**עיבוד שפות טבעיות**

**תרגיל בית 2**

**שם: רנים אבראהים (212920896), אסיל נחאס (212245096)**

**מבוא:  
במהלך התרגיל התבקשנו לבנות וליישם מודלי שפה המבוססים על טריגרמות קורפוס שנלקח מפרוטוקולים של הכנסת. המטרה היא לחשב הסתברויות הופעת משפטים וטוקנים לפי מודל סטטיסטי המבוסס על תדירות הרצפים בקורפוס, וכן לאתר קולוקציות נפוצות ולבצע משימות השלמה וחיזוי טוקנים חסרים במשפט.**

**הקורפוס מחולק לשני סוגים: משפטים שמקורם בפרוטוקולים מסוג "ועדה" ומשפטים שמקורם בפרוטוקולים מסוג "מליאה". בהתאם לכך, נדרשו שני מודלי שפה נפרדים: אחד לוועדות ואחד למליאות.**

**שלב 1:**

**בניית מודלי השפה:  
בשלב זה התבקשנו לבנות מודלים המבוססים על טריגרמות. רעיון הטריגרמה הוא שחיזוי הטוקן הבא במשפט מתבסס על שני הטוקנים הקודמים לו.**

**Trigram\_LM : מחלקת  
בנינו מחלקה המאגדת את כל המבנים והפונקציות הדרושים למודל השפה: שמרנו במבנה נתונים את ספירות היוני-גראמס (טוקן יחיד), הבי-גראמס (זוג טוקנים רציפים) והטרי-גראמס (שלישיות טוקנים רציפות) עבור כל אחד משני סוגי הפרוטוקולים: ועדות ו- מליאות.**

**עקבנו אחר מספר הטוקנים הכולל וספירת כל מילה, כדי שנוכל לחשב הסתברויות בהמשך.**

**הוספנו שני טוקני דמה בתחילת כל משפט, וכן טוקן סיום על מנת לטפל במקרים של תחילת המשפט וסופו לפי הדרישות.**

**וגם הגדרנו את הלמדות שיהיו המשקל של כל n-gram כך שבמהלך חישוב ההסתברויות של ה- ngrams נרצה לתת יותר משקל ל- trigrams ביחס ל unigrams / bigrams כי רצף של 3 מילים יכול להגיד לנו יותר על המילה הבאה ברצף משתי מילים או ממילה אחת.**

**בנינו פונקציה שמחשבת את כמות הטוקנים וגם מצרפת להם טוקני הדמה הפונקציה גם מחשבת את כמות ה- ngrams. הפונקציה נקראת fit\_model\_to\_sentences.**

**במחלקה שלנו גם קיימות הפונקציות הבאות:**

**calculate\_prob\_of\_sentenceפונקציית:**

**פונקציה זו מקבלת משפט ומחזירה את לוג ההסתברות שלו לפי המודל.**

**הוספנו למשפט את הטוקנים הדמה בתחילתו ובסופו.**

**עברנו על כל טוקן במשפט (החל מהטוקן השלישי) וחישבנו את ההסתברות שלו בהתבסס על שני הטוקנים הקודמים.**

**ההסתברות חושבה בשיטת נראות מקסימלית עם החלקת לפלס כדי למנוע הסתברויות אפס, ועשינו את החישוב הזה בעזרת הפונקציה compute\_smoothed\_probability  
כך שהפונקציה בודקת על פי איזה ngram אנחנו מחשבים את ההסתברות ומציבה בנוסחה שלוקחת את הכמות של ה- ngrams ומוסיפה 1 ומחלקת בכמות הכללית ועוד גודל השפה.**

**השתמשנו באינטרפולציה ליניארית: שילבנו את הסתברות היוני-גראם, הבי-גראם והטרי-גראם תוך שימוש במשקלים שנבחרו. בתחילה תכננו משקלים שונים, אך על פי ההנחיות ניתן לבחור משקלים ולהסביר בדו"ח. בדוגמה שלנו בחרנו משקלים כך שהטריגרמה מקבלת משקל גדול (0.99), הביגרמה משקל בינוני (0.009) והיוניגרמה משקל קטן (0.001), במטרה להסתמך יותר על ההקשר הרחב (שני טוקנים אחורה).**

**לבסוף סכמנו את לוג ההסתברויות של כל הטוקנים במשפט וקיבלנו לוג הסתברות כוללת למשפט.**

**generate\_next\_tokenפונקציית:**

**פונקציה זו מקבלת צירוף של טוקנים וחוזה את הטוקן הבא.**

**גם פה חישבנו הסתברות לכל מילה במילון לפי טריגרמה, בי-גראם ויוני-גראם בתוספת החלקת לפלס שמחשבים בעזרת הפונקציה compute\_smoothed\_probability שהזכרנו קודם.**

**בחרנו משקלי אינטרפולציה דומים לאלו ששימשו במשפט, אך בנוסף יישמנו "הענשה" לטוקנים מאוד נפוצים (כמו סימני פיסוק או מילים כמו "את", "על", "אני") כדי לא "ליפול" תמיד על הטוקן הנפוץ ביותר.**

**לאחר חישוב ההסתברויות המתוקנות עבור כל טוקן, בחרנו את הטוקן בעל ההסתברות הגבוהה ביותר והחזרנו אותו יחד עם לוג ההסתברות שלו.**

**שלב 2:**

**קולוקציות:**

**בשלב זה התבקשנו להחזיר את 10 הקולוקציות הנפוצות ביותר באורכי 2,3,4 בכל אחד משני סוגי הקורפוסים (ועדות ומליאות), לפי שני מדדים:**

**(frequency) ו-TF-IDF.**

**get\_k\_n\_t\_collocationsפונקציית:**

**חילקנו את המשפטים לפי סוג הפרוטוקול (committee, plenary) :**

**יצרנו n-grams מהמשפטים לדוגמה: (2-grams, 3-grams, 4-grams)**

**אם המדד הוא "frequency", סיננו רק את הנ-גראמס שמופיעות לפחות t פעמים, מיין אותם לפי תדירות בסדר יורד ולקחנו את ה- k המובילים.**

**אם המדד הוא "tfidf", חישבנו TF-IDF עבור כל נ-גראם, סיננו לפי הסף t, מיין לפי ערך TF-IDF, ולקחנו את k הגבוהות ביותר.**

**את התוצאות הדפסנו לקובץ knesset\_collocations.txt בפורמט הנדרש.**

**תוצאות לדוגמה מתוך הקובץ :knesset\_collocations.txt**

**ניתן לראות בקובץ כי הקולוקציות הנפוצות ביותר (במדד תדירות) כוללות סימני קריאה " < <", "> >", וכן מילות מפתח כמו "חבר הכנסת", "היושב - ראש". זה מצביע על כך שבקורפוסים יש הרבה תבניות טקסטואליות שחוזרות (כמו קריאות ביניים, שמות דוברים, סמלי פורמליות).**

**במדד ה-TF-IDF ניתן לראות הבדלים קלים: הקולוקציות עם ערכי TF-IDF גבוהים הן עדיין קולוקציות חוזרות, אך המיקום שלהן מבחינת TF-IDF מעיד על כך שהן "ייחודיות" יחסית בתת-קורפוס מסוים.**

**שלב 3:**

**יישום מודלי השפה:**

**mask\_tokens\_in\_sentences :פונקציית**

**בחרנו אקראית 10 משפטים מתוך קורפוס הוועדות, כל משפט באורך של לפחות 5 טוקנים. מסכנו 10% מהטוקנים בכל משפט (לפחות טוקן אחד) בעזרת "[\*]". את המשפטים המקוריים כתבנו לקובץ original\_sampled\_sents.txt ואת הממוסכים כתבנו לקובץ  
.masked\_sampled\_sents.txt**

**ניבוי הטוקנים החסרים (generate\_results) :  
לאחר שמסכנו את הטוקנים, השתמשנו במודל המליאה (plenary) לחזות מה הטוקן החסר.  
למשל, אם המשפט היה "רק ביום [\*] *ראיתי את הפקס", המודל ניסה לנבא את המילה במקום "[\*]*". בקובץ התוצאות sampled\_sents\_results.txt ניתן לראות שחזה "חמישי", כנראה כי "רק ביום חמישי ראיתי את הפקס" הייתה אופציה שהמודל העריך כבעלת הסתברות גבוהה יחסית.**

**כמו כן, חישבנו את ההסתברות של המשפט המשוחזר בכל אחד משני המודלים (מליאה וועדה) והדפסנו את התוצאות. ניתן לראות שהסתברויות הלוג משתנות, ולעיתים המשפט מתאים יותר למודל המליאה ולעיתים למודל הוועדה.**

**חישוב Perplexity:  
חישבנו את ה- Perplexity עבור הטוקנים הממוסכים בלבד Perplexity .הוא מדד לאיכות המודל: ככל שהוא נמוך יותר, כך המודל מתאים יותר לנתונים.  
בקובץ perplexity\_result.txt התקבל ערך גדול מאוד (5774.29). משמעות הדבר היא שהמודל, במימוש הנוכחי (עם הענשות ומשקלים מאוד מוטים לטריגרמות ולפיסוק) לא מנבא היטב את הטוקנים הממוסכים. זה צפוי, שכן המודל הוא די בסיסי, ובנוסף החלת ענישות על טוקנים נפוצים עלולה להקשות על המודל ולגרום לו לבחור טוקנים פחות מתאימים.**

**הסבר על הבחירות והמשקלות:  
בחרנו משקלים (λ1, λ2, λ3) המתעדפים באופן כבד את הטריגרמה, בתקווה שהקונטקסט הרחב יותר יסייע לחיזוי מדויק.**

**הוספת ענישה לטוקנים נפוצים הייתה ניסיון למנוע מצב בו המודל כמעט תמיד בוחר מילות עצירה, פיסוק או מילות מפתח חוזרות. עם זאת, הענשה כזו עלולה לפגוע באיכות הניבוי ובהתאמה לנתונים האמיתיים.**

**סיכום:  
בתרגיל זה בנינו מודלי שפה טריגרמיים עבור פרוטוקולים של הכנסת, למדנו להעריך הסתברויות למשפטים ולטוקנים, ולאתר קולוקציות נפוצות לפי תדירות או TF-IDF. ביצענו גם משימות מסכה וניבוי טוקנים חסרים, וחישבנו Perplexity כדי להעריך את ביצועי המודל. התוצאות מראות כי למרות שהמודל יכול לנבא טוקנים ולהציג הסתברויות, ההוספה של ענישות וחלוקת משקלים קיצונית ייתכן ופגעה באיכות החיזוי. בנוסף, הקולוקציות הנפוצות מראות את המבנה החוזר והפורמלי של הפרוטוקולים. כנראה שמודל מתוחכם יותר, או שימוש במשקלים מאוזנים יותר, עשויים לשפר את ביצועי המודל ולהפחית את ה-Perplexity.**