Steam game data set
Recommendation System

צבי פוצ׳נסקי

תוכן עניינים

פרק 1: הצגת נתונים ושלבי עיבוד מקדים

הצגת ה data set המקורי

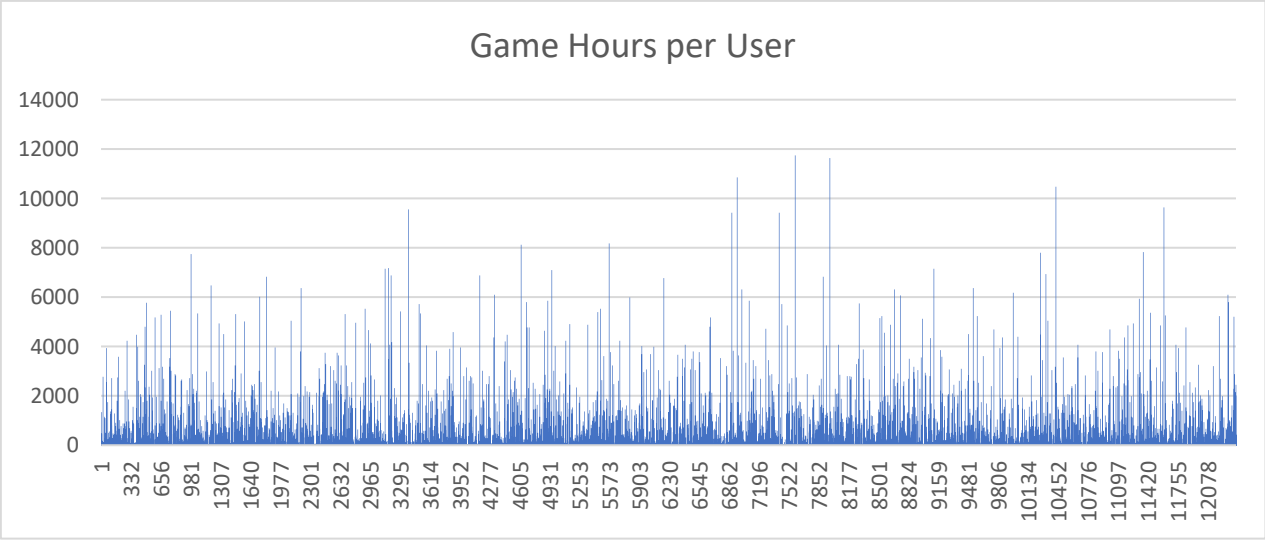
הפרויקט מתבסס על data set מקורי של מערכת המשחקים המפורסמת steam. לאחר בחירת ה data set הוצבה מטרה לבנות מערכת המלצה אשר מבצעת המלצה איכותית למשתמש על בסיס מספר פרמטרים שיפורטו בהמשך הדו"ח.

להלן ה data set המקורי[1]:
ה data set מורכב ממספר קבצי dat. אשר מספקים נתונים לגבי רכישות של משתמשים, זמני משחק ושמות המשחקים שנרכשו. כל הנתונים מקודדים במספרי קוד המייצגים את הנתונים בהתאם לאופי המידע הנדרש.

1. **game_play.dat**
   קובץ זה מכיל את קודי המשתמשים, קודי המשחק וזמני המשחק שהם שיחקו בכל אחד בהתאמה. (user_id, game_id, hours)

User_ID	Game_ID	Hours
1	1	273
1	2	87
1	3	14.9
1	4	12.1
1	5	8.9
1	6	8.5
1	7	8.1
1	8	7.5
1	9	3.3
1	10	2.8
1	11	2.5
1	12	2

Figure1 game play first rows



Total of 12,393 Users
Avg of 607.05 game play hours per user

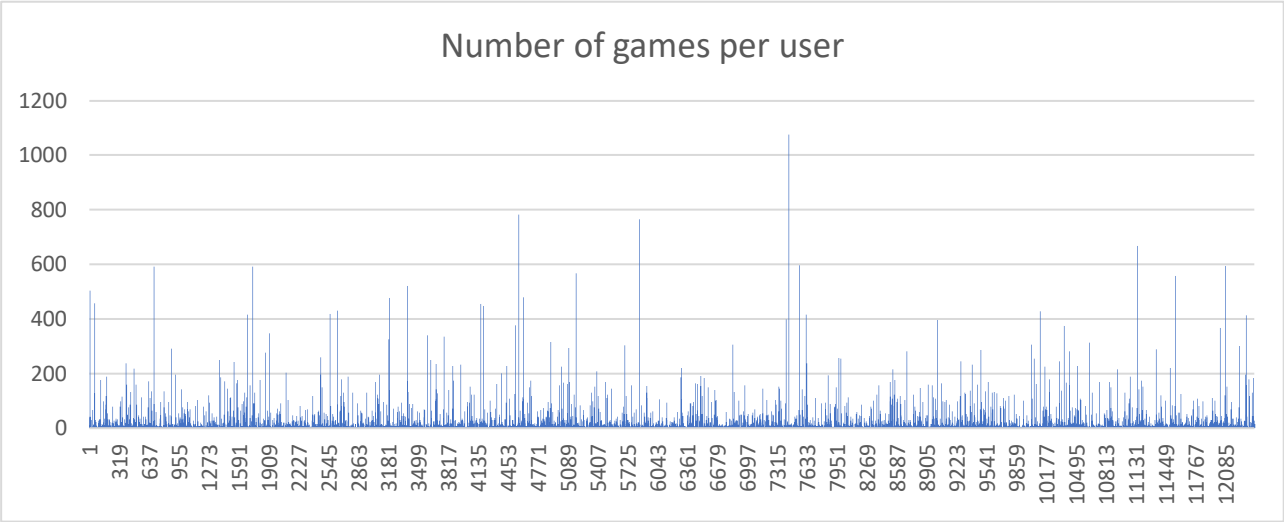
[1] https://github.com/caserec/Datasets-for-Recommender-Systems/tree/master/Processed%20Datasets/Steam

2. game_purchase.day

קובץ זה מכיל קודי משתמש, קודי משחק והאם נרכש. (user_id, game_id, purchase)

User_ID	Game_ID	Purchase
1	1	1
1	2	1
1	3	1
1	4	1
1	5	1
1	6	1
1	7	1
1	8	1
1	9	1
1	10	1
1	11	1
1	12	1
1	13	1
1	14	1
1	15	1

Figure2 game purchase first rows



Total of 12,393 Users
Avg of 20.89 game per user

3. item_info.dat

קובץ זה מכיל את הפירוש של הקודים של המשחקים כלומר לכל קוד מותאם השם של המשחק שאותו הוא מייצג.

Game_ID	Game Name
1	The Elder Scrolls V Skyrim
2	Fallout 4
3	Spore
4	Fallout New Vegas
5	Left 4 Dead 2
6	HuniePop
7	Path of Exile
8	Poly Bridge
9	Left 4 Dead
10	Team Fortress 2
11	Tomb Raider
12	The Banner Saga

Figure3 item info first rows

Total of 5155 games

ניתוח נתונים של ה data set המקורי

לאחר בחינת הנתונים כמו שהוצגו על מנת לבצע ניתוח נדרש היה להוסיף עוד פרמטרים לתיאור המשחקים שנקנו על מנת שנוכל לאפיין את המשחקים והרכישות, על בסיס מאפיינים (features) אלו נוכל לתת המלצות וביצוע ניתוח לנתונים.

על מנת להרחיב את ה data set הנתון ולאחר ביצוע חקר ובחינת אפשרויות נמצאה גישת API לאתר משחקים www.igdb.com הנותן אפשרות לבצע בקשות POST וקבלת מידע רלוונטי על בסיס מפרט ה API שהם מספקים. [קישור ל API doc]
האתר מאפשר ביצוע עד 50000 בקשות ומחזיר תשובות בפורמט json .

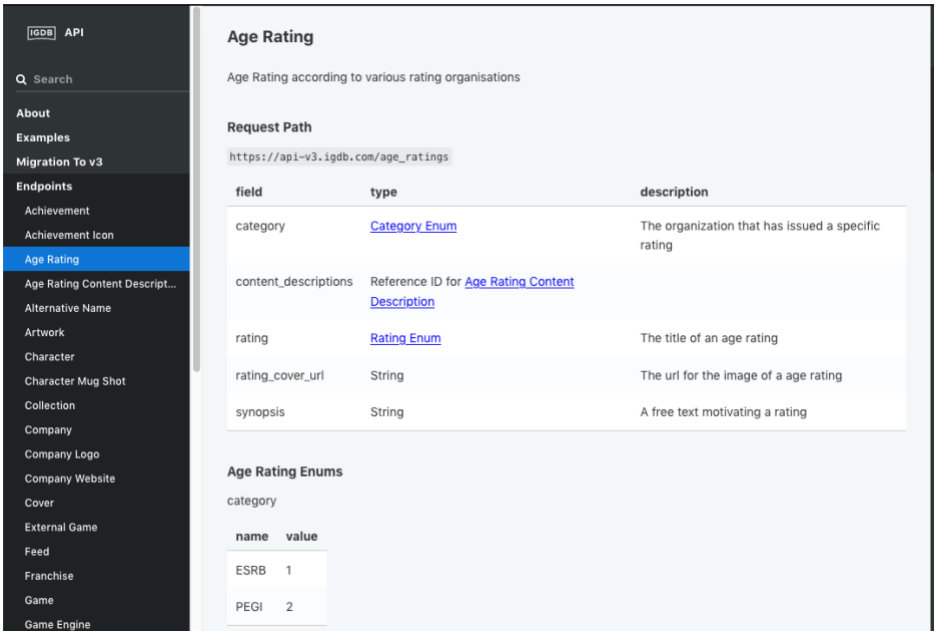


Figure4 IGDB API doc

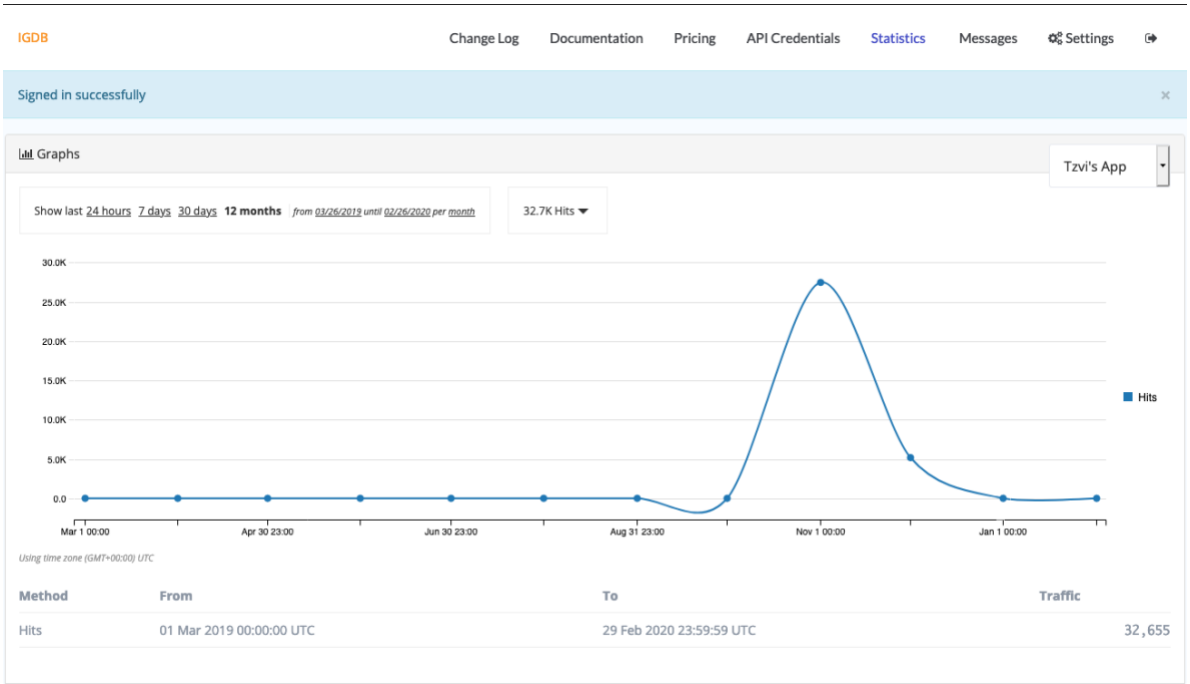


Figure5 Request log to the API

האתגר בשימוש בגישות ל API שכל הנתונים של האתר גם כן מקודדים בקודים, כלומר שבכל שדה שאנחנו רוצים להמיר את התשובה שלו למלל ממשי נדרש לבצע עוד בקשה לאתר על מנת למצוא את הפירוש המתאים לקוד המוצג.

הרחבת ה data set בוצע במספר שלבים:

1. ביצוע בקשות לאתר על בסיס כל השמות של המשחקים המופיעים ב data set המקורי בקובץ item_info.dat. כל בקשה נשלחה על בסיס השם של המשחק והתשובה נרשמה בקובץ טקסט להמשך עיבוד והבנת מבנה האתר והתוצאות שהוא נותן.

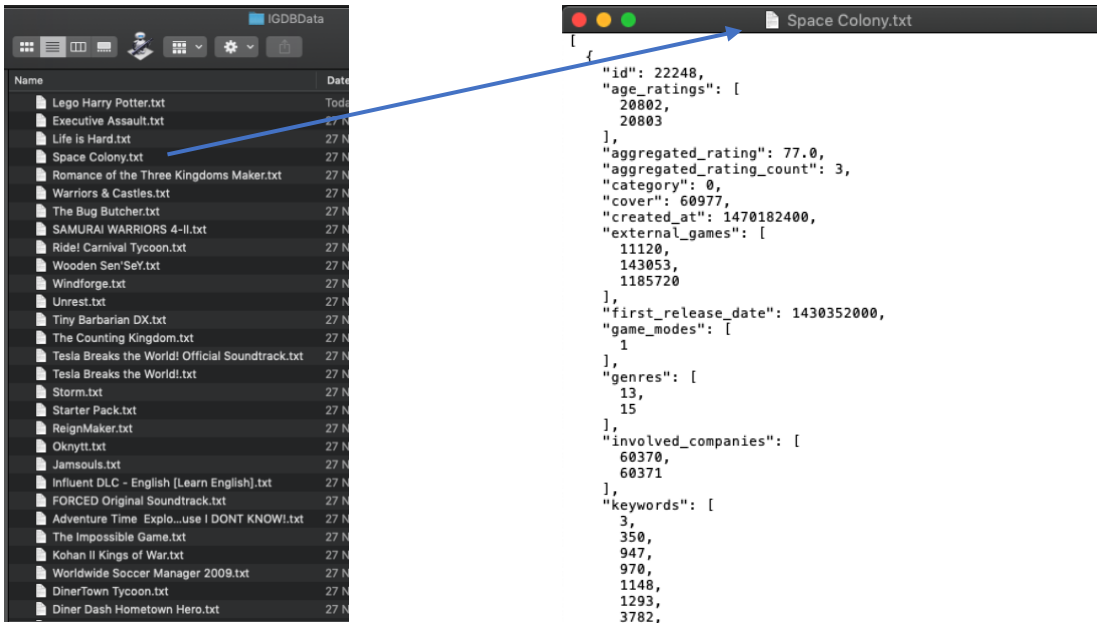


Figure6 API results example

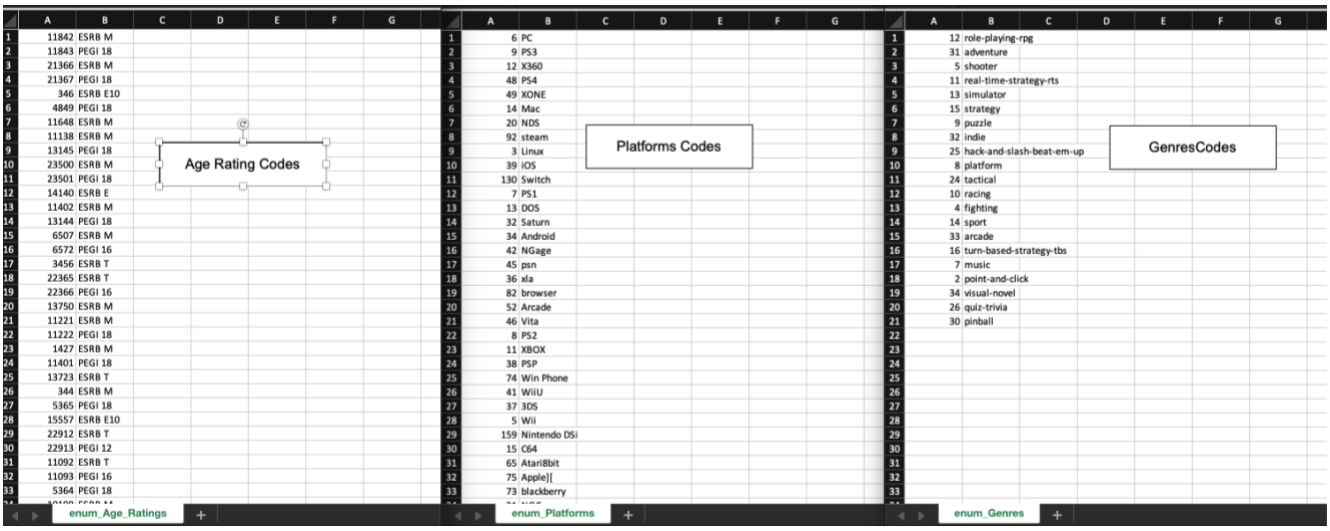
2. לאחר בחינת הקבצים והתוצאות המתקבלות מהאתר נבחרו מספר מאפיינים שניתן להתבסס עליהם על מנת לבנות מודל ולבצע המלצות.
המאפיינים שנבחרו:
   - first release date
   - genres
   - age rating
   - aggregated rating
   - rating
   - platforms
   - game modes
   - involved companies

כל התוצאות רוכזו לקובץ csv אחד לביצוע עיבוד נוסף והמרת הקודים שהתקבלו למלל.

steam_id	idgb_id	game_name	first_release_date	genres	age_ratings	aggregated_rating	rating	platforms	game_modes	involved_companies
1	472	The Elder Scr	1320969600	[12, 31]	[11842, 11843]	96.11111111	88.31594931	[6, 9, 12]	[1]	[3717, 3719]
2	9630	Fallout 4	1447113600	[5, 12]	[21366, 21367]	84.66666667	80.87232675	[6, 48, 49]	[1]	[24119, 24120]
3	1876	Spore	1220486400	[11, 13, 15, 31]	[346]	82	72.56041231	[6, 14, 20]	[1]	[5390, 5391]
4	16	Fallout New	1287446400	[5, 12]	[4849, 11648]	81.9	87.10116217	[6, 9, 12, 92]	[1]	[3714, 3715, 3716, 13512]

Figure7 Game extended data set before processing

3. ביצוע המרה של כל השדות למלל רלוונטי על מנת שיהיה ערך מובן למשתמש ולצורך תצוגה בממשק משתמש שנבנה בהמשך.
ההמרה בוצעה על ידי פונקציות אוטומטיות אשר פונות ל endpoints הרלוונטיות לכל שדה ומבצעות בקשה לאתר וניתוח התשובה המתקבלת ושמירת הקשר בין הקוד למלל בקובץ לשימוש חוזר (על מנת לחסוך בקשות נוספות לאתר).
כתוצאה מפונקציות נוצרו מספר קבצים המכילים את הקשרים על מנת לבצע את ההמשך.



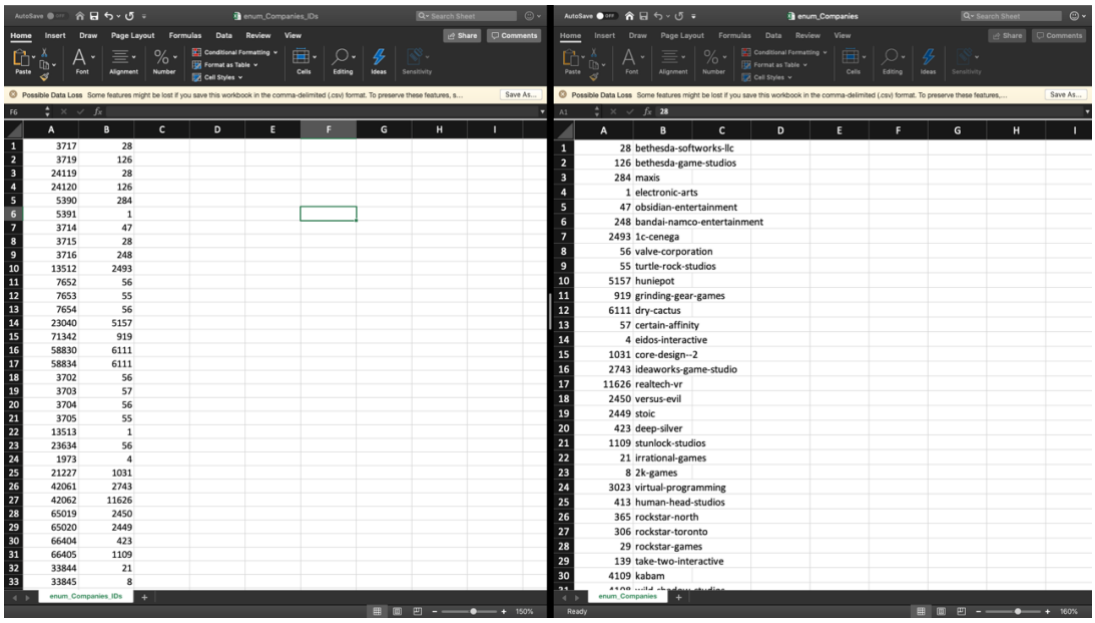


Figure8 Code to word files

4. לאחר ביצוע כל ההמרות ושמירת הנתונים בקבצים, בוצעה המרה לטבלה המקורית כך שנוכל לראות את כל המלל בצורה ברורה.
הערה: כאשר בוצעו הבקשות, היו מקרים שלא היה מידע באתר וחוסר במידע סומן כ -1 .

steam_id	idgb_id	game_name	first_release_date	genres	age_ratings	aggregated_rati	rating	platforms	game_modes	involved_companies
1	472	The Elder Sci	11/11/2011 2:00	['role-playing-rpg', 'adventure']	['ESRB M', 'PEGI 18']	96.11111111	88.3159493	['PC', 'PS3', 'X360']	['single player']	{'bethesda-softworks-llc', 'bethesda-game-studios'}
2	9630	Fallout 4	10/11/2015 2:00	['shooter', 'role-playing-rpg']	['ESRB M', 'PEGI 18']	84.66666667	80.8723268	['PC', 'PS4', 'XONE']	['single player']	{'bethesda-softworks-llc', 'bethesda-game-studios'}
3	1876	Spore	04/09/2008 3:00	['real-time-strategy-rts', 'simulator', 'strategy', 'adv	['ESRB E10']	82	72.5604123	['PC', 'Mac', 'NDS']	['single player']	{'maxis', 'electronic-arts'}
4	16	Fallout New	19/10/2010 2:00	['shooter', 'role-playing-rpg']	['PEGI 18', 'ESRB M']	81.9	87.1011622	['PC', 'PS3', 'X360', 'steam']	['single player']	{'obsidian-entertainment', '1c-cenega', 'bethesda-softworks-llc', 'bandai-namco-entertainment'}
5	124	Left 4 Dead 2	17/11/2009 2:00	['shooter']	['ESRB M', 'PEGI 18']	88.6	82.6931335	['Linux', 'PC', 'X360', 'Mac']	['single player', 'multiplayer',	{'valve-corporation', 'turtle-rock-studios'}
6	9655	HuniePop	19/01/2015 2:00	['puzzle', 'role-playing-rpg', 'simulator', 'strategy', 'i	-1	82.5	82.2170759	['Linux', 'PC', 'Mac']	['single player']	{'huniepot'}
7	1911	Path of Exile	23/10/2013 3:00	['role-playing-rpg', 'hack-and-slash-beat-em-up', 'ac	['ESRB M', 'PEGI 18']	78	81.2744864	['PC', 'PS4', 'XONE']	['single player', 'multiplayer',	{'grinding-gear-games'}
8	11597	Poly Bridge	12/07/2016 3:00	['puzzle', 'simulator', 'indie']	['ESRB E']	75	69.8514761	['Linux', 'PC', 'Mac', 'iOS', 'steam	['single player']	{'dry-cactus'}

Figure9 Game extended data set using code to word files

5. Feature selection – בחירת העמודות הטובות ביותר על מנת לבצע את הניתוח בהמשך.
בגלל שחלק מהתשובות היו חסרות וסומנו בתור 1- נדרש היה לבצע 'ניקיון' לטבלה כך שלא יהיו תאים ללא מידע.

לצורך כך נכתבה פונקציה אשר מבטלת שורות בהתאם למספר התאים הריקים שיש באותה שורה. אחד הפרמטרים של הפונקציה הוא מספר מקסימלי של תאים 'ריקים' – מכילים 1-, כלומר בכל הרצה של הפונקציה בוצע מעבר על הטבלה ומחיקת שורות בהתאם לפרמטר.
לדוגמא: כאשר הפרמטר עמד על 2 – כל שורה שמכילה לפחות 2 תאים 'ריקים' נמחקה מהמאגר במעבר הנוכחי.
בסיום ביצוע הפונקציה בוצע מעבר נוסף על הטבלה והפקת דוח לכל עמודה וכמות התאים הריקים הנשארים בטבלה.

```
Pre-process for 1
Number of missing values in current data set:
steam_id : 0/1187 | Percentage : 0.00%
idgb_id : 0/1187 | Percentage : 0.00%
game_name : 0/1187 | Percentage : 0.00%
first_release_date : 0/1187 | Percentage : 0.00%
genres : 0/1187 | Percentage : 0.00%
age_ratings : 0/1187 | Percentage : 0.00%
aggregated_rating : 0/1187 | Percentage : 0.00%
rating : 0/1187 | Percentage : 0.00%
platforms : 0/1187 | Percentage : 0.00%
game_modes : 0/1187 | Percentage : 0.00%
involved_companies : 0/1187 | Percentage : 0.00%
==============================================
Pre-process for 2
Number of missing values in current data set:
steam_id : 0/1989 | Percentage : 0.00%
idgb_id : 0/1989 | Percentage : 0.00%
game_name : 0/1989 | Percentage : 0.00%
first_release_date : 0/1989 | Percentage : 0.00%
genres : 0/1989 | Percentage : 0.00%
age_ratings : 430/1989 | Percentage : 21.62%
aggregated_rating : 342/1989 | Percentage : 17.19%
rating : 22/1989 | Percentage : 1.11%
platforms : 0/1989 | Percentage : 0.00%
game_modes : 3/1989 | Percentage : 0.15%
involved_companies : 5/1989 | Percentage : 0.25%
==============================================
Pre-process for 3
Number of missing values in current data set:
steam_id : 0/3089 | Percentage : 0.00%
idgb_id : 0/3089 | Percentage : 0.00%
game_name : 0/3089 | Percentage : 0.00%
first_release_date : 0/3089 | Percentage : 0.00%
genres : 0/3089 | Percentage : 0.00%
age_ratings : 1507/3089 | Percentage : 48.79%
aggregated_rating : 1402/3089 | Percentage : 45.39%
rating : 62/3089 | Percentage : 2.01%
platforms : 0/3089 | Percentage : 0.00%
game_modes : 6/3089 | Percentage : 0.19%
involved_companies : 25/3089 | Percentage : 0.81%
==============================================
Pre-process for 4
Number of missing values in current data set:
steam_id : 0/3495 | Percentage : 0.00%
idgb_id : 0/3495 | Percentage : 0.00%
game_name : 0/3495 | Percentage : 0.00%
first_release_date : 1/3495 | Percentage : 0.03%
genres : 22/3495 | Percentage : 0.63%
age_ratings : 1905/3495 | Percentage : 54.51%
aggregated_rating : 1795/3495 | Percentage : 51.36%
rating : 285/3495 | Percentage : 8.15%
platforms : 0/3495 | Percentage : 0.00%
game_modes : 15/3495 | Percentage : 0.43%
involved_companies : 197/3495 | Percentage : 5.64%
==============================================
```

```
==============================================
Pre-process for 5
Number of missing values in current data set:
steam_id : 0/3641 | Percentage : 0.00%
idgb_id : 0/3641 | Percentage : 0.00%
game_name : 0/3641 | Percentage : 0.00%
first_release_date : 15/3641 | Percentage : 0.41%
genres : 35/3641 | Percentage : 0.96%
age_ratings : 2048/3641 | Percentage : 56.25%
aggregated_rating : 1936/3641 | Percentage : 53.17%
rating : 408/3641 | Percentage : 11.21%
platforms : 13/3641 | Percentage : 0.36%
game_modes : 36/3641 | Percentage : 0.99%
involved_companies : 313/3641 | Percentage : 8.60%
==============================================
Pre-process for 6
Number of missing values in current data set:
steam_id : 0/3694 | Percentage : 0.00%
idgb_id : 0/3694 | Percentage : 0.00%
game_name : 0/3694 | Percentage : 0.00%
first_release_date : 33/3694 | Percentage : 0.89%
genres : 52/3694 | Percentage : 1.41%
age_ratings : 2099/3694 | Percentage : 56.82%
aggregated_rating : 1985/3694 | Percentage : 53.74%
rating : 452/3694 | Percentage : 12.24%
platforms : 31/3694 | Percentage : 0.84%
game_modes : 68/3694 | Percentage : 1.84%
involved_companies : 349/3694 | Percentage : 9.45%
==============================================
Pre-process for 7
Number of missing values in current data set:
steam_id : 0/3731 | Percentage : 0.00%
idgb_id : 0/3731 | Percentage : 0.00%
game_name : 0/3731 | Percentage : 0.00%
first_release_date : 59/3731 | Percentage : 1.58%
genres : 69/3731 | Percentage : 1.85%
age_ratings : 2132/3731 | Percentage : 57.14%
aggregated_rating : 2020/3731 | Percentage : 54.14%
rating : 488/3731 | Percentage : 13.08%
platforms : 57/3731 | Percentage : 1.53%
game_modes : 86/3731 | Percentage : 2.31%
involved_companies : 380/3731 | Percentage : 10.18%
==============================================
```

Figure10 Data summary output

סיכום התוצאות בטבלה לצורך הסקת מסקנות

Parameter	1			2			3		
	missing lines	total	percentage	missing lines	total	percentage	missing lines	total	percentage
steam id	0	1187	0.00%	0	1989	0.00%	0	3089	0.00%
idgb id	0	1187	0.00%	0	1989	0.00%	0	3089	0.00%
game name	0	1187	0.00%	0	1989	0.00%	0	3089	0.00%
firs release date	0	1187	0.00%	0	1989	0.00%	0	3089	0.00%
genres	0	1187	0.00%	0	1989	0.00%	0	3089	0.00%
age ratings	0	1187	0.00%	430	1989	21.62%	1507	3089	48.79%
aggregated rating	0	1187	0.00%	342	1989	17.19%	1402	3089	45.39%
rating	0	1187	0.00%	22	1989	1.11%	62	3089	2.01%
platforms	0	1187	0.00%	0	1989	0.00%	0	3089	0.00%
game modes	0	1187	0.00%	3	1989	0.15%	6	3089	0.19%
involved companies	0	1187	0.00%	5	1989	0.25%	25	3089	0.81%

Parameter	4			5			6			7		
	missing lines	total	percentage	missing lines	total	percentage	missing lines	total	percentage	missing lines	total	percentage
steam id	0	3495	0.00%	0	3641	0.00%	0	3694	0.00%	0	3731	0.00%
idgb id	0	3495	0.00%	0	3641	0.00%	0	3694	0.00%	0	3731	0.00%
game name	0	3495	0.00%	0	3641	0.00%	0	3694	0.00%	0	3731	0.00%
firs release date	1	3495	0.03%	15	3641	0.41%	33	3694	0.89%	59	3731	1.58%
genres	22	3495	0.63%	35	3641	0.96%	52	3694	1.41%	69	3731	1.85%
age ratings	1905	3495	54.51%	2048	3641	56.25%	2099	3694	56.82%	2132	3731	57.14%
aggregated rating	1795	3495	51.36%	1936	3641	53.17%	1985	3694	53.74%	2020	3731	54.14%
rating	285	3495	8.15%	408	3641	11.21%	452	3694	12.24%	488	3731	13.08%
platforms	0	3495	0.00%	13	3641	0.36%	31	3694	0.84%	57	3731	1.53%
game modes	15	3495	0.43%	36	3641	0.99%	68	3694	1.84%	86	3731	2.31%
involved companies	197	3495	5.64%	313	3641	8.60%	349	3694	9.45%	380	3731	10.18%

תצוגה גרפית של התוצאות

לאחר ביצוע הורדות שורות לפי הפרמטר בכל הרצה סך כל השורות גם קטן וה- data set קטן לכן היה נדרש לבחון את הנתונים כך שנוכל לצמצם את מספר השורות שבהן יש תאים ריקים אך עדיין לשמור על נפח מידע מקסימלי של הנתונים.
לאחר בחינת הטבלה והנתונים ניתן לזהות כי שני מאפיינים (המסומנים בצבע בטבלה):

- Age ratings
- Aggregated rating

לאורך כל הבדיקות ללא תלות בפרמטר ההורדה שומרים על כמות גדולה של חוסר במידע ולכן בביצוע ניסיון של ביטול שני המאפיינים האלו התקבלו תוצאות טובות כפי שניתן לראות בגרף.
בגרף קל לראות כי כאשר הפרמטר עומד על מינימום של 3 תאים ריקים בשורה מתקבל סך שורות גדול (מעל 3000) וערך השורות חסרות נמוך יחסית לפרמטרים הבאים.
על מנת לחסוך בזמן הוחלט להשתמש בפרמטר 3 ומילוי השורות החסרות בצורה ידנית כך שהתקבל data set שלם וללא חוסרים תוך כדי שמירה על נפח data set.

6. תוצאה סופית – data set מורחב למשחקים שבמאגר

התקבל data set המכיל 3089 משחקים עם מאפיינים נוספים הבאים:

- First release date - הוחלט לתת ייחוס רק לשנת יציאת המשחק.
- Genres – shooter, role playing RPG, adventure, etc.
- Rating - בוצע 'bining' לציונים בטווח של 1-5
- Platforms – PC, PS3, Linux, MAC etc.
- Game modes – single player, multiplayer, MMO etc.
- Involved companies – Bethesda softworks llc, valve corporation, electronic arts, etc.

steam_id	idgb_id	game_name	first_release_date	genres	rating	platforms	game_modes	involved_companies
1	472	The Elder Sc	2011	['role-playing-rpg', 'adventure']	5	['PC', 'PS3', '	['single player']	{'bethesda-softworks-llc', 'bethesda-game-studios'}
2	9630	Fallout 4	2015	['shooter', 'role-playing-rpg']	5	['PC', 'PS4', '	['single player']	{'bethesda-softworks-llc', 'bethesda-game-studios'}
3	1876	Spore	2008	['real-time-strategy-rts', 'simulator',	4	['PC', 'Mac',	['single player']	{'maxis', 'electronic-arts'}
4	16	Fallout New	2010	['shooter', 'role-playing-rpg']	5	['PC', 'PS3', '	['single player']	{'obsidian-entertainment', '1c-cenega', 'bethesda-sof
5	124	Left 4 Dead 2	2009	['shooter']	5	['Linux', 'PC',	['single player', 'multiplayer'	{'valve-corporation', 'turtle-rock-studios'}
6	9655	HuniePop	2015	['puzzle', 'role-playing-rpg', 'simulat	5	['Linux', 'PC',	['single player']	{'huniepot'}

Figure11 Final version of game extended data set

Content data set

על מנת לספק עוד מאפיין שניתן להתבסס עליו בסיווג משחקים מלבד המידע הכללי עליהם שהוצג קודם לכן, נוסף גם data set המכיל בתוכו כל משחק והתיאור המילולי שלו כ data set נפרד על מנת לבצע המלצה על בסיס תוכן – content recommendation.
ה data set גם נבנה על בסיס בקשות לאתר שהוזכר קודם לכן.

game_name	summary
The Elder Scrolls V Skyrim	The next chapter in the highly anticipated Elder Scrolls saga arrives from the makers of the 2006 and 2008 Games of the Year, Bethesda Game Studios. Skyrim reimagines and revolutionizes the open-world fantasy
Fallout 4	Bethesda Game Studios, the award-winning creators of Fallout 3 and The Elder Scrolls V: Skyrim, welcome you to the world of Fallout 4 ,Äì their most ambitious game ever, and the next generation of open-world gaming.
Fallout New Vegas	In this first-person Western RPG, the player takes on the role of Courier 6, barely surviving after being robbed of their cargo, shot and put into a shallow grave by a New Vegas mob boss. The Courier sets out to track down t
Left 4 Dead 2	Left 4 Dead 2 is a cooperative first-person shooter video game, the sequel to Valve Corporation's Left 4 Dead. The Game builds upon cooperatively focused gameplay and Valve's proprietary Source engine, the same game e
HuniePop	HuniePop is a 2015 adult dating sim/match-3 puzzle game published and developed by HuniePot. Funding for the game was raised via Kickstarter. It is available in two versions, one censored and one uncensored, although t
Path of Exile	The story component of Path of Exile's free 4.0 update, currently planned for release in December of 2020. The update will add a second campaign option to the game with 7 planned acts at launch.
Poly Bridge	Unleash your engineering creativity with an engaging and fresh bridge-building simulator with all the bells and whistles.

Figure12 content data set first rows

פרק 2: הסבר על המערכת

ארכיטקטורה כללית

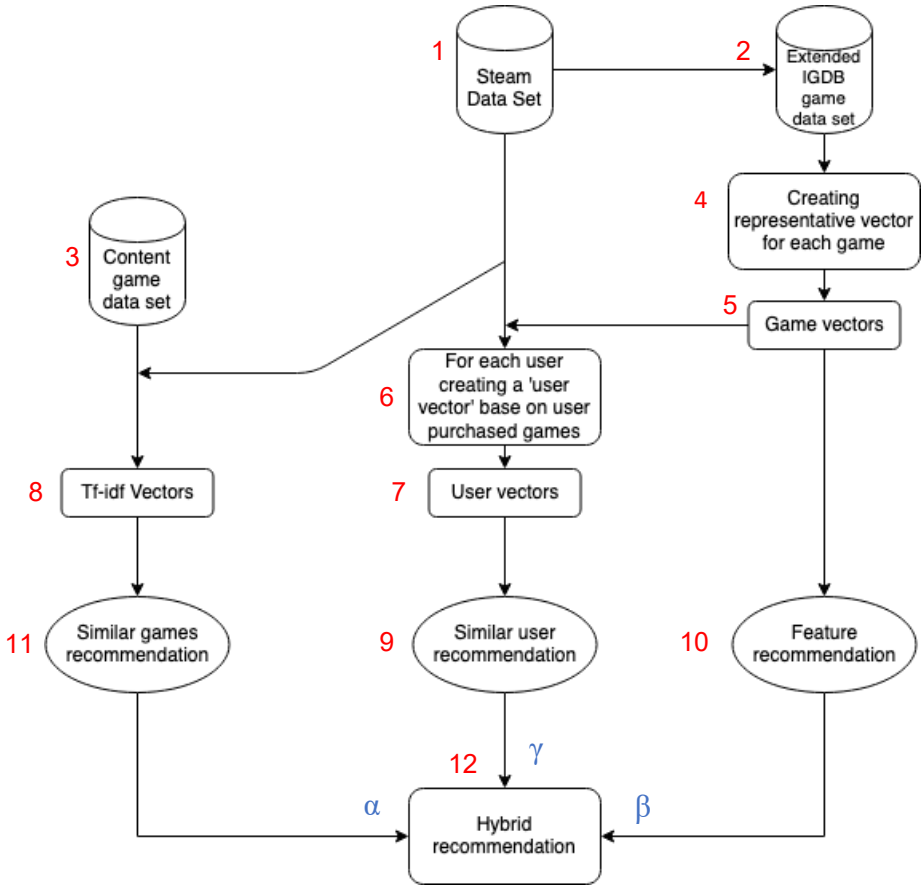


Figure13 Basic system architecture

1. **Steam data set –** הנתונים המקוריים כמו שהוצגו בפרק הקודם.
2. **Extended IGDB game data set –** מאגר המידע המורחב אחרי סינון המאפיינים כמו שהוצג בפרק הקודם.
3. **Content game data set –** מאגר המידע המכיל תיאור מילולי לכל משחק במאגר כמו שהוצג בפרק הקודם.
4. **Creating representative vector for each game –** חישוב ויצירת וקטור מייצג לכל משחק.
5. **Game vectors –** בשלב העיבוד המקדים נוצר לכל משחק ווקטור מייצג (המבנה מפורט בהמשך).
6. **For each user creating a 'user vector' based on user purchased games –** לכל משתמש נוצר אובייקט המייצג בתוכו וקטור יחיד המייצג אותו (אופן הייצוג והחישוב מפורט בהמשך).
7. **User vectors –** קובץ השומר בתוכו את כל הוקטורים המייצגים של המשתמשים במאגר.
8. **Tf-idf vectors –** יצירת וקטורים מייצגים לתקצירי המשחק על בסיס tf-idf.
9. **Similar user recommendation –** ביצוע המלצה על ידי מציאת משתמש הדומה ביותר למשתמש הנבדק.
10. **Feature recommendation –** ביצוע המלצה על ידי מציאת משחק הקרוב ביותר לוקטור המייצג של המשתמש.
11. **Similar games recommendation –** המלצה על בסיס מציאת משחקים דומים כאשר המאפיין הוא התיאור המילולי של המשחקים.
12. **Hybrid recommendation –** המלצה זו מתבססת על שילוב התוצאות של שלושת המערכות ונתינת משקלים לכל תוצאה של מערכת הניתנים לכיוון בהתאם הצורך ואחרי ביצוע ניסויים.
בשלב זה הנוסחה מוגדר באופן הבא:

$$\alpha \cdot (Content\ Rec) + \beta \cdot (Feature\ Rec) + \gamma \cdot (Similar\ game\ rec) = Recommendation$$

$$\alpha = 1\ ;\ \beta = 0.65\ ;\ \gamma = 1$$

Game vectors

על מנת שנוכל לאפיין את המשחקים ושהם יהיו ברי השוואה הוחלט להשתמש ב Vector Space Model (VSM). לכל משחק נבנה וקטור המייצג על בסיס המאפיינים שהוצגו קודם לכן.
כל וקטור מכיל את כל האפשרויות לכל מאפיין והוא בוליאני, זאת אומרת שכל וקטור מכיל רק אפסים ואחדות.
לאחר עיבוד כל הנתונים נוצרו וקטורים באורך של 2602 פרמטרים.

דוגמא לתחילת מספר וקטורים:

game_name	1987	1993	1994	2014	1989	2009	1997	1990	2012	1985	2015	1998
The Elder Sc	0	0	0	0	0	0	0	0	0	0	0	0
Fallout 4	0	0	0	0	0	0	0	0	0	0	1	0
Spore	0	0	0	0	0	0	0	0	0	0	0	0
Fallout New	0	0	0	0	0	0	0	0	0	0	0	0
Left 4 Dead 2	0	0	0	0	0	1	0	0	0	0	0	0
HuniePop	0	0	0	0	0	0	0	0	0	0	1	0
Path of Exile	0	0	0	0	0	0	0	0	0	0	0	0

Figure14 Game VSM example

User vectors

בנוסף לוקטור משחק נוצרו וקטורים מייצגים לכל משתמש על בסיס הנתונים הקיימים עליו. הוקטורים המייצגים נוצרו באופן הבא:

1. לכל שחקן:
   a. נוצר אובייקט המכיל את הפרטים שלו
      i. מס׳ סידורי של השחקן.
      ii. רשימת המשחקים שהוא קנה.
      iii. זמן משחק לכל משחק שהוא קנה.
   b. בניית הוקטור המייצג
      i. לכל משחק ברשימה של השחקן נשלף הוקטור המייצג של **המשחק** מ Game vectors וכל הפרמטרים נכפלים ב $\log(play\ time)$ וזאת על מנת לתת משקל והשפעה לזמן המשחק של כל משחק.
      ii. הוקטור נשמר בקובץ להמשך שימוש.

דוגמא לתחילת מספר וקטורי משתמש:

user_id	1997	2000	2013	2018	1996	1998	2006	2012	1983
7565	0	0	-0.7985077	0	0	0	0	0	0
4278	0	0	0	0	0	0	0	0	0
9554	0	0	0	0	0	0	0	3.93182563	0
11424	0	0	0	0	0	0	0	0	0
8512	0	0	0	0	0	0	0	0	0
543	0	0	1.5260563	0	0	0	0	0	0
7664	0	0	8.07410098	3.76120012	0	0	0	3.00071982	0
10946	0	0	0	0	0	0	-2.3025851	-0.3566749	0

Figure15 User vector example

ביצוע המלצה

המערכת נבנתה בשלבים ולכן גם מורכבת משלוש מצבי המלצה נפרדים (כמו שמוצג בארכיטקטורה), כל מצב המלצה עצמאי והמערכת יכולה להשתמש רק בו לצורך ביצוע המלצה למשתמש.
כעת נפרט על כל מערכת:

1. Feature recommendation

בשיטת המלצה זו המערכת מתבסס על וקטורי משחק ו-וקטורי השחקנים שנוצרו בשלבים הקודמים.

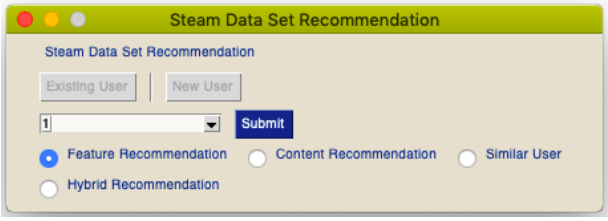


Figure16 Feature recommendation GUI

על מנת לייצר תוצאה למשתמש הנבחר המערכת בשיטה זו שולפת את הוקטור המייצג של השחקן ומחשבת את ה cosine similarity לכל הוקטורים של המשחקים הקיימים במאגר מלבד המשחקים שהשחקן הנבדק כבר קנה. לאחר ביצוע החישוב נבחרים רק TOP המשחקים הקרובים לוקטור של השחקן ומוצגים למשתמשים.

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

Figure17 Cosine similarity

דוגמא לתוצאה לשחקן מס׳ 1:

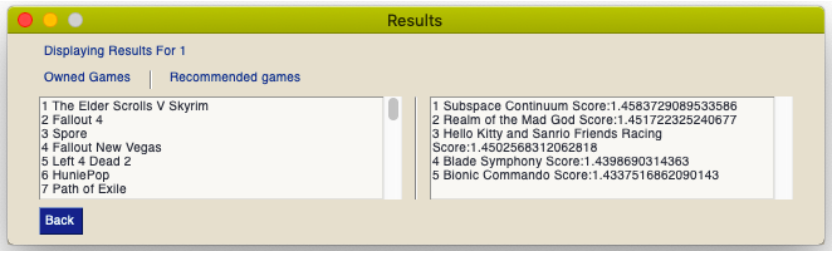


Figure18 Feature recommendation for user no. 1

2. Content recommendation

בשיטת המלצה זו המערכת מתבסס על תיאור מילולי של המשחקים על מנת לבחון עד כמה הם דומים. לצורך כך אנו משתמש במספר ספריות פייתון ייעודיות:

- **Pandas**
- **Sklearn**
  - **TfIdfVectorizer**
  - **Linear_kernel**

המערכת מחשבת וקטורי tf-idf לכל תיאור מילולי של משחק ומייצרת מטריצת tf-idf, לאחר מכן מחשבת cosine similarity בין וקטורים אלו כך שכעת אפשר לדרג את המרחק בין כל משחק ולשאר המשחקים על בסיס התיאור המילולי של המשחק.

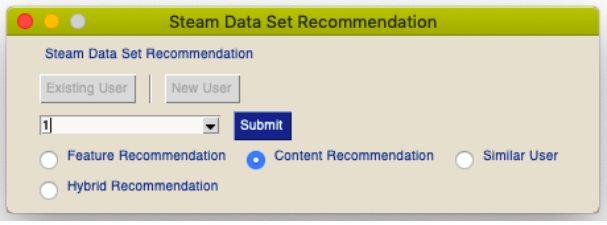


Figure19 Content recommendation GUI

בשלב של ביצוע ההמלצה למערכת יש כבר את כל הנתונים הדרושים על מנת לבצע את ההמלצה. המערכת שואלת את המשתמש האם הוא רוצה לבחור את המשחק שהוא הכי אוהב מהרשימה של המשחקים שהוא קנה **או** המערכת לוקחת את המשחק שיש לו הכי הרבה שעות משחק בו וממליצה על המשחקים הכי דומים לו.

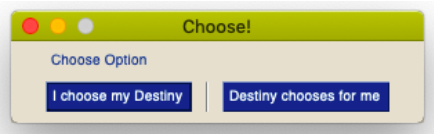


Figure20 Content user choice

'I choose my Destiny' – המשתמש בוחר את המשחק
'Destiny chooses for me' – המערכת בוחרת את המשחק

לאחר הבחירה המערכת שולפת את הנתונים שחושבו מקודם בהקשר למשחק שנבחר ומציגה למשתמש.

תוצאות עבור שחקן מס' 1 לאחר שהמערכת בחרה את המשחק עבורו:

Figure21 Content recommendation for user no. 1

3. User similarity

בשיטת המלצה זו המערכת מתבססת על הרעיון של מציאת המשתמש הדומה ביותר למשתמש הנבדק. גם בשיטה זו יש שימוש בספריות: **pandas, sklearn – cosine_similarity**.

המערכת משתמשת בקובץ שנוצר קודם לכן של תיאור וקטורי של השחקנים (user vectors) ומייצרת מטריצה של cosine similarity של כל השחקנים, כך שלכל שחקן יש רשימה של שחקנים אחרים הדומים לו בסדר יורד.
לאחר יצירת המטריצה נוצר מילון המכיל זוגות כך שלכל שחקן יש את מספר הסידורי של השחקן הכי דומה לו.

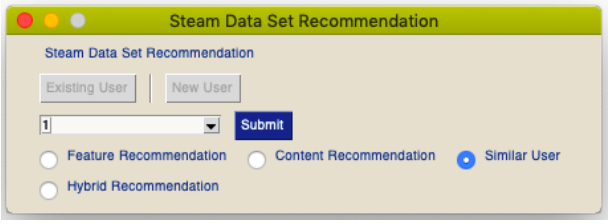


Figure22 Similar user GUI

בשלב זה המערכת רק ממתינה לקבל מס׳ סידורי של שחקן נבדק ואז שולפת מהמילון שהוכן מראש את השחקן הדומה ביותר לשחקן שהתבקש.
לאחר קבלת התוצאה, המערכת מציגה למשתמש את רשימת המשחקים שהשחקן **הדומה** קנה **ללא המשחקים שהמשתמש הנבדק כבר רכש.**

תוצאות למשתמש מס׳ 1:

Figure23 Similar user recommendation results for user no.1

4. Hybrid recommendation

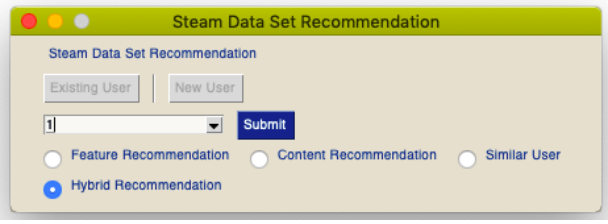


Figure24 Hybrid recommendation GUI

בשיטת המלצה זו אנו מנצלים את שלושת השיטות הקודמות ונותנים משקל לכל שיטה. כפי שצוין קודם המערכת מכוונת בשלב זה לתת משקלים באופן הבא:

$$\alpha \cdot (Content\ Rec) + \beta \cdot (Feature\ Rec) + \gamma \cdot (Similar\ game\ rec) = Recommendation$$
$$\alpha = 1\ ;\ \beta = 0.65\ ;\ \gamma = 1$$

המערכת בפועל מבצעת המלצות על בסיס כל אחת מהשיטות בנפרד ושומרת את התוצאות ברשימה כללית. כל התוצאות מוכפלות במשקל של המערכת בהתאמה ולאחר מכן הרשימה מסודרת בסדר יורד כך שהתוצאות הטובות ביותר נמצאות בראש הרשימה.
בסופו של התהליך ה TOP מוצג למשתמש.

דוגמא לתוצאות עבור משתמש מס׳ 1:

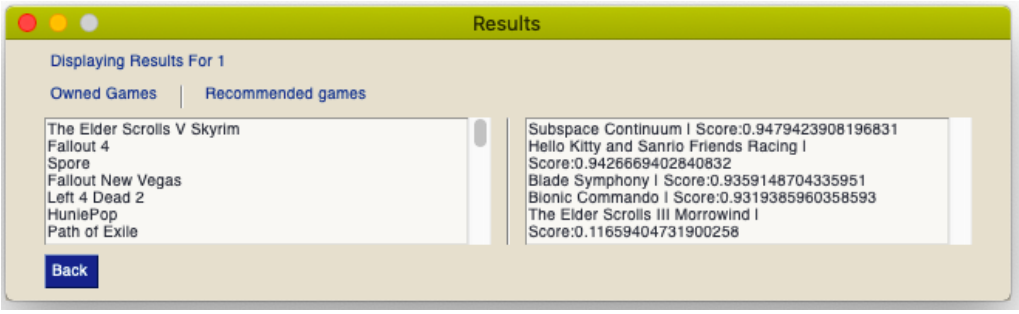


Figure25 Hybrid recommendation for user no.1

פרק 3: מדדי הערכה וניסויים

כל המערכת נבנתה על בסיס Vector space model (VSM), זאת אומרת שאת כל הנתונים שאפנו להפוך לוקטורים מייצגים על מנת שנוכל לבצע השוואה ביניהם ובעיקר להשתמש ב cosine similarity למציאת הזווית בין הוקטורים וסידורם בסדר יורד כך שיש לנו רשימה היררכית מהוקטור הכי קרוב לוקטור הנבדק עד להכי רחוק.

לכן על מנת לבצע הערכה נבחרה הגישה של leave one out לאחר בחינת האפשרויות ואת מורכבות המערכת בשלב ההערכה. בגלל שהמערכת בחלקה הגדול נבנתה ללא שימוש בספריות מוכנות ישנו קושי לבצע הערכה אוטומטית של התוצאות ושימוש בשיטות הערכה מתקדמות במסגרת הזמן.

ההערכה הוגדרה בשיטה הבאה, בכל ניסוי:

a. נבחר שחקן רנדומלי.
b. ניקח את רשימת המשחקים של השחקן הנבחר.
c. נמחק משחק אחד מהרשימה ונשמור אותו לצורך השוואה בסוף.
d. נבצע המלצה בכל אחת מהשיטות ונבחן האם המשחק הנמחק מופיע בהמלצות.

נכתב קוד המבצע את השלבים האלו מספר פעמים שנבחר בתחילת התהליך ושומר את התוצאות עבור כל אחד מהניסויים.

5 ניסויים

	Feature recommendation	Content recommendation	Similar user recommendation
1	0.323	0	1
2	0.443	0	1
3	0.068	1	1
4	0.018	0	1
5	0.436	0	1

15 ניסויים

	Feature recommendation	Content recommendation	Similar user recommendation
1	0.937	0	1
2	0.005	0	1
3	0.884	0	1
4	0.432	0	1
5	0	0	1
6	0.245	0	1
7	0.126	0	1
8	0.73	0	1
9	0.04	0	1
10	0.092	0	1
11	0.372	0	1
12	0.049	0	1
13	0	0	1
14	0.334	0	1
15	0	0	1

<u>הסבר לתוצאות:</u>
ב content ו similar user התוצאה הוערכה בצורה בוליאנית, כלומר אם המשחק שנמחק מהרשימה של השחקן שנבחר קיים או לא קיים ברשימת ההמלצה (0/1).
לעומת זאת, ב feature recommendation תוצאת ההערכה בוצע בשיטה מעט שונה. כאשר מבוצעת ההמלצה בשיטה זו מתקבלת רשימה של כל המשחקים בסדר יורד, כך שהמשחקים הכי מומלצים בראש הרשימה. תוצאת ההערכה מתבסס על מיקומו של המשחק שנמחק ברשימת ההמלצה, ככל שהוא רחוק יותר מראש ההמלצה המספר קטן וככל שהוא יותר קרוב לראש הרשימה המספר גדל על בסיס הנוסחה הבאה:
בהינתן רשימה של המשחקים לאחר ביצוע ההמלצה – result list :

$$score = 1 - \frac{index\ of\ game\ removed'\ in\ result\ list}{length\ of\ result\ list}$$

מסקנות:
על בסיס תוצאות הניסויים, ניתן להסיק כי המערכת הכי אמינה מתוך השלושה היא מערכת ה similar user והכי פחות אמינה ה content. אך גם לכך יש סייגים:

- נדרש לבצע מספר גדול מאוד של ניסויים על מנת לקבל תוצאות טובות יותר (לא בוצע עקב זמן ריצה ארוך).
- יכול להיות שנדרש היה לאפיין את תוצאות ההערכה של content recommendation באותו אופן כמו שבוצע ב feature recommendation – שימוש באינדקס מרחק ברשימת תוצאות.
- תוצאת ההערכה ב similar user, content מתבסס על תוצאה בוליאנית מה שמקשה על לזהות על מגמות בתוצאות.

פרק 4: סיכום ודיון בתוצאות

פרויקט המלצה זה מכיל בתוכו מספר מרכיבים קריטיים כמו שלוש שיטות שונות לביצוע המלצה ושילוב תוצאות במידת הנדרש על פי בחירת המשתמש. למשתמש במערכת ישנה חופשיות לבחירה על בסיס איזו שיטה הוא רוצה לקבל המלצה ואף אפשרות לייצר רשימה של משחקים אישית שעל בסיסה הוא יכול לקבל תוצאות.
מערכת ההמלצה מכילה ממשק משתמש נוח ויעיל לתצוגה למשתמש על מנת שהמשתמש יוכל לדעת באיזה שלב הוא נמצא בתהליך ואילו אפשרויות הוא יכול לבחור.

בפרויקט זה נתקלנו במספר רב של אתגרים החל ממציאת מאגר נתונים (date set) טוב מספיק על מנת לעבוד איתו ועד ביצוע הערכה בשלב הסופי של הפיתוח.

היינו צריכים להתמודד עם ניתוח מאגר הנתונים הקיים ובדיקת התפלגות הנתונים וסינון נתונים אשר יכולים לפגוע בתהליך. במקרה זה מאגר הנתונים הבסיסי היה מושלם וללא ׳חורים׳ במידע אך כאשר המאגר הורחב על בסיס הוספת מאפיינים של המשחקים היה אילוץ לסנן נתונים בהתאמה.

בניית מאגר נתונים נוסף למשחקים עצמם עם מאפיינים נוספים היה אתגר בפני עצמו אשר דרש למידה עצמית בתחום גישות API לאתר ואפשרויות ניתוח הנתונים וסידורם בצורה מסודרת במאגר חדש לצורך המשך עבודה.

אתגר נוסף התבטא כאשר הייתה התמודדות עם ספריות יותר מתקדמות שלא היה ניסיון עבודה איתן בתחילת העבודה כמו pandas, sklearn אשר דרש בחינת אפשרויות שניתן לבצע איתן, התאמת מבנה נתונים כך שנוכל לאפשר פעולה תקינה.

ביצוע ההערכה היה מאתגר משום שהמערכת נבנתה על בסיס רעיון שאינו מכיל בתוכו מודל מוגדר אשר אפשר להפעיל עליו שיטות הערכה אוטומטיות מתקדמות ולכן היינו צריכים לחשוב על דרך אפשרית לבצע הערכה שתיתן סוג של הערכה.

ממשק משתמש GUI – בוצע חקר מורחב על מנת למצוא חבילה יותר נוחה לבניית ממשק משתמש ולהימנע מלהשתמש בספריית ה tkinter המוכרת לאחר ניסיון עבודה מולה. לאחר ביצוע מספר בדיקות נמצאה סיפריית pySimpleGUI אשר חסכה המון זמן בפיתוח ממשק המשתמש אשר מוסיף להתנהלות מול התוכנית.

התמודדות עם דרישות משתנות ומתפתחות בזמן פיתוח המערכת. לאורך פיתוח המערכת הדרישות השתנו וגדלו בנפחם כך שהמערכת הורכבה בצורת מודלים נפרדים והיה אתגר לשלב את המודלים כך שיוכלו לפעול ולהרוויח מהתוצאות של כל מודל. אתגר זה גם השפיע המון על שלב ביצוע ההערכה הסופית משום שמנע אפשרות של תכנון המערכת בצורה יעילה לביצוע ההערכה.

אפשרויות לשיפור המערכת:
בניית המערכת על בסיס אותם עקרונות מחדש תוך כדי שימוש בכלים מתקדמים ובמודלים שלמים של ספריות מוכנות בעלות אפשרויות לביצוע הערכה בצורה יעילה יותר כך שנוכל לקבל תוצאות הערכה אשר מעידות על איכות המערכת בצורה אמינה.
שימוש בקוד או בתוצאות של הרחבות ה data set שבוצעו שכן חלק זה אינו קשור לבניית המודלים ובוצע בצורה טובה.

קישור לסרטון הרצה מלא

קישור לסרטון הרצה מלא
Google drive