

עיבוד שפות טבעיות - תרגיל 4

Word Embeddings

חלק א - יצירה של מודל Word2Vec ואימונו על קורפוס הכנסת

1. vector_size - ארגומנט זה קובע את גודל word vectors.

היתרון בבחירת ערך 100 לארגומנט זה הוא שהמודל יכול לזהות את הקשר בין המילים באופן סמנטי ללא צורך בחישובים מורכבים מדי.

החסרון הוא שהמודל עשוי לא לזהות פרטים קטנים יותר בין מילים דומות, במקרים שבהם יש הבחנות מדויקות אז נדרש שימוש בוקטורים יותר גדולים.

window - ארגומנט זה מציין את המרחק המקסימלי בין המילה הנוכחית והמילה החזויה בתוך משפט. Window גדול יותר משמעותו שהקשר נלקח מטווח רחב יותר של מילים מסביב.

היתרון בבחירת ערך 5 לארגומנט זה מאפשר למודל לזהות הקשר מקומי בצורה יעילה וזה מתאים לבעיה שלנו.

לעומת זאת, החסרון הוא שיייתכן ולא יתפוס תלות ארוכת טווח בין מילים בצורה יעילה באותה מידה.

min_count - ארגומנט זה קובע את התדירות המינימלית שמילה חייבת להיכלל באוצר המילים.

מילים עם תדירות נמוכה מ min_count מתעלמים מהן.

היתרון בבחירת ערכו להיות שווה ל 1 הוא שהמודל כולל את כל המילים הקיימות בקורפוס ללא קשר לתדירותן. וזה אכן מבטיח ששום מידע לא יאבד עקב אי הכללה של מילים שתדירותם היא נמוכה.

החסרון בבחירת ערך 1 לארגומנט זה הינה הכללת כל המילים גם אלו שתדירותם נמוכה שהיא יכולה להגדיל את גודל אוצר המילים והעלות החישובית. גם כן זה עלול להכניס רעש ממילים נדירות או שאינן רלוונטיות.

לפי דעתנו פרמטרים אלו הם מתאימים לבעייה שלנו .

מכיוון שערכים אלו מאזנים בין תפיסת המשמעויות בדאטה תוך שמירה על חישוב יעיל. בנוסף לכך, ערכים אלו מתיישרים היטב עם המאפיינים של הדאטה ודרישות המטלה שלנו, מה שמבטיח למידת ייצוג יעילה ללא מורכבות מיותרת. לכן לא ראינו צורך בשינוי פרמטרים אלו מכיוון שמספקים ביצוע טוב עבור המשימה שלנו.

2. הבעיות שיכולות לעלות משימוש במודל הנ"ל שאומן על הקורפוס שלנו הן:

-הקורפוס שלנו הוא לא כזה גדול ולכן ייתכן שהוא לא יכיל מספיק הקשרים שונים כדי ללמוד ממנו word embeddings שהם חזקות מה שיגרום לייצוגים פחות מדויקים.

מודל ה word2vec שלנו מסתמך על תדירות המילים בקורפוס של הכנסת שלנו כדי ללמוד embedding משמעותיות. מילים שלא מופיעות בתדירות גבוהה בקורפוס שלנו עשויות שלא להיות מיוצגות היטב מה שישפיע על יכולתו של המודל לתפוס את ההקשרים הסימנטיים של מילים אלה בצורה מדויקת.

איכות הקורפוס שמשמש לאימון המודל word2vec משפיעה ישירות על איכות תוצאות המודל. אם הקורפוס אינו מייצג נכון את תחום הנדרש או שהוא מכיל נתונים לא רלוונטיים, ייתכן שהמודל לא יצליח לתפוס בצורה מדויקת את ההקשרים הסמנטיים בין מילים.

חלק ב – דמיון בין מילים

1. כשאנחנו משתמשים ב most similar היינו מצפים לקבל מילים המשמשות בהקשרים דומים או שיש להן משמעויות קשורות למילת הקלט שלנו. אלו יכולות להיות מילים נרדפות, מונחים קשורים או מילים המופיעות לעתים קרובות במשפטים דומים.

אכן חלק מהמילים הכי קרובות שקיבלנו בסעיף א תואם את הציפיות שלנו מכך שהם היו דומות כמו למשל עבור מילת חבר קיבלנו : חברת , לחבר מבחינת משמעות או שהיו מילים שנמצאים באותו הקשר .

וחלק לא היה תואם את הציפיות שלנו מכיוון שהקורפוס שלנו לא כזה גדול ולכן הסיכוי להימצאות מילים באותו פירוש או הקשר לא כזה גדול ולכן מצופה לקבל מילים שהם לא קשורות כל כך.

2. אם ניקח שתי מילים שנחשבות להפכים היינו מצפים לקבל מרחק קצר בין שני וקטורי המילים, זה נובע מכך שמודל word2vec לומד את היחסים הסמנטיים בין מילים בהתבסס על הקשר ביניהן בטקסט, גם כן ההפכים ברוב המקרים באים באותו הקשר.

3. אז בחרנו בשני זוגות של מילה וההפך שלה שקיימות בקורפוס:

פתיחה – סגירה : עבור הזוג הזה קיבלנו $\text{similarity score} = 0.965$

$\text{distance} = 0.034$

טוב – רע : עבור הזוג הזה קיבלנו $\text{similarity score} = 0.853$

$\text{distance} = 0.146$

וכמו שיכולים לראות אכן קיבלנו כציפייה שלנו מרחק קצר.

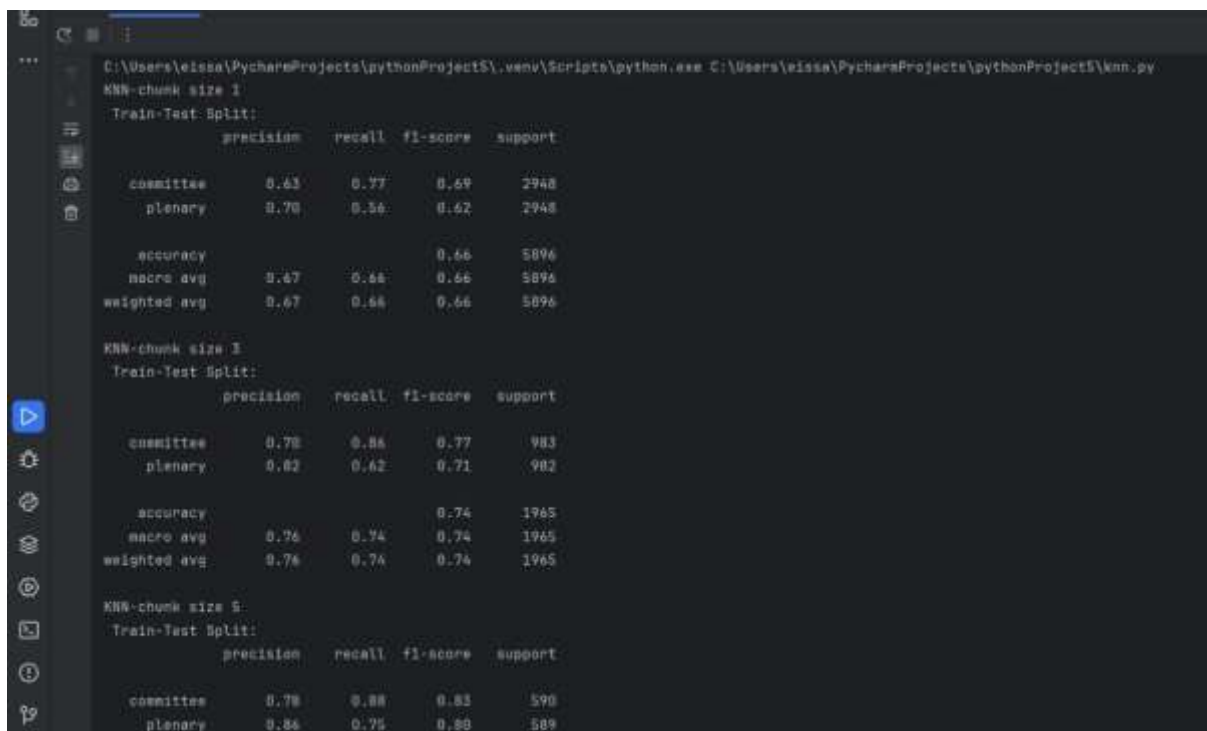
4. כאשר אנחנו משתמשים ב most similar כדי למצוא את המשפט הכי קרוב למשפט הקלט, אנו מצפים לקבל את המשפט הכי דומה למשפט הקלט. הדימיון יכול להתייחס למילים דומות למילים שבמשפט הקלט, קשרים סימנטיים או התייחסות לאותו נושא של המשפט או הקשר סמנטי.

אז יש חלק שכן תאם את הציפיות שלנו בכך שהיה אותה משמעות למשפט, או שימוש באותם מילים

אבל מצד שני יש חלק שלא היה תואם את הציפיות שלנו, לא היה קשר בין נושאי המשפטים או המילים. לרוב זה קרה בגלל שהקורפוס שלנו לא כזה גדול וייתכן ויש מילים או נושאים שאין להם משהו דומה.

חלק ג – סיווג

classification report:



```
C:\Users\elissa\PycharmProjects\pythonProject5\.venv\Scripts\python.exe C:\Users\elissa\PycharmProjects\pythonProject5\knn.py
KNN-chunk size 1
Train-Test Split:

```

	precision	recall	f1-score	support
committee	0.63	0.77	0.69	2948
plenary	0.70	0.56	0.62	2948
accuracy			0.66	5896
macro avg	0.67	0.66	0.66	5896
weighted avg	0.67	0.66	0.66	5896

```

KNN-chunk size 3
Train-Test Split:

```

	precision	recall	f1-score	support
committee	0.70	0.86	0.77	983
plenary	0.62	0.62	0.71	982
accuracy			0.74	1965
macro avg	0.76	0.74	0.74	1965
weighted avg	0.76	0.74	0.74	1965

```

KNN-chunk size 5:
Train-Test Split:

```

	precision	recall	f1-score	support
committee	0.70	0.88	0.83	590
plenary	0.86	0.75	0.80	589

```

accuracy          0.66  5896
macro avg         0.67  0.66  0.66  5896
weighted avg      0.67  0.66  0.66  5896

KNN-chunk size 3
Train-Test Split:
precision recall f1-score support

committee 0.78 0.66 0.77 983
plenary 0.82 0.62 0.71 982

accuracy          0.74  1965
macro avg         0.76  0.74  0.74  1965
weighted avg      0.76  0.74  0.74  1965

KNN-chunk size 5
Train-Test Split:
precision recall f1-score support

committee 0.78 0.88 0.83 590
plenary 0.86 0.75 0.80 589

accuracy          0.82  1179
macro avg         0.82  0.82  0.81  1179
weighted avg      0.82  0.82  0.81  1179

Process finished with exit code 0

```

1. קיבלנו תוצאות פחות טובות, השתמשנו באותו גודל צ'אנק $= 5$ ושיטת חלוקה שזו היא test train split ואותה כמות שכנים $= 51$.
2. Sentence embeddings דורשת לעתים קרובות דאטה גדול כדי ללמוד ייצוגים משמעותיים שיכולים להכליל היטב. לעומת זאת, TFIDF יכול לתפקד היטב אפילו על דאטה יותר קטן מכיוון שזוהי שיטה יותר ישירה ופשוטה. גם כן גודל ה sentence embeddings vector קטן יותר מגודל ה sentence embeddings vector. לרוב זה גם נובע מכך שפרמטרים אלו הם מתאימים ל feature vector שנבנה על ידי TFIDF (בעיית התרגיל הקודם) יותר מה sentence embeddings vector של התרגיל הנוכחי.
3. קיבלנו תוצאות טובות יותר עבור צאנק בגודל 5. וכן זה אכן נכון עבור וקטור המאפיינים שהשתמשנו בו בתרגיל 3. הסיבה לכך שעבור גודל $chunk=5$ אנו מקבלים תוצאות יותר טובות היא שהוא יכול להתייחס להקשר רחב יותר בין המילים במשפט וזה עשוי להקל על המודל לזהות קשרים סימנטיים עמוקים יותר בין המילים ולבצע סיווג טוב יותר. חשוב לציין: אם נקטין גודל chunk יותר מדי זה גורם ל underfitting ואם נגדיל אותו יותר מדי זה גורם ל overfitting.

חלק ד – שימוש במודלי שפה גדולים

1. כן קיבלנו משפטים הגיוניים ומובנים מבחינת התוכן, הקוהרנטיות והתחביר. המשפטים משתמשים במילים מדויקים וברורים .
 2. אכן המשפטים יש בהם שיפור בהשוואה לתוצאות שקיבלנו בתרגיל בית 2 . יש להם משמעות נכונה וברורה.
 3. ברוב המשפטים הוא עבד טוב.
- יש כמה סיבות לזה, dictabert עשוי להתקשה לבצע ביצועים יעילים בתחומים שהם מעבר לתחום שהתאמן עליו בשל הסתמכותו על נתונים שהתאמן עליהם אשר עשויים לא לכסות את ההקשרים שנמצאים בתחומים מסוימים כמו בתחום הרפואה או תחום המשפטים, שבהם ידע ספציפי לתחום הוא בעל חשיבות.
- בנוסף לכך, הוא עלול להיתקל בקשיים עם מילים נדירות או מחוץ לאוצר המילים בנתוני האימון שלו.
- גם המודל עשוי להתקשות בהבנת טקסטים עם שפה מיוחדת או תחביר פורמלי או משפטים מורכבים .
- המודל לא יעבוד בצורה מושלמת על כל משפט מתוך קורפוס הכנסת, הסיבה לכך היא ההרכב המגוון של משפטי הקורפוס שכולל תחומי נושא שונים כמו תחום המשפטים ומבני משפטים מגוונים.