI: 0.584542
```

بازنمایی ب)

حالت اول:

```
[[1.361e+03 4.780e+02 8.000e+01 5.540e+02 2.400e+01]
 [6.910e+02 7.440e+02 2.000e+00 1.400e+01 1.000e+00]
 [0.000e+00 4.000e+00 1.851e+03 4.000e+00 2.000e+00]
 [2.000e+01 3.500e+01 1.000e+00 1.608e+03 5.000e+00]
 [2.800e+01 5.390e+02 6.600e+01 1.900e+01 4.680e+02]]
```




```
{0: 'ادب و هنر' ,4: 'اقتصاد' ,3: 'ورزش' ,2: 'اجتماعی' ,1: 'سیاسی'}
accuracy: 0.701477
f1_score: 0.683545
NMI: 0.571621
```

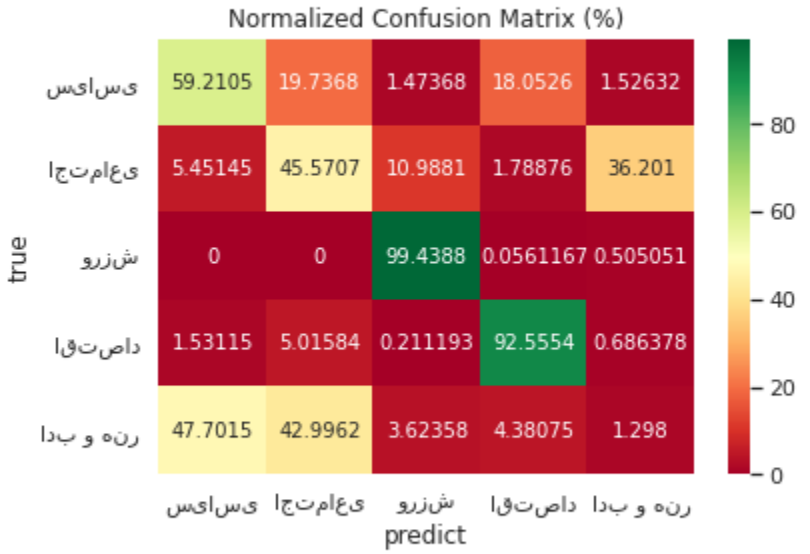
حالت دوم:

```
{0: 'اجتماعی' ,4: 'اقتصاد' ,3: 'ورزش' ,2: 'اجتماعی' ,1: 'سیاسی'}
accuracy: 0.709734
f1_score: 0.593860
NMI: 0.553297
```

بازنمایی ج)

حالت اول:

```
[[1.125e+03 3.750e+02 2.800e+01 3.430e+02 2.900e+01]
 [6.400e+01 5.350e+02 1.290e+02 2.100e+01 4.250e+02]
 [0.000e+00 0.000e+00 1.772e+03 1.000e+00 9.000e+00]
 [2.900e+01 9.500e+01 4.000e+00 1.753e+03 1.300e+01]
 [8.820e+02 7.950e+02 6.700e+01 8.100e+01 2.400e+01]]
```




```
{0: 'ادب و هنر' ,4: 'اقتصاد' ,3: 'ورزش' ,2: 'اجتماعی' ,1: 'سیاسی'}
accuracy: 0.605768
f1_score: 0.547275
NMI: 0.527844
```

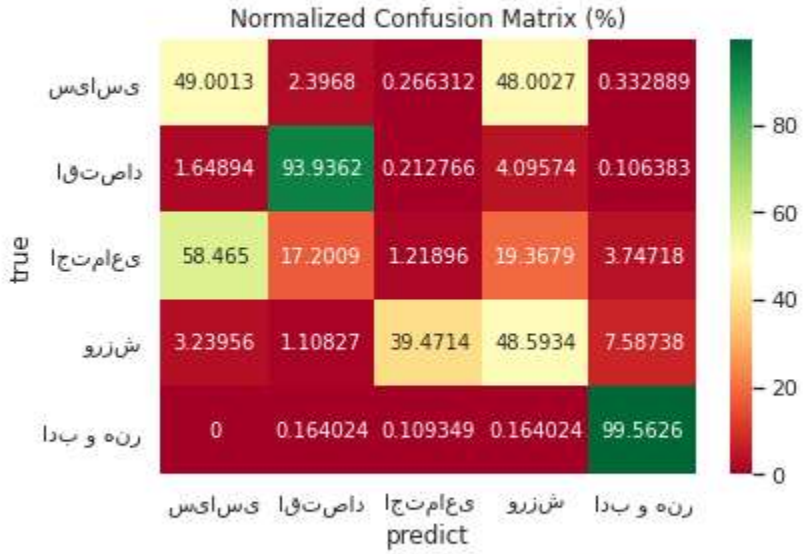
حالت دوم:

```
{0: 'سیاسی' ,4: 'اقتصاد' ,3: 'ورزش' ,2: 'اجتماعی' ,1: 'سیاسی'}
accuracy: 0.705547
f1_score: 0.567942
NMI: 0.568999
```

بازنمایی د)

حالت اول:

```
[[ 736.   36.    4.  721.    5.]
 [  31. 1766.    4.   77.    2.]
 [1295.  381.   27.  429.   83.]
 [  38.   13.  463.  570.   89.]
 [   0.    3.    2.    3. 1821.]]
```




```
{0: 'ادب و هنر', 1: 'اقتصاد', 2: 'ورزش', 3: 'اجتماعی', 4: 'سیاسی'}
accuracy: 0.572160
f1_score: 0.525813
NMI: 0.565587
```

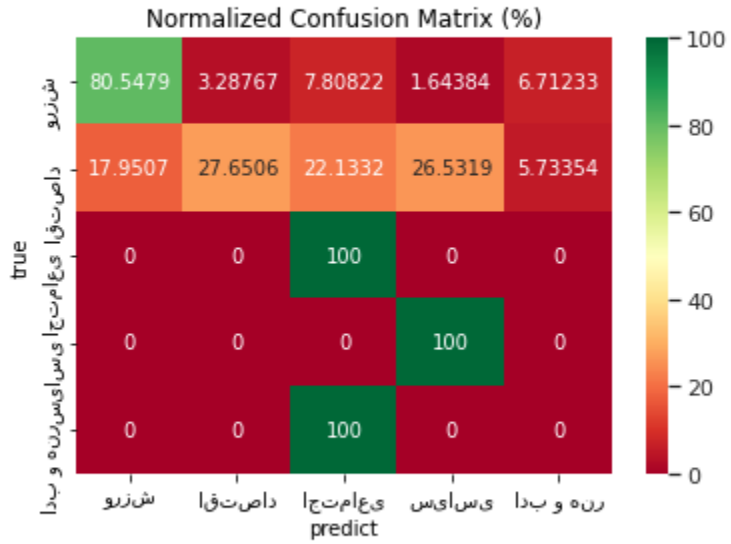
حالت دوم:

```
{0: 'ورزش', 1: 'اقتصاد', 2: 'سیاسی', 3: 'اجتماعی', 4: 'سیاسی'}
accuracy: 0.719619
f1_score: 0.579762
NMI: 0.602253
```

بازنمایی ه)

حالت اول:

```
[[5.880e+02 2.400e+01 5.700e+01 1.200e+01 4.900e+01]
 [1.412e+03 2.175e+03 1.741e+03 2.087e+03 4.510e+02]
 [0.000e+00 0.000e+00 1.000e+00 0.000e+00 0.000e+00]
 [0.000e+00 0.000e+00 0.000e+00 1.000e+00 0.000e+00]
 [0.000e+00 0.000e+00 1.000e+00 0.000e+00 0.000e+00]]
```




```
{0: 'ادب و هنر' ,4: 'سیاسی' ,3: 'اجتماعی' ,2: 'اقتصاد' ,1: 'ورزش'}
accuracy: 0.321549
f1_score: 0.173004
NMI: 0.085929
```

حالت دوم:

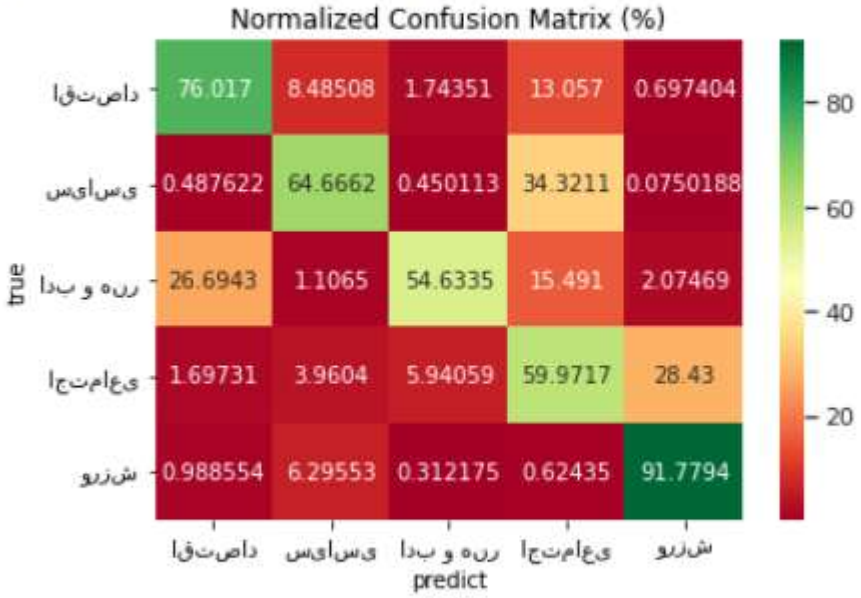
```
{0: 'اجتماعی' ,4: 'سیاسی' ,3: 'اجتماعی' ,2: 'اقتصاد' ,1: 'ورزش'}
accuracy: 0.321665
f1_score: 0.173226
NMI: 0.085936
```

بخش چهارم

در ابتدا مدل LDA را آموزش می‌دهیم و سپس توزیع topic هارا برروی هر سند بدست از مدل میگیریم و هر موضوعی که بیشترین احتمال را داشت به آن سند می‌دهیم و خوشه‌بندی به اینصورت انجام می‌شود. در ادامه نیز مانند حالت‌های قسمت قبل معیارهای ارزیابی محاسبه می‌شود.

حالت اول:

```
[[1962.  219.   45.  337.   18.]
 [  13. 1724.   12.  915.    2.]
 [ 193.    8.  395.  112.   15.]
 [  12.   28.   42.  424.  201.]
 [  19.  121.    6.   12. 1764.]]
```




```
{0: 'ورزش', 1: 'سیاسی', 2: 'ادب و هنر', 3: 'اجتماعی', 4: 'اقتصاد'}
accuracy: 0.729038
f1_score: 0.685625
NMI: 0.543536
```

حالت دوم:

```
{0: 'ورزش', 1: 'سیاسی', 2: 'ادب و هنر', 3: 'اجتماعی', 4: 'اقتصاد'}
accuracy: 0.729038
f1_score: 0.685625
NMI: 0.543536
```

بخش پنجم

تحلیل:

بهترین مدل برای خوشه‌بندی اسناد مدل LDA است زیرا این مدل توزیع کلمات و توزیع اسناد را بصورت موضوعی یاد می‌گیرد. در دیگر مدل‌ها برای بازنمایی اسناد تعداد زیادی از مولفه‌ها را حذف کردیم و اینکار باعث از بین رفتن اطلاعات می‌شد.

باتوجه به تصاویر ماتریس‌های درهم ریختگی در تمام حالت‌ها می‌توان مشاهده کرد در حالت LDA اعداد روی قطر اصلی همگی بیشتر از دیگر مولفه‌های ماتریس هستنند و این یعنی خوشه‌ها به خوبی جدا شده‌اند. همچنین در این مدل دسته‌ی ادب و هنر تشخیص داده شده‌است درحالی که در ۵ حالت قسمت دوم سوال این دسته جزو خوشه‌ها قرار نمی‌گرفت.

نکته‌ی دیگر که مشاهده می‌شود تفکیک پذیری بسیار خوب دسته‌های ورزش و اقتصاد است. و دلیل این امر تفاوت بسیار زیاد موجود در کلمات بکار رفته در اخبار ورزشی و اقتصادی است.

مقایسه‌ی حالت اول و دوم بازنمایی:

با استفاده از tf-idf ارزش کلماتی که تکرار زیادی دارند مانند stopwordها بسیار کم می‌شود و در میانگین‌گیری برای بازنمایی سند وزن کلمات باارزش‌تر بیشتر می‌شود. اما همانطور که در ابتدای گزارش ذکر شد ما در ابتدا کلمات پرتکرار را حذف می‌کنیم. درنتیجه دقت میانگین‌گیری معمولی در قسمت اول سوال خوب می‌شود.(۰.۷۲) در قسمت دوم که میانگین‌گیری وزن‌دار است کلمات اسناد اجتماعی بیشتر وزن گرفته‌اند و خوشه‌بندی شامل دو خوشه‌ی اجتماعی شده‌است. در حالی که در قسمت اول که میانگین‌وزن‌دار نبوده دو خوشه‌ی اقتصاد داشتیم.

مقایسه‌ی حالت اول و سوم از بازنمایی:

در حالت سوم، بردار کلمات بر روی پیکره‌ی سراسری و بزرگی بدست آمده، یعنی اسناد زیادی که در این پیکره‌ی ما وجود ندارند در وزن‌های بردارهای آماده تاثیرگذار بوده‌اند و این امر باعث می‌شود که دقت خوشه‌بندی کاهش بیابد. زیرا وزن‌های موجود متناظر با این اسناد نیستند. اما در حالت اول که بردارها دقیقا از روی داده‌های خودمان بدست آمده بود دقیق‌تر بودند.

مقایسه‌ی حالت سوم و چهارم از بازنمایی:

در حالت چهارم که میانگین وزن‌دار محاسبه می‌شود دقت اندکی بهتر از حالت سوم می‌باشد. زیرا ارزش هر کلمه در بازنمایی سند تاثیر می‌گذارد و این باعث می‌شود که اسناد در خوشه‌ها متناسب با کلمات مهم‌ترشان از هم تفکیک‌پذیری بهتری داشته باشند.

مقایسه‌ی حالت پنجم با دیگر حالات بازنمایی:

در این حالت بدلیل آنکه تعداد زیادی از مولفه‌های ماتریس درنظر گرفته نمی‌شوند و تنها ۳۰۰ مولفه نگه داشته‌می‌شود در نتیجه میزان زیادی از اطلاعات سند از بین می‌رود و خوشه‌بندی با دقت پایینی صورت می‌گیرد.

برای شهود بیشتر از بهتر بودن روش lda کافی است به قطر اصلی ماتریس‌ها توجه کنیم. همانطور که مشاهده می‌کنید در این حالت رنگ قطر اصلی شامل رنگ قرمز نیست و هر ۵ خوشه به خوبی پیدا شده‌اند.

توضیح بیشتر:

در ۵ روش اول خوشه‌ی ادب و هنر وجود ندارد ولی در lda این خوشه نیز وجود دارد. نکته‌ی دیگر: در ماتریس‌های رنگی خوشه‌ی تکراری به عنوان ادب و هنر درنظر گرفته شده و ماتریس رسم شده و اگر به خانه‌ی رنگ سبز(یا کرمی) مربوط به سطر ادب و هنر دقت کنیم متوجه می‌شویم خوشه‌ی واقعی که بیشترین تعداد تکرار برچسب را داشته چه بوده است. (همچنین با مقایسه‌ی دو دیکشنری چاپ شده‌ی مربوط به نام و شماره‌ی خوشه‌ها در دو حالت صحت عملکرد مشخص می‌شود.