

עיבוד שפה טבעית - תרגיל בית 1

תיאור המשימה

בתרגיל בית זה תממשו מודל MEMM (כפי שנלמד בשבוע 3), המשמש לתיג סדרתי של חלקי דיבר במשפט (Part of Speech Tagging), תבצעו משימות עיבוד שפה על נתונים אמיתיים ותנתחו את טיב הצלחתכם.

התרגיל יממש על ידכם בשפה Python 3, ועליו לרוץ באופן תקין בסביבת העבודה שתסופק לכם. בתרגיל תידרשו לממש שני מודלים – מודל 1 (הגדול) ומודל 2 (הקטן), לטובת כל מודל מצורפים בקובץ data.zip קבצי הנתונים הרלוונטיים לאותו מודל בלבד. לאורך התרגיל הכוונה במדד דיוק (Accuracy) היא לדיוק ברמת המילה. כלומר, שרשור כל הפרדיקציות של המודל שלכם והתיגים האמיתיים על פני כל המשפטים, וחישוב דיוק בין שתי הרשימות.

הסבר על מבנה הציון בתרגיל:

- **45%** - מימוש מלא של מודל 1 (הגדול) כפי שיפורט בהמשך ואימונו על קובץ ה-tag.train1, ודיווח אחוז הדיוק (Accuracy) על קובץ ה-tag.test1.
- **20%** - תחרות מבוססת Accuracy בתיג קובץ התחרות המתאים למודל 1 (הגדול) comp1.words
- **20%** - מימוש ואימון מודל 2 (הקטן) על קובץ ה-tag.train2, תיג קובץ התחרות המתאים למודל 2 (הקטן) comp2.words – **גם תחרות זו מבוססת Accuracy**.
- **5%** הערכת ביצועי מודל 2.
- **10%** - כתיבת דו"ח תמציתי (עד עמוד אחד בפונט אריאל בגודל לפחות 10 עם שוליים סטנדרטיים אשר יכלול את הסעיפים הנדרשים ועמידה בתנאי פורמט ההגשה (יפורטו בהמשך המסמך). דו"ח שלא יעמוד בדרישות ההגשה יקבל 0/10.

נתונים:

הסבר על הקבצים המצורפים –

מודל 1 (הגדול):

1. train1.wtag – קובץ המכיל 5000 משפטים מתויגים. עליכם להשתמש בקובץ זה בשלב האימון דוגמא לפורמט של הנתונים – בתרגול 3 ראינו את הדוג' הבאה -

the dog barks -> D N V STOP

בפורמט הנוכחי הדוג' תראה כך:

the_D dog_N barks_V .

שימו לב כי כל המשפטים (או רובם) מסתיימים בנקודה ('.'), אולם לא כל נקודה סוגרת משפט. הסכימה (scheme) לפיה המשפטים מתויגים נקראת Penn Treebank, וניתן למצוא עליה עוד אינפורמציה [כאן](#).

2. test1.wtag – קובץ המכיל 1000 משפטים מתויגים, בפורמט זהה לפורמט של הקובץ הקודם. עליכם להשתמש בקובץ זה בשלב המבחן של מודל 1 (הגדול)

3. comp1.words – קובץ המכיל 1000 משפטים לא מתויגים. המשפטים מופיעים בצורתם הטבעית, לדוג':

the dog barks .

מודל 2 (הקטן):

1. train2.wtag – קובץ המכיל 250 משפטים מתויגים בפורמט זהה לפורמט קובץ האימון ממודל 1 (הגדול) המתואר לעיל. עליכם להשתמש בקובץ זה (train2.wtag) בשלב האימון של מודל 2 (הקטן).

2. comp2.words – קובץ המכיל 1000 משפטים לא מתויגים. המשפטים מופיעים בצורתם הטבעית, לדוג':

the dog barks .

אימון (Train):

מודל 1 (הגדול):

כאמור, אמידת וקטור הפרמטרים תיעשה על הקובץ train1.wtag.

תידרשו לממש מודל המורכב מסט מאפיינים (Features) $f_{100} - f_{107}$, כפי שהוגדרו בהרצאות [(Ratnaparkhi, 96)] ומצורפים לנוחיותכם בסוף מסמך זה. בנוסף, אנא ממשו מאפיינים התופסים מספרים ומילים המכילות אותיות גדולות (Capital letters). את אלו לא נגדיר כאן באופן מפורש, אתם רשאים להגדיר אותם כרצונכם. כמו כן, אתם רשאים להגדיר מאפיינים נוספים כרצונכם, אך לא להחסיר מסט המאפיינים הנדרש.

הבהרה: באפשרותכם להחליט שמאפיינים מסויימים לא יקבלו ייצוג במודל הסופי שלכם, אך על המאפיינים שהגדרנו ($f_{100} - f_{107}$, מספרים ואותיות גדולות) להיות ממומשים בקוד שלכם.

שימו לב! גודל המודל המאומן צריך להיות 10,000 פרמטרים לכל היותר. אי עמידה בתנאי יגרור ציון 0 בתרגיל. הקוד המסופק לכם מדפיס את כמות הפרמטרים בתחילת הריצה, ויכול לשמש אותכם לבדיקה עצמית. (שימו לב שאינכם משנים את הפונקציה `get_features_idx`)

מודל 2 (הקטן):

כאמור, אמידת וקטור הפרמטרים תיעשה על הקובץ train2.wtag.

במודל זה, אתם רשאים להגדיר ולממש מאפיינים כרצונכם, כלומר ניתן להשתמש בכל קומבינציה של מאפיינים שמימשתם עבור מודל 1 (אך לשים לב שהאימון מתבצע על הקובץ המתאים) וכאלו שאתם מוסיפים עבור מודל 2.

בדו"ח שתכינו (עוד על כך בהמשך) יש לציין את סוג המאפיינים בהם השתמשתם (למשל באופן בו מתוארת f_{100} בהרצאה) בכל מודל.

את המאפיינים שבחרתם להוסיף לכל מודל יש להגדיר במפורש.

יש לציין ולהסביר כל שיפור שהכנסתם למודלים הסופיים (לדוג' קיצוץ של מאפיינים ש"לא הופיעו מספיק"), כולל המוטיבציה לבצע אותו.

שימו לב! על המודל המאומן להכיל לא יותר מ-5,000 פרמטרים. גם כאן, אי עמידה בתנאי זה יגרור ציון 0 על תרגיל הבית.

הסקה (Inference):

עבור שני המודלים, ההסקה תתבצע ע"י אלגוריתם Viterbi כפי שנראה בהרצאה ובתרגול. יש לציין כל חריגה ושיפור שהכנסתם לאלגוריתם הבסיסי, את המוטיבציה לחריגה וכן את תרומתה.

מבחן (Test):

עבור מודל 1 (הגדול) יש לבצע הסקה (Inference) על הקובץ test.wtag, ולדווח את תוצאות הדיוק (Accuracy) ברמת מילה.

נסו להעזר בקובץ test על מנת להבין מהן הטעויות הנפוצות של המודל (למשל ע"י שימוש ב Confusion Matrix), וחשבו כיצד אתם יכולים לשפר המודל בהתאם. אין צורך לדווח על התהליך, רק על השינויים שהוכנסו למודל הסופי.

עבור מודל 2 (הקטן) לא ניתן קובץ מבחן ייעודי, יש לחשוב כיצד להתמודד עם בעיות בחינת ביצועי המודל ומיעוט דוגמאות האימון ולתאר את דרך התמודדותכם בדו"ח בקצרה.

תחרויות:

לכל אחד מן המודלים, יש לבצע הסקה (Inference) על קובץ התחרות המתאים (אשר אינו כולל תיוגים), ולכתוב את תוצאות התיוג לתוך קובץ חדש בפורמט wtag (כמו קבצי האימון) (שמות הקבצים הרצויים מופיעים בהמשך). לדוג', עבור המשפט:

the dog barks .

יש לבצע הסקה, שתיתן לכם את תוצאות התיוג. בהנחה שהתיוג שהתקבל הוא "D N V", יש לכתוב עבור שורה זאת את השורה הבאה –

the_D dog_N barks_V .

שימו לב שהקבצים שאתם מגישים לא צריכים לכלול סימני כוכביות ו STOP, ושסדר המשפטים (הלא מתוויגים) בקובץ המקורי זהה לסדר המשפטים בקובץ הפלט.

יש לתאר במפורש מה עשיתם כדי לקבל את התוצאות שקיבלתם (שינויים שביצעתם בלמידה, בהסקה וכו').

בנוסף, אתם מתבקשים לכתוב: תחזית של אחוז הדיוק שאתם צופים לקבל, וכן להסביר מדוע עשוי להיות הבדל בין הדיוק על קובץ התחרות וה-test. הסברים חכמים אף עשויים לקבל בonus.

סביבת עבודה:

על הפרויקט לרוץ על המכונה שניתנה לכם.

הקוד שסופק לכם

הקוד שסופק לכם מכיל את הקבצים הבאים:

preprocessing.py – קובץ המכיל את יצירת הפיצרים וההיסטוריות. ניתן לכם קוד המממש את קבוצת הפיצרים 100, אתם נדרשים לממש את יתר הפיצרים כפי שתואר בפרקים של מודלים 1 ו-2.

Optimization.py – קובץ המכיל את הנדרש לאופטימיזציה. את קובץ זה אתם לא נדרשים לשנות, אך יכולים לעשות זאת אם תרצו.

Inference.py – קובץ בו אתם נדרשים לממש את הפונקציה memm_viterbi וכל פונקציית עזר שתמצו, המממשת את אלגוריתם ויטרבי כפי שתואר בפרק inference.

main.py – קובץ המריץ את התכנית אותה כתבתם ומוציא כפלט את קובץ התיוגים.

קוד חיצוני המותר לשימוש:

החבילות בהן מותר להשתמש הן החבילות הסטנדרטיות בשפת python, וכן החבילות numpy ו scipy בלבד. (מותקנות בסביבה הסטנדרטית המצורפת המוזכרת מעלה)

בשלב האימון ניתן (מומלץ, ואולי אפילו הכרחי) להשתמש בחבילה המממשת את אלגוריתם [LBFGS](#) (לדוגמה ב-scipy), בתנאי שהיא עושה אופטימיזציה על פונ' המטרה וגרדיאנט שאתם מספקים לה.

למען הסר ספק - אסור להשתמש ב:

1. חבילות הממשות אלגוריתם Viterbi.
2. חבילות המממשות MEMM.
3. חבילות העושות עיבוד על טקסט – ספירת חזרות, uni\bi\tri gram וכו'.

הגשה:

קובץ zip בלבד, בשם HW1_123456789_987654321.zip (עבור שני סטודנטים שמספרי זהות שלהם 123456789 ו 987654321). הקובץ הנ"ל יכלול:

1. **דו"ח קצר ותמציתי** (עד עמוד אחד בגודל 10, שוליים סטנדרטיים
report_123456789_987654321.pdf) המכיל הסברים תמציתיים, דיווח וניתוח תוצאות, הכולל:
 - a. שמות המחברים ות"ז
 - b. אימון - דיווח אחוז דיוק (Accuracy) והערות על אימון כל מודל (לפי הדגשים בסעיף "אימון")
 - c. הסקה - הערות על אלגוריתם ההסקה (לפי הדגשים בסעיף "הסקה")
 - d. מבחן - דיווח אחוז דיוק (Accuracy) על קובץ המבחן עבור מודל 1 (הגדול). יש לצרף צילום מסך מתהליך המבחן של מודל 1 המכיל הדפסה של אחוז הדיוק שהתקבל.
 - e. תחרות - הסבר קצר על שיפורים שעשיתם למודלים עבור תיוג קבצי התחרות (לפי ההסברים כפי שמפורטים בסעיף "תחרות")

מכוון שמדובר בדו"ח תמציתי - הקפידו לדווח אך ורק על השינויים הסופיים אותם הכנסתם למודל, והמנעו מלתאר שיפורים שניסתם אך בחרתם לא להכניס בסוף.

2. **קבצי הקוד של התרגיל.** על הקוד להיות מתועד וקריא. בנוסף, הקוד צריך להיות מסוגל לרוץ על כל מכונה שהיא. אנא כתבו ממשקי הרצה פשוטים לאימון, מבחן וייצור קבצי התחרות המתויגים. על קבצי הקוד להיות בתיקייה code.
3. **קבצי התחרות מתויגים** – על קבצי התוצאות להיות בפורמט wtag (כפי שמפורט בחלק "אימון"), הכולל את המילים והתגים. על מנת לוודא נכונות ולהימנע מאי נעימות בנוגע לציון, אנא ודאו כי אם משמיטים את הקו התחתון והתגים מהקבצים המתויגים שאתם מגישים, מקבלים בדיוק את אותם משפטים (ולפי אותו סדר) ובאותו פורמט כמו בקובץ התחרות המתאים. חוסר התאמה פירושו ציון 0 בחלק הזה.
על שמות הקבצים להיות – (123456789 הוא ת"ז של אחד הסטודנטים)
a. wtag.comp_m1_123456789_987654321 – קובץ wtag שתויג ע"י מודל 1 (הגדול).
b. wtag.comp_m2_123456789_987654321 – קובץ wtag שתויג ע"י מודל 2 (הקטן).
4. **המודלים המאומנים** - עליכם לצרף להגשה את המודלים המאומנים weights_1.pkl ו- weights_2.pkl (בהתאמה לסוג מודל), כפי שהם נשמרים ע"י הפונקציה get_optimal_vector. חריגה מצורת השמירה בפונקציה תגרור הורדת ציון. על המודלים להיות בתיקייה trained_models.
5. **ממשק לתיג קבצי התחרות** - על קבצי התחרות להיות ניתנים לשחזור (Reproducible). הדרישה היא שניתן יהיה לקחת את הקוד והמודלים המאומנים שהגשתם ולייצר באמצעותם קבצי תחרות מתויגים זהים לחלוטין לקבצים שהגשתם. לטובת שחזור הקבצים, יש לכתוב ממשק הרצה פשוט, בקובץ נפרד בעל השם generate_comp_tagged.py – להרצת Inference בלבד על המודלים המאומנים ויצירת קבצי התחרות המתויגים ע"י כל מודל.

לנוחיותכם, הקובץ check_submission.py יבדוק את תקינות ההגשה שלכם. יש להריצו מתיקיה בה נמצאים הקובץ HW1_ID1_ID2.zip והתיקייה data. שימו לב כי הקובץ מכיל את הפונקציה compare_files אך אינו משתמש בו - באפשרותכם להשתמש בו על קובץ test על מנת לבדוק את תקינות פורמט התיג.

העתיקות:

בשל אופי המשימה והמורכבות שלה, קל לבדוק העתיקות של קטעי קוד \ קבצים מלאים. למען הסר הספק אנו מדגישים כי אין להעביר קוד בין סטודנטים, בין אם להגשה ובין אם לא.

כפי שמצויין בסילבוס, באפשרותכם להשתמש בכלי בינה מלאכותית (כגון chatGPT) - אך הקוד הוא באחריותכם ועליכם לוודא את תקינותו ונכונותו. כמו כן, גם את קטעי הקוד שנוצרו ע"י שימוש בבינה מלאכותית או את הפרומפט ששימש כדי ליצור אותם אין להעביר בניכם.

The Full Set of Features in [(Ratnaparkhi, 96)]

- ▶ Word/tag features for all word/tag pairs, e.g.,

$$f_{100}(h, t) = \begin{cases} 1 & \text{if current word } w_i \text{ is base and } t = \text{Vt} \\ 0 & \text{otherwise} \end{cases}$$

- ▶ Spelling features for all prefixes/suffixes of length ≤ 4 , e.g.,

$$f_{101}(h, t) = \begin{cases} 1 & \text{if current word } w_i \text{ ends in ing and } t = \text{VBG} \\ 0 & \text{otherwise} \end{cases}$$

$$f_{102}(h, t) = \begin{cases} 1 & \text{if current word } w_i \text{ starts with pre and } t = \text{NN} \\ 0 & \text{otherwise} \end{cases}$$

The Full Set of Features in [(Ratnaparkhi, 96)]

- ▶ Contextual Features, e.g.,

$$f_{103}(h, t) = \begin{cases} 1 & \text{if } \langle t_{-2}, t_{-1}, t \rangle = \langle \text{DT}, \text{JJ}, \text{Vt} \rangle \\ 0 & \text{otherwise} \end{cases}$$

$$f_{104}(h, t) = \begin{cases} 1 & \text{if } \langle t_{-1}, t \rangle = \langle \text{JJ}, \text{Vt} \rangle \\ 0 & \text{otherwise} \end{cases}$$

$$f_{105}(h, t) = \begin{cases} 1 & \text{if } \langle t \rangle = \langle \text{Vt} \rangle \\ 0 & \text{otherwise} \end{cases}$$

$$f_{106}(h, t) = \begin{cases} 1 & \text{if previous word } w_{i-1} = \text{the and } t = \text{Vt} \\ 0 & \text{otherwise} \end{cases}$$

$$f_{107}(h, t) = \begin{cases} 1 & \text{if next word } w_{i+1} = \text{the and } t = \text{Vt} \\ 0 & \text{otherwise} \end{cases}$$