בס"ד

תאריך הגשה: 3.1.22

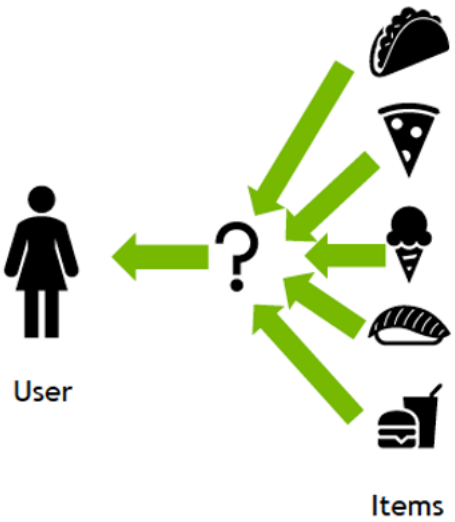
תרגיל 3 – מערכות המלצה

הקדמה

בתרגיל זה תבנו מערכת המלצה לסרטים.

התרגיל יחולק ל 3 חלקים

- ניתוח הנתונים
- בנית מערכת המלצה CF
- הערכות ביצועים



בקובץ zip המצורף נתונים לכם הקבצים הבאים:

תיקית plot ריקה לתוכה יכנסו תמונות הגרפים השונים שתתבקשו להציג.

main.py – הקובץ הראשי, דרכו נקרא לכל המימושים השונים.
אין לשנות קובץ זה!

data.py – כאן תממשו פו' שונות להבנת והצגת הנתונים.

collaborative_filtering.py – כאן תממשו מערכת המלצה CF מסוג user-based ו – item-based

evaluation.py – כאן תממשו פו' הערכה שונות למערכת ההמלצה.

בנפרד, קיים ZIP המכיל תיקיה בשם data עם קבצי הנתונים הרלוונטים. הורידו אותם ושימו בסיפרית השורש.

הגשה

במהלך התרגיל תערכו את הקבצים הבאים: data.py, collaborative_filtering.py ו - evaluation.py

עליכם לשלוח קבצים אלה עם הקוד והערות שלך. נא לא לשנות את הקבצים האחרים או לשלוח אף אחד מהקבצים המקוריים מלבד קבצים אלה.
בראש כל אחד מקבצים אלו נא לכתוב את שם הסטודנט ות.ז.

בנוסף לקבצים אלו עליכם להגיש דו"ח העונה על שאלות בתרגיל בקובץ בשם **report_<id>.pdf**,
כאשר id יהיה ת.ז. של הסטודנט וכן קובץ פרטים אישיים בשם detail.txt והוא יכיל את שם הסטודנט בשורה הראשונה ות.ז. בשורה השניה.

ההגשה דרך ה Moodle בלבד!! לא יתקבלו הגשות ב classroom משום סיבה.

חלק ראשון – נתונים

בקובץ data.py נתונה לכם הפו' watch_data_info(data). העזרו בפו' הנ"ל כדי להבין את קבצי הנתונים שצורפו לתרגיל.

נתייחס כעת לקובץ הדירוגים:

1. כמה משתמשים יחודיים דרגו את הסרטים? כמה סרטים יחודיות דורגו? כמה דירוגים קיימים בקובץ שניתן?
2. מהו מספר הדירוגים המינימלי והמקסימלי שניתן לסרט?
3. מהו מספר הדירוגים המינימלי והמקסימלי שמשתמש דירג?

את המימוש לקבלת התשובות הנ"ל יש לכתוב בפו' print_data(data). (הפו' מקבלת את קבצי הנתונים המתאימים כ tuple במשתנה data) מהי התפלגות הדירוגים? הציגו גרף מתאים.

את המימוש לקבלת הגרף הנ"ל יש לכתוב בפו' plot_data(data , plot = True) המקבלת את קבצי הנתונים המתאימים כ tuple במשתנה data, ומשתנה plot המקבל כברירת מחדל true – ובמצב זה הפעילו את פו' plot.show() עבור הגרף.

על כל השאלות הנ"ל והגרף שהתבקשתם להציג, יש לענות בקובץ הדו"ח אותו אתם מגישים.

חלק שני - collaborating-filltering

בקובץ collaborative_filtering.py ישנה מחלקה למימוש מערכת המלצה מסוג CF.
עבור בנית מטריצת החיזוי, נשתמש בשדה במדד הדמיון cosine - מדד המחשב את הזוית בין שני וקטורים במרחב:

$$\cos(\mathbf{x},\mathbf{y}) = \frac{\mathbf{x}\cdot\mathbf{y}^{\mathrm{T}}}{\|\mathbf{x}\|\cdot\|\mathbf{y}\|} = \frac{\sum_{i=1}^{n}\mathbf{x}_i\cdot\mathbf{y}^{\mathrm{T}}{}_i}{\sqrt{\sum_{i=1}^{n}(\mathbf{x}_i)^2}\sqrt{\sum_{i=1}^{n}(\mathbf{y}_i)^2}}$$

ממשו את שתי הפו':

create_user_based_matrix(data)

create_item_based_matrix(data)

המופיעות בקובץ את מטריצת החיזוי עבור CF-user based ו – CF-item based בהתאמה. הפו' מקבלות את המשתנה data – המכיל את קבצי הנתונים המתאימים כ tuple.
הפו' יבנו מטריצות חיזוי לתוך המשתנים user_based_matrix ו - item_based_matrix.

<u>טיפ</u> – לצורך נוחות, מומלץ לבצע המרה לשדות userId וmovieId למספרים עוקבים מ 0 עד למספר הערכים היחודיים של אותו שדה.

<u>שימו לב</u> – לא נבקש יותר מ 10 המלצות עבור משתמש.

בנוסף, ממשו פו' predict_movies (user_id, k, is_user_based) המקבלת 3 משתנים:
user_id – המכילה מספר ID של משתמש בעבורו נרצה לבצע חיזוי
k – מספר הסרטים המומלצים שנרצה לקבל.
is_user_based – המקבלת ערך true עבור חיזוי מסוג CF-user based (ברירת המחדל) וערך false עבור חיזוי מסוג CF-item based

הפו' תחזיר את רשימת שמות הסרטים המומלצים עבור אותו משתמש.

אתם רשאים להוסיף פו' ומשתנים בקובץ collaborative_filtering.py כרצונכם, מבלי לשנות את הקוד הקיים.

4. מה יהיה החיזוי עבור משתמש "**283225**" ב cf_user_based ו – k=5? האם התוצאות תואמות את אופי המשתמש? הציגו את תוצאות החיזוי עבור משתמש זה והסבירו את תשובתכם בדוח.
5. בקובץ collaborative_filtering.py קיימת פו' create_fake_user(rating) המקבלת את נתוני הדרוגים. בעזרת פו' זו, הוסיפו משתמש פקטיבי (userId = **283238**) לנתוני הדירוגים וחזו עבורו רשימת המלצות. מה תהיה ההמלצה הטובה ביותר עבור משתמש זה ב cf_user_based ו – k=5? הסבירו את המימוש ואופי המשתמש בדוח והציגו את תוצאות החיזוי עבור משתמש זה.

חלק שלישי – הערכות

קובץ test.csv הינו קובץ טסט של דרוגי משתמשים, בעזרתו תעריכו את מערכות ההמלצה CF השונות.

מדדי הערכה בהם נשתמש הם:

Precision@10 - מדד לדיוק ב- k, הוא החלק הרלוונטי של הפריטים המומלצים בערכת ה- top-k

במקרה שלנו נתייחס לדירוגים גבוהים בערכי 4 ו-5 כאשר K = 10

$$P@k = \frac{\#hits}{k}$$

ARHA - מדד אשר לוקח בחשבון רק היכן התוצאה הרלוונטית מתרחשת. אנו מקבלים יותר קרדיט על המלצה על פריט שבו משתמש מדורג בראש הדירוג מאשר בתחתית הדירוג. גבוה יותר זה יותר טוב.

$$ARHR = \frac{1}{\#users} \sum_{i=1}^{\#hits} \frac{1}{pos_i}$$

RMSE - מדד שנמצא בשימוש תכוף להבדלים בין ערכים חזויים על ידי מודל או אומדן לבין הערכים האמיתיים ומוגדר כך:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N} (Predicted_i - Actual_i)^2}{N}}$$

בקובץ evaluation.py מופיעות הפו' הבאות.:

precision_10(test_set, cf, is_user_based)
ARHA(test_set, cf, is_user_based)
RSME(test_set, cf, is_user_based)

ממשו כל אחת מהפו' הנ"ל של מחלקת collaborative_filtering, המקבלת את ה test_set, המתשנה cf ו - is_user_based שעבור ערך true תבצע חיזוי עבור מסוג CF-user based (ברירת המחדל) וערך false עבור חיזוי מסוג CF-item based

הפו' תדפיס את ערך ההערכה שחישבה (כפי שמופיע בתוך הפו')

גם כאן אתם רשאים להוסיף פו' ומשתנים כרצונכם, מבלי לשנות את הקוד הקיים.

6. הוסיפו בדוח את הטבלה הנ"ל המכילה את מדדי ההערכה שחישבתם.

	precision@10	ARHR	RMSE
User based			
Item based			

בהצלחה רבה!