שיטות בעיבוד שפה טבעית

דו"ח הסברים וניתוח תוצאות

מגישים: אופיר ברעם 314968595

אורי ברזאני 316137371

תאריך הגשה: 13.02.2024

האימון

המאפיינים (features) בהם השתמשנו באימון המודל הם המאפיינים שהוגדרו לנו במסגרת התרגיל, כלומר $f100 - f107$ כפי שהוגדרו בהרצאה וכן את המאפיינים שהיה נדרש לממש אשר תופסים מספרים ומילים המכילות אותיות גדולות. באופן פורמלי:

$$f_{is\_number}(h,t) = \begin{cases} 1, & if\ the\ current\ word\ (w_i)\ contains\ a\ capital\ letter \\ 0, & otherwise \end{cases}$$

$$f_{capital\_letter}(h,t) = \begin{cases} 1, & if\ the\ current\ word\ (w_i)\ contains\ a\ digit \\ 0, & otherwise \end{cases}$$

בנוסף לכך, הוספנו מאפיין אשר תופס את אורך המילה הנוכחית וכן את התיוג שלה:

$$f_{leng}\ \ (h,t) = \begin{cases} 1, & if\ the\ current\ word\ (w_i)\ contains\ a\ digit \\ 0, & otherwise \end{cases}$$

ראינו במחקר שנעשה באוניברסיטת Stanford על תיוג חלקי דיבר של מילים בשפה הסינית, כי מאפיין עבור אורך המילה הינו יעיל. לדוגמה, נראה במחקר כי מילים בעלות יותר מ-3 תווים הן בדרך כלל שמות עצם, מספרים וניבים. כלומר, מילים קצרות ומילים ארוכות לרוב משוייכות לחלקי דיבר שונים.
בעקבות זאת, החלטנו לבדוק האם המאפיין יכול להיות שימושי גם במקרה שלנו בו אנו מתייגים את השפה האנגלית מתוך הנחה כי ישנם מאפיינים "גלובליים" על אף השוני בן השפות. ראינו כי המאפיין אכן תרם בהבדלת מילים להחלקי הדיבר המתאים שלהן, והחלטנו להשתמש בו לאימון המודלים.
(מקור https://web.stanford.edu/~jurafsky/slp3/old_oct19/8.pdf)

כמו כן, הוספנו מאפיין אשר בודק האם קיים אוגד (be) לפני המילה הנוכחית:

$$f_{be\_before\_adj}(h,t) = \begin{cases} 1, & if\ the\ current\ word\ (w_i)\ comes\ after\ to-be\ verbs\ (copulas) \\ 0, & otherwise \end{cases}$$

בהרבה מהמקרים בשפה האנגלית, לפני שם תואר מופיע **אוגד** (למשל 'I **am** smart') ולכן ראינו בזה כמאפיין שיכול לתפוס שמות תואר. בנוסף, בבדיקת התוצאות ראינו כי לעיתים שמות תואר תוייגו במקרים אלו כשמות עצם ואכן ראינו כי הדבר תרם לדיוק המודל.

<u>נתונים על המודלים:</u>

	מודל 1	מודל 2
מס' המאפיינים שנוצרו	you have 120512 features 120512 f100 15415 f101 11090 f102 20392 f103 8150 f104 1060 f105 44 f106 32132 f107 30793 is_number 6 capital_letter 33 length 305 be_before_adj 1092	you have 15040 features! 15040 f100 1805 f101 2235 f102 3653 f103 1168 f104 316 f105 32 f106 2801 f107 2694 is_number 4 capital_letter 16 length 199 be_before_adj 117
משך האימון	train time: 415.0872073173523 seconds	train time: 4.932167291641235 seconds
אחוז הדיוק	Accuracy: 0.9873742970898494	Accuracy: 0.9983547219480092

<u>מפרט החומרה בה עבדנו:</u>

מעבד	11th Gen Intel(R) Core(TM) i7-1165G7 @ 2.80GHz 2.80 GHz
מותקן RAM	8.00 GB (7.65 GB ניתנים לשימוש)

הסקה

תחילה החלטנו לממש את האלגוריתם Viterbi כפי שראינו בכיתה. לאחר מספר נסיונות הרצה, נוכחנו לגלות כי זמן הריצה ארוך מאוד (מעבר לשעה) והדבר מקשה על בדיקת המאפיינים ונכונות מימוש האלגוריתם. לפיכך, החלטנו להוסיף beam search כפי שלמדנו. לאחר בחינת מספר אפשרויות לגודל התכונה, ראינו כי השפעת גודל ה beam search אינו משמעותי מעבר לגודל 2, ולכן החלטנו על ערך זה עבור beam search.

הוספת ה beam search הפחיתה את משך זמן הריצה בצורה משמעותית, מכ-50 דקות לכ-20 דקות על קובץ הבדיקה הראשון (test1), אשר אפשר לנו לבצע ייעול של הקוד בצורה טובה יותר.

כמו כן, בצענו בשני המודלים cross validation לקביעת ההיפר-פרמטרים – lambda וה- threshold. בתוצאות הבדיקה קיבלנו כי ערכי ההיפר-פרמטרים הברירות מחדל, כלומר הערך 1 עבור שניהם, הניב לנו את אחוזי הדיוק הגבוהים ביותר.

בנוסף, לצורך ייעול ריצת הקוד, הפחתנו את מספר האיטרציות לכל משפט ע"י תיוג מקרי הבסיס באיטרציות $k == 2$ ו- $k == 3$. כאשר $k == 2$, התיוגים המתאימים עבור המילה הקודמת $'u'$ והמילה הקודמת קודמת $'t'$ הם $'*'$ ובאופן דומה עבור $k == 3$, התיוג של $'t'$ הינו $'*'$. הפחתת מס' האיטרציות מוביל להפחתת במספר החישובים המבוצעים ע"י פונקציית ההסתברות q ובמציאת הציון המקסימלי כבאלגוריתם המקורי.

המבחן

<u>מודל 1</u>

<u>אחוזי דיוק</u>: Accuracy: 0.9599138295176143

<u>מטריצת הבלבול של עשר התיוגים שהמודל טעה בהם הכי הרבה</u>:

Confusion Matrix:

	NN	JJ	IN	VBN	NNP	RB	NNPS	VBD	VB	VBZ
NN	3166	82	0	2	27	6	0	1	10	0
JJ	72	1338	1	27	21	12	0	6	8	0
IN	3	1	2492	0	3	47	0	0	0	0
VBN	3	29	0	436	0	0	0	30	0	0
NNP	9	21	1	0	1928	0	13	1	1	0
RB	10	16	15	0	0	729	0	0	1	0
NNPS	0	4	0	0	28	0	28	0	0	0
VBD	3	1	0	35	0	0	0	791	2	0
VB	17	3	1	0	1	0	0	1	551	0
VBZ	0	0	0	0	0	0	0	0	0	476

את טעויות תיוג המילים החלטנו למדוד ע"י סכימת הטעויות (במטריצה ניתן לראות זאת על ידיי סיכמת הערכים בשורות מלבד ערכי האלכסון). השורות במטריצת הבלבול מייצגות את התגיום האמיתי של המילים בעוד שעמודות המטריצה מייצגות את התיוגים בעקבות החיזוי. נסתכל על שני התיוגים עבורם נוצר הבלבול (הטעויות) הרב ביותר: "NN" ו- "JJ" . על מנת למנוע בלבול בין התיוגים, נציע כפתרון הוספת משפחות מאפיינים נוספות אשר ינסו לתפוס מידע וקונטקסט על זוג התיוגים הנ"ל, למשל, תיוגים אשר בוחנים סיומות הידועות בשפה האנגלית כמשוייכות לשמות תואר, לדוגמה עבור הסיומות {'-ist','-esque','-able'}. מאפיינים ייעודיים אלו יאפשרו להבדיל בין המילים ויעזרו למודל ללמוד אותם בצורה מדוייקת יותר.
(דוגמאות לסיומות נפוצות לשמות תואר ושמות עצם בהתאמה: English adjectives - Wikipedia)

מודל 2

מאחר כי לא ניתן קובץ מבחן ייעודי למודל 2 וכן כי גודל המדגם במודל זה הוא קטן, השתמשנו בשיטת $cross\ validation$ אשר בעזרתה פיצלנו את המדגם למדגמי אימון וואלידאציה (ביחס של 70-30 כפי שמקובל) כדי לייצר קובץ $test$ באופן מלאכותי.
לצורך כך נעזרנו ב $k-fold\ cross-validation$ לשיפור המודל והתחזיות. בכל איטרציה נבחר ערך שונה להייפר פרמטר k, אשר לפיו חילקנו את מדגם האימון ל-k מדגמים באופן אקראי וללא סדר קבוע של משפטים ($shuffle$) כדי להמנע מ- $overfit$. בכל איטרציה, קבוצה אחת נבחרת עבור מדגם הוואלידאציה ומתאמנים על יתר הקבוצות. הערך k ייבחר לפי תוצאות הדיוק הממוצע הטובות ביותר מכל האיטרציות.
במקרה שלנו, בחרנו $k=7$. לפי תוצאות ה- $cross\ validation$, עבור ערך k זה נצפה לקבל כ93 אחוזי דיוק במודל 2.

תוצאות הדיוק במסגרת ה $k-fold\ cross\ validation$:

```
Accuracy for fold #0: 0.9354473386183465
Accuracy for fold #1: 0.927360774818402
Accuracy for fold #2: 0.9154228855721394
Accuracy for fold #3: 0.9247546346782988
Accuracy for fold #4: 0.9343065693430657
Accuracy for fold #5: 0.930622009569378
Accuracy for fold #6: 0.943728018757327
Average accuracy over 7 folds: 0.9302346044795654
```

התחרות

מעבר לשינויים שצויינו באימון ובהסקה, לא ביצענו שינויים נוספים. אנו מצפים לקבל כ-96 אחוזי דיוק על מודל 1 בהסתמך על תוצאות הדיוק באימון, שכן מדגם סט האימון של מודל 1 היה גדול מאוד (5000 משפטים) ומצופה כי הצלחנו למצוא קשרים סמנטיים רבים. יש לציין כי ייתכן פער בין אחוזי הדיוק, שכן קיימים הבדלים בתוכן הקבצים, ואלו יכולים להשפיע על דיוק החיזוי. ייתכנו קשרים סמנטיים בין מילים שהופיעו בקובץ אחד אך לא באחר, וכמו כן ייתכנו קיומם של מילים בקובץ אחד לעומת הקובץ השני.

התחזית שלנו לתחרות מודל 2 מתבססת על ה$cross\ validation$ שבצענו כפי שתואר בפרק המבחן של מודל 2. כאמור, התחזית שלנו למודל 2 היא כ- 93%.

חלוקת העבודה בתרגיל

העבודה בוצעה בשותפות מלאה בין שני השותפים – אורי ואופיר. בניית הקוד בוצעה יחדיו לאורך כל חלקי הקוד. בנוסף, באופן מקבילי, אופיר הכין את הדו"ח תוך התייעצות וקבלת משוב מאורי בזמן שאורי שיפר את מבנה הקוד, משך ריצתו וביצועיו.