**דוח חלק ג'**

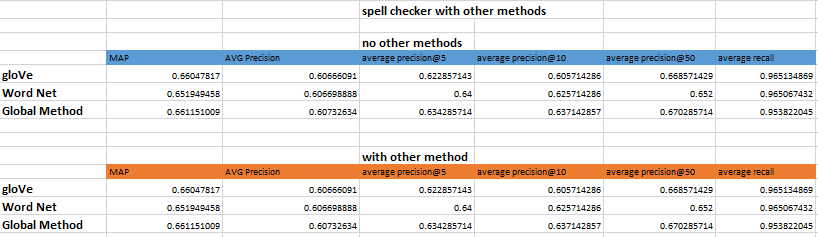
**שאלה 1**

בחלק זה, ניסינו שיטות שונות על מנת לשפר מדדים לאחזור המנוע. (כמו MAP, average recall, precision@n). השתמשנו בשיטות הבאות:

1. **GloVe** – מודל זה מסתכל על יחסי ההסתברויות המשותפות למילה בין מילים אחרות אשר יכולות להגיד לנו את הפוטנציאל של הקירוב ביניהן. השתמשנו בספריית genism. בעת עליית המנוע, ייבאנו מודל מאומן של אוקספורד (אומן על טוויטר) שמכיל 25 מימדים. בעת הרצת שאילתא, נרצה להרחיב אותה למילים נוספות אשר נמצאות אצלנו במילון. עבור כל מילה בשאילתא, ניצור לה וקטור שקיבלנו מהמודל המיובא, ונבדוק אותה מול שאר המילים אשר נמצאות במודל. המילה הכי קרובה אליה (בהנחה שזו לא אותה מילה) תיבדק במילון שלנו, ואם היא קיימת, נכניס אותה לשאילתא.
2. **Global method** – מודל זה בונה מטריצה של K המילים התדירות ביותר בזמן בניית המנוע, ובודק את ההסתברות שמילה הופיעה עם מילה אחרת.

בעת עליית המנוע, יצרנו מטריצה של 1330X1330. בעת הרצת שאילתא, נרצה להרחיב אותה למילים נוספות אשר נמצאות אצלנו במילון. עבור כל מילה בשאילתא, נמצא לה את המילה שמופיעה איתה הכי הרבה פעמים ונוסיף אותה לשאילתא.

1. **Spelling Correction** – מודל זה בודק אם כתבנו מילה בשאילתא שלנו אשר שלא נמצאת במילון ואולי התכוונו למילה אחרת או שנכתבה בצורה לא נכונה, מתקנת את המילה למילים קרובות שנמצאות במילון שלנו ומחזירה תוצאות על פי התיקון. ייבאנו את ספריית SpellChecker שנמצאת בpySpellChecker בכדי שנוכל להשתמש במאגר המילים שיש שם. מודל זה עובד בזמן ריצה. ניסינו להריץ מודל זה בשילוב עם כל אחת מהשיטות שביצענו כדי לבחון את טיב האיכות. אלו התוצאות שהגענו אליהן:



כפי שניתן לראות, כל המדדים שווים זה לזה. עבור מילה שלא נמצאת בשאילתא, ביצענו לה spell correction ואם לאחר מכן המילה הופיעה, אז הכנסנו אותה לשאילתא החדשה שמרחיבה את המילים. לאחר מכן הרחבנו את השאילתא עבור כל אחד מהמודלים.

1. **Word Net** – מודל זה מאופיין בזה שהוא מקשר מילים בצורה סמנטית ולקסיקלית, כך שלכל מילה יש לנו קשר למילים שהן נרדפות לה או דומות לה סמנטית. ייבאנו את ספריית wordnet(שנמצאת בnltk) בכדי שנוכל להשתמש במאגר המילים שיש שם. בזמן הרצת שאילתא, המודל מחזיר עבור כל מילה, את המילים הקרובות אליה סמנטית. אנו בודקים אם מילים אלה נמצאות במילון שלנו, ואם כן, מכניסים אותן לשאילתא.
2. **Advenced** **Parser** – בשיטה זו הרחבנו את הפרסר שלנו בחוקים נוספים אשר מחזירים מילים נוספות שבעבר הוצאנו אותן מהמילון או שהכנסנו אותן בצורה פחות תקינה. הסבר מפורט בשאלה 2.

**שאלה 2**

בחלק זה החלטנו להוריד חוקים שהיו קיימים בעבר וראינו שאין בהם שימוש ברוב המקרים, והחלטנו להוסיף מספר חוקים כדי לשפר את איכות הפרסור.

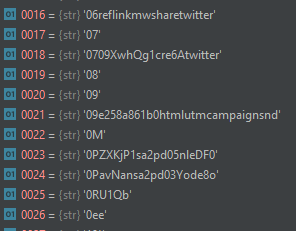
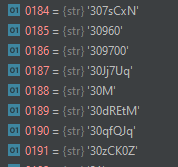
החוקים החדשים של הפרסר שלנו הם:

* **הורדת Hashtag** – דבר זה נובע מהעובדה שלרוב לא נרצה לחפש ממש את ההאשטג עצמו, אלא את התוכן שלו. בנוסף, נניח ונריץ שאילתא עם ההאשטג, אז בפרסור של השאילתא, נפרק אותו למילים, וכך הtrade off בין כמות המילים במילון שבחלק זה ויתרנו עליהן לבין דיוק השאילתא יהיה מובן. לכן, שמרנו את התוכן שלו במילון אך את ההאשטג עם המילה לא הכנסנו. לדוגמא: #DonaldTrump

בחלק א - #DonaldTrump, Donald, Trump

בחלק ג – Donald, Trump.

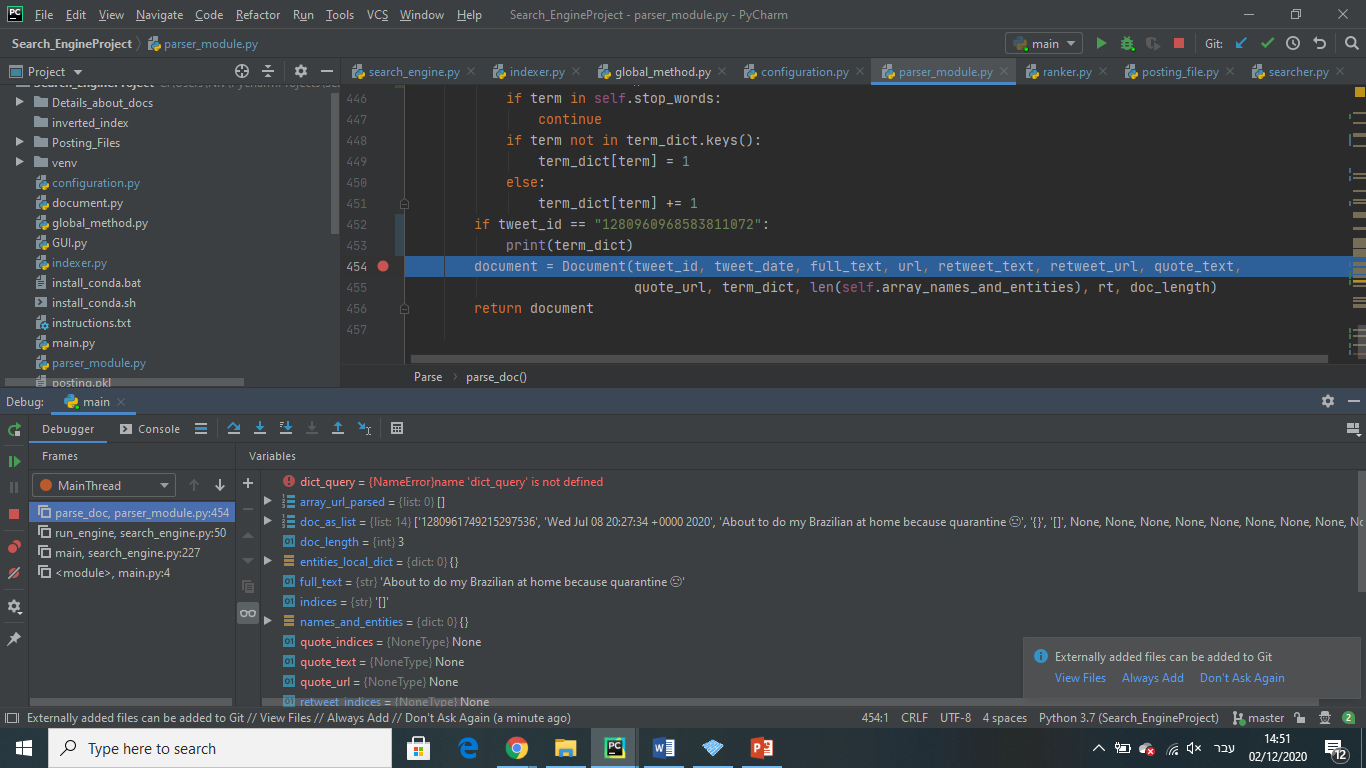
* **הורדת כתובות URL** – כל המוסיף גורע. לאחר בדיקה, ראינו שכמות המילים הלא רלוונטיות שנכנסות למילון באמצעות פרסור כתובות URL הוא גדול, והוא מגדיל משמעותית את גודל המילון שלא בצדק. החלטנו להוריד את פרסורו כדי לחסוך את זמן הפרסור והכנסת מילים רלוונטיות יותר למילון שכן אולי נקבל איכות תוצאות טיפה גבוהות יותר, אבל זמן פרסור כתובות URL הוא ארוך בקורפוס גדול ולא בהכרח יניב תוצאות איכותיות. כמה דוגמאות של מילים שנכנסו למילון:

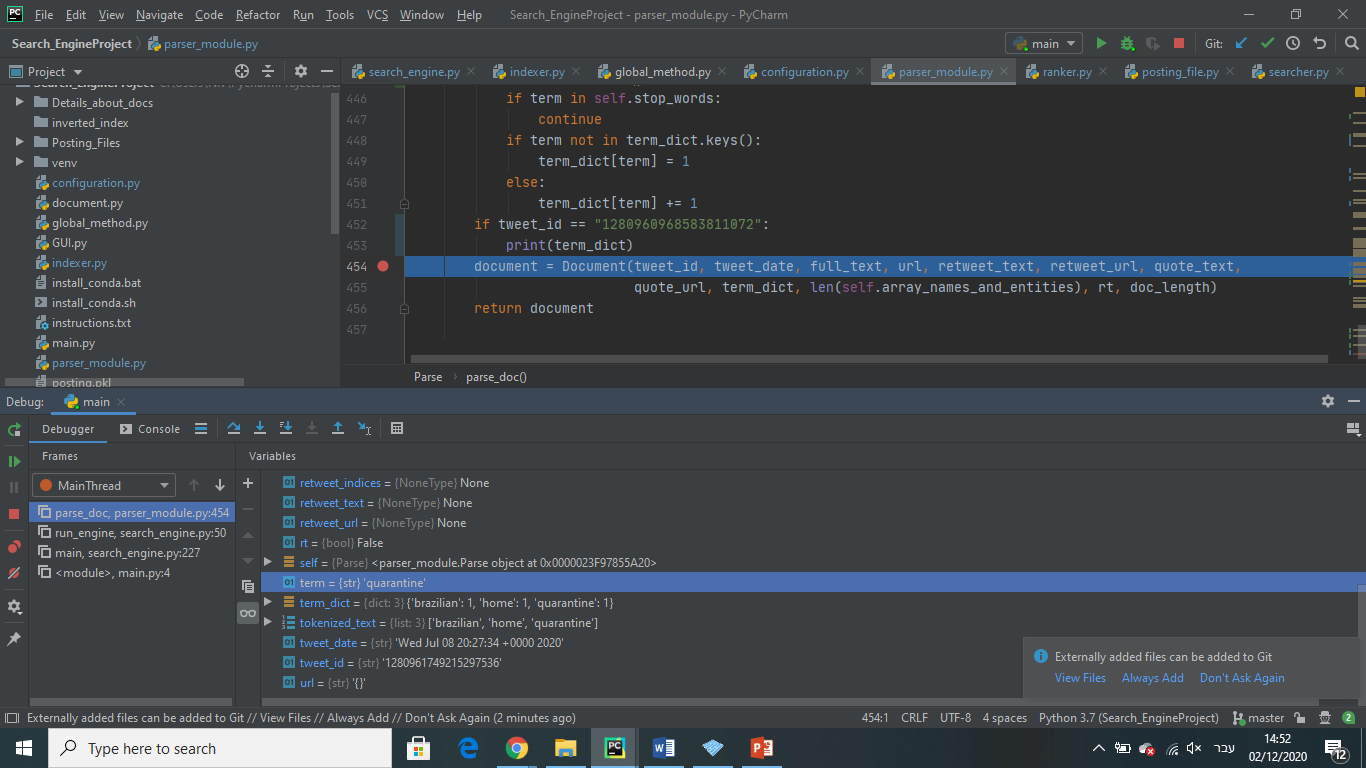
 

ישנם עוד מקרים ב-200 terms הראשונים שהוצגו לעיל שהוכנס ערך לא יעיל למילון.

במקרה שלנו כמות המילים שנכנסו למילון עם פרסור URL גדול בכ-700 .

* **הורדת אימוג'ים וסמיילים** – הורדנו את האימוג'ים והסמיילים שנמצאים בכל חלקי המשפט על מנת למנוע בעיות בפרסור הטקסטים ולחסוך מקום במילון שלנו עם מילים זהות. למשל אותה מילה מופיעה בשני מקומות שונים, ובאחד מהמקומות המילה מחוברת לאימוג'י. אז במקום להכניס אותה פעמיים למילון כאילו הן שתי מילים שונות, הורדנו את האימוג'י כי כאשר מחפשים שאילתא, בדר"כ היא ללא אימוג'י. לדוגמא:



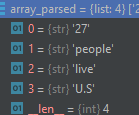


ניתן לראות שהאימוג'י לא נכנס למילון הלוקלי.

* **פירוק מילים המורכבות ממספרים ואותיות** – בחלק א שמנו לב שנכנסו לנו המון מילים שמחוברות כמו 27people, 100times. דבר זה מנע מאיתנו אחזור טוב יותר מפני שאנחנו מפספסים פה אותיות ומילים שאנחנו לא מחשיבים אותם בחישובים שלנו. בחלק ג פיצלנו מילים אלה ל-K מילים נפרדות. (בדוגמאות: 27 people, 100 times).

עבור השאילתא:

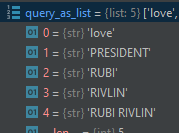
 יוחזר לנו:



* **הכנסת 100,000 מילים עם התדירות הגבוהה ביותר** – כאשר נבנה מילון מקורפוס גדול, ונרצה לנסות לייעל את הinverted index שלנו, אם נכניס מילים ללא רף מסוים, אז אולי תוצאות השאילתות יצאו יחסית מדויקות, אך גודל המילון ויצירתו ייקחו זמן רב ועל כן נצטרך לבצע trade-off כאשר נגביל את המילון לגודל מסוים של המילים בתדירות הכי גבוהה. דבר זה לא ישפר לנו את דיוק השאילתא אבל יחסוך לנו מקום רב .
* **הכנסת מילים עם נקודה בסוף משפט או מילים המכילות בתוכן נקודה** – בחלק א' שמנו לב שקיימות מילים רבות שנכנסות למילון עם נקודה שלא לצורך, כמו מילה אחרונה במשפט, אחריה יש נקודה ואז ללא רווה יש את המשפט החדש כבר. אז זה קורא אותו בתור word.word. על אחת כמה וכמה אם יש לי משפט שבסופו אותיות גדולות ובתחילת המשפט גם אותיות גדולות כמו :

I love the PRESIDENT.RUBI RIVLIN is…..

משפט זה לדוגמא ייקח את PRESIDENT.RUBI RIVLIN בתור יישות (עם אותיות גדולות) ודבר זה מונע מאיתנו תוצאות אחזור טובות יותר. דוגמא לפרסור המשפט:



חלק מהמילים יורדות בstop\_words. הוספנו את הישות כדי להמחיש את הפרסור לאחר השאילתא.

דוגמא נוספת: השם U.S מופיע רבות בקורפוס. בחלק א', כשהכנסנו את U.S למילון, הוא נוסף בתור US (ולאחר מכן שונה לus כי הופיע בצורה u.s). בחלק זה של הפרויקט הוספנו אותו כולל הנקודה באמצע כדי להבדיל בין U.S לבין us (לנו, איתנו)

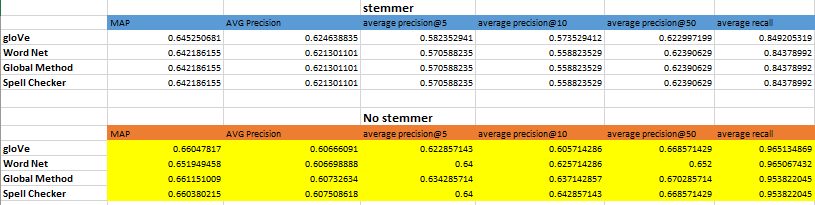
חלק א': חלק ג':

**שאלה 3**

**ללא** סטמר קיימות במילון שלנו: כ- 8600

**עם** סטמר קיימות במילון שלנו: כ- 6600



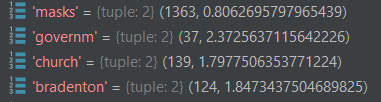
בטבלה למעלה – תוצאות המדדים עם סטמר. בטבלה למטה – תוצאות המדדים ללא סטמר.

בצהוב – השוואה בין אותו מדד עם אותה המתודה בין שתי הטבלאות. ניתן לראות שכל המדדים שלנו איכותיים יותר כאשר אנו לא משתמשים בסטמר.

**שאלה 4**

להלן מבנה ה- inverted indexוה-posting dictionary :

Inverted index :

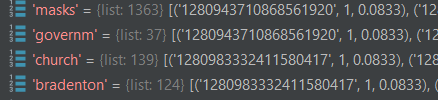


מבנה ה – inverted index שלנו שמור בתוך מבנה נתונים מסוג "מילון". כל כניסה במילון מהווה term ייחודי במילון. בכל בכניסה של term במילון ישנו tuple אשר בתוכו קיימים : מספר הופעות בקורפוס (tf) ו – idf .

בכל tuple בחרנו לשמור את נתונים אלו מכוון שבעת סיום יצירת ה – Inverted index, עלינו לחשב את ה-idf.

חישוב ה- idf :

Posting dictionary :

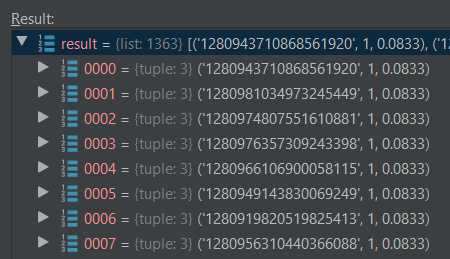


מבנה ה- posting dictionaryשלנו שמור בתוך מבנה נתונים מסוג "מילון" . כל כניסה במילון מהווה term ייחודי. בכל כניסה של term במילון, ישנה רשימה שמכילה בתוכה מספר tuple כמספר ההופעות של המילה בקורפוס. כל tuple מכיל בתוכו פרטים על הופעות של ה-term בקורפוס : tweet\_id, tf ו- תדירות המילה במסמך. להלן הנוסחה:

בחרנו למקם את תדירות המילה במסמך בקובץ ה- posting file מכוון שבעת אחזור המסמכים וחישוב ה-inner product נעשה שימוש בערך זה. בנוסף, שמרנו את ה- tweet id בPosting dictionary בposting file מכוון שבעת האחזור נרצה להימנע מהחזרת מסמכים כפולים ברגע האחזור.

כעת נעמיק ב-term הנמצא בתוך מילון ה- posting .

לדוגמא, המילה masks



בכניסה של המילה masks במילון ה- Posting ישנה רשימה שמכילה את כל הטוויטים בהם הופיעה המילה והנתונים שהוזכרו לעיל. כאמור, בכל תא ברשימה ישנו tuple כפי שהוסבר מעלה.

בחרנו במבנה זה של ה – Inverted index וה – posting dictionary מכוון שעל פי התיאוריה, בכל תא ב – inverted index ישנו pointer לעבר קובץ posting file אשר מכיל בתוכו רשימה של : מספר המסמך, כמות הופעות של המילה במסמך וחישוב התדירות שבו(שמסייע לאחזור המסמכים בהמשך התהליך).

מאחר ובחלק זה התבקשנו לשמור את שני המילונים בזיכרון של המחשב החלטנו להפוך כל מילה (כפי שהיא) למצביע במילון הposting והתא הרלוונטי בו. כלומר , בעזרת המילה ניתן להגיע למסמכים הרלוונטיים במילון ה- posting ולבצע אחזור מהיר יותר, כך שהמילה היא בעצם מפתח ב- posting dictionary וכך נקבל את הפרטים המבוקשים להמשך החישובים והאחזור עצמו.

**שאלה 5**

בחלק זה ביצענו מספר פעולות בכדי להביא לזמן ריצה מיטבי. ביצענו מספר תיקונים ועדכונים אשר הביאו לזמן ריצה יעיל יותר . לאחר קבלת ציון חלק א, הסקנו כי מודל ה- parser שברשותנו לא עומד בקריטריונים סבירים בכדי ליצור מילון inverted index ולכן ביצענו מספר שינוים במודל זה.

להלן הפעולות אשר שיפרו זמן ריצה :

1. כפי שהוזכר לעיל , שיפרנו את ה – parser ואף הוספנו\מחקנו חוקים שהוסברו מעלה.
2. שיפרנו את מודל ה- global method ביעילות זמן הריצה ואחסון הנתונים בו.
3. בחלק הקודם, בוצעה גישה לקבצי ה-posting file על ידי קריאה מקובץ שנמצא בדיסק. בחלק זה, נתוני ה- posting נשמרו בזיכרון הראשי מה שהביא לזמן ריצה מהיר הרבה יותר. כידוע, גישות לדיסק מאטות זמן ריצה באופן משמעותי.
4. בחלק הקודם, שמרנו את קבצי ה – posting file בקבצי txt והיה עלינו לעבור על כל הקובץ עד מציאת ה- term הרלוונטי בקובץ. בחלק זה היה ניתן לגשת לפרטי המילה ב"קובץ" ה- posting file על ידי גישה למבנה נתונים מסוג מילון ולהגיע לנתונים ב- O(1).

**שאלה 6**

לאחר פרסום חלק ג' התברר כי הקורפוס הוקטן לסביבות ה- 10,000 מסמכים וכתוצאה מכך ה-inverted index קטן בכ- 95%. החלטנו לעבור על המילון בעצמו בכדי לראות אילו מילים נמצאות בו. לאחר סקירה , גילינו כי הכנסנו למילון הרבה מילים ומספרים חסרי ערך שרק יפגעו באיכות המילון ואיכות אחזור המסמכים.

תחילה, החלטנו על שיפור ה- parser באופן מידי ומימוש advanced parser אשר תסייע לנו באיכות המונחים שיכנסו למילון. השיפורים שנעשו במודל ה- parser מופיעים לעיל (*שאלה 2*). כאשר השתמשנו ב- parser שהגשנו בחלק א' התקבל מילון inverted index בגודל 15K אשר אין ערובה לאיכותו כלל. לאחר תיקון המודל הקטנו את המודל לכ- 8.6K. לאור תיקון זה, קיבלנו מילון איכותי יותר מה שעזר באופן ישיר לאיכות התוצאות.

כפי שהוזכר מעלה, ביצענו מימוש למודלים : glove, wordNet ,global method, spell checker ובדקנו שילובים ביניהם. לבסוף, בחרנו ב- Global Method מכיוון שנראה כי הוא סיפר לנו את המדדים הגבוהים ביותר באחזור המסמכים.

**שאלה 7**

בכדי לבחור בשיטה המיטבית להרחבת השאילתה ושיפור איכות המסמכים המאוחזרים, ביצענו בדיקה עבור כל אחת מהשיטות והעלנו זאת על גרף אשר מסביר בצורה גרפית את טיב האחזור והמדדים.

להלן הגרף :

כפי שניתן לראות, ישנו יתרון קל לשיטת ה- global method .

לפיכך, החלטנו לבחור במודל שמממש את ה- **global method**. בחרנו במודל זה מכוון שהוא מציג מדדים גבוהים יותר. בנוסף, את המודל הזה מימשנו בעצמנו ואנחנו סמוכים ובטוחים כמה הוא יעיל ומיטבי גם מול קורפוסים גדולים יותר(חלק א'). שאר המודלים השתמשו בספריות חיצוניות אשר אין לנו מידע בנוגע למימושן ויעילותן. במצב הנתון הקורפוס הוא בסביבות העשרת אלפים מסמכים, אך אם וכאשר נריץ על המנוע קורפוס גדול יותר עלולה להיות ירידה בביצועים.

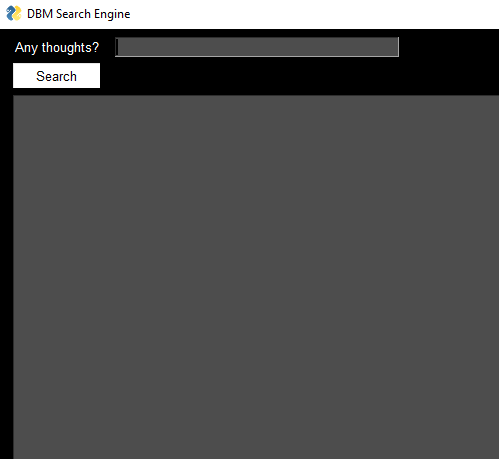
**שאלה 8**

בפרויקט שלנו לא השתמשנו בAPI חיצוני.

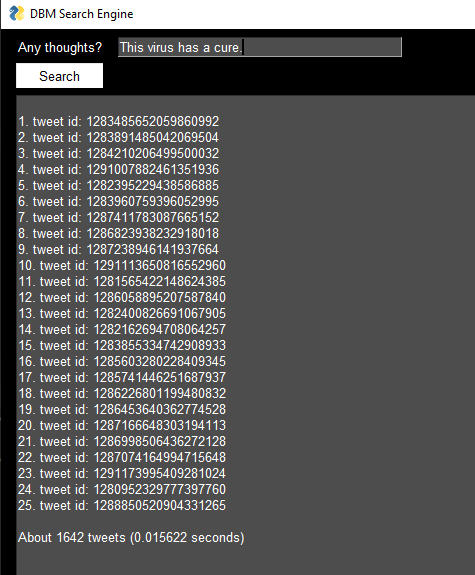
**שאלה 9**

יצרנו GUI לחווית משתמש נוחה יותר לקבלת תוצאות שאילתא אשר מציגה את מספרי הטוויטים מדורגים כך שמקום 1 הוא הכי רלוונטי, כמה זמן לקח למצוא אותם בקורפוס (מתחילת החיפוש ועד סופו) הנוכחי וכמה טוויטים זוהו כרלוונטים על פי המנוע שלנו.

יש לציין שהגבלנו ל25 התוצאות הראשונות אבל למטה אפשר לראות כמה מסמכים אוחזרו. להלן דוגמא:



לאחר חיפוש שאילתא מסוימת, נקבל:



הוספנו את הממשק רק במחלקת GUI, כלומר במידה ותרצו להפעילו, יש להיכנס לGUI.py וללחוץ על play. הוא עובד כמובן על search\_engine\_best.

על מנת להפעילו, השתמשנו בספרייה PySimpleGUI.

בנוסף, על מנת לממש את השיטות wordnet, glove, spell\_checker הורדנו את הספריות הבאות:

* wordnet
* הורדנו את הספרייה pySpellChecker. מspellchecker השתמשנו בSpellChecker.
* הורדנו את genism ומשם לקחנו את glove2word2vec ו- KeyedVectors.

יש לציין כי:

* gloVe – נקרא אצלנו בתור search\_engine\_1
* wordnet – נקרא אצלנו בתור search\_engine\_2
* Spell\_chcker – נקרא אצלנו בתור search\_engine\_3
* Global Method – נקרא אצלנו בתור search\_engine\_4 **וגם** search\_engine\_best

**שאלה 10**

תחילה נסביר על פי המודל הכי טוב שבחרנו, מודל ה- **global method** , את התוצאות שהתקבלו:

שאילתה 1 –

Dr. Anthony Fauci wrote in a 2005 paper published in Virology Journal that hydroxychloroquine was effective in treating SARS.

בשאילתה זו השיטה מרחיבה את השאילתה על ידי המילים הבאות :

*'doctor', 'director', 'died', 'data', 'following', 'hydroxychloroquine', 'forget', 'first', 'system', 'cure', 'early', 'everyone', 'everything', 'election', 'number', 'end'*

תוצאה 1 :

@weijia Hydroxychloroquine is the cure.

הסבר תוצאה 1 : Hydroxychloroquineהתרופה מופיעה בשאילתה ו- cure תואמת מבחינה סמנטית ובנוסף הורחבה על ידי השיטה.

תוצאה 2:

@TomFitton @realDonaldTrump Is it true Fauci wrote in virology journal in 2005 that hydroxychloroquine functions as a cure and a vaccine??

הסבר תוצאה 2 : השם של הדוקטור והשם של התרופה נמצאים יחדיו בתוצאת השאילתה.

תוצאה 3:

Dr. Anthony Fauci defends reopening schools

הסבר תוצאה 3 : השם של הדוקטור נמצא בתוצאת השאילתה.

תוצאה 4:

Dr. Anthony Fauci warns the coronavirus won’t ever be totally eradicated

הסבר תוצאה 4 :

השם של הדוקטור נמצא בתוצאת השאילתה.

תוצאה 5:

Dr. Anthony Fauci on COVID, HIV, and Being an 'Alarmist'

הסבר תוצאה 5 :

השם של הדוקטור נמצא בתוצאת השאילתה.

שאילתה 2 –

The seasonal flu kills more people every year in the U.S. than COVID-19 has to date.

בשאילתה זו השיטה מרחיבה את השאילתה על ידי המילים הבאות :

*‘far', '99', 'free', 'fear', 'still', 'status', 'dying', 'stop', 'positive', 'public', 'ever', 'single', 'number', 'everyone', 'even', '19', 'died', '50', 'day', 'deaths', '2.02K', 'new', 'cases', 'news', 'virus', 'covid', '07', '10'*

תוצאה ,2,3,4,51 :



הסבר תוצאה ,2,3,4,51 :

כל התוצאות שהוחזרו הם אותה תוצאה במסמכים שונים(tweet id שונים)

במסמך מופיעה השאילתה בשלילה. כלומר, סותר את השאילתה.

כל המילים בשאילתה נמצאות בתוצאה עצמה.

שאילתה 4 –

The coronavirus pandemic is a cover for a plan to implant trackable microchips and that the Microsoft co-founder Bill Gates is behind it

בשאילתה זו השיטה מרחיבה את השאילתה על ידי המילים הבאות :

*‘new', 'cases', 'news', 'status', 'city', 'nothing', 'people', 'please', 'still', 'president', 'support', 'staff', 'ppe', 'parents', 'place', 'closed', 'comes', 'clear', 'choice', 'nearly', 'block', 'ny', 'business’*

תוצאה 1,2,3,4:



הסבר : כל מילה שהופיעה בשאילתה מופיעה גם בתוצאה

תוצאה 5:

Bill Gates

הסבר: שמו של ביל גייטס מופיע בשאילתה והמשקל שניתן למסמך הוא גבוה מכוון שהוא מסמך קצר

שאילתה 7–

Herd immunity has been reached.

הסברי תוצאות על פי מודל ה – global method :

בשאילתה זו השיטה מרחיבה את השאילתה על ידי המילים הבאות :

*'infection', 'system', 'infections', 'question', 'ny', 'infected'*

*תוצאה 1 :*

NY has herd immunity.

*הסבר:* השאילתה הורחבה עם המילה ny ו- herd immunity הופיע בשאילתה עצמה.

תוצאה 2:

Pandemic is over in NY because you have reached herd immunity.

הסבר: השאילתה הורחבה עם המילה ny ו- herd immunity הופיע בשאילתה עצמה

תוצאה 3 :

How is this not herd immunity?

הסבר : לאחר פרסור והורדת stopwords נשארו המילים herd immunity מה שנתן למסמך זה דירוג גבוה ולכן הוא נשלף במסמכים הראשונים.

תוצאה ,54 :

Herd Immunity ....

הסבר: מילים אלו הופיעו בשאילתה עצמה וגם כאן התקבל דירוג גבוה.

שאילתה 8–

Children are “almost immune from this disease.”

בשאילתה זו השיטה מרחיבה את השאילתה על ידי המילים הבאות :

*'study', 'next', '19', 'never', 'via', 'deadly', 'system', 'cure', 'dying', 'director', 'died'*

*תוצאה 1:*

We are the disease.

הסבר: המילה disease הופיעה בשאילתה. לאחר פרסור היא קיבלה דירוג גבוה מכוון שהstopwords הוסרו ולכן אוחזרה ראשונה.

תוצאה 2:

No and never had a flu

הסבר: השאילתה הורחבה על ידי המילה never ולכן התקבל המסמך הנ"ל. גם כאן, לאחר הורדת Stopwords המסמך קיבל דירוג גבוה מכוון שהוא קצר.

תוצאה 3:

Children all almost immune...

הסבר: המילים הללו מופיעות בשאילתה

תוצאה 4:

Children are nearly immune from Covid disease!

הסבר: דומה מבחינה סמנטית לשאילתה

תוצאה 5:

No, children are NOT immune to Covid-19.

הסבר:

סותר את השאילתה ולכן התקבל המסמך הנ"ל.

כעת נסביר את התוצאות על פי מודל **wordNet** :

שאילתה 1 –

Dr. Anthony Fauci wrote in a 2005 paper published in Virology Journal that hydroxychloroquine was effective in treating SARS.

בשאילתה זו השיטה מרחיבה את השאילתה על ידי המילים הבאות :

*'write', 'publish', 'save', 'spell', 'paper', 'report', 'theme', 'issue', 'release', 'published', 'virology', 'journal', 'hydroxychloroquine', 'effective', 'efficient', 'good', 'treat', 'handle', 'process', 'cover', 'deal', 'address'*

תוצאה 1 :

Dr. Anthony Fauci defends reopening schools

הסבר תוצאה 1 : השם של הדוקטור נמצא בתוצאת השאילתה.

תוצאה 2:

Dr. Anthony Fauci warns the coronavirus won’t ever be totally eradicated

הסבר: השם של הדוקטור נמצא בתוצאת השאילתה.

תוצאה 3:

Dr. Anthony Fauci on COVID, HIV, and Being an 'Alarmist'

הסבר: השם של הדוקטור נמצא בתוצאת השאילתה.

תוצאה 3:

Dr. Anthony Fauci defends reopening schools

הסבר: השם של הדוקטור נמצא במסמך.

תוצאה 4:

Is it true Fauci wrote in virology journal in 2005 that hydroxychloroquine functions as a cure and a vaccine??

הסבר:

השם של הדוקטור והתרופה נמצאים במסמך.

תוצאה 5:

RT @TheMangoMan7: Fascist Fauci: Hydroxychloroquine Is 'Not Effective' in Treating the Coronavirus

הסבר:

מספר מילים שמופיעות בשאילתה אכן מופיעות במסמך.

שאילתה 2 –

The seasonal flu kills more people every year in the U.S. than COVID-19 has to date.

בשאילתה זו השיטה מרחיבה את השאילתה על ידי המילים הבאות :

*'seasonal', 'influenza', 'flu', 'killing', 'kill', 'defeat', 'pop', 'people', 'masses', 'mass', 'every', 'year', 'yr', 'class', '19', 'date', 'appointment', 'engagement', 'see'*

תוצאה 1,2,3 :



הסבר תוצאה ,2,31 :

כל התוצאות שהוחזרו הם אותה תוצאה במסמכים שונים(tweet id שונים)

במסמך מופיעה השאילתה בשלילה. כלומר, סותר את השאילתה.

כ המילים בשאילתה נמצאות בתוצאה עצמה.

תוצאה 4:

“The flu kills more people every year and we never shut down”

הסבר: התוצאה דומה מבחינה סמנטית לשאילתה.

תוצאה 5 :

Abortion kill millions every year

הסבר: המילה kill נמצאת בהרחבה של השאילתה.

שאילתה 4 –

The coronavirus pandemic is a cover for a plan to implant trackable microchips and that the Microsoft co-founder Bill Gates is behind it

בשאילתה זו השיטה מרחיבה את השאילתה על ידי המילים הבאות :

*'pandemic', 'screen', 'cover', 'covering', 'masking', 'back', 'top', 'continue', 'treat', 'handle', 'deal', 'address', 'embrace', 'track', 'cross', 'report', 'hide', 'breed', 'incubate', 'plan', 'program', 'design', 'project', 'implant', 'plant', 'trackable', 'chip', 'microchip', 'founder', 'father', 'collapse', 'give', 'break', 'bill', 'measure', 'account', 'note', 'posting', 'notice', 'card', 'peak', 'charge', 'gate', 'arse', 'butt', 'bum', 'seat', 'bottom', 'behind', 'ass', 'slow'*

תוצאה 1,2,3,4:



הסבר : כל מילה שהופיעה בשאילתה מופיעה גם בתוצאה

תוצאה 5:

Bill Gates

הסבר: שמו של ביל גייטס מופיע בשאילתה והמשקל שניתן למסמך הוא גבוה מכוון שהוא מסמך קצר

שאילתה 7–

Herd immunity has been reached.

הסברי תוצאות על פי מודל ה – global method :

בשאילתה זו השיטה מרחיבה את השאילתה על ידי המילים הבאות :

*'herd', 'crowd', 'immunity', 'resistance', 'exemption', 'reach', 'make', 'hit', 'gain', 'contact', 'achieve', 'touch', 'pass', 'hand', 'give', 'strain'*

*תוצאה 1 :*

What is herd immunity and how do we reach it?

*הסבר:* לאחר פרסור השאילתה התקבלו המילים herd,immunity,reached ובהרחבה התקבלה המילה reach אשר מופיע במסמך

תוצאה 2:

How is this not herd immunity?

הסבר : לאחר פרסור והורדת stopwords נשארו המילים herd immunity מה שנתן למסמך זה דירוג גבוה ולכן הוא נשלף במסמכים הראשונים.

תוצאה 3 :

Herd Immunity ....

הסבר : מילים אלו הופיעו בשאילתה עצמה וגם כאן התקבל דירוג גבוה.

תוצאה ,54 :

Herd immunity

הסבר: מילים אלו הופיעו בשאילתה עצמה וגם כאן התקבל דירוג גבוה.

שאילתה 8–

Children are “almost immune from this disease.”

בשאילתה זו השיטה מרחיבה את השאילתה על ידי המילים הבאות :

*'child', 'kid', 'baby', 'immune', 'resistant', 'disease'*

*תוצאה 1:*

We are the disease.

הסבר: המילה disease הופיעה בשאילתה. לאחר פרסור היא קיבלה דירוג גבוה מכוון שהstopwords הוסרו ולכן אוחזרה ראשונה.

תוצאה 2:

Children all almost immune...

הסבר: המילים הללו מופיעות בשאילתה

תוצאה 3:

Children are nearly immune from Covid disease!

הסבר: דומה מבחינה סמנטית לשאילתה

תוצאה 4:

TS A DISEASE NOW NOT A VIRUS

הסבר: המילה disease הופיעה בשאילתה.

תוצאה 5:

"Children almost immune from COVID!"

הסבר: ציטוט שמוכל בשאילתה

**שאלה 11:**

שאילתה 1:

להלן נתוני המדדים והגרף ביחס לשיטות השונות:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Glove** | **wordNet** | **spellChecker** | **Global Method** |
| **percision@5** | *0.8* | *0.8* | *0.8* | *0.8* |
| **percision@10** | *0.5* | 0.5 | *0.5* | *0.5* |
| **percision@50** | *0.82* | *0.84* | *0.82* | *0.82* |
| **recall** | 0.984 | *0.984* | *0.984* | *0.984* |

שאילתה 2:

להלן נתוני המדדים והגרף ביחס לשיטות השונות:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Glove** | **wordNet** | **spellChecker** | **Global Method** |
| **percision@5** | *0.6* | *0.8* | *0.8* | *0.8* |
| **percision@10** | *0.8* | 0.9 | *0.6* | *0.9* |
| **percision@50** | *0.8* | *0.78* | *0.74* | *0.82* |
| **recall** | *1* | *1* | *1* | *1* |

שאילתה 4:

להלן נתוני המדדים והגרף ביחס לשיטות השונות:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Glove** | **wordNet** | **spellChecker** | **Global Method** |
| **percision@5** | *1* | *1* | *1* | *1* |
| **percision@10** | *1* | 1 | *1* | *1* |
| **percision@50** | *0.94* | *0.94* | *0.92* | *0.92* |
| **recall** | *1* | *1* | *1* | *1* |

שאילתה 7:

להלן נתוני המדדים והגרף ביחס לשיטות השונות:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Glove** | **wordNet** | **spellChecker** | **Global Method** |
| **percision@5** | *1* | *1* | *1* | *1* |
| **percision@10** | *0.9* | *0.9* | *0.9* | *0.9* |
| **percision@50** | *0.9* | *0.9* | *0.9* | *0.9* |
| **recall** | *1* | *1* | *1* | *1* |

שאילתה 8 :

להלן נתוני המדדים והגרף ביחס לשיטות השונות:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Glove** | **wordNet** | **spellChecker** | **Global Method** |
| **percision@5** | *0.8* | *0.8* | *0.8* | *0.8* |
| **percision@10** | *0.9* | 0.9 | *0.9* | *0.9* |
| **percision@50** | *0.94* | *0.94* | *0.94* | *0.94* |
| **recall** | *0.985* | *0.988* | *0.985* | *0.996* |

**שאלה 12 :**

בפרויקט זה בחרנו את המודל global method להיות המנוע הטוב ביותר עבורנו.

כעת נפרט את היתרונות והחסרונות של מנוע זה.

יתרונות:

* המטריצה אשר נוצרת כחלק מהשיטה, נוצרת על סמך מילים תדירות אשר הופיעו בקורפוס הרלוונטי למנוע ולכן המילים שיתקבלו כתוצאה מההרחבה נמצאות בinverted index .
* מימוש יעיל ומוכר מבלי להשתמש בספריות חיצוניות אשר מימושן לא ידוע
* כפי שנראה בשאלה 7, בכל המדדים שיטה זו מתעלה על שאר השיטות. precision@[5,10,50]
* שיפור ה- parser לזמן ריצה יעיל ומיטבי

חסרונות:

* בקורפוס גדול עלולה להיווצר מטריצה עצומה אשר תאריך את זמן האחזור כתוצאה מהרחבת השאילתה.
* בהנחה ונהיה מוגבלים בזיכרון, גודל המטריצה מוגבל מה שיביא לצמצום מילים במטריצה עצמה מה שלא קורה במודלים שממומשים על ידי ספריות.
* אף על פי שאיכות התוצאות השתפרה, לא הגענו לדיוק מקסימלי שרצינו להגיע אליו ועדיין ישנן תוצאות שנמצאו כרלוונטיות בגלל מילה או שתיים, אבל הכוונה שלה הייתה שונה.