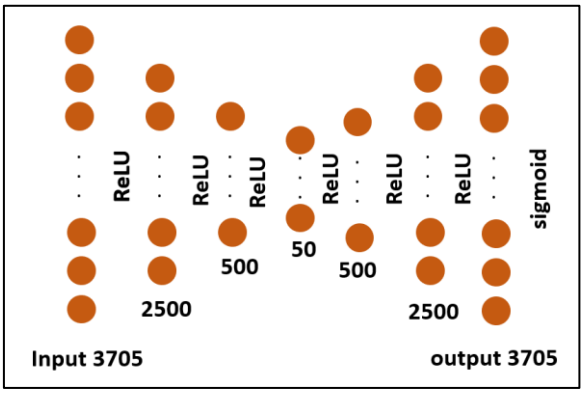
מבוא ללמידה עמוקה- תרגיל בית 3

סעיף 1 – ארכיטקטורת ה-AE

a. הגדרת ה-encoder וה-decoder:

בנינו autoencoder עבור סט הטסט (הן עבור שיטת הפופולריות והן עבור שיטת האקראיות) כך שגם ה-encoder וגם ה-decoder יורכבו מטרנספורמציות ליניאריות ובין השכבות הללו יהיה שימוש בפונקציית אקטיבציה ReLU. עם זאת בפלט של ה-decoder בחרנו בפונקציית אקטיבציה של סיגמואיד כדי לקבל פלט שמנורמל בטווח מ-0 ועד 1, כלומר הסתברויות.

באיור מטה ניתן לראות את רשת ה-AE, כך שגודל הקלט הוא כגודל הפלט שזה למעשה מספר הפריטים הכולל במערך הנתונים. השכבה הראשונה כוללת את הקלט הנכנס (מספר הפריטים) ולאחר מכן יש שכבה חבויה בגודל 2500, לאחריה שכבה חבויה בגודל 500 ולבסוף הקידוד של ה-encoder שזה שכבה חבויה בגודל 50. לאחר מכן חוזרים חזרה לשכבה חבויה בגודל 500 ולאחריה גודל 2500 ולבסוף חוזרים לממד המקורי.



b. פונקציית העלות:

בחרנו להשתמש ב-BCELoss יחד עם משקול. בחרנו במדד זה בגלל שהבעיה היא בינארית. בחרנו למשקל מכיוון שמטריצת פריט-משתמש הינה מטריצה דלילה שכן משתמש לא צרך את כל הפריטים אלא חלק קטן לרוב. כדי שנמנע מתוצאות מוטות למחלקה השלילית שכן היא תופסת נפח גדול במטריצה אז נתנו משקל רב יותר למחלקה החיובית. במשקולת זו נשתמש בפונקציית loss כך שהמשקל הטוב ביותר שנבחר בכוונון היפר הפרמטרים הוא 0.6.

c. רגולריזציה:

על מנת להכניס רגולריזציה לרשת בחרנו כי צוואר הבקבוק יהיה מורכב מ-50 נוירונים כך שבקלט ובפלט יש 3705 (כפי שמוצג בתמונה בסעיף a). עם זאת למדנו כי כדאי להכניס שיטות רגולריזציה נוספות שכן הכנסת צוואר בקבוק אינה בהכרח מונעת אובר פיטינג. לכן החלטנו להשתמש גם בשיטת early stopping, כך שנעצור את אימון הרשת על פי מדד הדיוק של סט הולידציה. זאת כך שנדרש מינימום epochs לפני עצירה כדי שהמודל יתאמן, אך במידה ואין שינוי כלל במדד הדיוק החל ממספר ה-epochs המינימליים נעצור את האימון. כמו כן נשתמש באופטימייזר מסוג adam עם weight decay, כלומר רגולריזציה של l2 בכל שכבה.

d. הצדקת בחירת הארכיטקטורה:

לבחירת הקונפיגורציה הטובה ביותר, הרצנו קונפיגורציות שונות ומדדנו לכל אחת את מדד הדיוק על סט הולידציה. בחרנו בקונפיגורציה שהניבה דיוק גבוה ביותר. לאחר בחירת קונפיגורציה זו ומציאת כמות epochs אשר מונעת אובר פיטינג, אימנו את המודל על כל הדאטה (ללא פיצול ל-validation נפרד) וזאת לצורך חיזוי על סט הטסט אשר נעשה ל-2 השיטות בנפרד (אקראיות ופופולריות).

הקונפיגורציה הטובה ביותר שהתקבלה ובה השתמשנו היא כמתואר בסעיף a לעיל. ניתן לראות את הקונפיגורציות השונות והמדדים שהתקבלו לכל שיטה בנספחים.

נציין כי מבחינת היפר פרמטרים קבועים, החלטנו שהמודל יופעל עם גודל batch של 256, ומספר epochs מוגבל ל-50. כמו כן, בחרנו בפונקציית אקטיבציה של סיגמואיד בשכבת הפלט כדי שנקבל פלט של הסתברויות וכך נוכל להשוות לאיזה מבין 2 הפריטים יש סיכוי גבוה יותר להיצרך ע"י המשתמש.

סעיף 2 – תיאור תהליך האימון

a. פיצול הנתונים ל-train&validation:

סט הולידציה ישמש אותנו כדי לכוון את היפר הפרמטרים השונים שיש ברשת, כמו גם למנוע התרחשות של אובר פיטינג. למעשה, לאחר אימון המודל עם סט האימון, נבצע הערכה עם סט הולידציה עבור בחירת היפר הפרמטרים כדי לבחור את הטובים ביותר. כפי שהוזכר סט זה יעזור לנו לבדוק האם אנו בשלב בו המודל התכנס, כלומר הדיוק על סט הולידציה ללא שינוי, בשלב זה נעצור וזאת כדי למנוע מצב של אובר פיטינג. מכאן קיים הצורך להשתמש בסט ולידציה וביצירתו שכן אינו נתון לנו מראש.

נרצה לייצר סט ולידציה שיהיה דומה כמה שניתן לסט הטסט. לצורך כך סט הולידציה נוצר ע"י בחירת שורה אקראית אחת לכל משתמש אשר תיכנס לסט הולידציה ע"י דגימה של פריט חיובי אקראי לכל משתמש, כלומר פריט שהמשתמש אכן צרך והוצאת פריט זה מסט האימון. לאחר שלב זה יש לנו שורה עם משתמש ופריט שצרך, לכן דגמנו גם פריט שלילי, אשר המשתמש לא צרך, לכל משתמש ע"י התפלגות מתאימה, כלומר לפי פופולריות או לפי אקראיות. נחזור על תהליך זה עבור כל המשתמשים ולבסוף נוצר סט ולידציה שבו יש לכל משתמש מסט האימון רשומה אחת ובה פריט אחד שצרך (פריט חיובי) ופריט אחד שלא צרך (פריט שלילי).

את התכנסות המודל בדקנו ע"י הגדרת מדד דיוק שמשמעותו היא הפרדה מוצלחת בין המדגם החיובי והשלילי לפי הסתברויות שהן פלט המודל/הרשת שנוצרה. מדד הדיוק נבחר עקב כך שמדובר בבעיית קלסיפיקציה אשר מתאים לבעיות אלו, כך שמדד גבוה יותר הוא יותר טוב. המודל רץ עם הקונפיגורציה שנבחרה עד שינוי הדיוק לפי סף שהוגדר מראש, כך שלצורך מניעת אובר פיטינג אימון המודל נעצר לפני שינוי זה. בשלב זה שמרנו את מספר ה-epochs המתאים והרצנו איתו על המודל הסופי.

נציין גם כי לפני פיצול הנתונים לאימון וולידציה ביצענו עיבוד מקדים לנתונים. העיבוד כלל אינדוקס מחדש ל-lds כך שיתחילו מ-0 ולא מ-1, הן עבור המשתמשים והן עבור הפריטים. זאת על מנת להיות מיושרים עם מטריצת משתמש-פריט. יצרנו את מטריצת משתמש פריט לסט האימון אשר כללה את הפריטים והמשתמשים וייצגה עבור כל משתמש את כל הפריטים שלו באופן הבא: 1 אם המשתמש צרך את הפריט ו-0 אם לא. מטריצה זו תהווה קלט לרשת ע"י ייצוגי המשתמש. בנוסף יצרנו מילון להסתברויות הדגימות השליליות עבור כל משתמש לצורך הולידציה בהמשך. הדגימות השליליות יכלו להיבחר מהתפלגות אקראית (בצורה אחידה) או פופולרית (לפי הנתונים). לצורך כך יצרנו 2 מערכי הסתברויות מנורמלות למשתמש לכל הפריטים שלא צרך בסט האימון (מה שצרך היה בהסתברות 0 שייצג פריט שלא ניתן לבחור). המילון היה עם מפתח של מזהה המשתמש והערך זה טאפל עם (הסתברות לפי פופולריות, הסתברות לפי אקראיות ומספר פריטים שדירג).

תהליך יצירת סט הולידציה החל כך שהגדרנו validation ratio של 0.2 ופיצלנו את המשתמשים מסט האימון לאימון וולידציה. לפי המשתמשים שנבחרו עפ"י יחס זה חילקנו את המטריצה של פריט-משתמש למשתמשי אימון ומשתמשי ולידציה.

b. דגימת פריטים שליליים:

אין לנו דוגמאות שליליות. בסט הנתונים שלנו יש בכל שורה מזהה משתמש ומזהה לפריט כאשר הם מייצגים פריטים שהמשתמש צרך. כלומר, יש לנו דירוגים מרומזים ועל כן עלינו להתמודד עם זה. לכן, על מנת ליצור דוגמאות לפריטים שהמשתמש לא צרך שייקראו דוגמאות שליליות. נגריל דוגמאות שליליות מתוך הפריטים שאין תיעוד לצריכתם על ידי אותו המשתמש.

כפי שהוסבר לעיל, בעיבוד המקדים יצרנו מילון להסתברויות הדגימות השליליות עבור כל משתמש כך שלהגרלת הפריטים יש 2 אפשרויות של הסתברויות:

1. הגרלה בצורה אחידה: מבין כל הפריטים שהמשתמש לא צרך, נגריל בצורה אחידה פריט אחד, כך שלכל פריט יש הסתברות שווה להיבחר.
2. הגרלה מדורגת: נמשקל את הפריטים שהמשתמש לא צרך לפי הפופולריות שלהם – לפי כמות הפעמים שהמשתמשים האחרים צרכו אותם. לאחר מכן, נגריל מההתפלגות הממושקלת את הפריט, כך שפריט פופולרי יותר יהיה בעל הסתברות גבוהה יותר לקבלו.

<u>c. בחירת ה-loss למודל ועדכון הסכמה:</u>

כפי שצוין לעיל בסעיף 1b, בחרנו במדד BCELoss יחד עם משקול. עבור סכמת העדכון, ביצענו התאמות בפונקציית ה-loss ע"י משקול כדי לאזן בין דגימות חיוביות ושליליות כפי שמוסבר גם כן באותו הסעיף.

סעיף 3 – תיאור ההסקה שלנו: איך נשתמש במודלים המאומנים ליצירת תחזיות

על מנת לייצר תחזיות ולחזות את סט הטסט ל-2 השיטות, לאחר בחירת היפר הפרמטרים נאמן את המודל עם כל הדגימות (ללא פיצול לאימון וולידציה) עם מספר ה-epochs הטוב ביותר שמצאנו עד עצירה מוקדמת אם התרחשה או עד כמות ה-epochs שנקבעה מראש. מאימון המודל על כל נתוני האימון נקבל וקטור הסתברויות לכל משתמש לרכישת כל פריט. לכל שורה בסט הטסט ניקח את ההסתברויות שקיבלנו ל-2 הפריטים הרלוונטיים שמופיעים ונבדוק לאיזה פריט יש הסתברות גבוהה יותר. אם לפריט הראשון יש הסתברות גבוהה ביותר אז נגדיר תגית 0 ואחרת 1.

סעיף 4 – ההבדל בין 2 קבצי ה-test

<u>a. האם ראינו הבדלים בין הקבצים? איך נסביר זאת?</u>

ראשית, ניתן לראות כי אכן יש הבדלים בין 2 קבצי הטסט שקיבלנו. על פי מדד הדיוק שקיבלנו מסט הולידציה (ראה נספח) ראינו כי התוצאה של השיטה האקראית השיגה תוצאות טובות בהרבה (0.922) מאשר השיטה הפופולרית (0.815). כלומר אנו רואים כי קשה יותר ללמוד את הסט של השיטה הפופולרית מאשר האקראית. התוצאות היו בניגוד לציפיות שלנו, כך שציפינו כי השיטה הפופולרית תניב תוצאות טובות יותר שכן לוקחת את התפלגות הפריטים בהתאם לפופולריות ולא משייכת הסתברות זהה לבחירה לכל אחד באופן שלא בהכרח תואם את המציאות. למעשה, התפלגות 2 הסטים של הטסט שונה זה מזה וזה יכול להסביר את הפערים. ראינו כי עבור השיטה האקראית ההסתברות לדגימת פריטים שליליים הינה שווה לכל הפריטים בעוד בפופולריות ככל שהפריט פופולרי יותר כך הסתברותו לדגימה גבוהה יותר. אנו מניחים כי הסיבה שדגימת הפופולריות מקשה יותר על לימוד פריטים אשר יותר ייחודיים ונמצאים במיעוט, כלומר מופיעים פחות וזה יכול להקשות על ההבחנה בין העדפות המשתמשים. בסט הטסט של הפופולריות, המודל מקבל 2 דגימות כך שאחת חיובית ואחת שלילית, כך שהדגימה החיובית הייתה בסבירות גבוהה פופולרית מאוד בקרב כל המשתמשים. המשמעות היא שהמודל יצטרך להתאמץ יותר לציין את ההעדפות הייחודיות שאינן טריוויאליות ע"י המשתמשים לצורך הבחנה בין הפריטים.

<u>b. איך נוכל להתאים את האימון כדי להתאים טוב יותר לקובץ המבחן השני (פופולריות)</u>

כפי שציינו בסעיף הקודם יש פער בין כמות המופעים של פריטים כך שיש פריטים פופולריים שמופיעים בתדירות הרבה יותר גבוהה. על מנת להתגבר על פער זה נרצה לתת משקל גדול יותר עבור הפריטים הלא פופולריים, אשר מופיעים בתדירות נמוכה. ננרמל את הפריטים ע"י חלוקה בפרופורציית המופעים שלהם בסט האימון.

עבור סט הפופולריות, בדקנו את הצעה זו (עם/ללא נרמול) במחברת הקוד. קיבלנו כי אכן המדדים הינם טובים יותר כאשר אנו מבצעים נרמול, ולכן בחרנו להשתמש בשיטה זו.

תובנות מהעבודה

ראשית, נציין כי למדנו המון מתרגיל זה והבנו כי מודלי ה-AE חזקים בתור מודלי חיזוי עבור מערכות המלצה שונות. זו הפעם הראשונה בה התנסנו במודל זה ומעצם החוזקה שלו נוכל להשתמש בו בעתיד בעבודות שונות. הרעיון של מודל ה-AE והאופן שבו יש לחלק את סט האימון לולידציה היווה דרך חדשה שלא הכרנו ומאתגרת וכך הבנו איך מיישמים מודלים כאלה בעולם האמיתי שבו לא הכל בהכרח נתון מראש. בנוסף לתובנות כלליות אלו, למדנו כי עלינו לייצר סט ולידציה שעליו נחשב את הדיוק ונוודא שלא נוצר אובר פיטינג אך זאת בהתאם למבנה של סט הטסט שנרצה לחזות בסוף תהליך בניית המודל גם אם אינו בהכרח זהה לסט האימון. בפרט, בעבודה זו יצרנו למשתמש גם פריט שלילי על ידי דגימה לפי הסתברות מתאימה, שאינו קיים למעשה בסט האימון. למדנו גם כי עלינו לתת את הדעת במקרה שיש אי איזון בתיוגי הפריטים השונים ובפרט כאן במטריצה דלילה. כלומר, יש המון פריטים אך כל משתמש לא צורך את הכל ואפילו לא את הרוב ומכאן יש הרבה פריטים שליליים אך פחות חיוביים לכל משתמש. על כן בחרנו לתת משקל רב יותר בחישוב ה-loss עבור התיוג המועט יותר. דרך זו תמנע מהמודל ליצור הטיה לכיוון התיוג הגבוה יותר מבחינת מספר הרשומות.

נספחים

תוצאות עבור חיפוש מבנה הרשת הטוב ביותר. השורה הצבועה בצהוב הינה הנבחרת.

מבנה השכבות החבויות	רנדומלי	פופולריות ללא נרמול	פופולריות עם נרמול
2500X500X50	0.922	0.776	0.815
500X100X50	0.904	0.711	0.813
2000X100X500	0.914	0.715	0.815
8500X750X650	0.918	0.718	0.820
1000X500X100X50	0.909	0.773	0.789
1000X500X250X100	0.895	0.704	0.752
1500X800X400X100	0.9	0.738	0.748
1500X800X200X50	0.917	0.766	0.729
2500X1800X1000X500X100	0.875	0.874	0.873
2300X1600X850X400X75	0.879	0.878	0.88
2700X1600X500X200X40	0.866	0.86	0.864
2850X1700X400X100X25	0.836	0.842	0.841