מיני פרויקט - עיבוד שפה טבעית

משימה 2: סיכום פסקאות מכתבי הרב קוק באמצעות מודל גנרטיבי

**שאלת המחקר: באיזו מידה מודל גנרטיבי עבור משימות סיכום פסקאות שעבר fine-tuning על בסיס כתבי הרב קוק יחזיר סיכומים המתאימים לאופי כתבי הרב?**

רצינו לבחון את השאלה באמצעות ביצוע fine-tuning למודל גנרטיבי אשר אומן על עברית על מנת לבחון את התוצאות.

ההשערות שלנו הן שתהליך הfine-tuning ישפר משמעותית את איכות הסיכומים לפסקאות שנכתבו על ידי הרב קוק, אך פסקאות אחרות שלא נכתבו במקור בקירוב לאופי כתבם של הרב יסוכמו לרוב בצורה אקסטרקטיבית על ידי שאיבת מילים מנחות מהטקסט לצורך הסיכום, מאשר בצורה אבסטרקטיבית של ניתוח סמנטי של אופי הפסקה וסיכום על פי כן, עקב גודל מקור המידע המצומצם עליו אומן המודל.

**לינקים:**

* [GitHub Repository for project files](https://github.com/LiorLevi15/NLP-MINI-PROJECT)
* [Hugging Face model](https://huggingface.co/ThatGuyVanquish/mt5-small-rabbi-kook)
* [Hugging Face dataset](https://huggingface.co/datasets/NLP-MINI-PROJECT/rabbi_kook)

## סיכום חומר הקריאה:

* הסיכום הנ״ל נכתב לאחר קריאת המאמר ״ROUGE: A Package for Automatic Evaluation of Summaries״ מעת Chin-Yew Lin, מאוניברסיטת דרום קליפורניה.

### ספריית ROUGE להערכת סיכומים

### Recall-Oriented Understudy for Gisting Evaluation

במאמר זה מתארים את החבילה ROUGE, שהיא כלי הערכה אוטומטי עבור סיכום. הכלי כולל שיטות מדידה כמו ROUGE-N, ROUGE-L, ROUGE-W, ו-ROUGE-S כדי לקבוע את איכות הסיכום על ידי השוואתו לסיכומים אידיאליים שנוצרו על ידי אנשים. שיטות המדידה לוקחות בחשבון מספר סוגים של רצפים תואמים, כגון n-gram, רצף מילים וזוגות מילים, בין הסיכום שנוצר על ידי המחשב לסיכומים האידיאליים שנוצרו על ידי אנשים. המאמר הנ״ל מציג תוצאות המראות שיש קשר טוב בין שיטות המדידה השונות להערכות שנתנו בני אדם כאשר ביצעו השוואה בין הסיכום שנוצר על ידי המחשב לסיכומים האידיאליים שנוצרו על ידי אנשים. ההערכה הידנית המסורתית של סיכומים היא יקרה וקשה לבצע בתדירות רבה, לכן כלים אוטומטיים להערכה כמו ROUGE קיבלו הרבה תשומת לב בקהילת המחקר לסיכום.

כעת נפרט בקצרה על כל אחת מהשיטות השונות שמציעה החבילה, על היתרונות והחסרונות של כל אחת מהן והמשמעות של הציון שניתן על ידי כל אחת מהשיטות:

### ROUGE-N

### סטטיסטיקות על בסיס N-grams משותפים - N-gram Co-Occurrence Statistics

משמעות הביטוי N-grams הוא רצפים רציפים של מילים או סמלים או טוקנים במסמך. במונחים טכניים, ניתן להגדיר אותם כ-״רצפים סמוכים של פריטים במסמך״.

בצורה פורמלית, ROUGE-N מחשב את היחס שבין מספר ה-N-grams בסיכום המוערך לבין סיכום (או מספר סיכומים) המשמש כרפרנס, והוא מחושב באופן הבא:

כאשר n מייצג את אורך ה-N-gram ו- הוא המספר המרבי של N-grams המופיעים בסיכום המוערך ובקבוצת סיכומי הרפרנס.

### ROUGE-L

### תת-הרצף המשותף הארוך ביותר - Longest Common Subsequence

שיטה זו ניתן לחישוב הן ברמת המשפט והן ברמת הסיכום כולו באופן הבא:

* ברמת המשפט החישוב עבור 2 משפטים, משפט רפרנס X באורך m ומשפט מוערך Y באורך n הוא

כאשר LCS הוא האורך של תת-הרצף המשותף הארוך ביותר בין X ל-Y ו-.

* ברמת הסיכום כולו, נלקח בחשבון האיחוד של הLCS בין משפט בתוך סיכום הרפרנס ri וכל משפט בסכום המוערך ci. כך שבהינתן סיכום רפרנס באורך u משפטים וסה״כ m מילים ומשפט מוערך באורך v משפטים וסה״כ n מילים, החישוב מתבצע באופן הבא:

כאשר הוא ציון הLCS של איחוד תתי-הרצפים המשותפים.

אחד היתרונות של שיטה זו הוא שהשיטה אינה מצריכה התאמות רצופות אלא התאמות של תתי-רצפים. בנוסף, שיטה זו לוקחת בחשבון את ה-N-gram הארוך ביותר המשותף שנמצא בתוך רצף ולכן אין צורך להגדיר מראש את אורך ה-N-gram.

יתרון נוסף של שיטה זו הוא שהיא לוקחת בחשבון רק התאמות בתוך רצף, כלומר unigrams משותפים נלקחים בחשבון אך ורק אם הם נמצאים בתוך רצף משותף. בצורה זו, ROUGE-L מבטאת בצורה טבעית את ההשוואה במבנה בין הסיכומים ברמת המשפט.

### ROUGE-W

### תת-הרצף המשותף הארוך ביותר הממושקל - Weighted Longest Common Subsequence

ל-LCS יש תכונות טובות, אולם ישנה בעייתיות הבאה לידי ביטוי בכך ששיטה זו אינה מבדילה בין LCSs שונים באותו האורך, שיתכן כי אחד יותר תואם מהשני, כמו בדוגמא הבאה -

X: A B C D E F G

Y1: A B C D H I K

Y2: A H B K C I D

קל לראות ש-Y1 תואם יותר ל-X מאחר והתווים התואמים נמצאים באותו המיקום ובעקביות, בעוד שבY2 אותו רצף נמצא במיקומים שונים, אך ציון ה-ROUGE-L של Y1 וY2 יהיה זהה.

לכן בשיטה הזו, ניתן משקל לכל רצף עוקב תואם, כך שרצפים עוקבים תואמים זוכים למשקל גבוה יותר מאשר רצפים תואמים שאינם עוקבים.

## ROUGE-S

**bigram (צמד מילים) דילוג משותף Skip-Bigram Co-Occurrence** **Statistics**

בשיטה זו נעשה שימוש במונח bigram דילוג. משמעותו הוא צמד כלשהו של מילים במשפט וביניהם מילים עליהן מדלגים (למשל במשפט ״דני הלך לגן החיות״ יש את ה-bigrams ״דני לגן״, ״דני החיות״, ״הלך החיות״ וכו׳). בשיטה זו מודדים סטטיסטיקות בנוגע לbigrams תואמים בשני סיכומים בצורה הבאה:

כאשר SKIP2 הוא מספר ה-bigrams דילוג התואמים בין X ל-Y, ו-C מודד את מספר הקומבינציות.

אחד היתרונות של שיטה זו הוא שהיא אינה דורשת התאמות עוקבות אך עדיין רגישה לסדר המילים.

בנוסף, בהשוואה לLCS, שיטה זו סופרת את כל תתי-הרצפים של זוגות מילים במשפט בעוד ש-LCS סופרת רק את הרצף הארוך ביותר.

### ROUGE-SU

### הרחבה לשיטה הקודמת

אחת הבעיות בשיטה הקודמת היא ששיטה זו אינה נותנת קרדיט למשפט אם במשפט זה אין כלל צמד מילים תואם למשפט הרפרנס. למשל אם המשפט הפוך לגמרי בסדר המילים ממשפט הרפרנס הרי שאין לו אף צמד מילים תואם אך זהו בעצם כמעט אותו משפט ולכן נרצה להבדילו ממשפטים אחרים אשר להם אין מילים תואמות בכלל, ולא רק כאלה ללא רצפים תואמים. השיטה הנ״ל נותנת אפשרות זו ע״י כך שהיא סופרת גם unigrams - כלומר סופרת גם מילים בודדות תואמות בין המשפטים.

## ממצאי המאמר

במאמר זו מפורט ניסוי שבוצע על בסיס השיטות המפורטות לעיל ומספר מסדי נתונים בהם טקסטים רבים וסיכומים שנכתבו על ידי תוכנה ועל ידי בני אדם, וציוני הערכה שניתנו על ידי בני אדם.

לפי ממצאי ניסוי הזה, נמצא השיטות ROUGE-2, ROUGE-L, ROUGE-S ו-ROUGE-W (עם פקטור משקל 1.2) עובדות היטב עבור משימות של סיכום מסמך יחיד. בעוד ש-ROUGE-1, ROUGE-L, ROUGE-W, ROUGE-SU4 ו-ROUGE-SU9 (משמעות המספר <SU<num הוא אורך הדילוג המקסימלי) פעלו היטב עבור משימות של סיכום טקסט במשפט קצר (כמין כותרת לטקסט).

לסיכום, מאמר זה מציג כיצד ספריית ROUGE משמשת לביצוע הערכת אוטומטית ואפקטיבית של סיכומים.

* הסיכום הנ״ל נכתב לאחר קריאת המאמר ״The Pyramid Method: Incorporating Human Content Selection Variation in Summarization Evaluation״ מעת ANI NENKOVA, REBECCA PASSONNEAU, ו-KATHLEEN MCKEOWN, מאוניברסיטת קולומביה.

# שיטת ה-Pyramid לערכת סיכומים

המאמר מציע את שיטת הפירמידה כשיטת דירוג כדי לטפל בשונות שבבחירת תוכן בסיכום שנכתב על ידי אדם, שכן כאשר אדם ניגש למטלת סיכום הוא ברוב המקרים מסכם את החומר הנדרש בצורה אבסטרקטיבית ולא בצורה אקסטרקטיבית, דבר הגורם לשוני רב בין סיכום של אדם אחד למשנהו, ולעיתים אף בין 2 סיכומים של אותו אדם שנכתבו בזמנים שונים. השיטה נותנת דגש ליחידות סמנטיות של התוכן ומקצה להם משקל חשיבות אמפירי, ומאפשרת לכמת את השונות האנושית שבבחירת תוכן הסיכום אך עם זאת הייצור של סיכומים הזהים במידת המידע שהם מעבירים. המאמר מדבר גם על המהימנות של הגדרת יחידות התוכן ועל היחס בין ציוני הפירמידה לשיטות דירוג אחרות.

המטרה העיקרית בעבודה זו היא להתמודד עם בעיות הבדיקה וההערכה של סיכומים אוטומטיים בהשוואה לסיכומים אנושיים. כיום הגישה המקובלת היא להשוות בין הסיכומים האוטומטיים לסיכומים שנכתבו על ידי אנשים ולהשתמש במדדים כמו דיוק וזיהוי של משפטים חשובים. אולם, התגלו כמה בעיות כגון התלות בקבוצת האנשים וההבדלים בבחירת המשפטים החשובים, הבעיה במורכבות הנדרשת לניתוח של המשפטים וכן השאלה האם משפטים שונים בתוכן או עושים שימוש במילים שונות יכולים להיות שווים זה לזה מבחינה סמנטית כולם מעוררים שאלות נוספות לגבי הדרכים הנכונות לבדוק ולהעריך סיכומים. שיטת הפירמידה המתוארת במאמר זה מאפשרת לטפל בבעיות אלו ומציעה מדדים עבור הכרת חשיבות התוכן ומניעת תלות בקבוצת אנשים ספציפית.

הערכה כמותית של מודלי סיכום באמצעות בחירת תוכן מצריכה ״תקן זהב״ על בסיסו יהיה ניתן להשוות מודלים אוטומטיים שונים. הפירמידה היא תצוגה של סיכום העומד בתקן עבור קבוצת מסמכים. מאחר והפירמידה משמשת להערכה של תוקן הסיכום, בניגוד להערכה של המילים בסיכום, יחידות ההשוואה בפירמידה הן יחידות של משמעות והן נקראות ״(Summary Content Units (SCUs״. שיטת הפירמידה מייצגת דעות של מגוון כותבי סיכום (בני אדם) שכל אחד מהם כתב סיכום עבור קבוצת המסמכים. אחד התכונות המרכזיות בשיטה זו היא שהיא מכמתת את ההסכמה של כל המסכמים האנושיים - SCU שמופיע ביותר סיכומים אנושיים יקבל משקל גבוה יותר.

המאמר מגדיר את ה-SCUs כיחידות סמנטיות שאורכן משתנה אך לא עולה על משפט. ה-SCUs נובעות מתוך אנוטציות של קבוצת סיכומים אנושיים של אותו המסמך, ומוגדרות כל אחת על ידי ציון המידע שהן מציגות. במהלך התהליך, מומחים מסווגים SCUs במילים שלהם ויכולים לשנות את המשמעות של SCU תוך כדי התהליך. משפטים המתאימים למידע המופיע רק בסיכום אחד מחולקים לביטויים, כל אחד

מתוכם SCU אחד בפירמידה. משקלים משויכים לכל SCU ומציינים את מספר הסיכומים שבהם הוא הופיע.

כעת נציג דוגמא שניתנה במאמר עצמו התורמת להבנת השיטה:

1. The industrial espionage case involving GM and VW began with the hiring of Jose Ignacio Lopez, an employee of GM subsidiary Adam Opel, by VW as a production director.
2. However, he left GM for VW under circumstances, which along with ensuing events, were described by a German judge as “potentially the biggest-ever case of industrial espionage”.
3. He left GM for VW in March 1993.
4. The issue stems from the alleged recruitment of GM’s eccentric and visionary Basque-born procurement chief Jose Ignacio Lopez de Arriortura and seven of Lopez’s business colleagues.
5. On March 16, 1993, with Japanese car import quotas to Europe expiring in two years, renowned cost-cutter, Agnacio Lopez De Arriortua, left his job as head of purchasing at General Motor’s Opel, Germany, to become Volkswagen’s Purchasing and Production director.
6. In March 1993, Lopez and seven other GM executives moved to VW overnight.

תהליך הסיווג מתחיל בזיהוי משפטים דומים, כמו השישה לעיל, וממשיך בזיהוי חלקי משפטים שיותר זהים אחד לשני. ניתן לראות 2 SCUs (בין היתר, בסופו של דבר כל החלקים בכל משפט מסוגגים לSCUs) במשפטים לעיל:

1. SCU1 (w=6): Lopez left GM for VW
   1. the hiring of Jose Ignacio Lopez, an employee of GM ... by VW
   2. he left GM for VW
   3. He left GM for VW
   4. recruitment of GM’s ... Jose Ignacio Lopez
   5. Agnacio Lopez De Arriortua, left his job ... at General Motor’s Opel ... to become Volkswagen’s ... director
   6. Lopez ... GM ... moved to VW
2. SCU2 (w=3) Lopez changes employers in March 1993

C. in March, 1993

E. On March 16, 1993

F. In March 1993

ניתן לראות כיצד יחידות בעלות משמעות זהה כמעט לחלוטין נתנות לביטוי בכל כך הרבה דרכים שונות ובאמצעות מילים שונות. בנוסף המומחים יכולים להבין הרבה מן המשמעות של משפטים אלו כיוון שיש להם קונטקסט - למשל כאשר באחד המשפטים יש התייחסות לאדם מסויים, המומחים יודעים במי מדובר כיוון שישנו קונטקסט. כל SCU מקבל משקל לפי כמות הסיכומים שהוא מופיע בהם. לכל SCU יש סיווג קונטקסטואלי בשפה הטבעית, שניתן לשינוי תוך כדי תהליך הסיווג, והוא משמש על מנת לשחרר את תהליך הסיווג מהתלות בשפת ייצוג סמנטית. בנוסף, סיווג זה מצריך את המומחים המסווגים להיות מודעים למשמעות הספציפית המשותפת לכל המסכמים ולבסוף, מכיוון שהSCUs מוצאים מהקשרם, הסיווג עוזר להזכיר את הקונטקסט של היחידה.

כאשר תהליך הסיווג הסתיים, ניתן לקחת את כל יחידות הSCU ולסדרם בחלוקות במבנה של פירמידה על בסיס המשקל של כל יחידה. כל שלב בפירמידה מכיל את כל היחידות מאותו המשקל. מספר הסיכומים לאותו מסמך הוא גם מספר השלבים המקסימלי לפירמידה המייצגת את סיכומי מסמך זה. מכאן שככל שהשלב נמוך יותר בפירמידה, כך היחידות הנמצאות בו פחות משמעותיות עבור סיכום איכותי של המסמך.

לאחר סידור זה ניתן לייצר ציון עבור סיכום נתון אשר מבטא את מספר הSCUs אשר הוא מכיל מתוך אלו הנמצאות בפירמידה, כמו גם את איכותן.

אחת המסקנות של מאמר זה היא כמות הסיכומים הדרושים עבור מסמך יחיד על מנת לייצר ציון פירמידה אשר מבטא בצורה נכונה את איכות הסיכום. ראשית, קל להבין כי כאשר נתון סיכום יחיד, לא ניתן ליצור פירמידה עבור המסמך, מאחר ולכל יחידת SCU יש משקל 1, ולכן משמעות הציון מתבטאת אך ורק בהשוואה סמנטית שבין סיכום אנושי לבין סיכום אוטומטי של מודל. בנוסף לכך, לפי ממצאי המחקר שנעשה כחלק מכתיבת המאמר, על מנת להציג ציון אשר אינו מושפע מבחירת הסיכומים לייצור הפירמידה, יש צורך ב-5 סיכומים לפחות לכל מסמך.

# תיאור השיטות שנבחרו:

לצורך העבודה על הפרויקט חילקנו את העבודה לשלושה חלקים:

### חלק ראשון – עבודה על ה-Dataset:

קיבלנו את כתבי הרב קוק ואת סיכומי הכתבים בצורה אשר מקשה על פרסור הקבצים במקביל לטובת יצירת Dataset שמכיל 3 קטגוריות: תגית מספרית (ID), פסקה וסיכום הפסקה הרלוונטית.

על מנת להקל על עבודת הפרסור ערכנו את הקבצים במעט כך שיתאימו למבנה:

פסוק (ID) <טאב> פסקה/סיכום

קבצים אלו שמורים בgithub של הפרויקט תחת התיקייה [raw\_files](https://github.com/LiorLevi15/NLP-MINI-PROJECT/tree/master/DATA/raw_files).

לאחר עבודה על הקבצים כך שיתאימו לפירסור, כתבנו סקריפט פייתון ([parser.py](https://github.com/LiorLevi15/NLP-MINI-PROJECT/blob/master/DATA/parser.py)) שתפקידו לפרסר קובץ אחר קובץ לתוך קבצי JSON בהם 3 הקטגוריות של כל רשומה בdataset, כל הקבצים המפורסרים נמצאים תחת התיקייה [parsed\_files](https://github.com/LiorLevi15/NLP-MINI-PROJECT/tree/master/DATA/parsed_files).

לאחר מכן לקחנו את כל קבצי ה-JSON ואיחדנו אותם ל-3 קבצים המגדירים את ה-Dataset והם Train, Test ו-Validation, אותם ניתן למצוא בתיקיה [parsed\_data\_splits](https://github.com/LiorLevi15/NLP-MINI-PROJECT/tree/master/DATA/parsed_data_splits).

ה-Dataset נמצא כעת ב Hugging Face Hub[[1]](#footnote-0).

### חלק שני – עבודה על המודל:

תחילה בחרנו מודל קיים על מנת לבצע על בסיסו את תהליך הfine-tuning.

לצורך כך, בהתאם להנחיות של מני, ניסינו ליצור מודל Sequence to Sequence מהמודל הקיים alephbert, ולעבוד במקביל למדריך לSummarization- מאתר Huggingface[[2]](#footnote-1) על מנת לאמן את המודל באמצעות הDataset שבנינו.

לאחר נסיונות כושלים לאימון המודל באמצעות alephbert קיבלנו ממני גישה למודל אחר שאומן על עברית, BEREL, ועל כן מכיוון שכמו alephbert המודל BEREL אינו מודל Seq2seq נאלצנו לנסות להגדיר ממנו מודל Seq2seq על מנת לאמן אותו למשימות סיכום.

כאשר גילינו קשיים בקבלת תוצאות כלשהן מהמודל שנבנה הן מalephbert והן מBEREL, ולאחר בדיקה של מני מול יוצרי המודלים שהעלתה כי לא ניתן לייצר מהם מודל Seq2seq, הופננו לעבוד בעזרת המודל MT5 שגם כן אומן על עברית.

בניסיון לבצע fine-tuning ל-MT5 באמצעות הDataset של כתבי הרב קוק שיצרנו נתקלנו במספר קשיים, שכן נראה כי המודל לא מצליח לייצר תוצאות טובות כלל. לאחר התייעצות עם מתן אייל מהמחלקה הופננו לאמן את המודל עם ניתור באמצעות wandb[[3]](#footnote-2).

באמצעות העבודה עם הכלי wandb גילינו שהמודל שלנו לא מתאמן כשורה, ועל כן כל ניסיון אימון בעזרת MT5 כשל. בעקבות שיחה עם מתן גילינו שהבעיה נובעת ממימוש לא מתאים של הפונקציה preprocess\_function.

לאחר חיפוש פתרון מצאנו מדריך אחר[[4]](#footnote-3) אשר עבד לצורך סיכומים באמצעות MT5, ועל כן זהו הקוד שעל בסיסו אימנו וביצענו fine-tuning לMT5 באמצעות הDataset שבנינו מפסקאות וסיכומי הרב קוק.

האימון התבצע ב-3 epochs שעבדו בbatch של 2, ולקח 5352 צעדים. בנוסף, מספר הפרמטרים הניתנים לאימון הוא 300176768.

האימון לקח שעתיים ו-22 דקות, כאשר בכל שניה אומנו 1.469 דוגמאות, עם loss של 3.4519.  
תהליך הוידוא לקח 380 שניות, בו וודאו 1.2 דוגמאות בשניה עם loss של 2.76773.

### חלק שלישי – עבודה על האתר

את האתר בנינו כאפליקציית Flask עם שרת הכתוב בפייתון ומשתמש בשני הממשקים לעיל על מנת לקבל שאילתות מהמשתמשים ,מחזיר תשובות בהתאם ושומר את הדירוג של המשתמש על הסיכום המתקבל.

קיימים שני סוגים של api אותם כתבנו:

1. ממשק המשמש לשליחת שאילתות למודל השמור ב-huggingFace וקבלת התשובה מהמודל.
   1. הממשק מממש את האובייקט query המקבל טקסט בבנאי ומשתמש בשיטה ()query.query על מנת לקבל סיכום של הטקסט.

1. ממשק המשמש לשמירת המידע כולו בDB של FireBase וחילוץ המידע, הממשק מממש אובייקט Database ולו השיטות הבאות:
   1. שמירת רשומה חדשה - (uploadData(paragraph, result, השיטה מחזירה recordId.
   2. קבלת על הרשומות שנשמרו עד כה - getData()

**הרצת האתר**

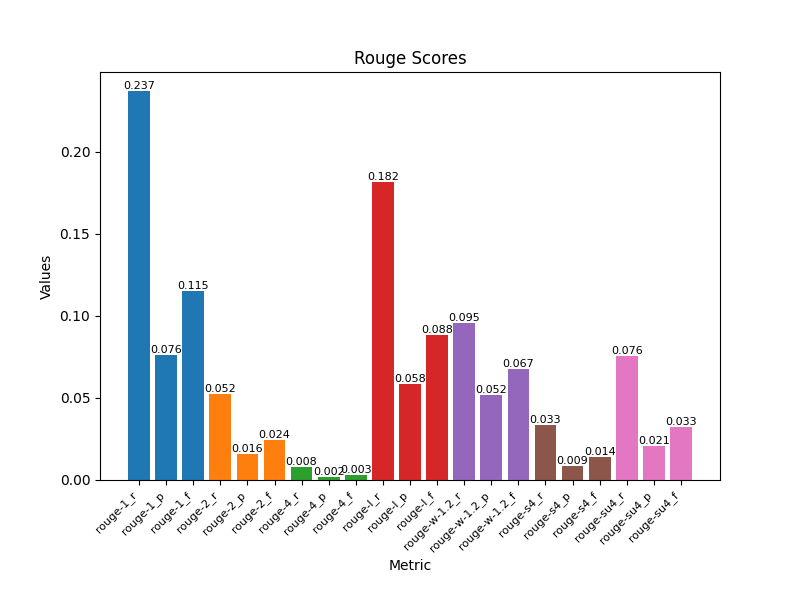
1. יש לשנות את הקובץ FirsestoreAPI שיצביע על הקובץ עם ההרשאות המתאימות לדאטה בייס של המשתמש (firebase)
2. יש לשנות את המשתנים - q1,q2,q3,q4,q5 לשאלות הרלוונטיות שאותו תרצה לשאול לגבי הסיכום שחוזר מהמודל.
3. כשנכנסים לאתר יש לשים את הטקסט שאותו רוצים שהמודל יסכם.
4. יש להכניס פרטים אישיים ולדרג את הסיכום לפי השאלות שנשאלות.

# תיאור נתוני ההערכה של התמצות:

לשם הערכת המודל, נעזרנו במאמרים שקראנו על שיטות הערכה - [ROUGE](https://aclanthology.org/W04-1013.pdf), [Pyramid](https://repository.upenn.edu/cgi/viewcontent.cgi?referer=&httpsredir=1&article=1771&context=cis_papers), בשיטות ההערכה השונות שכל מאמר מציע ובמסקנות מהניסויים שפורטו במאמרים.

החלטנו לבצע הערכה על בסיס מספר שיטות Rouge אשר נמצאו כמתאימות ביותר להערכת סיכום של מסמך יחיד (מין פסקה) - השיטות ROUGE-2, ROUGE-L, ROUGE-S ו-ROUGE-W (עם פקטור משקל 1.2). נוסף על כך החלטנו להשתמש גם בשיטות ROUGE-1, ROUGE-4 ו-ROUGE-SU4.

לצורך כך נעזרנו בספרייה PyRouge המציע את כל שיטות ההערכה השונות שיש למאמר להציע, וחישבנו עבור כל מטריקה גם את ערך הrecall וה-percission. את הקוד מאחורי הגרף הנ״ל ניתן למצוא בתיקייה [ROUGE SCORE](https://github.com/LiorLevi15/NLP-MINI-PROJECT/tree/master/ROUGE%20SCORE).

להלן התוצאות:

ניתן לראות שכל התוצאות נמצאות מתחת ל-0.3, מה שנותן את התחושה שביצועי המודל אינם טובים, אך חשוב לזכור כי מדובר בהשוואה אל מול סיכום אנושי אחד ויחיד, ויתכן כי בהינתן סיכומים נוספים, תוצאות המודל היו עולות. ובכל אופן, קרוב לוודאי שעל מנת להגיע לביצועים טובים יותר יש לאמן את המודל באמצעות dataset גדול משמעותית מזה הקיים בידינו.

את ההערכה על בסיס שיטת הפירמידה החלטנו שלא לבצע במפורט המאמר, זאת מאחר וממצאיו מציגים בצורה ברורה את הצורך בלפחות 5 מסכמים אנושיים ו-2 מומחי לשון לצורך סיווג ובניית פירמידה גדולה דיה. לחלופין, החלטנו להוסיף לאתר את האפשרות לענות על מספר שאלות הבאות לבטא את דעתו של המשתמש ביחס לסיכום שהתקבל על ידי המודל עבור הפסקה אותה הכניס המשתמש. שאלות אלו נתונות לשינוי ומטרתן לספק פידבק של המשתמש על הסיכום שחזר מהמודל.

את ממצאי הדירוג הזה יהיה ניתן לבחון כאשר יתקבלו מספיק ביקורות באתר.

# מסקנות:

במהלך העבודה על הפרויקט עלו מספר מסקנות, בנוסף למסקנות שהתקבלו לאחר שאימון המודל הסתיים וממצאיו התקבלו:

1. לפי הסיכומים אותם בחנו וגם לפי ציוני ה-rouge הגענו למסקנה שיש עוד מקום לשיפור על מנת להגיע למודל המסכם בצורה טובה את פסקאות הרב קוק, אך עם זאת, בהתחשב בכמות הדאטה היחסית דלה שהייתה ברשותינו, אנו מרוצים מהתוצאות.
2. בהמשך למסקנה הראשונה, על מנת להגיע לתוצאות טובות באימון מודל מסוג טרנספורמר, יש להגדיל משמעותית את גודל הדאטה סט.
3. אחת ההשערות שעלו במהלך אימון המודל היא שהמודלים הקיימים היום, אמנם מותאמים להמון שפות ובניהם גם לעברית, אך לא אומנו על מספיק דאטה בעברית, מה שמקשה על פעולת הfine-tuning למימוש משימת סיכום תוכן בעברית, על אחת כמה וכמה כאשר מדובר בשפה עתיקה כמו שבכתבי הרב קוק.
4. קראנו את המאמרים בנושא הערכת סיכומים והשוואת סיכומים אנושיים לסיכומים אשר יוצרו על ידי מודל, והגענו למסקנה ששיטת הפירמידה אינה מתאימה למחקר בהיקף שלנו, מאחר והיא מצריכה כוח אדם מומחה רב, אך עם זאת מאוד מתאימה כאשר אנו רוצים לדעת האם הסיכום מביא לידי ביטוי בצורה טובה את הפסקה אותה הוא מסכם, ולא רק בהשוואה לסיכום אנושי אחד יחיד. בניגוד לכך, שיטת rouge הינה קלה למימוש ויעילה, אך לא תמיד מדויקת ולא בהכרח תדע להעריך סיכום טוב שאינו תואם בתוכנו את הסיכום האנושי אלא רק מבחינה סמנטית.

1. [Rabbi Kook Dataset](https://huggingface.co/datasets/NLP-MINI-PROJECT/rabbi_kook) [↑](#footnote-ref-0)
2. [Hugging Face Summarization tutorial](https://huggingface.co/docs/transformers/tasks/summarization) [↑](#footnote-ref-1)
3. [wandb](https://wandb.ai/) [↑](#footnote-ref-2)
4. [The tutorial for summarization](https://github.com/huggingface/transformers/tree/main/examples/pytorch/summarization) [↑](#footnote-ref-3)