סחר אלקטרוני - דו"ח פרויקט מערכות המלצה

מגישים:
איתי בוגנים, 305278384
סהר ויה, 205583453

מבוא

בפרויקט זה נפתח מודלים שונים לצורך בניית מערכות המלצה. המודלים שניצור יחזו את ה rating הצפוי עבור כל סרט שנמצא ב dataset הנתון. ה dataset שנשתמש בו במהלך פרויקט זה הוא MovieLens 100K. עבור כל מודל שניצור ננתח את התוצאות וננסה לשפר כל מודל בכדי לחזות תוצאות טובות יותר, בסוף התהליך ניצור השוואה של המודלים שלנו בכדי לקבוע איזה מודל הכי מומלץ למימוש ואיזה מן המודלים הכי פחות מומלץ(בהתאם לתוצאות שנקבל). בפרוייקט זה השתמשנו בספריות שונות של Keras,Turi Create, DeepCTR – python. בכל שלב של הפרוייקט נציג את התוצרים בצורה מסודרת באמצעות דיאגרמות ותרשימים מתאימים. בנוסף, עבור כל מודל נחשב את תוצאות החיזוי, את ה MAE\RMSE הממוצע ואת ה Precision/Recall @K. ערכי ה Precision/Recall @ K נקבעים בצורה הבאה:

$$Precision@k = \frac{number\ \ of\ recommended\ items\ @k\ that\ are\ relevant}{number\ of\ recommended\ items\ @k}$$

$$Recall@k = \frac{number\ of\ recommended\ items\ @k\ that\ are\ relevant}{total\ number\ of\ relevant\ items}$$

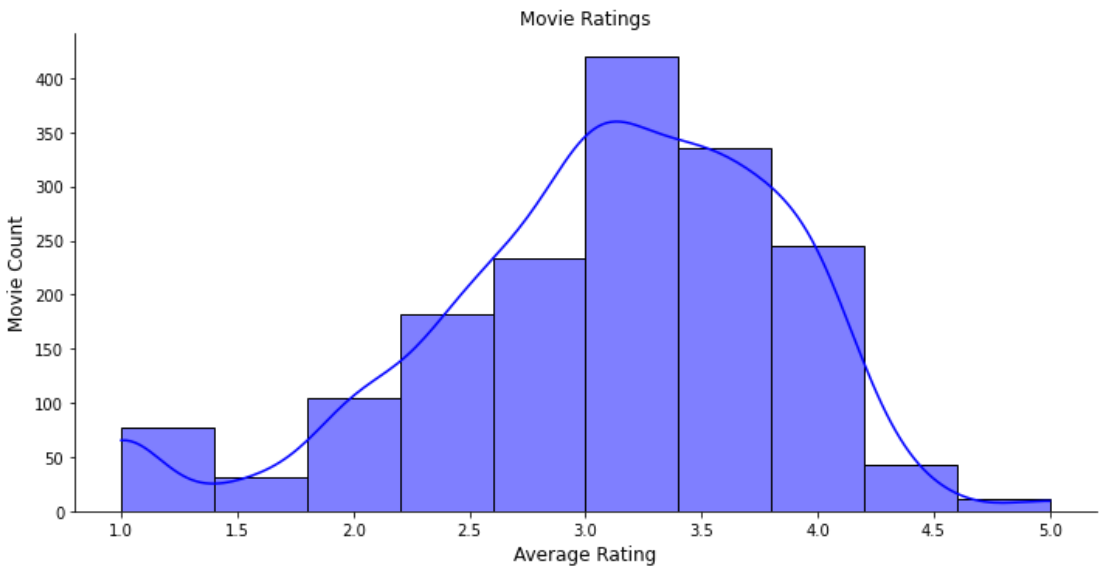
איך נקבע אם Item הוא רלוונטי ?
item רלוונטי נקבע לפי הנחה שאנחנו הנחנו בפרויקט זה שלפיה כל item שעבורו יש רייטינג חזוי שגדול או שווה מ 4.0 (threshold) הוא item רלוונטי, אחרת מדובר ב item לא רלוונטי.

חלק א ניתוח מידע

תרגיל 1

א. *בסעיף זה נדרשנו לחשב עבור כל סרט את ה* rating *הממוצע שלו ולהציג בהיסטוגרמה התוצאות – כאשר בציר* x *זה ה* rating *הממוצע, ובציר* y *זה מספר הסרטים בעלי הרייטינג הנ"ל. הדיאגרמה שקיבלנו לאחר החישוב :*

-

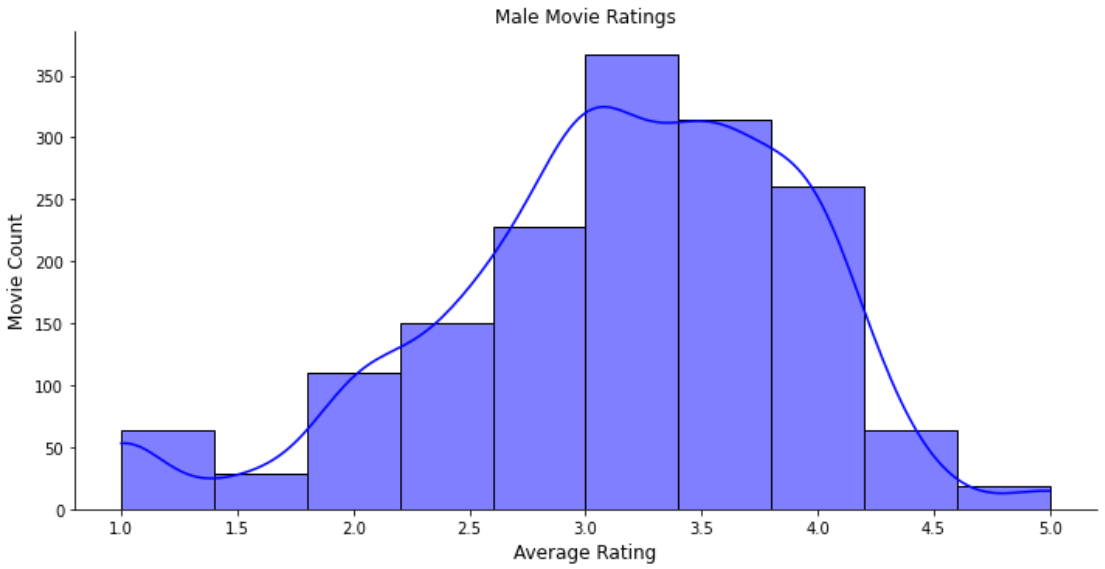


לפי ההתפלגות הנ"ל ניתן לראות כי רוב הסרטים מדורגים בין 3.0 ל 3.5 וכי יש מספר קטן מאוד של סרטים שמדורגים גבוה מאוד (4.5-5.0). לאחר מיון הסרטים לפי רייטינג, שלושת הסרטים בעלי הדירוג הממוצע הגבוה ביותר שהתקבלו הם :

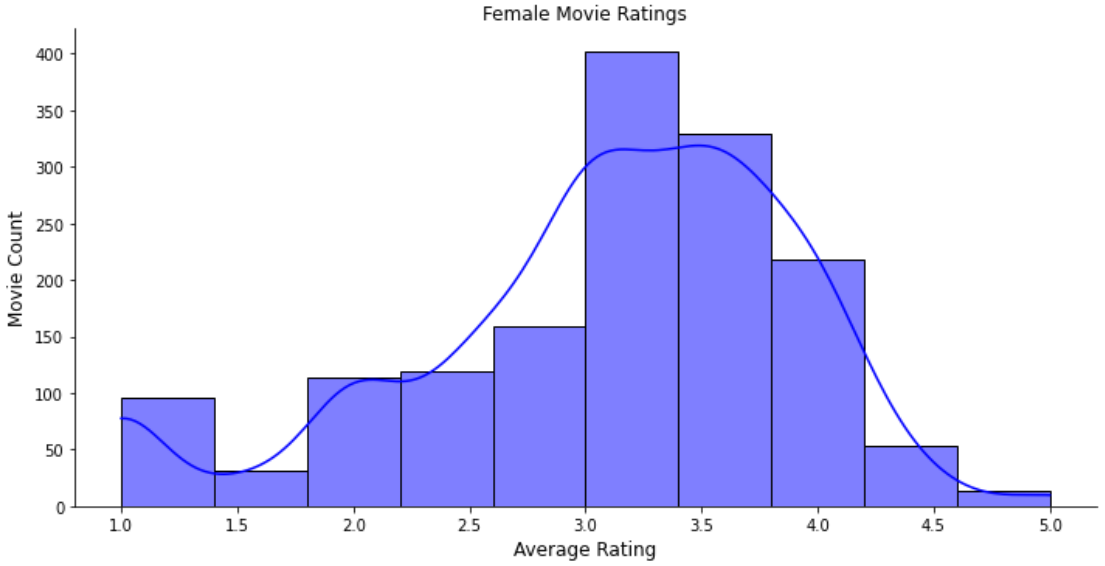
	movie_id	avg_rating	title	count
1188	1188	5.0	Prefontaine (1997)	3
1292	1292	5.0	Star Kid (1997)	3
1466	1466	5.0	Saint of Fort Washington, The (1993)	2

נשים לב שלא מדובר בהכרח בסרטים הפופולריים ביותר, מכיוון שהדירוג פה לא מתחשב בכמות המדרגים של הסרט (כלומר יכול להיות שסרט דורג על ידי אדם אחד בציון 5.0 והוא עדיין נמצא בראש הרשימה).

ב. *בסעיף זה חזרנו על סעיף א' עם התחשבות באוכלוסיית הנשים בנפרד ואוכלוסיית הגברים בנפרד. בכדי לחשב את הרייטינג הממוצע בחתך לאוכלוסיות התייחסנו רק לעמודות שעבורן מתקיים* ('sex == "M"') *ו* ('sex == "F") *עבור גברים ונשים בהתאמה. התוצאה שקיבלנו עבור אוכלוסיית הגברים היא התוצאה הבאה :*



התוצאה שקיבלנו עבור אוכלוסיית הנשים היא התוצאה הבאה:



נשים לב כי ישנה התפלגות דומה ביחס לנתונים עבור כל האוכלוסיה וגם התפלגות דומה בין הגברים לנשים.

את ההבדלים בממוצעים בין 2 האוכלוסיות הצגנו בטבלה הבאה:

	movie_id	rating_male	rating_female	title	rating_abs_diff
2	1305	5.000000	1.0	Delta of Venus (1994)	4.000000
8	850	4.666667	1.0	Two or Three Things I Know About Her (1966)	3.666667
10	1428	4.500000	1.0	Sliding Doors (1998)	3.500000
19	640	4.419355	1.0	Paths of Glory (1957)	3.419355
53	1591	4.250000	1.0	Magic Hour, The (1998)	3.250000
154	1572	4.000000	1.0	Spirits of the Dead (Tre passi nel delirio) (1...	3.000000
1451	838	1.000000	4.0	Loch Ness (1995)	3.000000
156	1557	4.000000	1.0	Aparajito (1956)	3.000000
1435	1025	1.000000	4.0	Lay of the Land, The (1997)	3.000000
1432	912	1.000000	4.0	Love and Death on Long Island (1997)	3.000000

טבלה זאת כוללת את 10 ההפרשים הגדולים ביותר בממוצעים עבור סרט נתון, כלומר הסרטים עם הפערים הגדולים ביותר בדירוג שלהם בין הגברים לנשים.

שלושת הסרטים בעלי הדירוג הגבוה ביותר בקרב הגברים הם :

	movie_id	avg_rating	title	count
1290	1292	5.0	Star Kid (1997)	3
1172	1174	5.0	Hugo Pool (1997)	2
1186	1188	5.0	Prefontaine (1997)	2

בעוד ששלושת הסרטים בעלי הדירוג הגבוה ביותר בקרב הנשים הם :

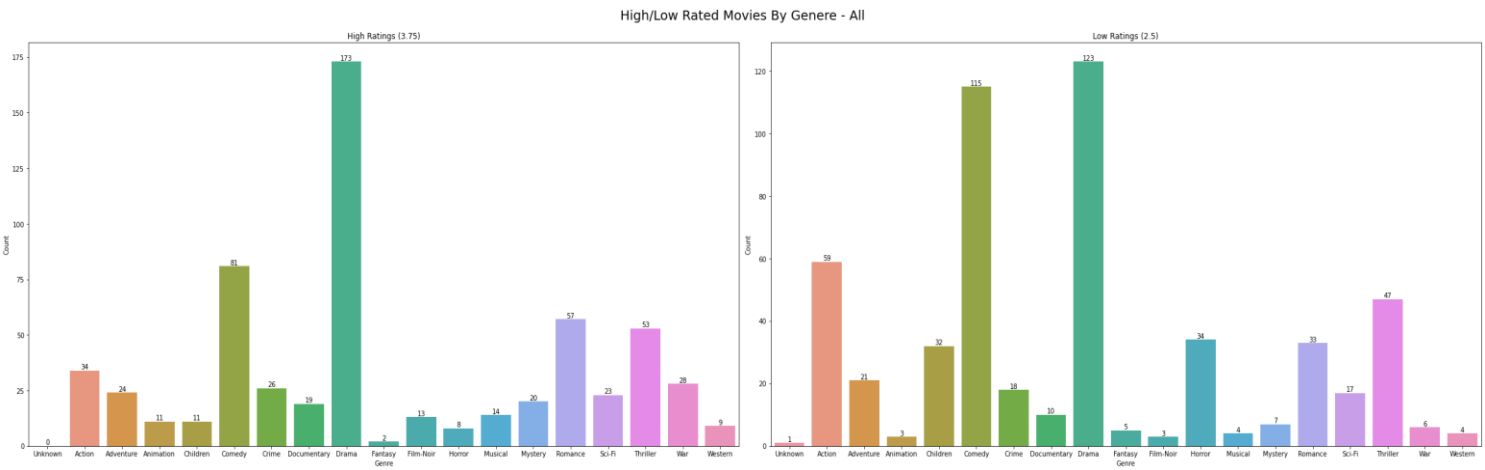
	movie_id	avg_rating	title	count
1302	1367	5.0	Mina Tannenbaum (1994)	2
73	73	5.0	Faster Pussycat! Kill! Kill! (1965)	1
117	118	5.0	Maya Lin: A Strong Clear Vision (1994)	1

מכנה משותף בין הסרטים הוא שרובם סרטי דרמה\רומנטיקה. כמו כן ניתן לראות שברוב הפערים הגדולים ביותר מדובר בסרטי דרמה שאותן נשים דירגו 1.0 בעוד שהגברים דירגו גבוה, דבר שמעיד על כך שנשים ביקורתיות יותר עבור סרטים מסוג זה. למרות התוצאות שראינו בפערים בין גברים לנשים, יש לשים לב כי מדובר בסרטים שדורגו על ידי כמות קטנה מאוד של אנשים ולכן הרייטינג לא מעיד כלום על הפופולריות של הסרט, לכן ביצענו חישוב של הפער עבור סרטים שקיבלו כמות דירוגים גבוהה מממוצע כמות הדירוגים עבור גברים ונשים בהתאמה. התוצאות שקיבלנו נמצאות בטבלה הבאה:

	movie_id	avg_rating_male	count_male	avg_rating_female	title	count_female	rating_abs_diff
1090	719	2.746032	63	3.782609	First Knight (1995)	23	1.036577
1073	154	2.774194	62	3.666667	Dirty Dancing (1987)	36	0.892473
47	524	4.250000	52	3.476190	Big Sleep, The (1946)	21	0.773810
1091	475	2.742574	101	3.491525	First Wives Club, The (1996)	59	0.748951
91	155	4.127119	118	3.433333	Reservoir Dogs (1992)	30	0.693785
314	692	3.791667	72	3.105263	Casino (1995)	19	0.686404
981	553	2.927711	83	2.263158	Waterworld (1995)	19	0.664553
440	484	3.602410	83	4.238095	My Fair Lady (1964)	42	0.635686
1190	28	2.538462	91	3.173913	Batman Forever (1995)	23	0.635452
809	4	3.140625	64	3.772727	Copycat (1995)	22	0.632102

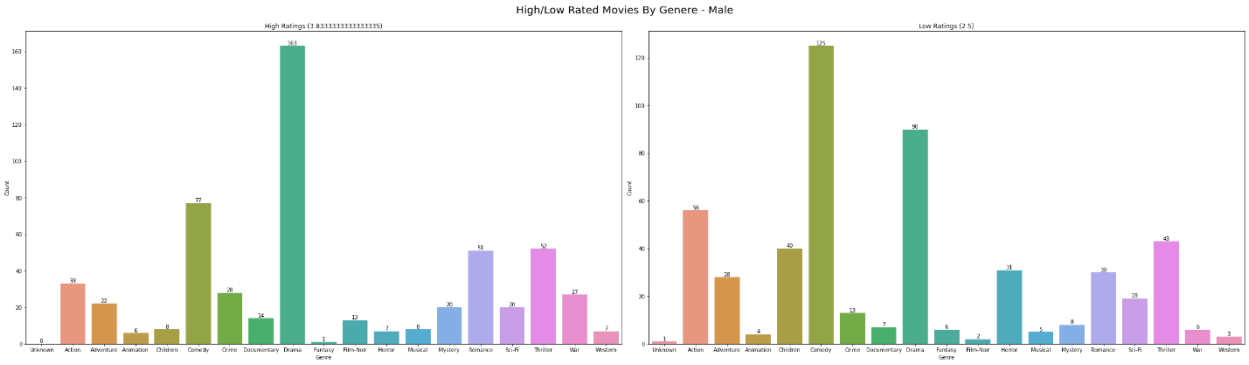
נשים לב כי כאן יש תוצאות שהן הגיוניות יותר. הסרט ריקוד מושחת שהוא סרט דרמה שכל העלילה שלו סובבת סביב ריקודים , דורג הרבה יותר גבוה על ידי הנשים מאשר על ידי הגברים. תרחיש דומה מתקיים עבור הסרט "My Fair Lady". כמו כן, דירוג הסרט "באטמן לנצח" שדורג נמוך יותר על ידי הגברים יכול להעיד על כך שגברים ביקורתיים יותר עבור סרטים המבוססים על גיבורי על.

ג. *בסעיף זה חישבנו תחילה את התפלגות הקטגוריות (genre) של הסרטים בעלי הדירוג הגבוה ביותר והנמוך ביותר עבור האוכלוסייה כולה. הבחירה של הסרטים בעלי הדירוג הגבוה ביותר והנמוך ביותר נקבעה על פי חישוב של אחוזון. בחירה של הסרטים בעלי הדירוג הגבוה ביותר חושבה על פי האחוזון ה 80, ובחירה של הסרטים בעלי הדירוג הנמוך ביותר חושבה על ידי האחוזון ה 20. התוצאות שקיבלנו עבור התפלגות הקטגוריות:*

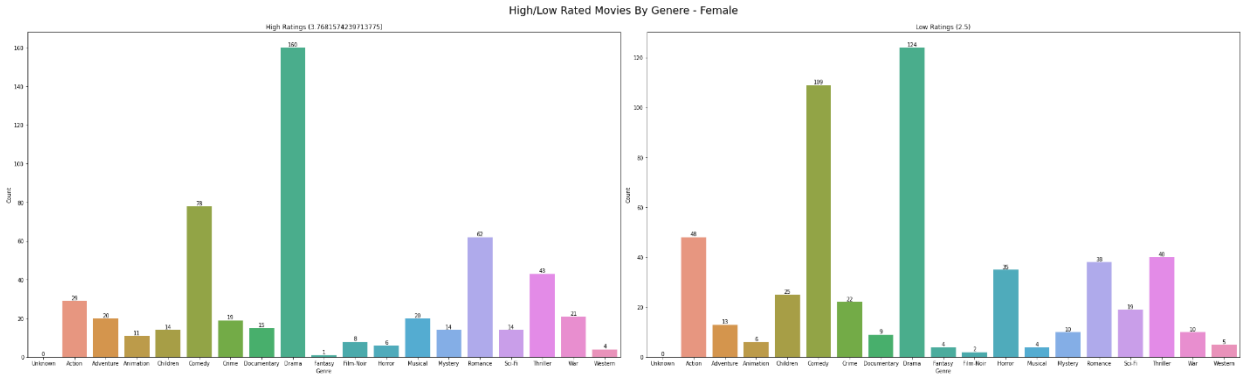


עבור הסרטים בעלי הדירוג הגבוה ביותר קיבלנו 606 השתייכויות של סרטים לז'אנרים ועבור הסרטים בעלי הדירוג הנמוך ביותר קיבלנו 542 השתייכויות של סרטים לז'אנרים. פער משמעותי מאוד שזיהינו עבור האוכלוסיה כולה הוא בסרטי האימה. עבור הסרטים המדורגים גבוה, סרטי האימה משתייכים רק ל $\frac{8}{606} \approx 1\%$, בעוד שעבור הסרטים המדורגים נמוך, סרטי האימה משתייכים רק ל $\frac{34}{542} \approx 6\%$. נתון זה מצביע על כך שסרטים מסוג אימה מבוקרים יותר לרעה.
את ההתפלגות הנ"ל חישבנו גם עבור נתונים דמוגרפיים (מין וגיל) והתוצאות שקיבלנו הן:

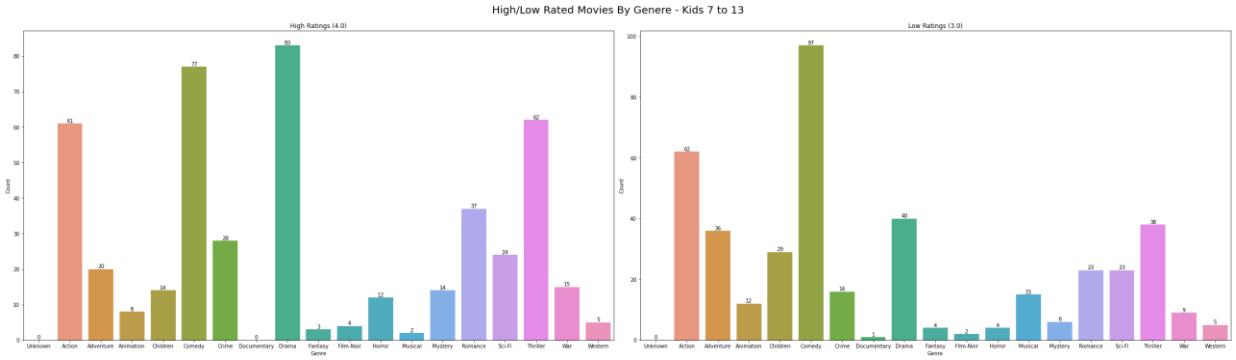
עבור גברים



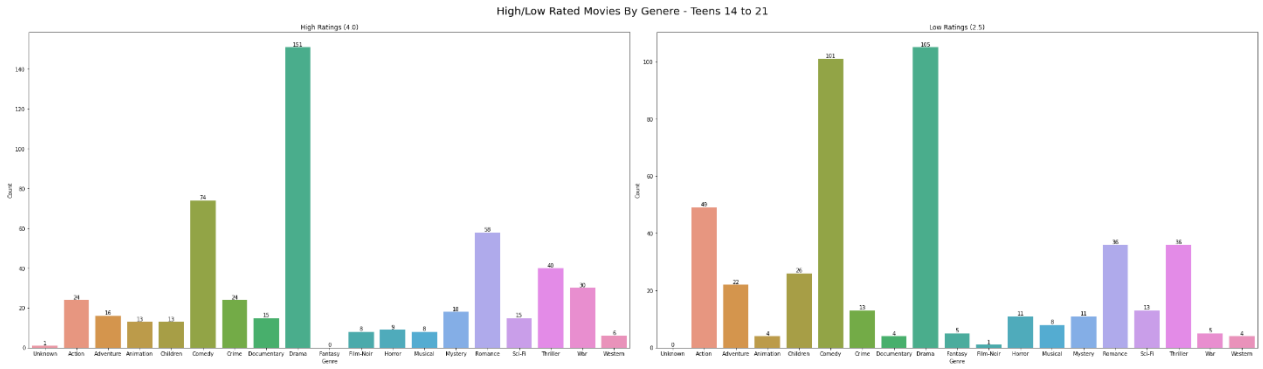
עבור נשים



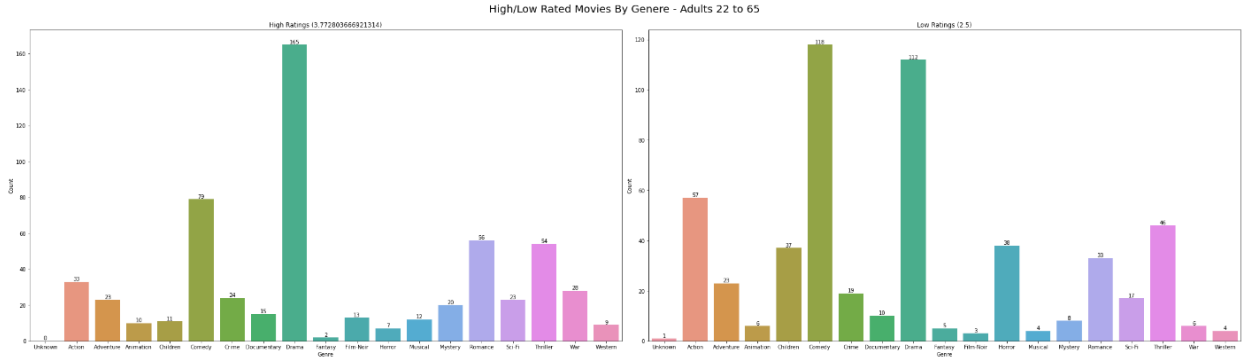
עבור ילדים (גילאים 7-13)



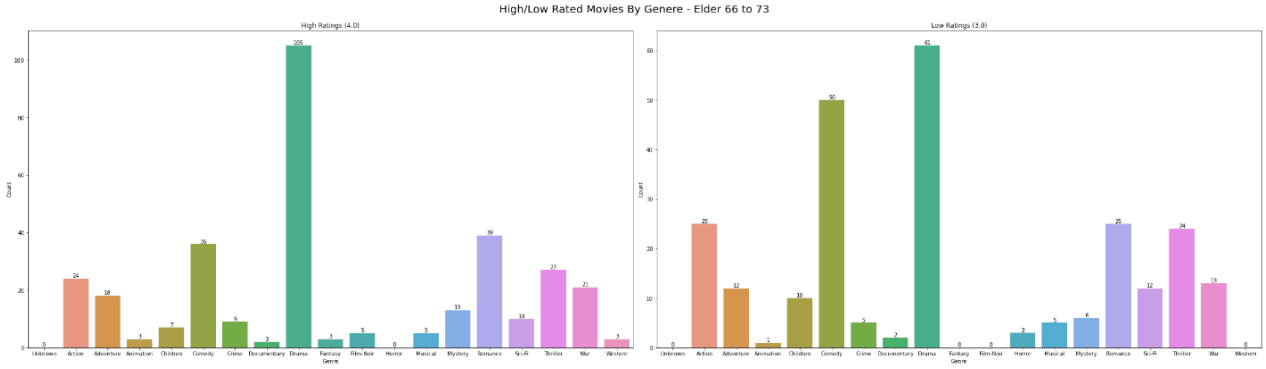
עבור נערים (14-21)



עבור בוגרים(22-65)



עבור מבוגרים(66-73)



מסקנות שניתן להגיע אליהן מההתפלגויות השונות שחישבנו :

- *התפלגות הנשים מול הגברים די דומה ולא זוהה פער גדול מלבד פער בסרטים מסוג דרמה.*
  בהתפלגות הגברים- עבור הסרטים המדורגים נמוך, סרטי הדרמה משתייכים ל $\frac{90}{517} \approx 17\%$
  בהתפלגות הנשים - עבור הסרטים המדורגים נמוך, סרטי הדרמה משתייכים ל $\frac{124}{523} \approx 23\%$
  נתון זה יכול להעיד על כך שבקרב הנשים סרטי דרמה מבוקרים קצת יותר לרעה.
- *נתון מעניין נוסף שניתן לראות הוא שעבור האוכלוסייה המבוגרת והצעירה ערך הרף שנקבע עבור סרטים שמדורגים גבוה הוא 4.0 וערך הרף שנקבע עבור סרטים שמדורגים נמוך הוא 3.0 בעוד שעבור בוגרים ערך הרף הוא 3.7 ו 2.5 בהתאמה.*
- *פער גדול נוסף שזוהה הוא באוכלוסיית הילדים מול הנערים ביחס לסרטי דרמה.*

בהתפלגות הילדים- עבור הסרטים המדורגים נמוך, סרטי הדרמה משתייכים ל $\frac{40}{422} \approx 9\%$
בהתפלגות הנערים - עבור הסרטים המדורגים נמוך, סרטי הדרמה משתייכים ל $\frac{105}{454} \approx 23\%$
ולכן מדובר בפער גדול מאוד שמצביע על כך שנערים ביקורתיים יותר עבור סרטים מסוג זה.

ד. *בכדי למדוד פופולריות של סרט לא מספיק רק לבדוק את ממוצע הדירוגים, זאת מכיוון שיכול להיות סרט שדורג על ידי אדם אחד בציון 5.0 והוא ייחשב כפופולרי ביותר. לכן בכדי למדוד פופולריות של סרט התייחסנו לכמה פרמטרים שונים :*

- *מס' הדירוגים עבור הסרט*
- *מס' מינימלי של דירוגים שעבורו אפשר להיכנס לרשימה*
- *דירוג ממוצע של הסרט*
- *דירוג ממוצע של כל הסרטים בקורפוס*

מס' הדירוגים המינימלי שעבורו סרטים נכנסו לרשימה הוא האחוזון ה 90 , כלומר כמות דירוגים שקיימים 90% שקטנים ממנה. לאחר שנותרה רשימה של סרטים שנשארו מהסינון של האחוזון ה 90, חישבנו ממוצע משוקלל בכדי לקבוע איזה סרט הוא הפופולרי ביותר. החישוב התבצע בצורה הבאה:

Movie Popularity calculation

The formula as used in IMDB for calculating the Top Rated movies gives a true Bayesian estimate:

weighted rating (WR) = (v ÷ (v+m)) × R + (m ÷ (v+m)) × C

where:

- v is the number of ratings for the movie.
- m is the minimum ratings required to be listed in the chart.
- R is the average rating of the movie.
- C is the mean ratings across the whole data.

לאחר החישוב קיבלנו את התוצאות הבאות :

	movie_id	avg_rating	title	count	rating_count	popularity
5	49	4.358491	Star Wars (1977)	583	583	4.070281
0	317	4.466443	Schindler's List (1993)	298	298	3.963279
2	63	4.445230	Shawshank Redemption, The (1994)	283	283	3.933300
11	126	4.283293	Godfather, The (1972)	413	413	3.932735
9	97	4.289744	Silence of the Lambs, The (1991)	390	390	3.922811
...	...	...	...	...	...	...
165	545	3.031496	Broken Arrow (1996)	254	254	3.049294
166	288	2.980695	Evita (1996)	259	259	3.018345
167	322	2.933333	Dante's Peak (1997)	240	240	2.992302
168	234	2.847926	Mars Attacks! (1996)	217	217	2.947802
169	677	2.808219	Volcano (1997)	219	219	2.924875

ה. *כפי שלמדנו בהרצאות, ה sparsity מחושב באמצעות הפרמטרים הבאים :*

- *כמות הרייטינגים*
- *כמות המשתמשים*
- *כמות הסרטים*

החישוב התבצע בצורה הבאה :

```
sparsity = 1 - len(ratings) / (len(users) * len(movies))
```

והתוצאה שקיבלנו היא - `Sparsity: 0.9369533063577546`

מאחר והתבקשנו לחשב את מספר ה rating הממוצע למשתמש, תחילה חישבנו כמה דירוגים כל משתמש נתן והצגנו את המשתמשים המדרגים ביותר בטבלה הבאה:

user_id	
404	737
654	685
12	636
449	540
275	518
415	493
536	490
302	484
233	480
392	448

כלומר משתמש מספר 404 דירג הכי הרבה פעמים (737 דירוגים סה"כ). לאחר מכן הצגנו את הנתונים בטבלה מפורטת שמפרטת גם את הממוצע, סטיית התקן וגם את ערכי האחוזונים השונים:

count	943.000000
mean	106.044539
std	100.931743
min	20.000000
25%	33.000000
50%	65.000000
75%	148.000000
max	737.000000

ניתן לראות ע"פ טבלה זו כי כמות הדירוגים הממוצעת למשתמש היא 106.

חלק ב – המלצות לא אישיות

תרגיל 2

א. *בסעיף זה התבקשנו להציג מודל לחיזוי* rating *לכל סרט. בכדי לחזות רייטינג עבור כל סרט ביצענו טעינה של הקובץ* u1.test *והכנסנו ל*dataframe *כל* movie id *עם הרייטינג שהוא קיבל. יצרנו מחלקה* BaselineRecommender *שבתוכה שמרנו* data *שמחזיק עבור כל סרט את הממוצע המשוקלל שלו ובמחלקה זאת נמצא המודל שבעצם מדובר במיפוי בין כל סרט לממוצע שלו. בפונקציה* fit *ביצענו חישוב של הממוצע המשוקלל עבור כל סרט ושמרנו את המידע כמודל שלנו, בפונקציה* predict *אנו מחזירים עבור כל סרט את הרייטינג שקיבל ממשתמש מסוים ולצד עמודה זאת אנו מחזירים את החיזוי לרייטינג שהסרט אמור לקבל(בעזרת הממוצע המשוקלל שחישבנו קודם לכן), התוצאות שקיבלנו נשמרו ב* dataframe *הבא :*

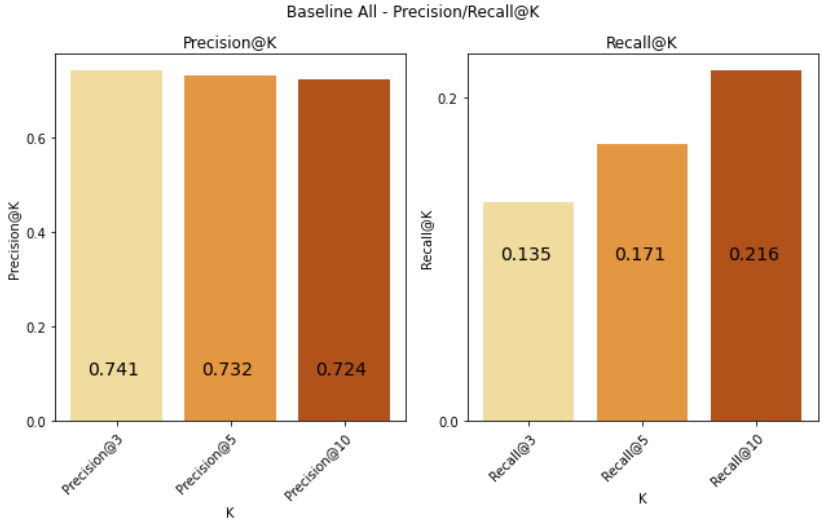
	user_id	movie_id	rating	rating_pred
0	0	5	5	3.576923
1	0	9	3	3.831461
2	0	11	5	4.385768
3	0	13	5	3.967213
4	0	16	3	3.119565
...	...	...	...	...
19995	457	647	4	4.029851
19996	457	1100	4	3.770270
19997	458	933	3	2.926471
19998	459	9	3	3.831461
19999	461	681	5	3.060000

כמו כן ביצענו חישוב של ה MEA *ושל* MSAE *והתוצאות הן :*

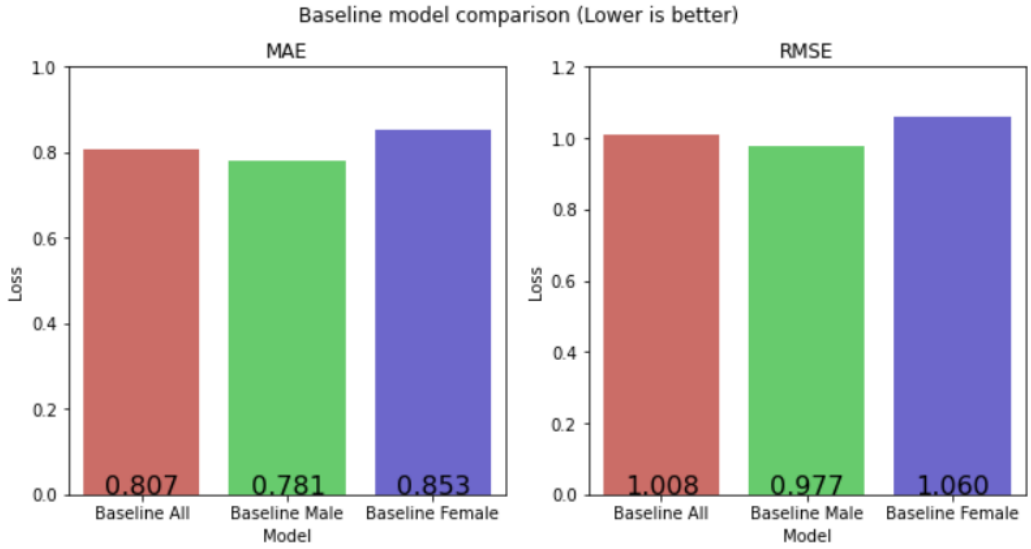
MAE: 0.8072284653453374

$RMSE$: 1.0083009538696794

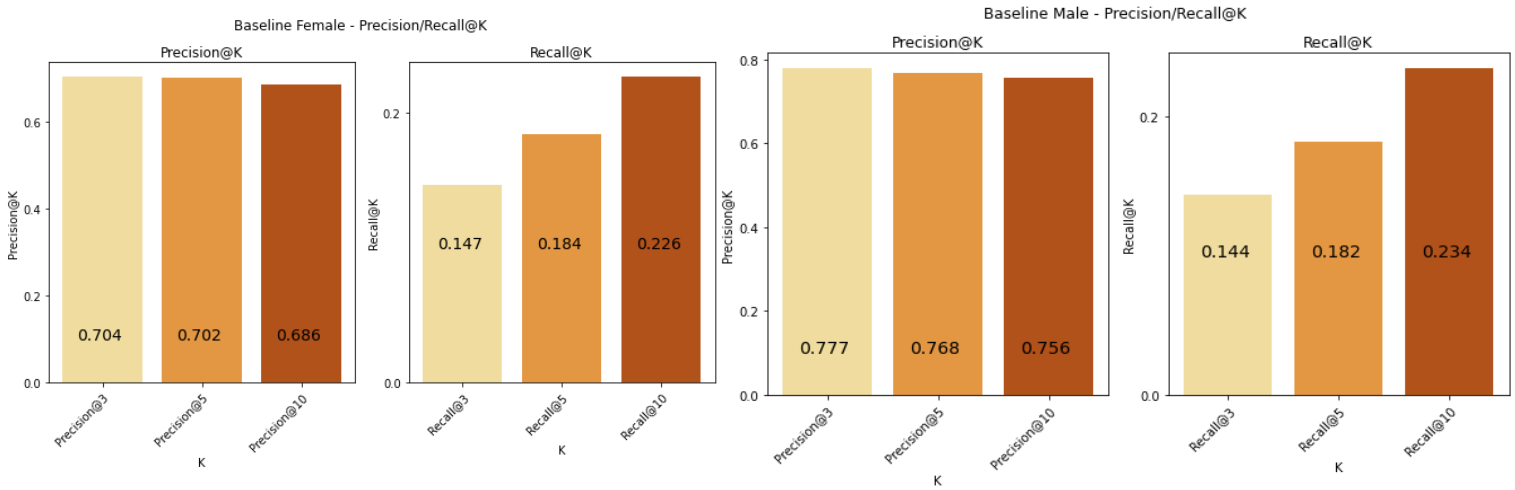
עבור ה Precision\Recall@k קיבלנו את התוצאה הבאה :



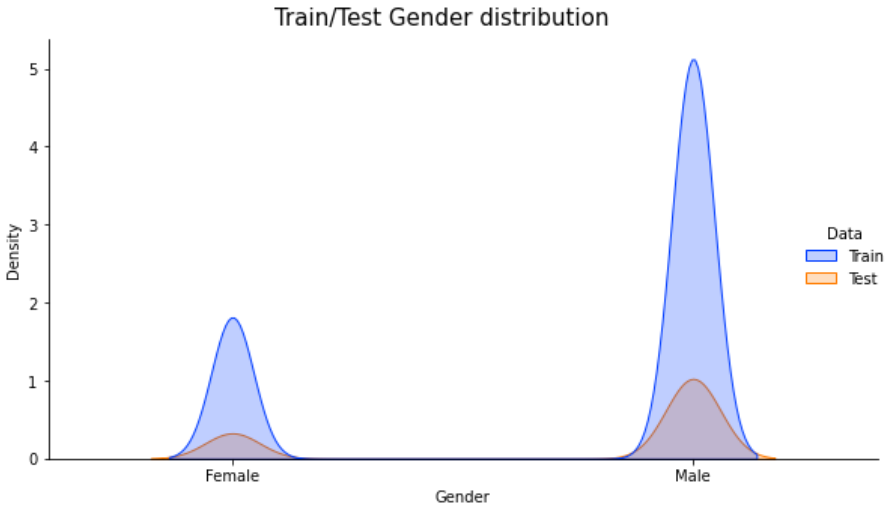
ב. *כפי שנדרשנו, חזרנו על סעיף א' רק שהפעם ביצענו את החיזוי עבור גברים ונשים ולא עבור כל האוכלוסייה. החישוב שבוצע בסעיף א' הוא זהה למעט זה שבפונקציה fit ו predict התייחסנו אך ורק לרשומות בהן המין של המדרג הוא זכר או נקבה בהתאמה. לאחר חישוב החיזוי עבור גברים ונשים הצגנו את ההבדלים בדיאגרמה הבאה:*



כמו כן ערכי ה Precision/Recall @K עבור הגברים והנשים מתוארים בדיאגרמות הבאות:



ניתן לראות כי עבור הגברים קיבלנו את התוצאות הטובות ביותר ועבור הנשים קיבלנו את התוצאות הגרועות ביותר. הסיבה לכך היא שהיה פחות דירוגים של נשים, כלומר המדגם של הנשים קטן יותר. מבדיקה שעשינו, ישנם 4833 דירוגים של נשים בקובץ ה test לעומת 15167 דירוגים של גברים בקובץ ה test. המדגם של הגברים גדול ביותר מפי 3 וזאת הסיבה לכך שעבור הגברים קיבלנו תוצאות טובות יותר. כמו כן על פי גרף של התפלגות המגדר עבור ה Train\Test :



ניתן לראות בבירור שהגברים ב train\test גדולה יותר וזאת הסיבה העיקרית לכך שעבור הגברים קיבלנו תוצאה הרבה יותר טובה.

חלק ג – המלצות אישיות

תרגיל 3

א. בתרגיל זה נדרשנו לממש מודל לחיזוי rating לסרט עבור user על פי מודלים שונים שנלמדו בכיתה. בתרגיל זה השתמשנו בספריית Turi create.
עבור הtrain במודל זה השתמשנו בקובץ u1.base , ועבור ה test השתמשנו בקובץ u1.test.
תחילה יצרנו את המודלים באמצעות פונקציית create של ספריית Turi create. עבור item similarity יצרנו מודל עבור 2 שיטות שונות – pearson ו cosine.
בשיטת ה cosine similiraty, הדמיון בין 2 items נקבע על פי חישוב של מרחק וקטורי בצורה הבאה:

$$similarity(A,B) = \frac{A \cdot B}{\|A\| \times \|B\|} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \times \sqrt{\sum_{i=1}^{n} B_i^2}}$$

לעומת שיטת pearson שלפיה דמיון בין 2 items נקבע על פי החישוב הבא:

$$r_{xy} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}$$

ב. תוצאות שקיבלנו עבור כל אחד מהמודלים :

- Item similarity-cosine:

$$\text{MAE} = 3.265873325082315$$
$$\text{Overall RMSE} = 3.451457615312013$$

- Item similarity-pearson:

$$\text{MAE} = 0.8258111410295526$$
$$\text{Overall RMSE} = 0.8258111410295526$$

ניתן לראות בבירור כי המודל ה Item similarity החזיר תוצאות טובות יותר בשיטת pearson.

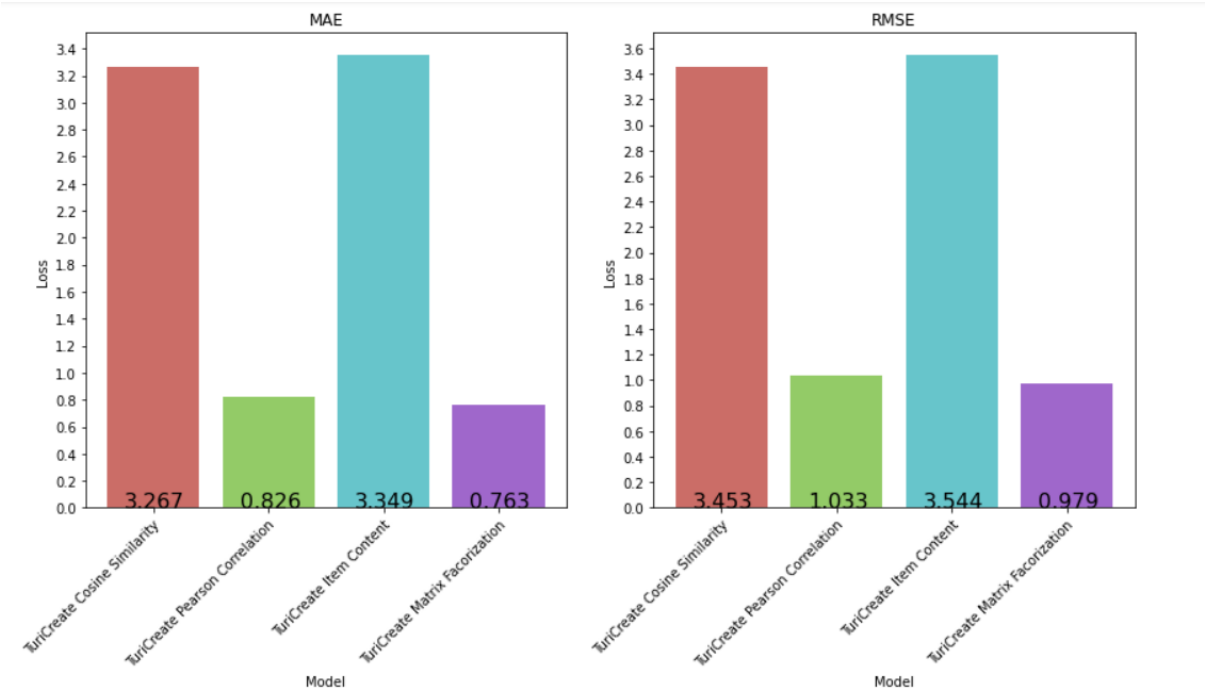
- Item content:

$$\text{MAE} = 3.348997427662275$$
$$\text{Overall RMSE} = 3.5437546598043004$$

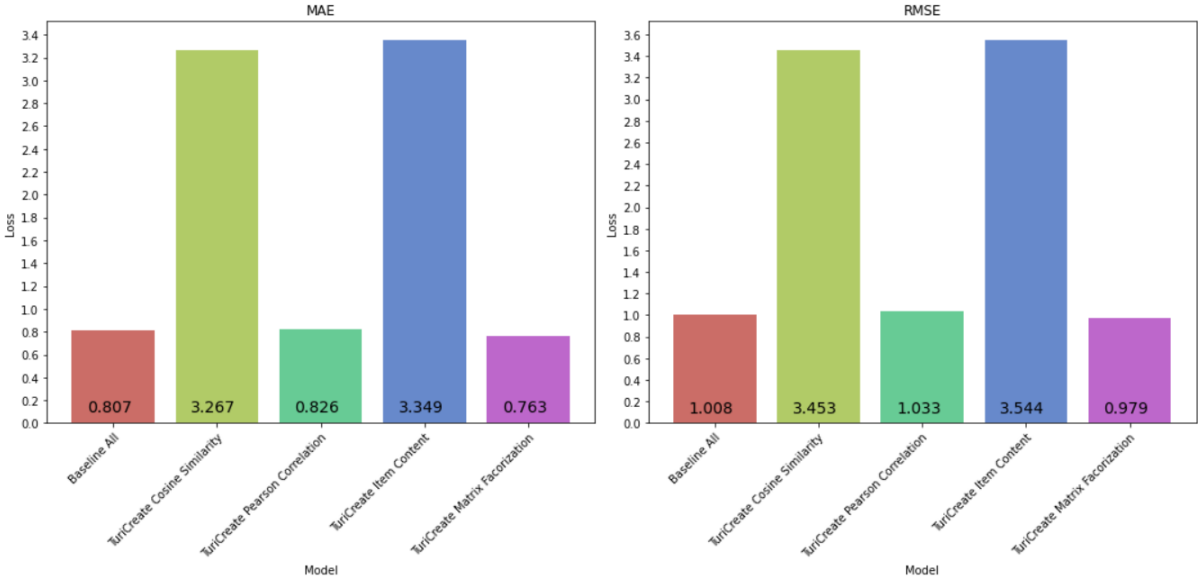
- Matrix factoralization:

$$\text{MAE} = 0.8115838801784592$$
$$\text{Overall RMSE} = 1.0590465422889352$$

נשים לב כי ערך ה MAE שהתקבל בשיטת ה matrix factoralization הוא הערך הנמוך ביותר ולכן מדובר במודל עם התוצאות הטובות ביותר לחיזוי דירוג. התוצאות הגרועות ביותר התקבלו במודל ה item content. הצגנו את ההשוואה בין המודלים בדיאגרמה מסודרת:



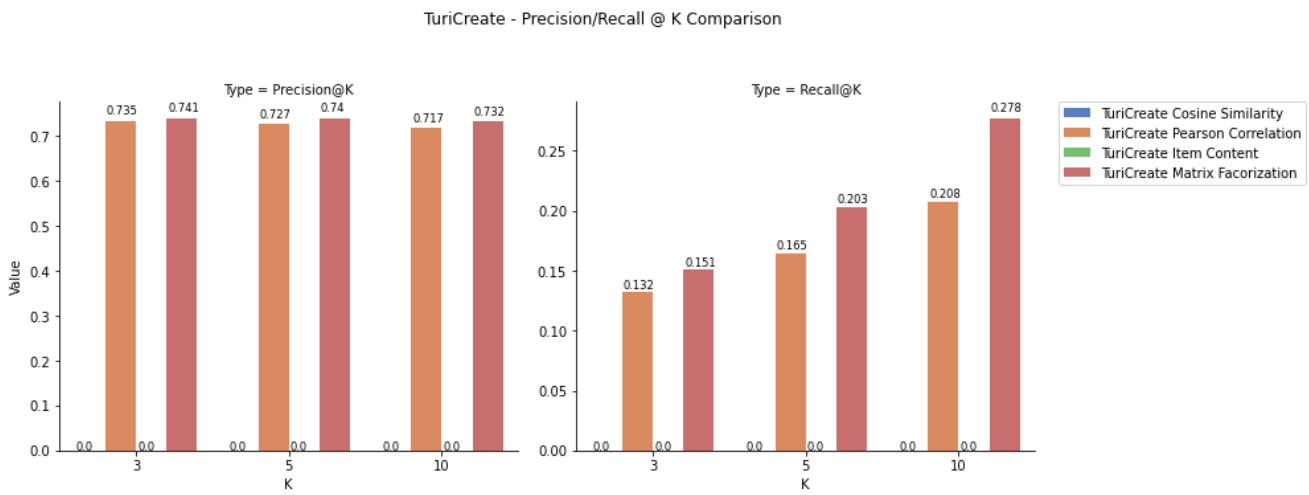
ג. בהשוואה שערכנו למודלים שבסעיף זה למודל בתרגיל 2, עבור MAE ו RMSE קיבלנו את התוצאות הבאות :



כאשר Baseline All מייצג את הערכים שקיבלנו בתרגיל 2, ניתן לראות כי על פי ה MAE ו RMSE , התוצאות של מודל ה matrix factoralization עדיין הטובות ביותר גם ביחס למודל שבתרגיל 2. לעומת זאת, המודל בתרגיל 2 מציג תוצאות טובות יותר בהשוואה למודלים האחרים שחושבו בתרגיל זה. משך האימון עבור כל מודל שהתבקשנו לחשב :

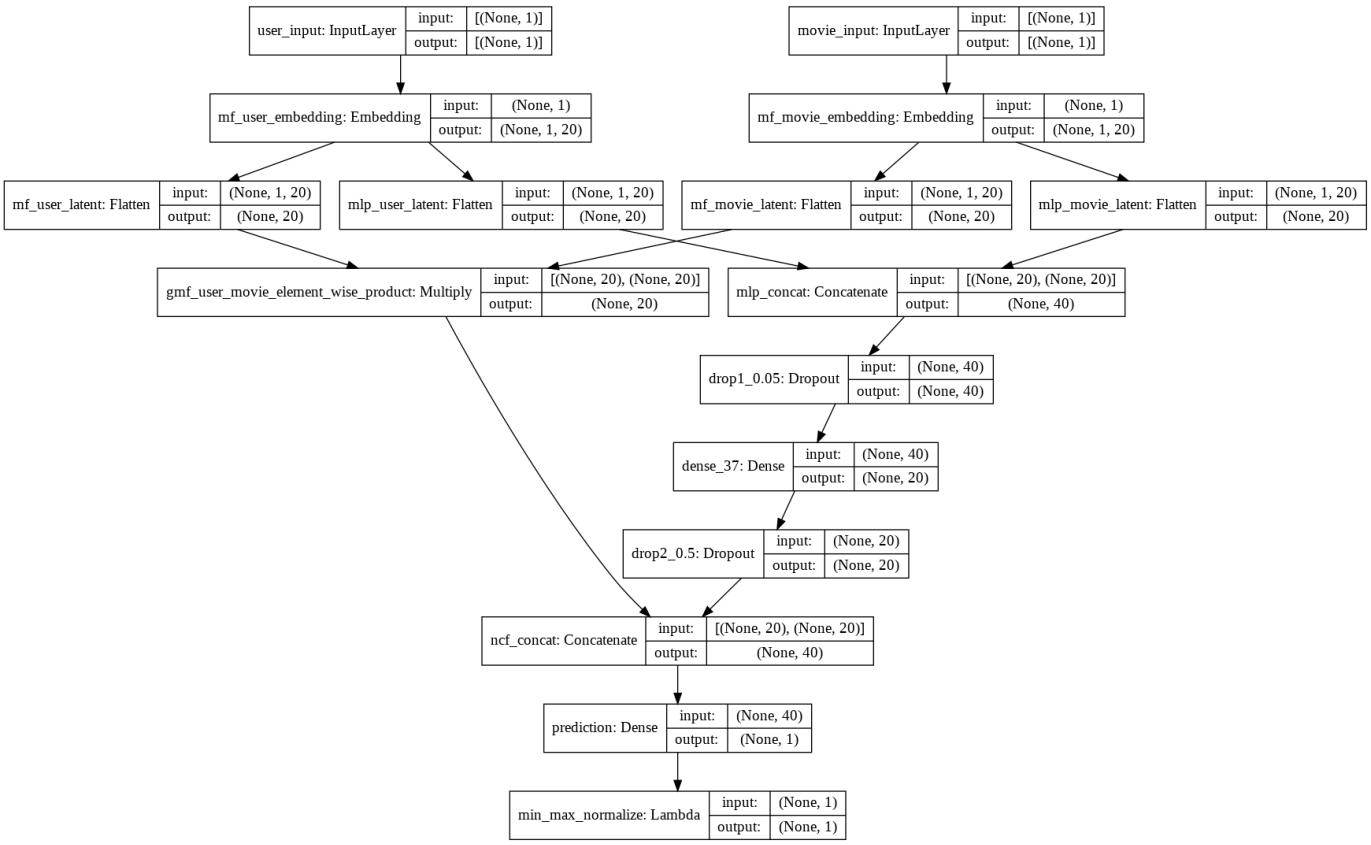
- 0.2507 sec - Cosine similarity
- 0.3179 sec - Pearson correlation
- 0.033861 sec - Item content
- 2.9341 sec - Matrix factoralization

לפי הזמנים שקיבלנו ניתן לראות כי משך האימון המהיר ביותר היה עבור המודל item content בעוד שמשך האימון האיטי ביותר היה עבור המודל matrix factoralization. למרות שמשך האימון הארוך ביותר היה עבור המודל הנ"ל, הוא הציג את התוצאות הטובות ועבורו התקבלו החיזויים הקרובים והטובים ביותר. בנוסף לכך, לפי השוואה של ערכי ה Precision/Recall ניתן לראות בדיאגרמה הבאה כי התוצאות הטובות ביותר התקבלו גם כן עבור matrix factoralization:

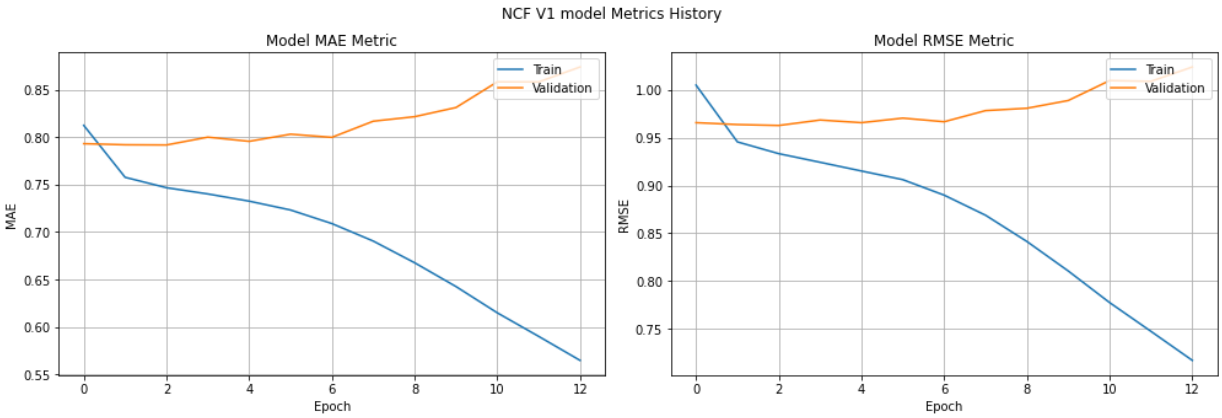


תרגיל 4

א. *בתרגיל זה נדרשנו לממש מודל לחיזוי* rating *על פי מודל ה* neural collaborative filtering. *כפי שנדרש התחלנו ממודל עם שכבת* hidden *אחת. המודל שמימשנו כולל* Matrix factorization embedding layer *שכוללת וקטור אחד עבור ה* users *ווקטור אחד עבור המידע על הסרטים. שכבה זו עוברת דרך שכבה שמבצעת שיטוח(*flatten*) בכדי להוריד מימד מיותר עבור ערכי ה* input. *אחרי השיטוח אנו מבצעים הכפלה של 2 הוקטורים הנ"ל כאשר כל ערך במקום ה*i *בווקטור הראשון מוכפל בערך במקום ה* i *בוקטור השני בהתאמה. עבור ה* MLP *קיימת שכבה אחת כנדרש בשאלה עם פונקציית* $activation$ RELU. *לאחר מכן מתבצע שרשור של התוצאה משכבת ה* MLP *ושכבת ה* GMF*(השכבה שעבורה התבצעה ההכפלה של הווקטורים).בנוסף, לשכבת ה* MLP *הוספנו 2 שכבות שמבצעות* DROPOUT *בכדי למנוע* Overfitting *המודל שבנינו בצורה הוויזואלית שלו נראה כך :*



התוצאות שקיבלנו עבור המודל הנ"ל הן :



ניתן לראות בבירור שמתקיים Overfitting. ככל שהאיטרציות עולות ה Loss ב Train יורד אך ניתן לראות כי ה Loss ב Validation עולה עם האיטרציות, מה שמצביע על כך שהמודל לומד את הערכים שנמצאים ב Train אך הוא לא מצליח לחזות דברים חדשים שטרם ראה. בסעיף הבא ננסה לשפר מודל זה באמצעות שיטות שונות שנפרט עליהן בהרחבה.

ב. *עבור המודל מסעיף א' קיבלנו כי ה MAE ו RMSE הממוצעים הם :*

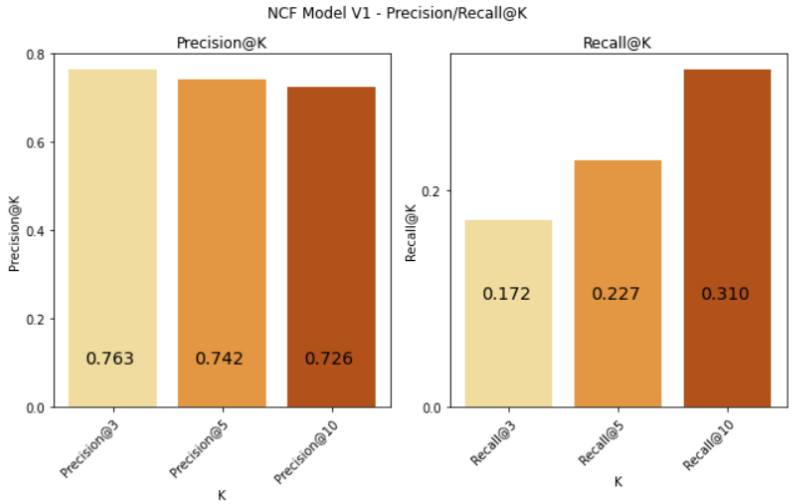
$MAE = 0.7796024680137634$

$RMSE = 0.9821773171424866$

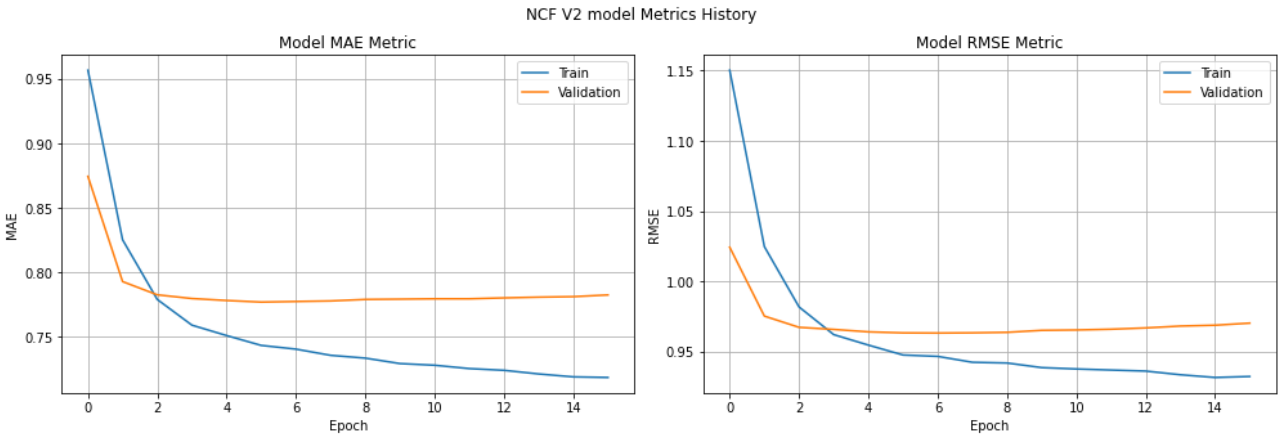
כמו כן תוצאות החיזוי עבור מודל זה מציגות את הטבלה הבאה:

	user_id	movie_id	rating	unix_timestamp	rating_pred
0	0	5	5	887431973	3.800531
1	0	9	3	875693118	4.134938
2	0	11	5	878542960	4.930750
3	0	13	5	874965706	4.294693
4	0	16	3	875073198	3.214992
...	...	...	...	...	...
19995	457	647	4	886395899	3.928123
19996	457	1100	4	886397931	3.894881
19997	458	933	3	879563639	3.131896
19998	459	9	3	882912371	3.934884
19999	461	681	5	886365231	3.312722

ועבור חישוב precision/recall @k קיבלנו את הדיאגרמה הבאה:



- *כעת, בכדי לשפר את המודל הנ"ל ניצור מודל חדש בו נקטין את ה learning rate מ 0.001 ל 0.0001 בכדי למנוע את התבדרות המודל ולמנוע את ה Overfitting שנוצר במודל הקודם מסעיף א'. כמו כן בסעיף א' ה loss function הוגדר להיות MSE, כעת נשנה זאת ל MAE. נשים לב כי שכבות המודל נשארו זהות והשינויים שבוצעו הם רק השינויים שציינו כעת. לאחר שהרצנו את המודל החדש קיבלנו שיפור משמעותי מאוד – כעת כאשר האיטרציות עולות ה Loss של ה Validation נמצא במגמת ירידה ולא במגמת עלייה כמו קודם לכן. נשים לב לשינויים בגרף הבא:*



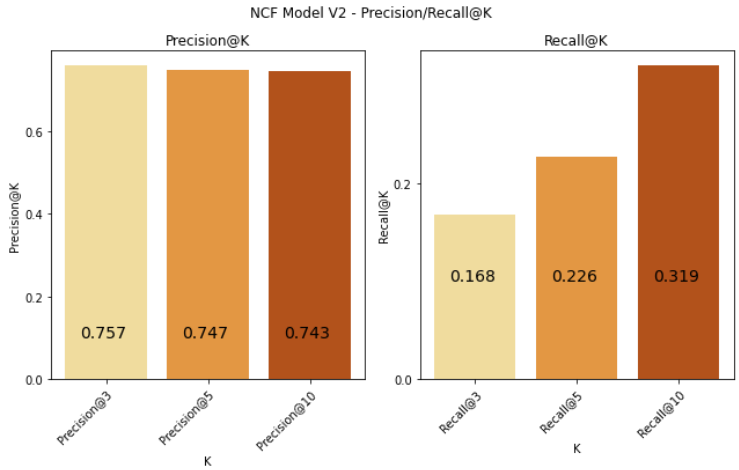
נשים לב על פי הגרף שמנענו את ההתבדרות של המודל שהייתה בסעיף א'. ישנו שיפור משמעותי במודל והמודל הנ"ל סובל פחות מ Overfitting אך עדיין ניתן לשפר אותו מעט.
עבור מודל זה קיבלנו RMSE/MAE ממוצעים :

$$MAE = 0.7393589019775391$$
$$RMSE = 0.9469417929649353$$

וניתן לראות שערך ה MAE הממוצע ירד מעט.
את החיזויים של המודל הדפסנו לטבלה מסודרת שניתן לראות את חלקה כאן :

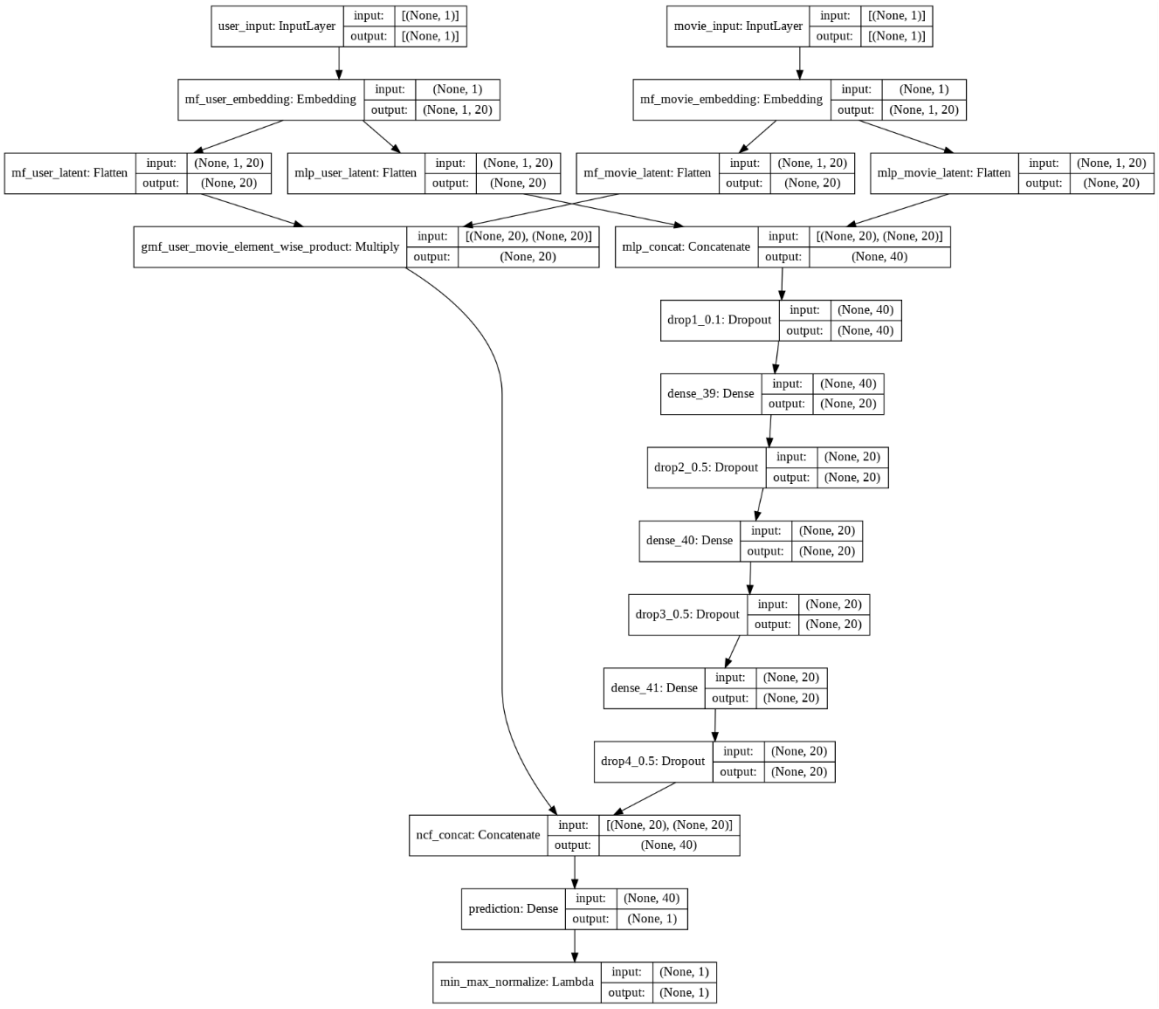
	user_id	movie_id	rating	unix_timestamp	rating_pred
0	0	5	5	887431973	4.339095
1	0	9	3	875693118	4.568455
2	0	11	5	878542960	4.988988
3	0	13	5	874965706	4.396618
4	0	16	3	875073198	3.649216
...	...	...	...	...	...
19995	457	647	4	886395899	4.133552
19996	457	1100	4	886397931	4.113035
19997	458	933	3	879563639	2.999519
19998	459	9	3	882912371	3.334958
19999	461	681	5	886365231	4.054100

ועבור חישוב precision/recall @k קיבלנו את הדיאגרמה הבאה:

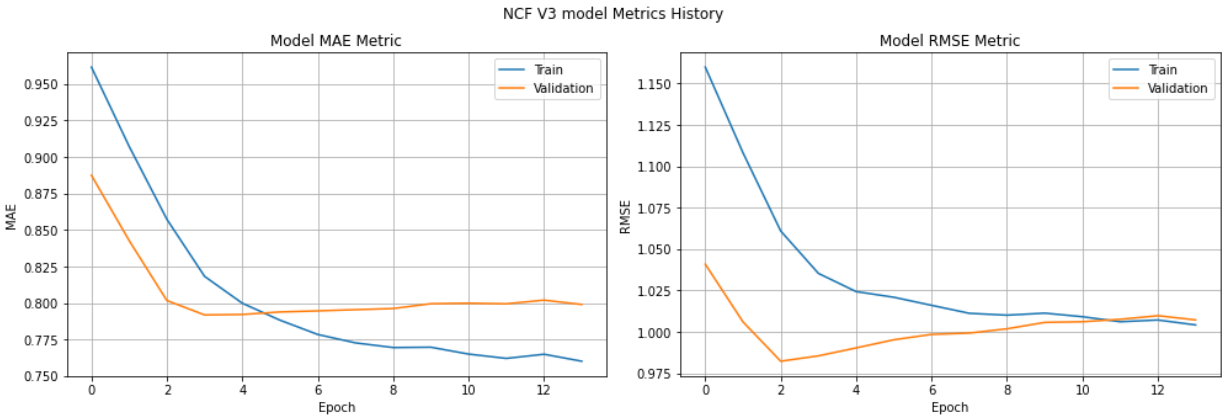


ניתן לראות על פי הדיאגרמה כי ערכי ה precision וה recall של מודל זה גבוהים יותר, דבר אשר מצביע על שיפור של המודל מסעיף א'.

- *לאחר שיפור המודל מסעיף א', ננסה ליצור מודל נוסף עם פרמטרים שונים ונראה אם נצליח לשפר יותר את המודלים הקיימים. כעת ננסה ליצור מודל עמוק יותר באמצעות הוספת 2 שכבות נוספות ל MLP , כלומר המודל שיצרנו כעת הוא מודל עם 3 hidden layers בגודל 20 , במקום שכבה אחת בגודל 20. המטרה היא ליצור מודל עמוק יותר בכדי לראות אם זה ישפר יותר את המודלים הקודימים שעשינו. ניתן לראות את המודל שיצרנו באמצעות המחשבה ויזואלית שיצרנו :*



ניתן לראות כי ההבדל העיקרי מהמודל הקודם הוא בכך שנוספו 3 שכבות ל MLP שביניהם קיים DROPOUT שמטרתו להקטין את ה Overfitting. במודל זה ה learning rate מוגדר על 0.0001 כמו הקודם. התוצאות שקיבלנו על פי מודל זה הן :

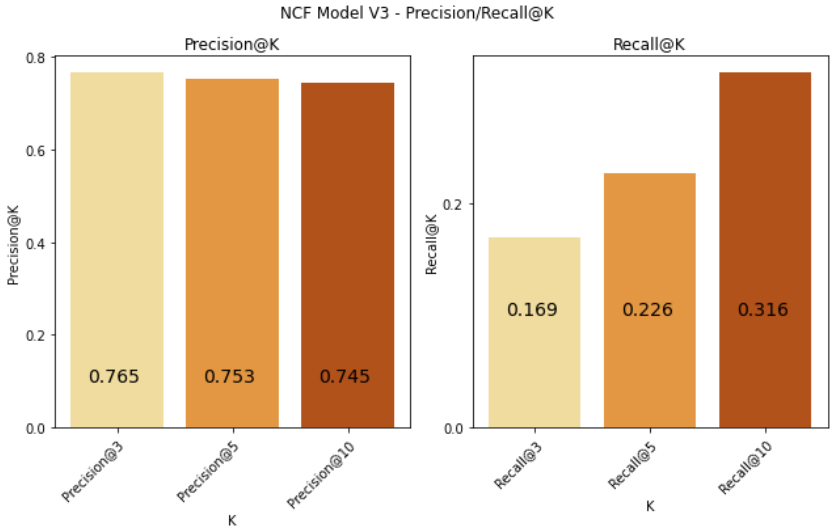


על פי הגרפים ניתן לראות כי ה MAE נמצא במגמת ירידה, עם כי בשלב מסויים לאורך האיטרציות הוא מתחיל לעלות קצת. בנוסף, ערך ה RMSE נמצא תחילה במגמת ירידה ולאחר מכן נמצא במגמת עלייה מתמדת. ניתן לראות מודל זה גם

הוא פחות סובל מ Overfitting מהמודל הראשון מסעיף א' אך עדיין מהתוצאות שקיבלנו עבור ה RMSE\MAE הממוצעים :

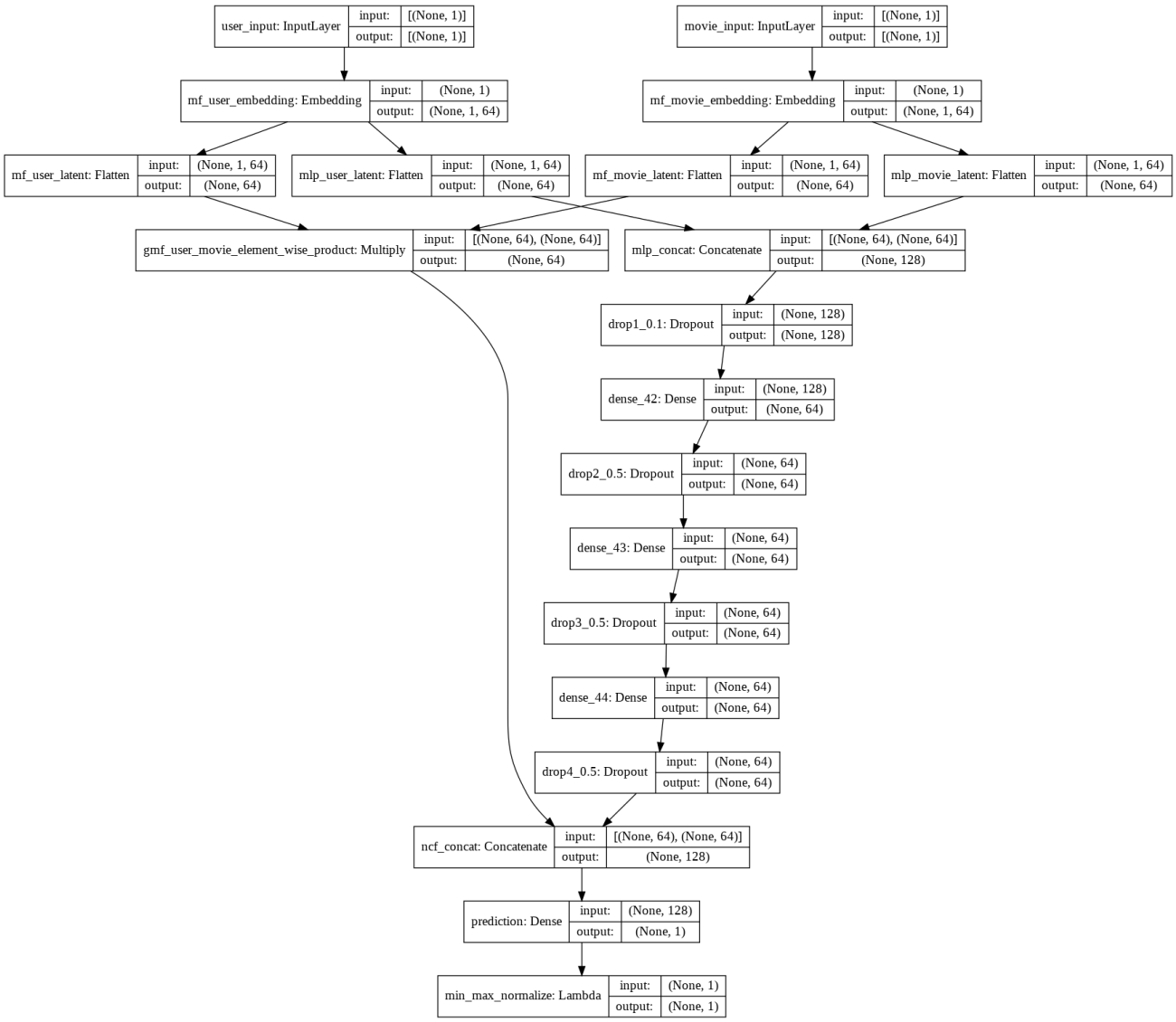
$$MAE = 0.7854375243186951$$
$$RMSE = 1.0197725296020508$$

ניתן להסיק כי מעבר של המודל הקודם למודל עמוק יותר לאו דווקא שיפר את יכולת החיזוי אלא להפך, ערך ה MAE ו ה RMSE עלו וניתן להסיק כי מודל זה הוא עם תוצאות פחות טובות. עם זאת, הדיאגרמה של ערכי ה precision/recall @k :

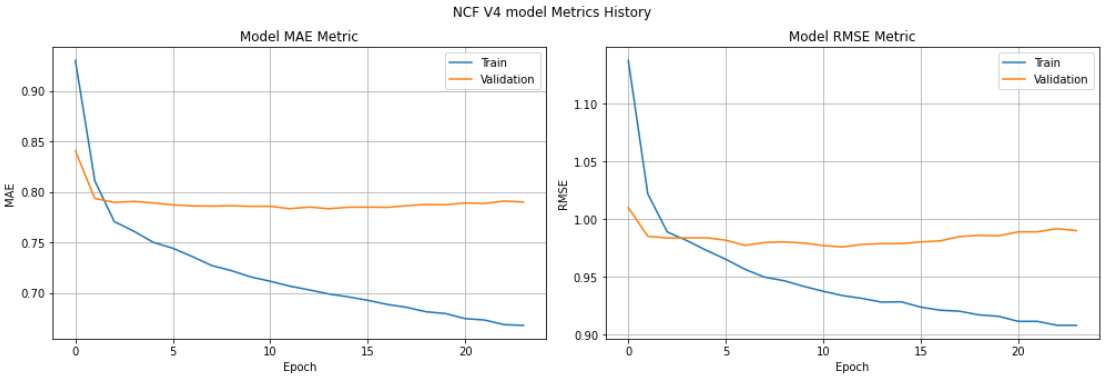


מצביעים על ערכי recall ו precision גבוהים במעט.

- *כעת בפעם השלישית ננסה ליצור דווקא מודל רחב יותר ולהגדיל את כמות הנוירונים מ 20 ל 64. יצרנו מודל זה דווקא בכדי לנסות לשפר את המודל הקודם שבנוי מ3 hidden layers. המודל שיצרנו נראה כך בצורה ויזואלית :*



ניתן לראות כי מדובר באותו מודל כמו המודל העמוק הקודם בעל 3 שכבות ה MLP רק שהפעם מדובר במודל רחב יותר עם כמות גדולה יותר של נוירונים .(אפשר לראות כי המימדים של כל השכבות גדלו יותר ב input וב output). התוצאות שקיבלנו על פי מודל זה הן :

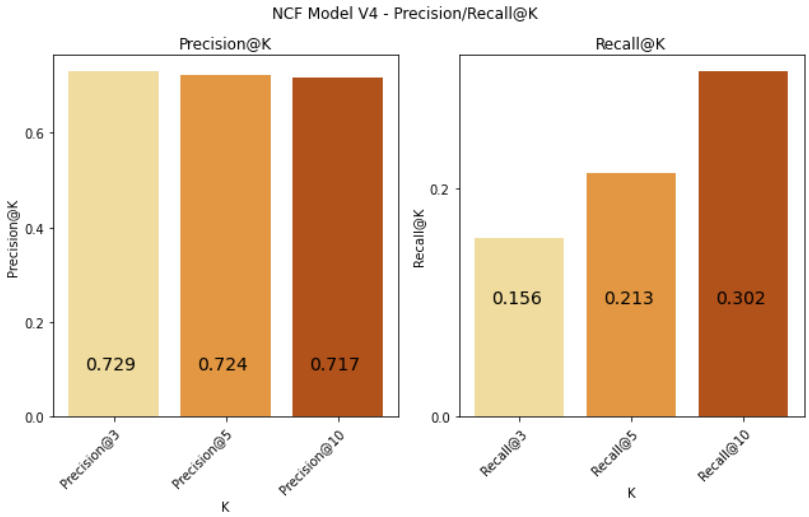


הגרפים מצביעים על ערכים נמוכים של MAE ו RMSE. ערך הMAE נמצא במגמת ירידה עבור ה Validation אם כי בשלב מסויים מתייצב לאורך האיטרציות. הגרפים מציגים שיפור משמעותי מהמודל הקודם שכן ערך RMSE עלה בצורה קיצונית יותר וכן ערכי ה RMSE \MAE הממוצעים עבור מודל זה הם:

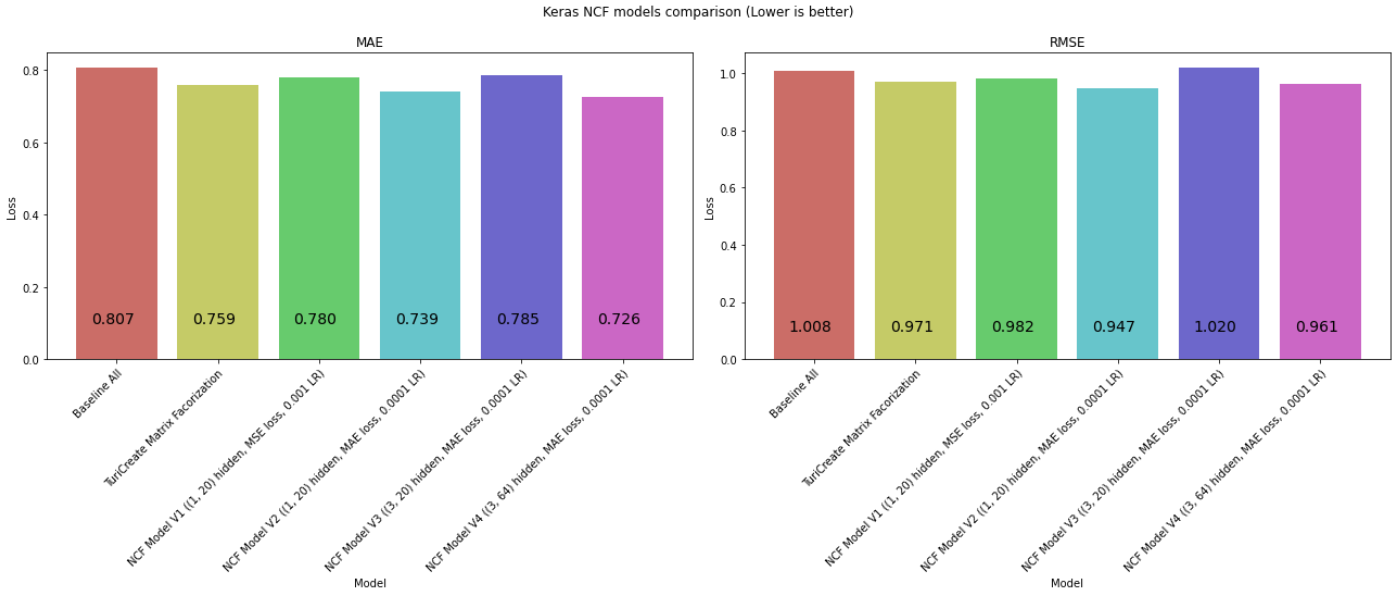
$$MAE: 0.7257158160209656$$

$$RMSE: 0.9612079858779907$$

ערכים אלו מצביעים על שיפור של המודל הקודם. עם זאת, הדיאגרמה של ערכי ה precision/recall @k מציגים כי ערך ה precision קטן יותר אך ערך ה recall גדול יותר עבור מודל זה:

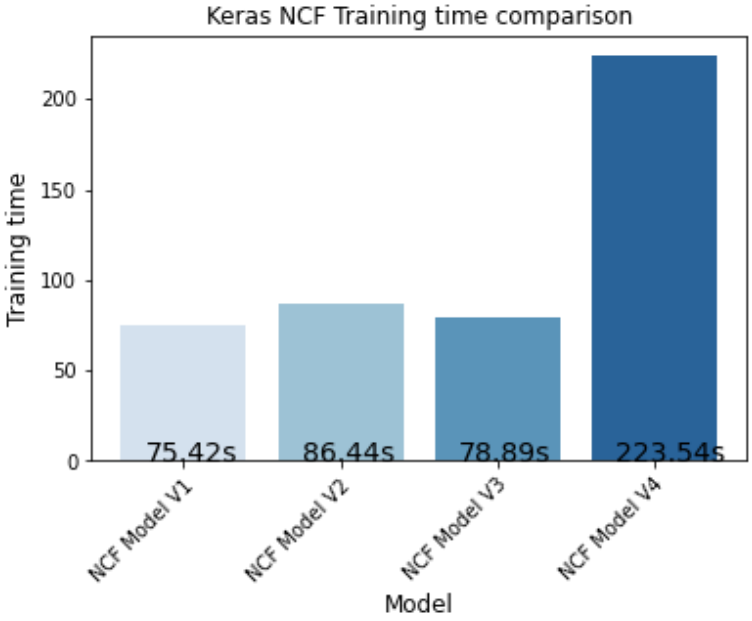


ג. *לסיכום, ביצענו השוואה של כל המודלים שעשינו עד כה (בשאלה זאת, שאלה 3 ושאלה 2) את ההשוואה יצרנו בצורה ויזואלית באמצעות דיאגרמה מתאימה. עבור ערכי ה MAE וערכי ה RMSE :*



ע"פ הדיאגרמה ניתן לראות כי ערך ה MAE הנמוך ביותר(הטוב ביותר) שקיבלנו הוא עבור מודל ה neural collaborative filtering *האחרון שיצרנו (המודל העמוק והרחב). כמו כן ערך ה* RMSE *הנמוך ביותר מתקבל גם הוא עבור המודלים של* neural collaborative filtering *(השני והאחרון). מהנתונים ניתן להסיק כי השיטה שעבדה בצורה הכי טובה היא* neural collaborative filtering *עם כמות נוירונים רחבה ועם מס' hidden layers.*

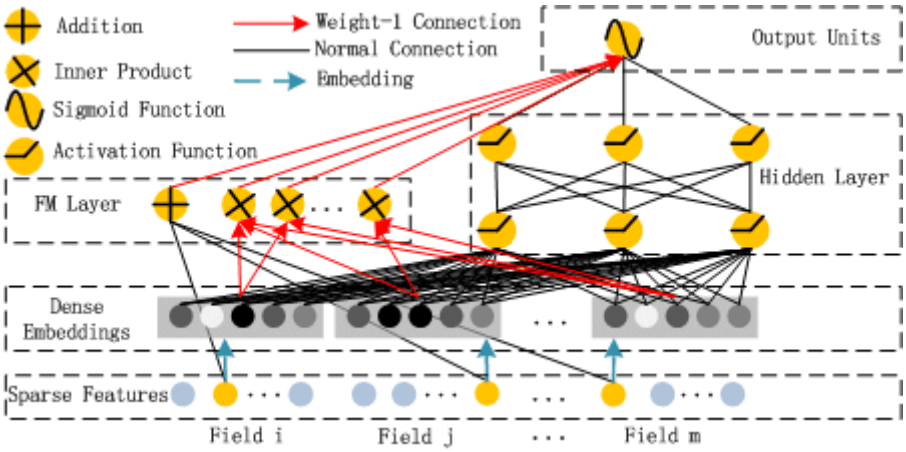
במבחן הזמן קיבלנו את התוצאות הבאות :



ניתן לראות כי המודל האחרון שיצרנו הוא גם הכי בזבזני מבחינת זמן בפער עצום. למרות שהוא מציג תוצאות טובות יותר, לוקח לו המון זמן לבצע את הלמידה. לעומתו, המודל הראשון של neural collaborative filtering *שהצגנו הוא המהיר ביותר. המודל האחרון שהצגנו מציג את התוצאות הטובות ביותר מאחר והוא לומד בצורה עמוקה יותר את ה* train *והוא בעל מספר שכבות ונוירונים גדול יותר מהמודלים האחרים .*

תרגיל 5

בתרגיל זה נדרשנו לממש מודלים נוספים לחיזוי הרייטינג המתחשבים במאפיינים נוספים של הסרט או של הצופה. בתרגיל זה יצרנו מודלים של DeepFM, *מודלים אלו משלבים* factorization machines *לצורך המלצות ו* deep learning *לצורך למידה באמצעות רשתות נוירונים. בסופו של דבר 2 יחידות נפרדות אלו מקבלות את אותו* input *וה* output *של כל אחד מהיחידות הנ"ל נסכם לצורך החיזוי הסופי. מבנה מודל זה נראה בצורה הזאת :*



כאשר בצד שמאל ניתן לראות את ה factorization machines (FM layer) *ומצד ימין מדובר בשכבת הלמידה באמצעות רשתות נוירונים. ניתן לראות על פי ציור זה כי ה*output *של כל יחידה נפרדת נסכם לבסוף לכדי תוצאה סופית של החיזוי.*

הנחה: *כל מודל* DeepFM *שניצור יהיה במבנה הבא שפירטנו במחברת הפרוייקט שלנו :*

DeepFM model architecture

- 1st order embedded layers to have overall characterization of individual features.

$$y = \sum w_i x_i$$

- 2nd order shared embedded layers for both deep and FM parts, from which dot product between pairs of embedded features address the 2nd order feature interactions.

$$y = \sum w_{i,j} x_i x_j$$

1st order factorization machines (summation of all 1st order embed layers)

- numeric features with shape (None, 1) => dense layer
- categorical features (single level) with shape (None,1) => embedding layer (latent_dim = 1)
- categorical features (multi level) with shape (None,L) => embedding layer (latent_dim = 1)
- output will summation of all embedded features

2nd order factorization machines (summation of dot product between 2nd order embed layers)

- numeric features => dense layer
- categorical features (single level) => embedding layer (latent_dim = k)
- categorical features (multi level) with shape (None,L) => embedding layer (latent_dim = k)
- shared embed layer will be the concatenated layers of all embedded features
- shared embed layer => dot layer => 2nd order of FM part

Deep part (DNN model on shared embed layers)

- shared embed layer => series of dense layers => deep part

א. לצורך בניית המודל שלנו נתייחס למאפיינים שונים עבור כל משתמש :

- *גיל*
- *מין – יש צורך לבצע* preprocessing *לצורך מעבר מ* "F" *ו* "M" *לייצוג בינארי.*

ומאפיין שונה עבור כל סרט:

- *ז'אנר - יש צורך לבצע* preprocessing *כך שעבור כל סרט יהיה רשימה של הז'אנרים ששייכים אליו, רשימה זאת תהיה באורך מס' הז'אנרים המקסימלי לסרט (במקום להחזיק עבור כל סרט ערכים בינאריים עבור כל ז'אנר). בתהליך ה* preprocessing *ניקח תחילה את הרשימה הבינארית של הזא'נרים עבור כל סרט ונהפוך אותה לרשימה של ז'אנרים בצורה הבאה:*

	movie_id	genres
0	0	[Animation, Children, Comedy]
1	0	[Animation, Children, Comedy]
2	0	[Animation, Children, Comedy]
3	0	[Animation, Children, Comedy]
4	0	[Animation, Children, Comedy]
...	...	...
79995	1678	[Romance, Thriller]
79996	1679	[Drama, Romance]
79997	906	[Comedy]
79998	1680	[Comedy]
79999	1681	[Drama]

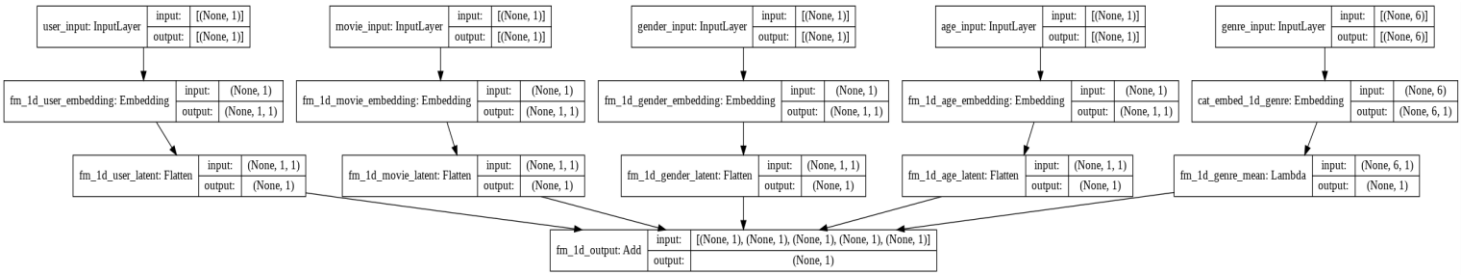
לאחר מכן כל ז'אנר ייוצג על ידי מספר וניצור רשימה של מספרים(ז'אנרים) כך שהרשימה תהיה באורך של מס' הז'אנרים המקסימלי לסרט . הייצוג ייראה בצורה הבאה:

	movie_id	genres
0	0	[15, 7, 2, 0, 0, 0]
1	1	[3, 6, 4, 0, 0, 0]
2	2	[4, 0, 0, 0, 0, 0]
3	3	[3, 2, 1, 0, 0, 0]
4	4	[8, 1, 4, 0, 0, 0]
...	...	...
1677	1677	[1, 0, 0, 0, 0, 0]
1678	1678	[5, 4, 0, 0, 0, 0]
1679	1679	[1, 5, 0, 0, 0, 0]
1680	1680	[2, 0, 0, 0, 0, 0]
1681	1681	[1, 0, 0, 0, 0, 0]

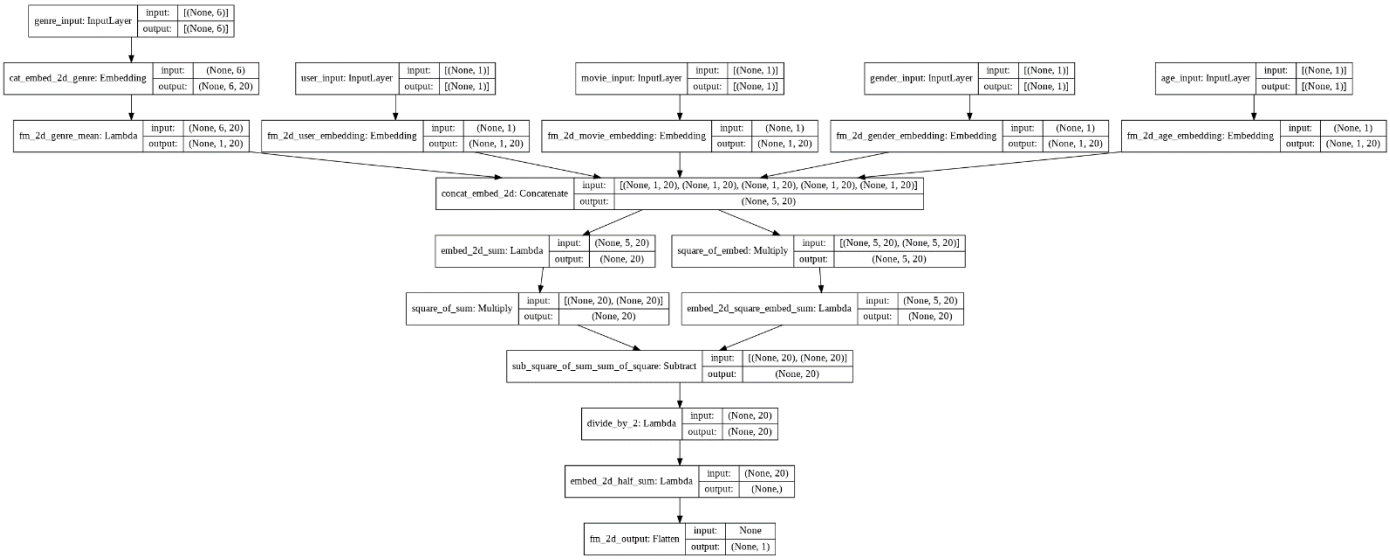
זהו בעצם הייצוג סופי של מאפיין הז'אנר עבור כל סרט.

המודל שבנינו מורכב מהחלקים שהצגנו בהנחה בתחילת התרגיל וניתן לראות זאת בצורה ויזואלית באמצעות הדפסות שעשינו :

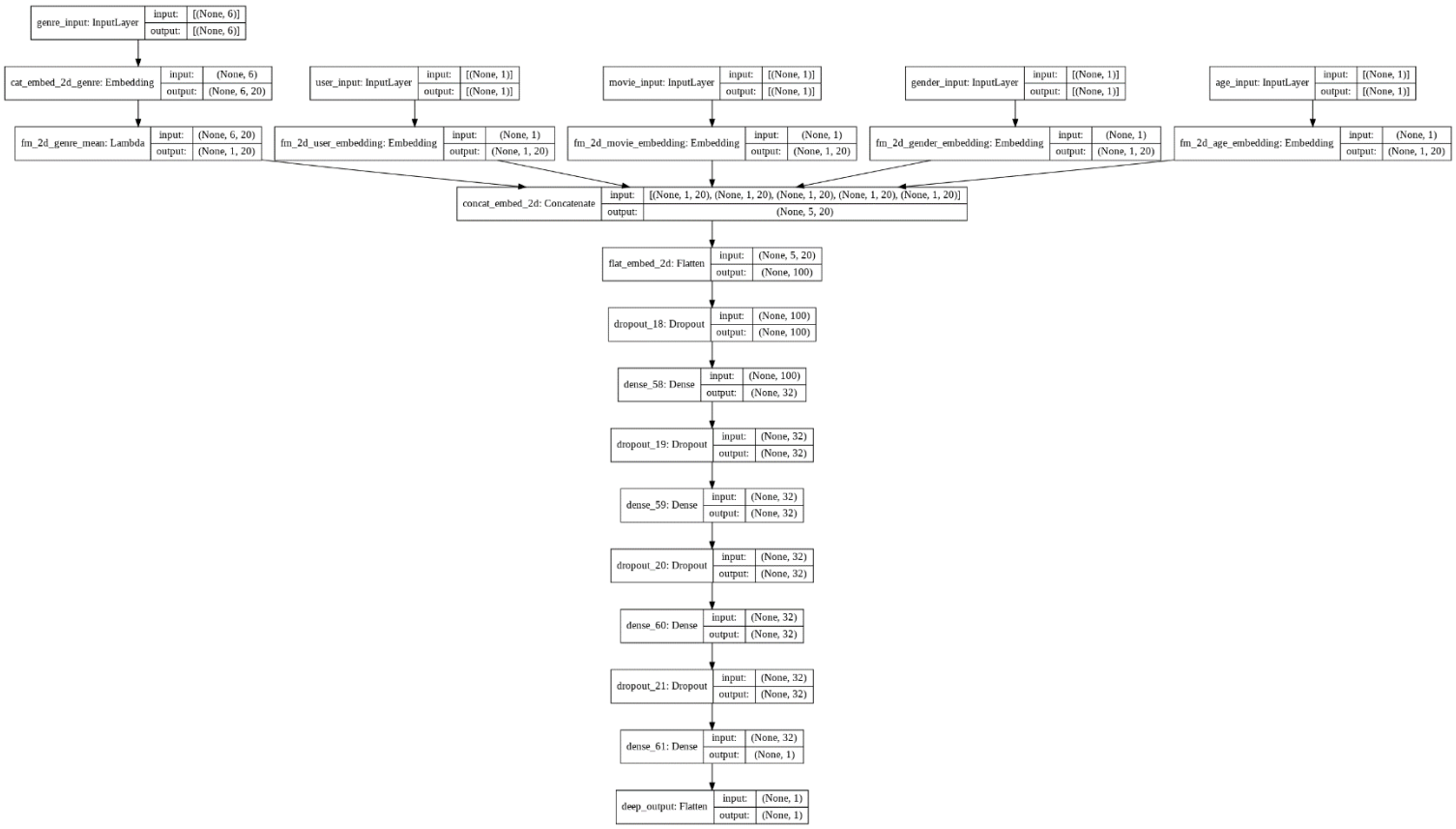
- *עבור 1st order factorization machines:*



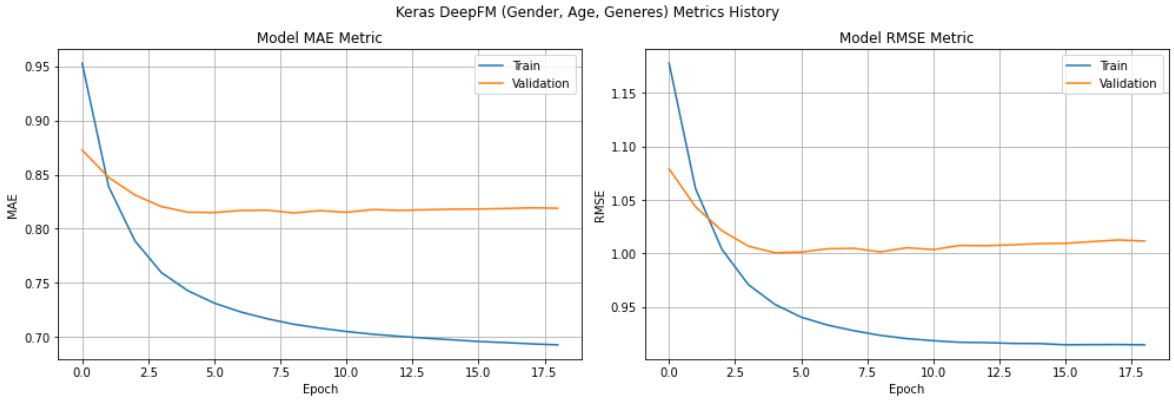
- עבור *2nd order factorization machines:*



- ועבור ה *Deep part:*



המודל הכולל לוקח את הoutput מכל component ומחבר את התוצאות לכדי חיזוי סופי.
לצורך מודל זה השתמשנו בספריית keras כאשר פונקציית ה loss שלנו היא MAE
והאופטימייזר מוגדר על learning rate = 0.0001 . התוצאות שקיבלנו עבור מודל זה :

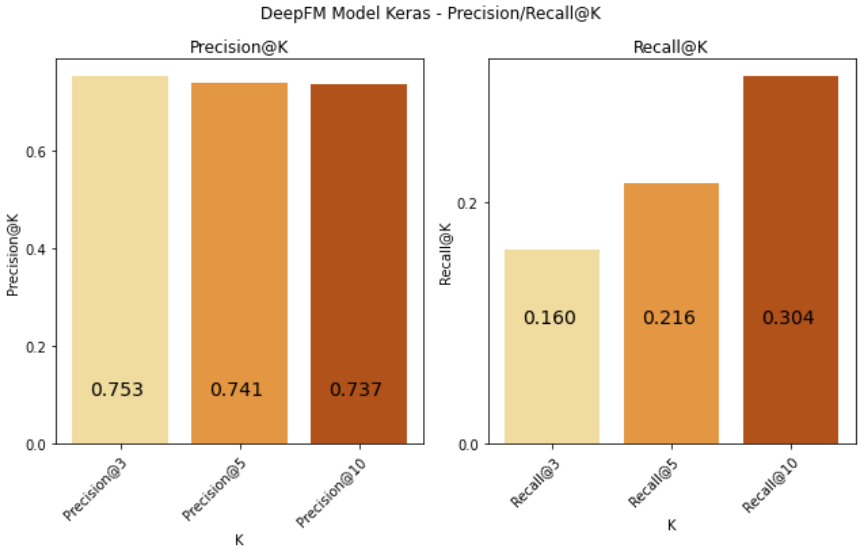


כך שערך ה RMSE\MAE הממוצע הוא :

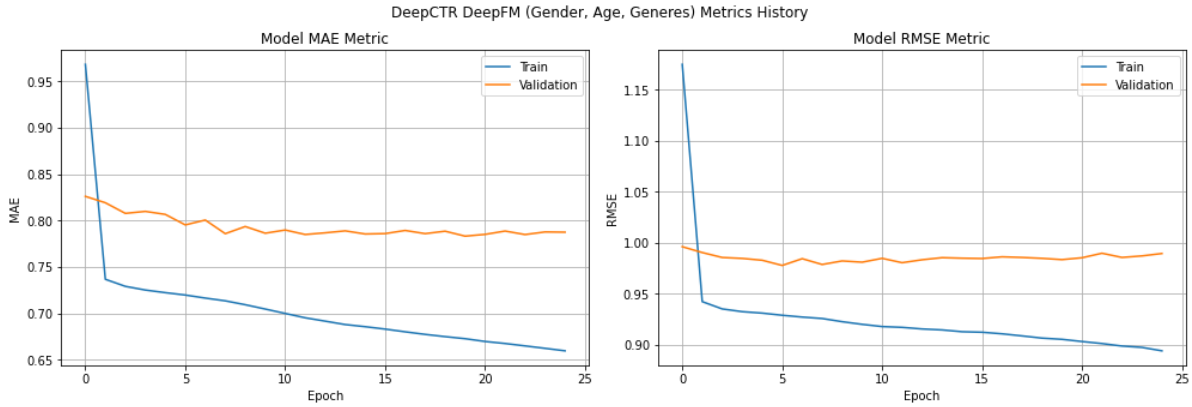
$$MAE: 0.7530499696731567$$

$$RMSE: 0.9630998969078064$$

ניתן לראות על פי התוצאות כי ערך ה MAE נמצא במגמת ירידה ככל שהאיטרציות עולות.
עבור חישוב precision/recall @k קיבלנו את הדיאגרמה הבאה:



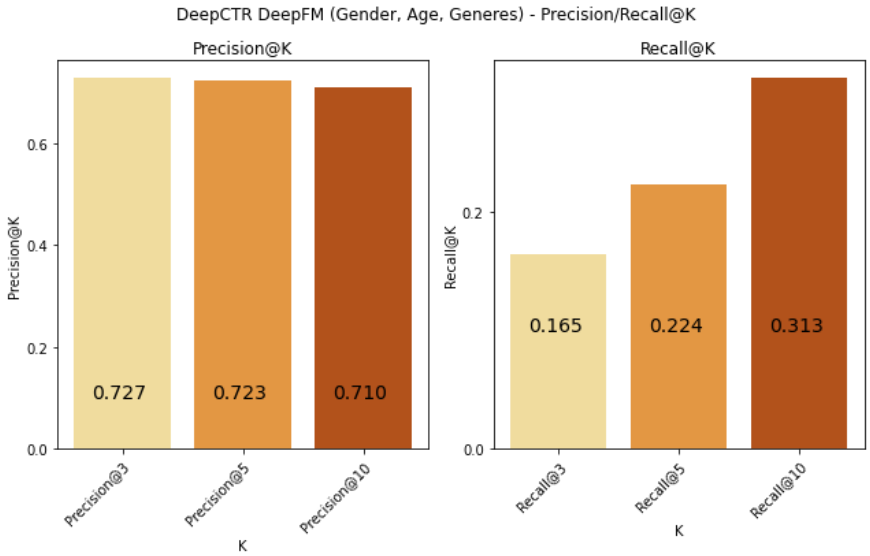
ב. כעת לצורך ניסיון שיפור של המודל הנ"ל, נשתמש בספריית DeepCTR שהוצגה בכיתה. המודל השני הוא מודל זהה למודל הנ"ל(בעל אותן תכונות) שמתייחס למאפייני גיל ומין עבור כל משתמש, ומתייחס למאפיין הז'אנר עבור כל סרט. כאשר הרצנו מודל זהה עם ספריית DeepCTR קיבלנו את התוצאות הבאות:



כך שערך ה RMSE\MAE הממוצע הוא :

$$MAE: 0.7297515869140625$$
$$RMSE: 0.954951345920562$$

כבר כאן ניתן לשים לב כי קיים שיפור משמעותי בתוצאות. על פי הגרף ניתן לראת כי ערך MAE\RMSE נמצא בירידה ככל שהאיטרציות עולות וערכי ה MAE וה RMSE הממוצעים במודל זה קטנים יותר מהערכים במודל הקודם ולכן קיבלנו שיפור. עם זאת, ניתן לראות כי ערך ה precision/recall @k שקיבלנו במודל זה הוא קטן יותר :

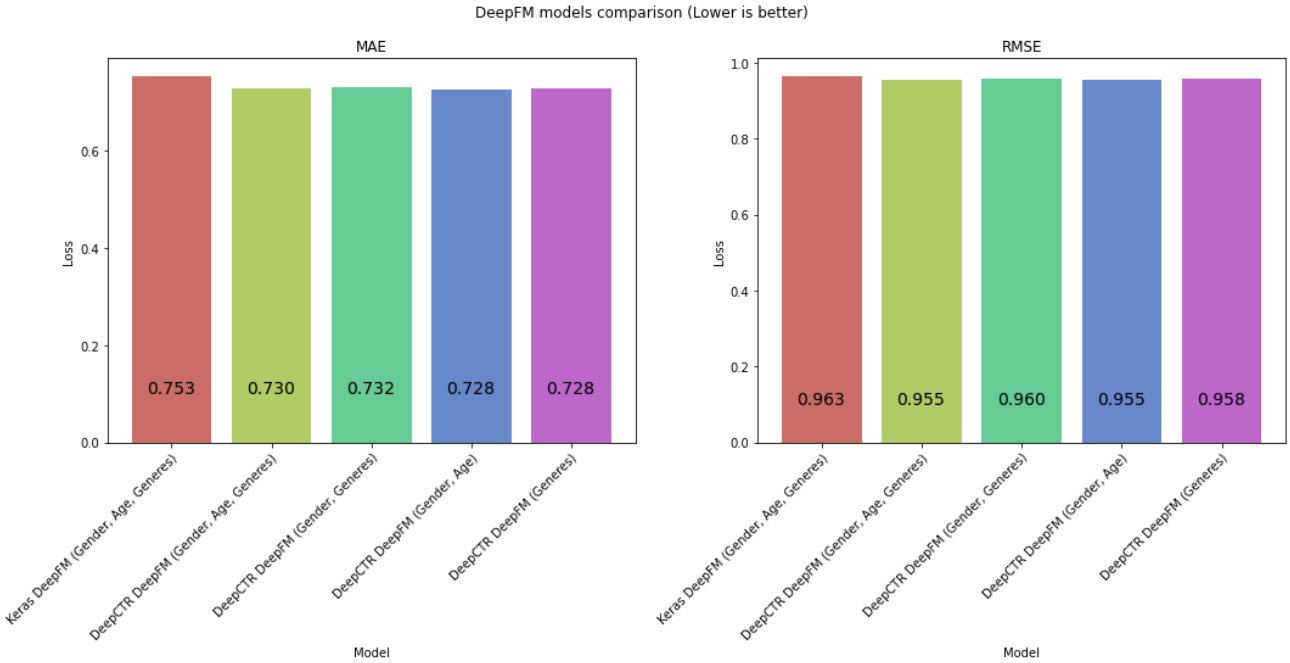


ג. בסעיף זה ניסינו ליצור מודלים שונים באמצעות קומבינציות שונות של המאפיינים בכדי לראות מה המודל המוצלח ביותר. המודלים השונים שיצרנו :

- מודל DeepCTR DeepFM עם התייחסות למין המשתמש וז'אנר הסרט
- מודל DeepCTR DeepFM עם התייחסות למין וגיל המשתמש
- מודל DeepCTR DeepFM עם התייחסות רק לז'אנר הסרט
- מודל DeepCTR DeepFM עם התייחסות למין המשתמש וז'אנר הסרט

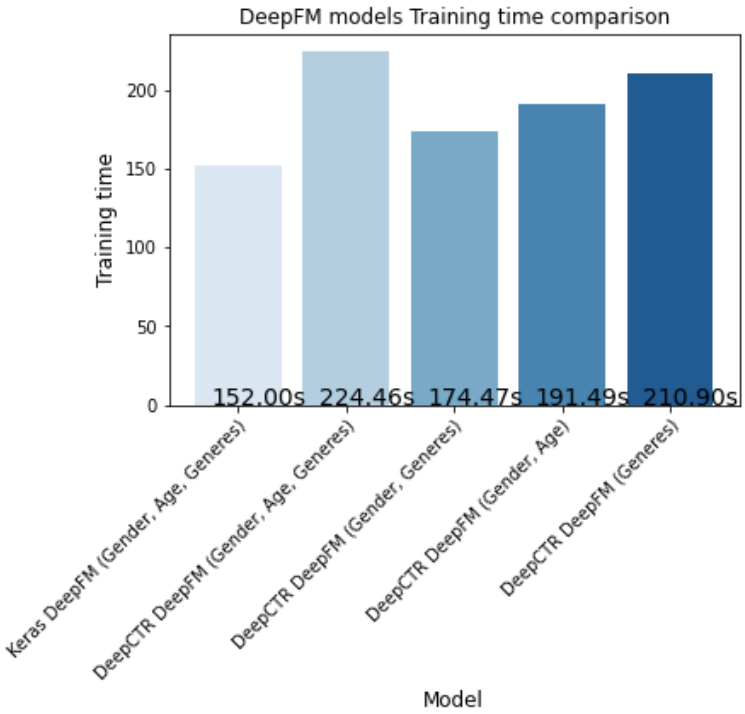
את ההשוואות הסופיות בין המודלים והצגת ה MAE הממוצע עבור כל מודל נציג בסעיף הבא.

ד. השוואת RMSE\MAE ממוצע עבור כל מודל שיצרנו בתרגיל זה:



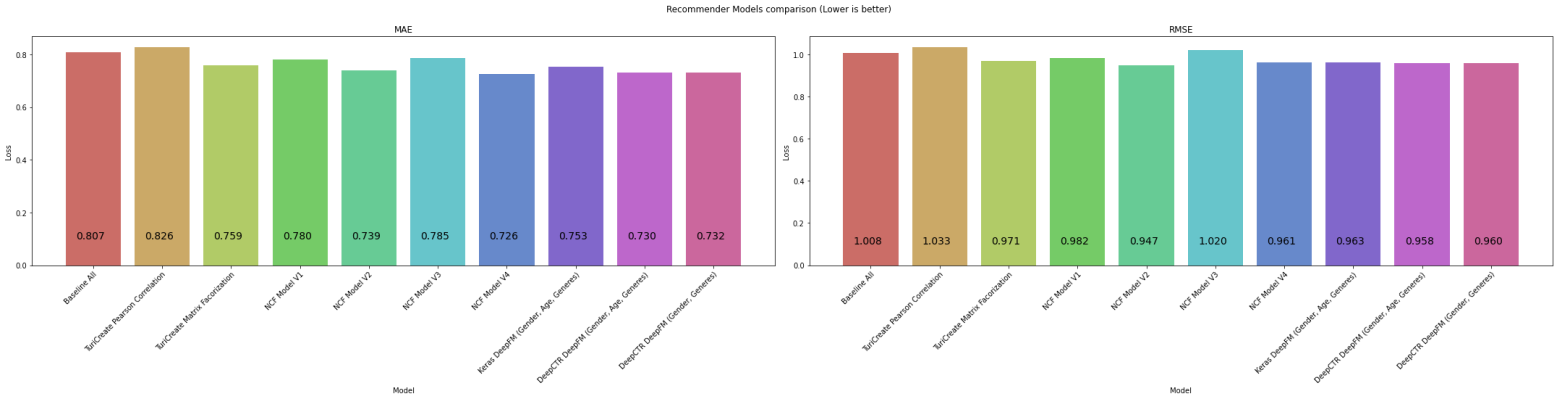
ניתן לראות בבירור שערך ה MAE הממוצע הטוב ביותר התקבל במודלים:
DeepCTR DeepFM עם התייחסות למין וגיל המשתמש
DeepCTR DeepFM עם התייחסות רק לז'אנר הסרט
ערך ה RMSE הממוצע הטוב ביותר התקבל במודלים:
DeepCTR DeepFM עם התייחסות למין וגיל המשתמש
DeepCTR DeepFM עם התייחסות למין וגיל המשתמש וז'אנר הסרט

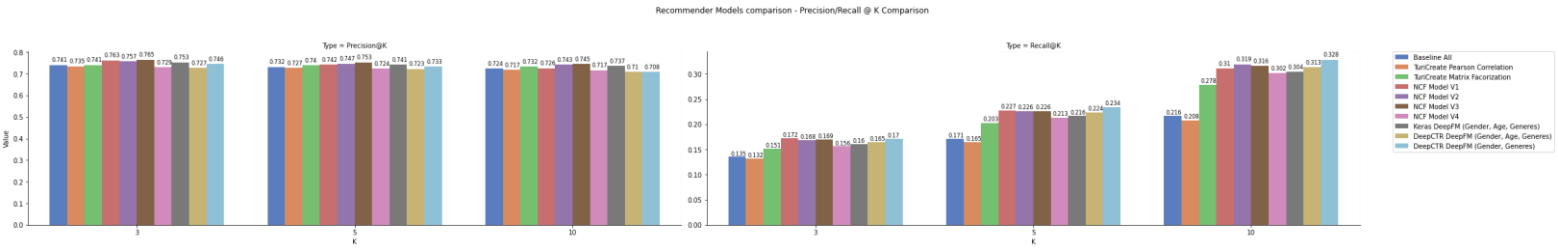
מבחינת השוואת זמן אימון קיבלנו את התוצאה הבאה:



וכאן ניתן לראות בבירור שהמודלים שמשתמשים בספרייה של DeepCTR הם בעלי יכולת למידה איטית יותר, אך לבסוף הם מוציאים תוצאות טובות יותר.

ה. לסיכום, ביצענו השוואה של כל המודלים שביצענו בפרוייקט זה, ההשוואה שעשינו מוצגת בצורה ויזואלית בדיאגרמות הבאות:





לדעתנו ההבדלים בין המודלים הם לא משמעותיים ולא בעלי פער עצום מכיוון שגודל ה dataset הוא די קטן, אם ניקח dataset גדול יותר עם יותר מ 100k רשומות, ההבדלים שנראה יהיו משמעותיים יותר ויותר.
סה"כ שמנו לב כי ע"י שימוש ב DeepFM , מקבלים תוצאות טובות עבור dataset גדול יותר ולכן היינו ממליצים על שימוש של DeepFM עם ספריית DeepCTR שמשלב את מאפיין הז'אנר עבור הסרטים ומאפיין המגדר עבור המשתמשים, ע"י שימוש במאפיינים אלו זיהינו בשיפור המשמעותי ביותר במודלים.