**יבש 1 – NLP**

מגישים: אמיר בלדר – 204179659, רועי גנץ - 204506349

חלק 1

1. בעיית ה- OOV הינה בעיה מוכרת, כיצד נוכל לסווג מילים שלא ראינו קודם לכן באימון?   
   ישנם מספר סוגים של "משפחות" מאפיינים שיכולים לעזור בכדי לנתח בעיית OOV.
2. מאפיינים שתמסתמכים על ה-tags הקודמים למילה שלא נראתה.
3. מאפיינים שתמסתמכים על ה-tags האוחרים למילה שלא נראתה.
4. מאפיינים שתמסתמכים על המילים שלפני המילה שלא נראתה.
5. מאפיינים שתמסתמכים על המילים שלפני המילה שלא נראתה.

כמו שבני אדם לא בהכרח מבינים את כל המילים במשפט שהם קוראים ויכולים להבין מילים מההקשר שלהן, כך גם המודל יכול לעשות בעזרת פרמטרים מתאימים. כלומר, אפשר להתסמך על המילים והתיוגים שהיו לפני ואחרי המילה הלא מוכרת בכדי לסווגה נכונה.

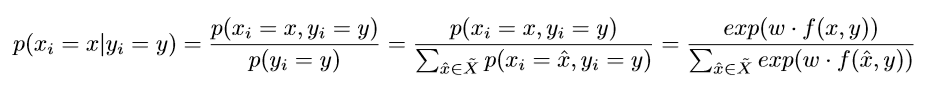
בנוסף, אנחנו יכולים להסתכל גם על משפחות מאפיינים שבהן יש משמעות לחלקים מסוימים במילה, לדוגמא:

1. שימוש באותיות גדולות/קטנות – מה שיכול לרמז על שם של דבר כלשהו.
2. שימוש בתחיליות/סופיות שחוזרות על עצמן.
3. שימוש במספרים בתוך מילים.

שימוש במאפיינים מעין אלה יהווה פתח לניתוחן של מילים רבות שלא ראינו עד כה, ולהוות לפחות ניתוח ראשוני ובסיסי עבור אלה בעיקר השימוש בתחיליות/סופיות שחוזרות על עצמן יכול להיות יעיל, שכן לרוב מדובר בתוספת למילה אחרת, שיתכן שכן הופיעה מוקדם יותר במודל.

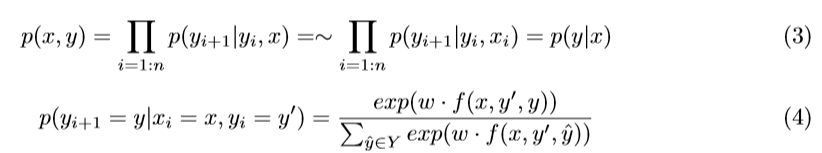
1. מדובר בעצם במרחב האפשרויות של וקטורי הייצוג. מרחב זה יהיה מורכב מכמות התיוגים והמילים האפשרויות. לכל מילה יש 100,000 אפשרויות, ולכל תיוג 25 אפשרויות ולכן:

כלומר, שמספר המאפיינים האפשריים הוא .

1. הפרמטרים עבור המודל לא יכולים להיות מחושבים בצורה יעילה בעבור מודל בגודל שכזה, שכן מרחב האפשרויות של כל מאפיין הינו גדול **מאוד**. במודל "סביר" ישנם בערך אפשרויות לכל וקטור מאפיינים. אנחנו מבינים שבמקרה זה קיבלנו מספר אפשרויות גדול בעשרות מונים, כלומר במספר סדרי גודל לא מבוטל. המשמעות תהיה כמובן שברוב הכולל של המקומות בכל וקטור מאפיינים יהיו אפסים. זה המצב ברוב וקטורי המאפיינים, אך כפי שציינו במקרה זה יהיו **מספר סדרי גודל יותר של אפסים**, מה שיהפוך את החישובים עצמם ללא יעילים ואת ריצת האלגוריתם ללא יעילה.
2. נתחיל מלציין עובדה חשובה – פונקציית ה-exp הינה פו' מונטונית עולה, שהינה גדולה מאפס. על כן, השימוש בה, יחד עם הנרמול במכנה, משאיר את התוצאה בתחום [0,1], כיאה להסתברות. בנוסף, בשל היותה מונוטונית עולה, השימוש בה לא ישנה את ה"מגמה" ואת מציאת הכיוון הנכון של הגרדיאנט לו נזדקק לעדכון המשקלים בהמשך. אם כן, השימוש ב-exp משאיר אותנו בתחום בין [0,1], ואם לא היינו משתמשים בה היינו יכולים לקבל גם תוצאות שליליות, מה שיכול היה לסבך את חישוב הגרדיאנט בהמשך.
3. בהנתן בחירה בין שימוש בייצוג בינארי עבור מאפיינים לבין ייצוג בעזרת מספרים, נעדיף להשתמש בייצוג בינארי. זאת ממספר טעמים. הראשון שבהם הוא עניין "קרבת" המאפיינים. כלומר, בהנתן שיש לנו onehot vector, כל המאפיינים קרובים זה לזה במידה שווה, בהיבט של מרחק האמינג. לעומת זאת, בייצוג מספרי, נקבל כי מאפיינים מסוימים קרובים יותר למאפיינים אחרים ללא הצדקה בהכרח. לדוגמא, אם התיוג NN (שם עצם) יקבל את הספרה 1, התיוג JJ (תואר) את הספרה 2 והתיוג VB (פועל) את הספרה 3. נקבל בפועל כי תואר דומה יותר לפועל מאשר ששם עצם דומה לפועל. כמובן שזאת לא בהכרח הנחה נכונה...   
   הייצוג הבינארי ימנע מצבים מעין אלה. בנוסף, הייצוג הבינארי ניתן לדחיסה בצורה יעילה יחסית בעזרת ספריות מוכרות בפייתון, כך ששימוש נבון בו הוא לרוב יעיל למדי.
4. נבחן את הנוסחא:  
   

נשים לב כי החישוב של המכנה הוא בעל אפשרויות רבות מאוד:  


כלומר, החישוב של המכנה ידרוש מעבר על המון אפשרויות שונות, ביצוע exp על כל אחת מהן ולאחר מכן סכימתן. כפי שציינו קודם לכן, מדובר על סדרי גודל גדולים למדי של אפשרויות שונות, ועל כן, החישוב עצמו יארך זמן רב מאוד. על כן, ככל הנראה שלא יהיה משתלם להשתמש בנוסחא המדוברת.

1. עתה אנו מסתכלים על נוסחאות אחרת:  
   

כלומר שבמקרה הזה אנו מסתכלים על המכנה:   
כאשר הוא קטן משמעותית מאשר . מהווה את מרחב התיוגים השונים, כאשר הדוגמא שראינו בשאלה 2, גודל המרחב היה 25 תיוגים, ובתרגיל הבית שבו אנו השתמשנו במרחב תיוגים סטדנדרטי הוא היה בגדול 45. על כן, מדובר עתה בבעיה שניתן באמת לחשבה בצורה סבירה.

1. המודל של HMM הינו מודל גנרטיבי והמודל של MEMM הינו דיסקרימינטיבי. ככלל אצבע, מודל גנרטיבי משערך את המודל ההסתברותי של כל אחד מה-class-ים האפשריים בבעיה, בעוד שמודל דיסקרימנטיבי משערך את הגבולות המבדילים בין ה-class-ים האפשריים. אם כן, שימוש במודל גנרטיבי יכול להיות טוב יותר למשימות מסוימות, לדוגמא אם נרצה ליצור משפטים.   
   בנוסף, HMM הוא מודל פשוט הרבה יותר להבנה, בדיוק בשל הסיבה שהוא משערך את ההסתברות של כל class ישירות. כך בעצם קל יותר לנתח את התוצאות ולהבין איפה המודל טעה, לפחות קל יותר מאשר MEMM.

חלק 2

1. נציב את משוואה 7 לתוך משוואה 8 ונפתח את הביטוי:

*המעבר הראשון הינו הצבה של משוואה 7 ואילו המעבר השני הינו שימוש בחוקי לוגריתמים.*

1. *כעת, נגזור את הביטוי מהסעיף הקודם לפי :*
   1. *במידה וישנן 500,000 מילים וכל אחת מיוצגת ע"י וקטור באורך 500, מספר הפרמטרים הינו מכפלת שני המספרים הללו:*
   2. *באופן כללי בלמידת מכונה ישנו כלל אצבע לפיו על כמות המידע להיות גדולה מכמות הפרמטרים. הדבר חיוני להצלחת תהליך האופטימיזציה וללמידת המודל. היות ובמודל שלנו יש סדר גודל של פרמטרים, נזדקק לכמות דוגמאות בסדר גודל של לפחות .*
   3. *מודל שכזה אינו מעשי מבחינה חישובית, בשל גודלו העצום. למשל, לצורך חישוב הביטוי במכנה ה-softmax, עלינו לחשב אקספוננט של מ"פ בין וקטורים באורך 500, חצי מיליון פעמים ולסכום את כולם. זהו כאמור רק חלק קטן מתהליך האופטימיזציה שכן יש לבצעו עבור כל אפשרות ל ולעבור על כל הזוגות האפשריים . לכן, מודל זה אינו מעשי חישובית.*
2. *להלן הביטוי אותו נתבקשנו לפתח:*
3. *כעת נגזור את הביטוי לפי :*
4. *ישנו קושי חישובי מהותי במודל זה, היות שסיבוכיות החישוב לעיל הינה . גודל זה הינו עצום ולא מעשי מבחינה חישובית היות וקבוצה V הינה כל המילים בשפה.*
5. *לפי הסעיף הקודם, ישנו קושי חישובי בתהליך האופטימיזציה של המודל. נשים לב כי הגרדיאנט שפותח בסעיף 5 מורכב מ-2 ביטויים – הראשון סוכם על פני סט זוגות מילים עוקבות מתוך הקורפוס הנתון, בעוד שהשני עוקב אחר כל יתר זוגות המילים. גודל הסט השני גדול משמעותית מגודל הסט הראשון שכן כמות המילים השפה ענקית וכמות הזוגות שיופיעו יחדיו קטנה משמעותית מסך כל האפשרויות לצימוד מילים. על כן, הביטוי השני הינו הביטוי היקר יותר חישובית.*

*יתר על כן, היות והקבוצה גדולה משמעותית מ D, לביטוי השני בפונקציית ה-loss חשיבות גדולה יותר מהביטוי הראשון, דבר אשר עלול להביא לכך שהמודל יקנה חשיבות יתרה ללמידה של ייצוגים וקטוריים רחוקים עבור מילים שאינן הופיעו צמודות זו לזו ב-D מאשר למידת ייצוגים דומים עבור מילים שכן הופיעו בצמידות. ניתן לחשוב על פונקציית הloss כבעלת 2 חלקים – הראשון הוא הobjective – לייצג מילים דומות ע"י וקטורים דומים ואילו השני הוא הרגולריזציה– לייצג מילים שלא הופיעו בסמיכות ע"י וקטורים שונים. תחת הסתכלות זו, פונקציית הloss מקנה חשיבות יתר לרגולריזציה מאשר ל-objective האמיתי, כלומר, חשיבות יתר ללמידת ייצוגים שונים עבור מילים מ- מאשר למידת ייצוגים דומים עבור זוגות מילים מ-D.*

1. *שיטת ה negative sampling מסייעת הן עם הבעיה החישובית אשר הוצגה בסעיף 6, והן עם הבעיה שהוצגה בסעיף 7. במקום לחשב סכום של כל זוגות הקבוצה , נחשב סכום של כל זוגות הקבוצה- D’, אשר קטנה ממנה משמעותית (אנו בוחרים את גודלה). כמו כן, היות ו-D’ קטנה משמעותית מ, המשקל של ביטוי זה בביטוי הLoss הכולל קטן אף הוא, דבר אשר עתיד לאפשר למודל להקנות חשיבות יתרה ללמידת ייצוגים וקטוריים דומים עבור מילים שהופיעו בהקשרים דומים.*
2. *אנו צופים כי מילים בעלות ייצוגי word2vec דומים תהיינה associated ולא similar מכיוון שהביטוי אותו אנו ממזערים מתחשב בזוגות מילים עוקבות. מילים דומות במשמעות בד"כ לא תופענה זו בסמוך לזו אלא זו במקום זו, אי לכך, מילים כאלו יופיעו לרוב בקבוצה D’ – קבוצת מילים עבורן לא נדרוש דמיון בווקטורי הייצוג. לעומת זאת , מילים שהן associated תופענה לעיתים קרובות בסמוך זו לזו, כלומר, בקבוצה D, דבר שיביא לכך שווקטורי הייצוג שלהן יהיו דומים.*