محمد امین رامی

تمرین اول درس تحلیل داده های حجیم

پاییز ۱۴۰۱

سوال اول

الف)

برای حل مسئله داده شده به صورت زیر عمل میکنیم:

گام اول:

ابتدا جدول R و S رو جوین میکنیم و حاصل را RS مینامیم.

گام دوم:

جدول T و U را ادغام میکنیم و آن را UT مینامیم.

گام سوم:

جدول RS و UT را جوین میکنیم.

عملا داریم مانند رو به رو عمل میکنیم: $(R \bowtie S) \bowtie (T \bowtie U)$

حال الگوریتم Map Reduce برای جوین($\bowtie$) را معرفی میکنیم. فرض کنید میخواهیم جدول T1(B, C) و T2(A, B) را جوین کنیم. به صورت زیر عمل میکنیم.

Map Operation:

به ازای هر سطر T1 یک جفت (key, value) به صورت زیر تولید میکنیم:

For each row in T1:

emit <key = b, value=(a, b, 'T1')>

به ازای هر سطر T2 یک جفت (key, value) به صورت زیر تولید میکنیم:

For each row in T2:

emit <key = b, value=(c, b, 'T2')>

در Reducer نیز به صورت عمل میکنیم:

Reducer Operation:

تمام (key, value) هایی که key یکسان دارند را به یک Reducer مشخص ارسال میکنیم. در Reducer نیز تمام اعضایی که b آنها یکسان هست را در نظر میگیریم و آن دسته ای که مربوط به T1 هستند را در آن دسته ای مربوط به T2 هستند ضرب دکارتی میکنیم:

For each <key=b, value=([a1, ‘T1’], [a2, ‘T1’], …, [an, ‘T1’], [c1, ‘T2’], [c2, ‘T2’], …, [cm, ‘T2’])>

Calculate cartesian product of (a1, a2, …, an) × (c1, c2, …, cm)

حال هزینه الگوریتم را محاسبه میکنیم. این هزینه برابر با هزینه Communication است. هزینه یک جوین برای دو جدول با سایز t1 و t2 با الگوریتم ذکر شده برابر است با: $2t_1 + 2t_2$

پس هزینه کل در مسئله اصلی برابر است با:

$$Cost = (2r + 2t) + (2t + 2u) + (2rtp + 2utp)$$

پرانتز اول هزینه (R⋈S)، پرانتز دوم هزینه (T⋈U) و پرانتز سوم هزینه (RS⋈TU) است.

محاسبه پارامترها:

Replication rate = r = 1

Reducer size of first step = r + s

Reducer size of second step = t + u

Reducer size of third step = rsp + utp

ب)

در این بخش، جوین را به صورت single step انجام میدهیم. برای اینکار از hash function مناسب استفاده میکنیم. Hash function مورد استفاده ما، باقی مانده تقسیم است. فرض کنید تعداد reducer های ما k باشد. فرض کنید key های ما به صورت (i, j, k) باشند.

حال در مرحله Map به صورت زیر عمل میکنیم:

فرض کنید x و y و z سه عدد هستند به قسمی که : $xyz = M$ داریم:

For each row in R:

    For j in range(y):

        For k in range(z):

            emit <key=(mod(b, x), j, k), value = (a, b, ‘R’) >

For each row in S:

For k in range(z):

emit <key=(mod(b, x), mod(c, y), k), value=(b, c, 'S')>

For each row in T:

For i range(x):

emit <key=(i, mod(c, y), mod(d, z), value=(c, d, 'T'))>

For each row in U:

For i in range(x):

For j in range(y):

emit <key = (i, j, mod(d, z), value(d, e, 'U'))>

به این صورت عملیات Map را روی دیتا انجام میدهیم.

حال به سراغ عملیات Reduce میرویم.

Reduce Operation:

در این مرحله، همه عناصری که key یکسان دارند به یک reducer هدایت میشوند. حال در هر reducer به value عناصر نگاه میکنیم. سپس به ترتیب از عناصر جدول R شروع کرده و b آنهارا با b عناصر جدول S مچ میکنیم. و به همین ترتیب برای c و d جداول T و U انجام میدهیم. در آخر تمام a و e هایی که مچ شده اند را ضرب دکارتی میکنیم:

Calculate Cartesian product of: (a1, a2,...,an)× (e1, e2, ...,em)

حال هزینه الگوریتم را محاسبه میکنیم:

Cost = r + s + t + u + ryz + sz + tx + uxy

Such that: xyz = M M: number of reducers

حال باید x و y و z را به گونه ای انتخاب کنیم که هزینه مینیمم شود. داریم:

Min r + s + t + u + ryz + sz + tx + uxy

Subject to xyz = M

این مسئله معادل است با:

Min ryz + sz + tx + uxy

Subject to xyz = M

حال برای حل مسئله از ضرایب لاگرانژ استفاده میکنیم.

داریم:

$$\mathcal{L}(x,y,z,\lambda) = ryz + sz + tx + uxy + \lambda xyz - \lambda M$$

$$\frac{\partial \mathcal{L}}{\partial x} = t + uy + \lambda yz = 0$$
$$\frac{\partial \mathcal{L}}{\partial y} = rz + ux + \lambda xz = 0$$
$$\frac{\partial \mathcal{L}}{\partial z} = ry + s + \lambda xy = 0$$
$$\frac{\partial \mathcal{L}}{\partial \lambda} = xyz - M = 0$$

حال برای سادگی فرض میکنیم: $r = s = t = u$

معادلات به صورت زیر میشوند:

$$\frac{\partial \mathcal{L}}{\partial x} = r + ry + \lambda yz = 0$$
$$\frac{\partial \mathcal{L}}{\partial y} = rz + rx + \lambda xz = 0$$
$$\frac{\partial \mathcal{L}}{\partial z} = ry + r + \lambda xy = 0$$
$$\frac{\partial \mathcal{L}}{\partial \lambda} = xyz - M = 0$$

از حل معادلات بالا مقادیر زیر حاصل میشوند:

$$x = \sqrt{M}, \quad z = \sqrt{M}, \quad y = 1$$

هزینه نهایی با استفاده از پارامترهای به دست آمده برابر است با:

$$Single\ Step\ Cost = 4r + 4r\sqrt{M}$$

هزینه الگوریتم بخش الف برابر است با:

$$Multiple\ Step\ Cost = 8r + 4r^2p$$

واضح است که که اگر r بزرگ باشد (که در داده های حجیم بسیار بزرگ است) هزینه روش Single Step بسیار کمتر از هزینه روش Multiple Step است.

سوال دوم

الف) این بخش عملی است و در کد تحویلی انجام شده است.

ب) درصد مقادیر NaN در هر ستون:

```
Percentage of NaN values in each column:

color                      0.38
director_name              2.06
num_critic_for_reviews     0.99
duration                   0.30
director_facebook_likes    2.06
actor_3_facebook_likes     0.46
actor_2_name               0.26
actor_1_facebook_likes     0.14
gross                     17.53
genres                     0.00
actor_1_name               0.14
movie_title                0.00
num_voted_users            0.00
cast_total_facebook_likes  0.00
actor_3_name               0.46
facenumber_in_poster       0.26
plot_keywords              3.03
movie_imdb_link            0.00
num_user_for_reviews       0.42
language                   0.24
country                    0.10
content_rating             6.01
budget                     9.76
title_year                 2.14
actor_2_facebook_likes     0.26
imdb_score                 0.00
aspect_ratio               6.52
movie_facebook_likes       0.00
dtype: float64
```

پ) علت وجود مقادیر NaN این است که ممکن است برخی اطلاعات مربوط به یک فیلم در سایت IMDB موجود نباشد و یا در سایت ثبت نشده باشند.

درصد مقادیر NaN در هر ستون:

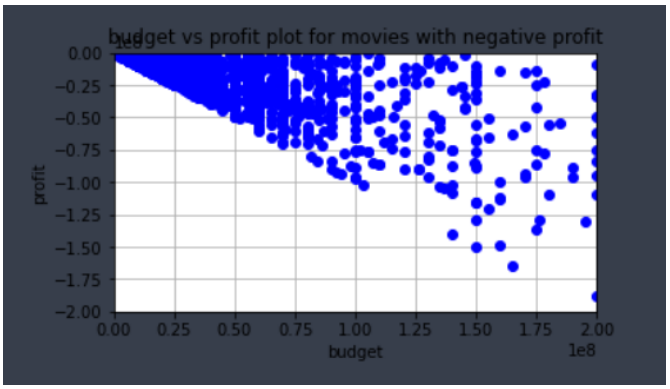
```
Percentage of NaN values in each column:

director_name              2.06
num_critic_for_reviews     0.99
gross                     17.53
genres                     0.00
actor_1_name               0.14
movie_title                0.00
num_voted_users            0.00
num_user_for_reviews       0.42
language                   0.24
budget                     9.76
title_year                 2.14
imdb_score                 0.00
movie_facebook_likes       0.00
dtype: float64
```

ت) برای تحلیل بهتر داده ها، نمودار فیلم هایی که سود آنها مثبت است و فیلم هایی که سود آنها منفی است، جداگانه رسم میکنیم.

نمودار سود بر حسب بودجه برای فیلم ها با سود مثبت:

نمودار سود بر حسب بودجه برای فیلم ها با سود منفی:



تحلیل نمودارها:

فیلم هایی که سود آنها مثبت است، موفق بوده اند. همچنین به طور کلی، هرچه بودجه بیشتری خرج شده باشد، فیلم بهتر ساخته شده و موفق تر بوده است فلذا سودآوری بیشتری داشته است. پس همانطور که دیده میشود،برای فیلم ها با سود مثبت، با افزایش بودجه، سود نیز افزایش داشته است.

برای فیلم های با سود منفی عکس این قضیه برقرار است. چون فیلم موفق نبوده، سود ناخالص آن عددی کوچک است. پس هرچه بودجه بیشتری خرج شده باشد، سود خالص آن کمتر است. پس همانطور که در نمودار دیده میشود، با افزایش بودجه سود کاهش پیدا میکند.

لیست پرسود ترین فیلم ها:

	movie_title	profit
0	Avatar	523505847.0
1	Jurassic World	502177271.0
2	Titanic	458672302.0
3	Star Wars: Episode IV - A New Hope	449935665.0
4	E.T. the Extra-Terrestrial	424449459.0
5	The Avengers	403279547.0
6	The Lion King	377783777.0
7	Star Wars: Episode I - The Phantom Menace	359544677.0
8	The Dark Knight	348316061.0
9	The Hunger Games	329999255.0

ث)

اطلاعات خواسته شده به صورت زیر است:

genres	movie count	mean profit
Action	6	-3.264576e+07
Action|Adventure	9	6.507458e+07
Action|Adventure|Animation|Comedy|Crime|Family|Fantasy	1	-5.640309e+07
Action|Adventure|Animation|Comedy|Drama|Family|Sci-Fi	2	5.323508e+06
Action|Adventure|Animation|Comedy|Family	5	4.714108e+07
...	...	...
Romance|Sci-Fi|Thriller	1	1.225332e+07
Sci-Fi	1	-2.318050e+05
Sci-Fi|Thriller	7	2.065219e+06
Thriller	3	-5.618203e+05
Western	3	1.320459e+07

ج)

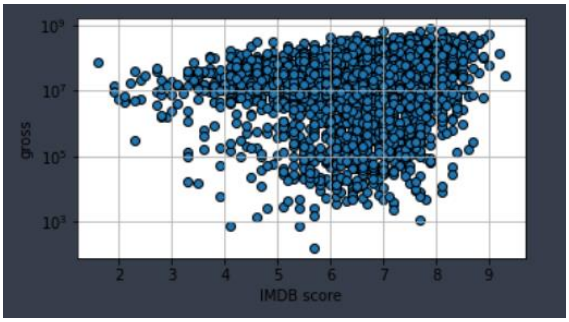
اطلاعات خواسته شده به صورت زیر است:

```
0                        Action|Adventure|Fantasy|Sci-Fi
3                                      Action|Thriller
7       Adventure|Animation|Comedy|Family|Fantasy|Musi...
8                         Action|Adventure|Sci-Fi
9                       Adventure|Family|Fantasy|Mystery
                                ...
4748                                 Documentary|Sport
4890                                      Documentary
4904                       Documentary|History|Music
4922                                  Drama|Romance
4931                                Drama|Music|Romance
```

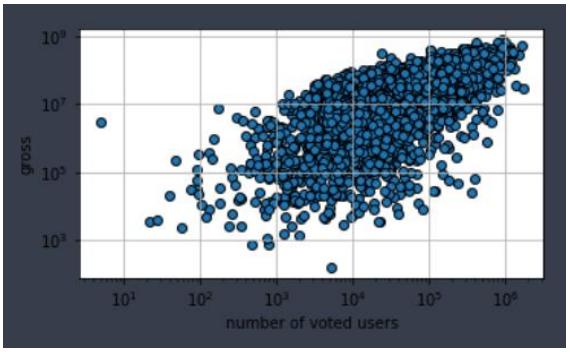
چ)

برای اینکه به تاثیر نظر مردم بر سود فیلم ها پی ببریم، نمودار سود برحسب IMDB score و number of voted users رسم میکنیم.

نمودار سود بر حسب IMDB score :



نمودار سود بر حسب number of voted users :



تحلیل نمودارها:

همانطور که مشاهده میشود، به ازای یک مقدار ثابت از number of voted users یا IMDB score، هرچه این مقدار ها بزرگ تر باشد، سود فیلم بیشتر است. یعنی به صورت تقریبی میتوان گفت که اثر نظر مردم با سود فیلم correlation مثبت دارد. یعنی هرچه نظر مردم بهتر باشد، به طور میانگین سود فیلم نیز بیشتر است. همچنین به صورت چشمی نیز یک ارتباط نسبتا خطی بین نظر مردم و سود فیلم وجود دارد.

سوال سوم

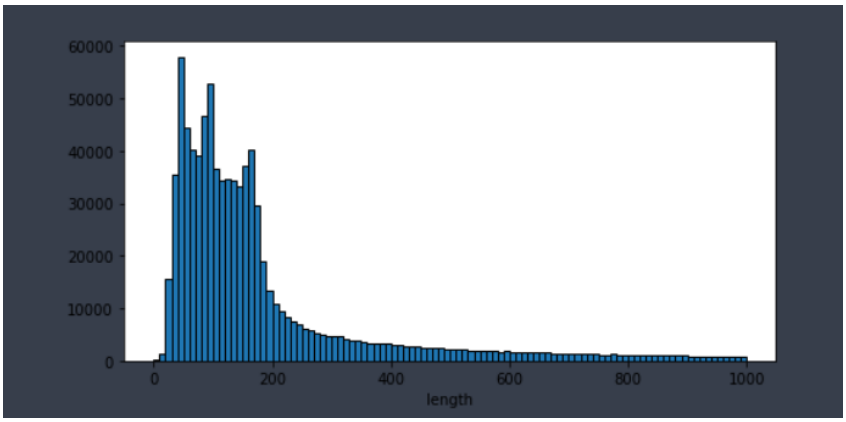
برای پیدا کردن stop word ها، کلمات با بیشترین دفعات تکرار را لیست میکنیم. اما مشاهده خواهیم کرد که بعضی کلماتی که stop word تشخیص داده شده اند، stop word نیستند. برای مثال کلمه "ایران" جزو این کلمات است. بنابراین باید stop word ها را به صورت دستی از کلمه های به دست آمده جدا میکنیم.

لیست کلمه های پرتکرار سه حرفی انگلیسی به صورت زیر است:

most common three letter English words

```
[('The', 10114),
 ('the', 9211),
 ('amp', 8271),
 ('and', 7048),
 ('org', 6816),
 ('php', 3758),
 ('mol', 3656),
 ('www', 3646),
 ('com', 3476),
 ('for', 2906),
 ('DNA', 2186),
 ('web', 1751),
 ('ref', 1637),
 ('int', 1615),
 ('New', 1537),
 ('end', 1533),
 ('bdi', 1222),
 ('III', 1158),
 ('San', 1152),
 ('new', 865)]
```

همچنین توزیع طول مقالات نیز به صورت زیر است:



همچنین طولانی ترین مقالات نیز مقالات زیر هستند:

```
title: نادرشاه
url: https://fa.wikipedia.org/wiki?curid=10787
length: 101772
============================
title: ایران
url: https://fa.wikipedia.org/wiki?curid=163930
length: 89892
============================
title: تبرد اسمولنسک
url: https://fa.wikipedia.org/wiki?curid=2322946
length: 88049
============================
title: جمهوری وایمار
url: https://fa.wikipedia.org/wiki?curid=26519
length: 84469
============================
title: تبرد مسکو
url: https://fa.wikipedia.org/wiki?curid=1501875
length: 84277
============================
```

درصد و تعداد مقالاتی که حاوی کلمات ذکر شده هستند:

```
2.68      25071      :%سیاست
10.58      99011      :%تاریخ
2.13      19969      :%قانون
1.81      16900      :%اقتصاد
0.83      7737      :%مهندسی
1.19      11118      :%پزشکی
```

حال به شرح قسمت آخر سوال میپردازیم. در این قسمت از ما خواسته شده است که مقالات مرتبط با کلمات خواسته شده را بدست بیاوریم. الگوریتمی که من استفاده کردم به این صورت است که برای هرکدام از کلمات خواسته شده، یک لیست که شامل خود کلمه و همچنین کلمات مرتبط به آن را به صوت دستی تهیه کرده ام.

سپس برای اینکه معیاری برای ارتباط یک مقاله با یکی از کلمات داده شده داشته باشیم، به هر مقاله یک امتیاز نسبت میدهیم. این امتیاز به این صورت است که جمع tf_idf لیست کلمه های مرتبط به آن کلمه و خود کلمه است. سپس مقالات را بر اساس امتیازشان sort میکنیم و در نهایت لیستی از مقالات که از مرتبط ترین تا کمترین مرتبط مرتب شده اند، خواهیم داشت. سپس ۵ تای اول لیست را به عنوان ۵ مرتبط ترین مقالات معرفی میکنیم.

شایان به ذکر است که من تابع tf_idf را به دلایل تجربی کمی تغییر دادم و به صورت زیر تعریف کردم:

$$tf - idf = \frac{tf_{word} \times \ln\left(1 + \frac{number\ of\ articles}{df_{word}}\right)}{1 + 0.003 \times d}$$

با این روش مقالات مرتبط به هر کلمه را پیدا میکنیم:

اقتصاد:

سیاست:

قانون:

تاریخ:

مهندسی:

پزشکی:

همچنین میخواهیم مشخص کنیم که آیا یک مقاله با یک موضوع مرتبط است یا خیر. برای اینکار ابتدا میانگین امتیازات مقالات برای آن موضوع مشخص را برای تمام مقالات حساب میکنیم. سپس انحراف معیار را حساب میکنیم. سپس مقالاتی که امتیاز آنها از میانگین به اندازه سه برابر انحراف معیار بیشتر است، به عنوان مقاله مرتبط به کلمه درنظر میگیریم.

به این حساب، درصد مقالات مرتبط به هر کلمه به صورت زیر مشخص میشوند:

```
articles related to:
====================
اقتصاد:
2.33%
====================
سیاست:
2.00%
====================
قانون:
1.96%
====================
تاریخ:
3.73%
====================
مهندسی:
2.01%
====================
پزشکی:
1.32%
====================
```

نتایج به دست آمده تقریبا با نتایج بخش سه همخوانی دارد به جز برای کلمه **"تاریخ"**. زیرا در بخش سه تعداد مقالات مرتبط با این کلمه ۱۰ درصد گزارش شده بود درحالی که اینجا این عدد به ۳.۷۳ کاهش یافته است. دلیل آن این است که ممکن است در یک جای یک مقاله یک بار از کلمه تاریخ استفاده شده باشد اما این به این معنا نیست که مقاله لزوما با تاریخ ارتباط دارد. به همین دلیل است که این تفاوت وجود دارد.