**דו"ח תרגיל בית 5**

Transformer and LLMs

**שמות המגישים + ת"ז:**

עוביידה חטיב, 201278066

מאיה עטואן, 314813494

**חלק א'**

הפונקציה data\_load, שנקראת מה- main, אחראית על טעינת ה- subset מתוך המאגר. אם יש subset שנשמר מקודם, היא טוענת אותו מהדיסק, הדבר נעשה ע"י בדיקת אם התיקייה של הפרויקט מכילה מאגר נתונים בשם ' 'imdb\_subset. אם כן, היא טוענת אותו. אחרת, הפונקציה טוענת את מאגר הנתונים השלם של IMDB, בוחרת באופן אקראי 500 דגימות ממנו, ושומרת אותו לדיסק. לאחר שמירתו לדיסק, היא טוענת אותו מחדש.

**חלק ב'**

המודל bert-base-uncased נטען, והוגדר לביצוע סיווג לשתי קטגוריות (ע"י num\_labels=2) בהתאם לסיווג שעושה המשימה שלנו לביקורות לחיוביות ושליליות. המודל עבר טוקנזציה, ושם העמודה 'label' הוחלף ל- 'labels'. מאגר הנתונים שלנו חולק לשני סטים: אימון (80%) ובדיקה (20%). לאחר מכן, אובייקט Trainer אותחל, כאשר קיבל לצד המודל את הארגומנטים של האימון, שני הסטים של האימון והבדיקה, הטוקנייזר, והפונקציה compute\_metrics המחשבת את דיוק המודל. הוא עבר אימון ואיבלואציה, כאשר את תוצאות האיבולואציה מודפסות למסך. **התוצאות מצורפות להלן**:

**האוגרמטים שנבחרו עבור האימון**:

**evaluation\_strategy = "no"** , נבחר בשל כך שאנחנו לא נדרשים להדפסה במהלך האימון, אלא רק את התוצאות שהתקבלו בשלב האיבולאציה.

**learning\_rate= 2e-5**, זהו הערך הסטנדרטי והיה חשוב להשתמש בו על מנת לשמור על איזון בין יציבות המודל ומהירות הלמידה שלו.

**per\_device\_train\_batch\_size = 8, per\_device\_eval\_batch\_size = 8**, הערכים האלה התאימו למאפיינים של המחשב שלנו ולכן לא היה צורך בלשנות אותם.

**num\_train\_epochs = 3**, הערך נבחר על סמך תצפית שלנו לתוצאת שהראו שמעבר ל- epoch השלישי אחוז הדיוק וה- function loss לא משתנות בהרבה, ומזה הוסק ששימוש במספר גדול יותר עלול להוביל ל- overfitting.

**weight\_decay = 0.01**, גם כן נבחר על סמך תצפית שלנו, כך שהערך הראה איזון והתאמה למשתנה num\_train\_epochs.

הפונקציה compute\_metrics שהוזכרה קודם לכן מחשבת את הביצוע של המודל ע"י השוואת הפלט logits לתוויות האמיתיות labels. היא משתמשת בפונקציה np.argmax() על מנת להמיר את התוויות של הפלט לבינאריות (0,1) ומשווה אותם לתוויות האמיתיות ע"י הפונקציה accuracy\_score() מהספרייה sklearn.

**שאלות**

1. **האם התוצאות היו טובות? האם הן תאמו לציפיות שלכם? הסבירו.**

לדעתנו, התוצאות טובות, התוצאה של 91% מצביעה על דיוק גבוה. הם כן תאמו במידה רבה לציפיות שלנו, בשל כך שהמודל יותר דינמי בתפיסת המבנה התחבירי של מילה במשפט, ואת ההקשר הסמנטי בין המלים לעומת מודלים שהשתמשנו בהם בעבר כמו TF-IDF.

1. **אם נרצה לסווג ביקורות ספרים באמצעות אותו מודל שאימנו, האם לדעתכם התוצאות יהיו טובות? הסבירו.**

אנחנו מצפים שהמודל יצליח להתמודד עם סיווג ביקורות ספרים, אולם במידה פחות טובה. יש הרבה משותף בין שתי המשימות. דוגמה לכך היא השימוש בשם תואר על מנת לתאר את החוויה או היבטים בה. שם התואר תמיד יצביע על חוויה שהיא כפן שלו, כמו במקרה של המילה "מאכזב" שנוטה להופיע בחוות דעת שלילית יותר מאשר חיובית ללא קשר להקשר. מצד שני, יש אלמנטים שכן ייחודיים להקשר של סרטים, כמו האלמנטים הוויזואליים הקשורים לצילום והעריכה, או המשפט שהוזכר בהרצאה " the value I got from the two hours watching it was the sum total of the popcorn and the drink" שיש לה פחות משמעות כאשר היא מופיעה בביקורת של ספר.

1. **אם נרצה לסווג ביקורות סרטים ממאגר אחר, שאינו IMDB , באמצעות המודל שאימנו, האם לדעתכם התוצאות יהיו טובות? הסבירו.**

אנחנו מצפים שהתוצאות יהיו יותר טובות מאלה של הספרים, אולם פחות בהשוואה לביקורות של מאגר ה- IMDB, כאשר מידת הדיוק תהיה בהתאם לכמה המאגר הזה דומה למאגר של ה- IMDB. אלמנטים שיכולים להיות שונים בין שני המאגרים: סגנון השפה, למשל אם הביקורות לקוחות ממגזין שבו אנשים מקצועניים כותבים את הביקורות אנחנו נצפה לשפה יותר פורמלית ושימוש באוצר מלים שונה. ז'אנרי הסרטים שהוא מכיל והתפלגות הז'אנרים השונים מתוך כלל הסרטים, כי ז'אנרים שונים מושכים קהלים שונים בתחביב, בגיל וכתוצאה מכך המלים, האורך של הביקורת ועוד מאפיינים עשויים להיות מושפעים.

1. **עבור כל אחד מהמקרים מסעיפים 2,3 , תארו מה נוכל לעשות כדי לשפר (עוד יותר) את התוצאות?**

להן הצעות שאמורות לשפר את התוצאות של המודל. ההצעות תקפות לשתי המשימות מהסעיפים 2,3:

* להוסיף דאטה שהיא ביקורות ספרים לסט האימון (להוסיף דאטה ממאגר הסרטים האחר), ובכך ליצור מודל המסוגל לסווג את שני הסוגים של ביקורות.
* לעשות fine tuning למודל ולהתאים אותו מחדש למשימה של סיווג ספרים (להתאים אותו לסגנון ותוכן הביקורות של המאגר האחר).
* להגדיל את הרגולריזציה, ע"י לתת ערך גדול יותר לארגומנט weight\_decay אשר מונע למידת יתר של סגנון מסוים, דבר שאמור להשתקף ביכולת המודל להכליל טוב יותר למשימה של ביקורות הספרים (להכליל טוב יותר לתבניות אחרות של ביקורות סרטים).

**חלק ג'**

הקוד מתחיל בטעינת מאגר הנתונים של IMDB, ומתוכו הוא בוחר באקראי 100 ביקורות חיוביות ו- 100 שליליות. הקוד לאחר מכן טוען שני מודלים נפרדים של GPT-2, אחד ישמש לאימון על ביקורות חיוביות והשני על ביקורות שליליות. כמו כן, לכל מודל, טוענים גם את הטוקנייזר שלו. השלב שאחרי הוא ביצוע טוקנזציה על המאגרים שבחרנו. הפעולה מתבצעת ע"י הקריאה לפונקציה tokenize\_reviews(), אשר ממירה את הביקורות לרצפי טוקנים קבועי אורך. התהליך מתבצע עבור כל סוג של ביקורות (חיוביות ושליליות) בנפרד.

בשלב שאחרי, נוצרו שני מופעים של TrainingArguments, שהגדירו את פרטי האימון של המודל. **בחרנו את הערכים של הפרמטרים** באופן זהה לערכים שהשתמשנו בהם עבור המשימה בחלק ב' מטעמים דומים: evaluation\_strategy = "no" כי לא התבקשנו להדפיס בשלב האימון, ה- learning\_rate= 2e-5 הוא הערך הכי בשימוש, ובשל היות המאגר שלנו לא גדול במיוחד העלאתו יכולה להביא ל- Overfitting. per\_device\_train\_batch\_size = 8, ו- per\_device\_eval\_batch\_size = 8 התאימו למאפיינים של המחשב שלנו. באשר ל- num\_train\_epochs ו- weight\_decay נבחרו להיות 3 ו- 0.01 בהתאמה גם כאן על סמך תצפית שערכנו, ערכים אלו הראו איזון טוב בין Overfitting ל- Underfitting. יש לציין שהערכים של הפרמטרים זהים עבור שני המודלים בשל החשיבות שהתהליך יתבצע באופן אחיד עבור שני סוגי הביקורת.

לאחר מכן, נוצרו שני אובייקטי Trainer, שכל אחד שימש לאימון אחד המודלים. המודלים והטוקנייזר נשמרו לשימוש עתידי, ונטענו שוב מהדיסק. בשלב שאחרי, הוגדר prompt להיות " The movie was", הוא עבר קידוד והתווסף אליו attention mask. הפעולה התבצעה בנפרד עבור כל מודל.

לאחר מכן, הפונקציה reviews\_generate נקראה פעמיים עבור כל סוג של ביקורת. הפונקציה מפיקה (generate) חמשה ביקורות ע"י השימוש בפונקציה generate() של המודל עם **הפרמטרים והערכים הבאים**:

* **max\_length = 100**, הערך של הפרמטר נבחר להיות כזה משום שערכים גבוהים יותר הראו נטייה של הביקורת לכלול ציטוטים יותר מאשר היא מביעה דעה. הציטוטים מובאים כדברים שנאמרו בתוך הסרט ע"י אחד הדמויות. מנגד, ערך max\_length קטן יותר הציג ביקורות שטחיות שלא מביעות דעה מעמיקה כך שלפעמים קשה לסווג אותה גם עבור מישהו אנושי.
* **do\_sample = True**, מאפשר רנדומליות בבחירת המילה בכל שלב שלא בהכרח המילה בעלת ההסתברות הגדולה ביותר. זה חשוב על סמך כך שהמשימה שלנו היא הפקת משפטים בעלי משמעות וחשוב לתת מקום ליצירתיות ולא משימה דטרמיניסטית שבה יש מילה נכונה בהכרח. כמו כן, זה מתבסס על תרגילי הבית הקודמים שראינו בהם שיש לפעמים השלמות שהן פחות שכיחות מהשלמות אחרות.
* **temperature=0.5, top\_p=0.9**, שני הפרמטרים קשורים אחד לשני, וקובעים את מידת אקראיות\שמרנות המודל. הערך נבחר על סמך התצפיות שלנו, ערכים גבוהים יותר אולם יצרו משפטים נכונים תחבירית, אך לא היו נכונים מבחינה עובדתית. דוגמה לכך "… Anne Rice and her husband's dog Harry Potter when he met William Shakespeare …".
* repetition\_penalty=1.2, הבחירה נועדה מהמודל לחזור על מלים באופן מיותר ובכך לשפר את הנראות של הטקסט. לפעמים, אכן אין ברירה אלא להשתמש באותה מילה. למשל, במקרה של שם של מישהו.
* num\_return\_sequences = 5, זהו מספר המשפטים שהתבקשנו להפיק.

לאחר ההפקה, התוצאות עוברות decode כדי שהתוצאה תהיה טקסט קריא, ונשמרות בקובץ הטקסט generated\_reviews.txt לפי הפורמט הנדרש.

**שאלות**