



به نام خدا
دانشگاه تهران



نام و نام خانوادگی	پرهام بیچرانلو – آناهیتا هاشم زاده
شماره دانشجویی	۸۱۰۱۰۰۵۰۲ – ۸۱۰۱۰۰۳۰۳
تاریخ ارسال گزارش	۱۴۰۱،۰۸،۱۹

دانشکده مهندسی برق و کامپیوتر

درس شبکه‌های عصبی و یادگیری عمیق
تمرین اول

پاسخ ۳ – Restricted Boltzmann Machine ۱

۳-۱. سیستم توصیه گر ۱

شکل‌ها

شکل ۱. نمودار خطای شبکه..... ۶

جدول‌ها

- جدول ۱. دیتاست فیلم **Error! Bookmark not defined.**
- جدول ۲. دیتاست امتیازات ۲
- جدول ۳. دیتافریم ادغام شده ۳
- جدول ۴. فیلم‌های توصیه شده به کاربر ۷۵ ۶

پاسخ ۳ – Restricted Boltzmann Machine

۳-۱. سیستم توصیه‌گر

A. از دستور read_csv برای خواندن دیتاست‌ها استفاده می‌کنیم.

دیتاست فیلم‌ها:

movielid	title	genres	List Index
0	1	Toy Story (1995)	Adventure Animation Children Comedy Fantasy 0
1	2	Jumanji (1995)	Adventure Children Fantasy 1
2	3	Grumpier Old Men (1995)	Comedy Romance 2
3	4	Waiting to Exhale (1995)	Comedy Drama Romance 3
4	5	Father of the Bride Part II (1995)	Comedy 4
...
9737	193581	Black Butler: Book of the Atlantic (2017)	Action Animation Comedy Fantasy 9737
9738	193583	No Game No Life: Zero (2017)	Animation Comedy Fantasy 9738
9739	193585	Flint (2017)	Drama 9739
9740	193587	Bungo Stray Dogs: Dead Apple (2018)	Action Animation 9740
9741	193609	Andrew Dice Clay: Dice Rules (1991)	Comedy 9741

جدول ۱- دیتاست فیلم

دیتاست امتیازات:

movieId	title	genres	List Index	userId	rating	timestamp
0	1	Toy Story (1995)	Adventure Animation Children Comedy Fantasy	0	1	4.0 964982703
1	1	Toy Story (1995)	Adventure Animation Children Comedy Fantasy	0	5	4.0 847434962
2	1	Toy Story (1995)	Adventure Animation Children Comedy Fantasy	0	7	4.5 1106635946
3	1	Toy Story (1995)	Adventure Animation Children Comedy Fantasy	0	15	2.5 1510577970
4	1	Toy Story (1995)	Adventure Animation Children Comedy Fantasy	0	17	4.5 1305696483
...
100831	193581	Black Butler: Book of the Atlantic (2017)	Action Animation Comedy Fantasy	9737	184	4.0 1537109082
100832	193583	No Game No Life: Zero (2017)	Animation Comedy Fantasy	9738	184	3.5 1537109545
100833	193585	Flint (2017)	Drama	9739	184	3.5 1537109805
100834	193587	Bungo Stray Dogs: Dead Apple (2018)	Action Animation	9740	184	3.5 1537110021
100835	193609	Andrew Dice Clay: Dice Rules (1991)	Comedy	9741	331	4.0 1537157606

جدول ۲-دیتاست امتیازات

Movies dataset shape: (9742, 3)

Ratings dataset shape: (100836,4)

برای اضافه کردن ستون 'List Index' از دستور زیر استفاده کردیم:

```
df_movie['List Index'] = df_movie.index
```

B. از دستور زیر استفاده می‌کنیم:

```
df = pd.merge(df_movie,df_rating,on = 'movieId')
```

جدول ادغام شده:

movie Id	title	genr es	List Index	userl d	ratin g	timesta mp	
0	1	Toy Story (1995)	Adventure Animation Children Comedy Fantasy	0	1	4.0	964982703
1	1	Toy Story (1995)	Adventure Animation Children Comedy Fantasy	0	5	4.0	847434962
2	1	Toy Story (1995)	Adventure Animation Children Comedy Fantasy	0	7	4.5	1106635946
3	1	Toy Story (1995)	Adventure Animation Children Comedy Fantasy	0	15	2.5	1510577970
4	1	Toy Story (1995)	Adventure Animation Children Comedy Fantasy	0	17	4.5	1305696483
...
100831	193581	Black Butler: Book of the Atlantic (2017)	Action Animation Comedy Fantasy	9737	184	4.0	1537109082
100832	193583	No Game No Life: -	Animation Comedy Fantasy	9738	184	3.5	1537109545

(2017)									
10083	1935	Flint		Drama	9739	184	3.5	15371098	
3	85	(2017)						05	
10083	1935	Bungo		Action Animation	9740	184	3.5	15371100	
4	87	Stray Dogs: Dead Apple						21	
		(2018)							
10083	1936	Andre		Comedy	9741	331	4.0	15371576	
5	09	w Dice Clay: Dice Rules						06	
		(1991)							

جدول ۳ - دیتا فریم ادغام شده

C. ستون title چون از روی movieId می‌توان فیلم‌ها را از هم متمایز کرد و دیگر نیاز به عنوان فیلم نیست.

ستون ژانر چون الگوریتم ما بر اساس مشابهت امتیاز دهی کاربران فیلم‌ها را پیشنهاد می‌دهد و ژانر فیلم تاثیری روی این پیشنهاد ندارد. ستون timestamp هم در الگوریتم استفاده نمی‌شود.

D. از دستور زیر برای این کار استفاده می‌کنیم:

```
'Group_df = df.groupby('userId
```

E. از روش نرمال سازی MinMax استفاده می‌کنیم و امتیاز هر کاربر برای هر فیلم را در یک لیست قرار می‌دهیم پس یک لیست دو بعدی داریم. که هر سطر مربوط به یک کاربر است و هر ستون مربوط به یک فیلم. طبیعی است که خیلی از خانه‌ها صفر باشند چون کاربر به خیلی از فیلم‌ها رای نداده است.

F. در کلاس RBM با TensorFlow1 پیاده سازی شده است. (توضیح صفحه بعد)

G. در کلاس RBM با TensorFlow1 پیاده سازی شده است. (توضیح صفحه بعد)

H. در کلاس RBM تابع recommender پیاده سازی شده است.

پیاده سازی: الگوریتم در ۲۰ اپاک آموزش دیده است. از روش min-batch با سایز ۱۲۸ استفاده شده است. همچنین از نسخه TensorFlow 1 برای توسعه استفاده کردیم.

توضیح الگوریتم: RBM از دو قسمت تشکیل شده است. لایه قابل مشاهده و لایه پنهان.

که وزن‌ها مشترک ولی بایاس هر لایه متفاوت است. هر بار طی دو مرحله forward feed و backward الگوریتم آموزش داده می‌شود و ارور از تفاوت مقدار واقعی با مقدار بدست آمده از شبکه مقایسه شده و با توجه به تفاوت آن‌ها (ارور) وزن‌ها و بایاس‌ها آپدیت می‌شوند و این روند به تعداد ایپاک‌ها ادامه خواهد داشت.

استفاده برای سیستم توصیه‌گر: هر کاربر به چندین فیلم امتیاز داده است. هر بار به شبکه تمام فیلم‌ها را با امتیاز داده شده توسط کاربر به عنوان ورودی می‌دهیم و اگر کاربر به آن نمره نداده باشد برای آن فیلم صفر در نظر می‌گیریم. پس لایه قابل مشاهده شامل امتیازهایی که کاربر به فیلم داده است می‌شود. لایه هیدن هم در واقع عامل‌هایی هستند که وابستگی‌ها را در خود دارند اما برای مشخص نیست که دقیقاً چیست مثلاً می‌تواند چیزی شبیه ژانر را نمایندگی کنند یا اینکه موضوع فیلم چیست. هر چه هست فرقی نمی‌کند مهم این است که می‌تواند کمک کند از وابستگی‌های کاربران استفاده کنیم.

که مقادیر بایاس لایه هیدن و لایه قابل مشاهده و وزن‌ها با ترین کردن شبکه بدست می‌آیند.

سپس برای بدست آوردن یا پیش بینی نمره فیلم‌ها برای کاربر جدید فقط یکبار دیگر مرحله forward و backward را انجام می‌دهیم تا نمرات را برای فیلم‌ها بسازیم. دقت کنید که با داشتن وزن و بایاس با توجه به ساختار شبکه می‌توان ورودی را تولید کرد پس نمره فیلم‌هایی که نداده است را هم شبکه به ما می‌دهد.

در نهایت برای اینکه بهترین فیلم‌ها را به کاربر پیشنهاد دهیم خروجی را نسبت به امتیاز پیش بینی شده به ترتیب نزولی سورت می‌کنیم و پانزده‌تای بهتر را نمایش می‌دهیم.

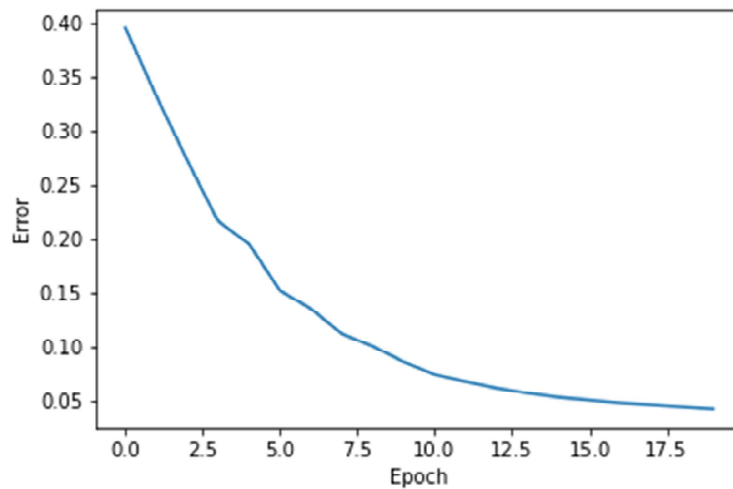
چند نکته:

- اگر شبکه را برای هر کاربر فقط با توجه به نمراتی که داده ترین می‌کردیم احتمالاً بهتر بود. چون وزن صفر دادن به فیلم‌هایی که کاربر ندیده است اطلاعات غلط به شبکه می‌دهد. به علت کمبود وقت نشد که با این روش پیاده سازی کنم.
- شبکه RBM در اینجا اساساً در حال یادگیری توزیع احتمال در مجموعه داده است. یعنی اگر احتمال امتیاز یک کاربر به یک فیلم زیاد باشد وزن هم متناسب با آن افزایش پیدا می‌کند.

خروجی‌های الگوریتم:

ارور نهایی: 0.041977

نمودار ارور:



شکل ۱- نمودار خطا شبکه

۱۵ تا بهترین فیلم پیشنهادی برای کاربر ۷۵:

movieid	title	genres	List Index	Recommendation Score
314	356	Forrest Gump (1994)	Comedy Drama Romance War	314 1.000000
277	318	Shawshank Redemption, The (1994)	Crime Drama	277 1.000000
257	296	Pulp Fiction (1994)	Comedy Crime Drama Thriller	257 1.000000
510	593	Silence of the Lambs, The (1991)	Crime Horror Thriller	510 0.999945
1939	2571	Matrix, The (1999)	Action Sci-Fi Thriller	1939 0.999707

224	260	Star Wars: Episode IV - A New Hope (1977)	Action Adventure Sci-Fi	224	0.78726 6
461	527	Schindler's List (1993)	Drama War	461	0.53777 5
97	110	Braveheart (1995)	Action Drama War	97	0.53261 8
2226	295 9	Fight Club (1999)	Action Crime Drama Thriller	2226	0.48145 7
507	589	Terminator 2: Judgment Day (1991)	Action Sci-Fi	507	0.45680 4
0	1	Toy Story (1995)	Adventure Animation Children Comedy Fantasy	0	0.44485 8
123	150	Apollo 13 (1995)	Adventure Drama IMAX	123	0.44400 7
659	858	Godfather, The (1972)	Crime Drama	659	0.44254 4
2145	285 8	American Beauty (1999)	Drama Romance	2145	0.44160 2
4137	595 2	Lord of the Rings: The Two Towers, The (2002)	Adventure Fantasy	4137	0.42905 9
جدول ۴ - فیلم‌های توصیه شده به کاربر ۷۵					