דו"ח תרגיל בית 3

סיווג

שמות המגישים + ת"ז:

עוביידה חטיב, 201278066

מאיה עטואן, 314813494

שלב 1

מציאת שני הדוברים בעלי המספר הכי גדול של משפטים בקורפוס התבצעה ע"י כך שעשינו חילוץ לכל השורות בקובץ ה- JSON לתוך מבנה נתונים מסוג list, בו כל תא הכיל מילון המייצג את המשפט והמיתא דאטה שלו. לאחר מכן, עברנו על השורות, ולכל דובר ספרנו את כמות המשפטים שהוא אמר, סידרנו את המונים של הדוברים החל מהגדול, והחזרנו את השניים הראשונים. התוצאה שהתקבלה, המופיעה למטה, הם של "ראובן ריבלין" ו- "א' בורג".

בקוד, חילוץ השורות מתבצע ע"י הפונקציה 'json_lines_extract', וספירת הדוברים והחזרת השמות של השניים בעלי כמות המשפטים הגדולה ביותר מתבצעת ע"י הקריאה לפונקציה 'top_two_speakers'.

```
[('3146 ,'ראובן ריבלין'), ("2016 ,"א' בורג")]
```

שלבים 1.1, 1.2

על סמך התוצאה הנ"ל, "ראובן ריבלין" ו- "א' בורג" יהוו את שתי המחלקות speaker1, speaker2 בהתאמה.

על מנת לענות על דרישת המשימה, יצרנו מתודה בשם 'split_data_by_speaker'. המתודה מקבלת כקלט את רשימת השורות של קובץ ה- JSONL ושתי מחרוזות של שמות דוברים, ומחזירה 3 רשימות כך שבאחת את השורות שמשויכות לדובר הראשון, בשנייה את השורות המשויכות לדובר השני, ובשלישית את שאר השורות.

על מנת למקסם את כמות המשפטים עבור כל דובר, ועל סמך האבחנות שלנו מהמטלה הראשונה, הגרסאות הנפוצות בהם יכול שם הדובר להופיע מפורטות להלן ביחד עם איך התמודדנו עם כל אחת מהם:

- **כשם מלא** (שם פרטי + שם משפחה. למשל, ראובן ריבלין): שני שמות זהים בדיוק מבחינת האלגוריתם הם אותו אדם.
- **כקיצור השם הפרטי ע"י האות הראשונה בו ואחריה ' + שם המשפחה** (למשל, ר' ריבלין): אם שני השמות מורכבים משתי מלים לפחות, והמילה האחרונה בשם אחד זהה למילה האחרונה בשם השני, והאות הראשונה במילה הראשונה דומות בשניהם, ובאחד מהם האות השנייה במילה הראשונה היא ', האלגוריתם מתייחס לשניהם כאותו אדם.
- **כשם משפחה בלבד** (למשל, ריבלין): אם אחד לפחות משני השמות מורכב ממילה אחת, והיא דומה למילה האחרונה בשם השני, האלגוריתם מתייחס אליהם כאותו אדם.
- **כשם מלא הכולל את השם האמצעי**: אם שני השמות מורכבים מיותר ממילה אחת, ובשניהם המילה הראשונה זהה והמכילה האחרונה גם כן זהה, האלגוריתם מתייחס אליהם כאותו אדם.
- **ככינוי הידוע בו + שם המשפחה** (למשל, רובי ריבלין): זהו מקרה פרטי בו מתייחסים לפוליטיקאים ראובן ריבלין כ- "רובי ריבלין". אם אחד מהשמות בגודל יותר משתי מלים והמילה הראשונה בה היא "רובי", והמילה האחרונה דומה בשני השמות, אז הם אותו אדם מבחינת האלגוריתם.

בכל שאר המקרים האלגוריתם מתייחס לשני השמות כאל שונים, והמשפט מסווג לקטגוריית "other"

על סמך כך, האלגוריתם מסווג את הדוברים הבאים לאותה קטגוריית דובר מבין שתי הקטגוריות:

ראובן ריבלין

ר' ריבלין

רובי ריבלין

ריבלין

הפונקציה 'split_data_by_speaker' נקראה מה- ()main.

שלב 2

ה- down-sampling מתבצע ע"י הפונקציה 'down_sample' אשר מקבלת כקלט אוסף של רשימות של רשומות, מוצאת את האורך המינימלי מבין הרשימות, ומתוך כל אחת בוחרת

באופן אקראי שורות כאורך המינימלי שמצאה. הפונקציה מחזירה רשימות חדשות בעלות מספר זהה של רשומות. הפונקציה נקראה עבור כל אחת מהמשימות בנפרד (שתי הרשימות של הדוברים במשימה הסיווג הבינארית, ושתי הרשימות של הדוברים ביחד עם רשימת האחרים במשימת הסיווג מרובת המחלקות).

<u>עבור משימת הסיווג הבינארית</u>:

מספר השורות בכל אחת מהמחלקות לפני ביצוע הפעולה:

```
Sentences count of each binary classification class before the down sampling:
first_sentences: 3456
second_sentences: 2159
```

מספר השורות בכל אחת מהמחלקות לאחר ביצוע הפעולה:

```
Sentences count of each binary classification class after the down sampling:
first_sentences: 2159
second_sentences: 2159
```

<u>עבור משימת הסיווג מרובת המחלקות</u>:

מספר השורות בכל אחת מהמחלקות לפני ביצוע הפעולה:

```
Sentences count of each multiclass classification class before the down sampling:
first_sentences: 3456
second_sentences: 2159
other_sentences: 104800
```

מספר השורות בכל אחת מהמחלקות לאחר ביצוע הפעולה:

```
Sentences count of each multiclass classification class after the down sampling:
first_sentences: 2159
second_sentences: 2159
other_sentences: 2159
```

למרות שיכולנו לעשות את ה- Down-sampling של שתי המשימות בבת אחת, הוחלט לעשות את זה באופן נפרד לכל משימה, בשל אפשרות תיאורטית בה מספר הפריטים במחלקת 'other' יכול להיות פחות משתי המחלקות האחרות.

שלב 3

שלב 3.1

לשם יישום דרישות השלב, יצרנו את הפונקציה 'tfidf_vector_creator', אשר מקבלת אוסף של שורות כקלט, ומחלצת מתוכן את המשפטים (הטקסט). לאחר מכן, היא יוצרת ווקטוריזר מסוג Tf-idf, מתאימה אותו לטקסט ויוצרת מטריצה שמורכבת מהווקטורים של המשפטים, ומחזירה את המטריצה ביחד עם הווקטוריזר.

העדפנו להשתמש ב- Tf-idf על פני CountVectorizer, מפני שהוא ממשקל את הטוקנים לא רק לפי כמות הופעתם אלא גם לפי הייחודיות שלהם על פני המסמך, ומפני שחשוב לנו להדגיש את המלים הייחודיות של הדובר, דבר היכול לעזור בסיווג של משפט, בנוסף לרצון שלנו בלהמעיט מהערך של טוקנים נפוצים כמו "של" ו- "את" ו- "," אשר מצמצמים את ההבדלים בין משפטים, מכאן הסקנו ש- Tf-idf הוא הבחירה העדיפה.

שלב 3.2

על מנת לענות על דרישות המשימה, יצרנו את הפונקציה 'custom_vector_creator' אשר מקבלת כקלט אוסף של שורות (המשפט עם המטא-דאטה שלו), ורשימה של שמות הדוברים, ומייצרת וקטור מאפיינים לכל משפט המכיל את המאפיינים הבאים עם ההסבר של ההיגיון מאחורי בחירת כל אחד:

1. **מספר הכנסת**: זהו מאפיין חשוב, מפני שחברי הכנסת נבדלים במספרי הכנסת בהם השתתפו, ולא נרצה שמשפט יסווג לחבר כנסת שלא היה חלק מהכנסת בה המשפט נאמר.
2. **מספר הפרוטוקול**: בדומה למספר הכנסת, ובאופן יותר ספציפי, משפט נאמר בפרוטוקול בעל מספר שהחבר הכנסת השתתף בו מעלה את הסבירות שהוא שייך לחבר הכנסת ההוא.
3. **סוג הפרוטוקול**: מאפיין זה יכול להיות משמעותי בשל היותו משקף את ההקשר בו הדובר נוהג להשתתף. למשל, דובר בעל תפקיד מרכזי בכנסת, כמו יו"ר, צפוי להשתתף יותר בדיוני המליאה, לעומת חבר כנסת שהוא חבר ועדה אשר נוטה יותר להשתתף בדיונים של הועדות.
4. **אורך המשפט**: מאפיין זה יכול להבדיל בין שני דוברים בשל כמה סיבות, העיקרית בהם היא אופי אישי, יש דוברים שנוטים לדבר משפטים יותר ארוכים מאחרים. סיבה
5. **אם המשפט מכיל ספרה**: הכלת המשפט לספרה או שלא יכולה גם כן להעיד על סגנון אישי, למשל דוברים מסוימים יכול להיות עם נטייה של להזכיר עובדות או סטטיסטיקות יותר מאחרים. המאפיין יכול גם לנבוע מההקשר וסוגי הדיונים בהם הדובר נוטה להשתתף, כאשר למשל, השפה המקצועית של דיוני הועדות יכולה לכלול יותר מספרים מאשר הדיונים של המליאות.

6. **מאפיינים הנוגעים לכמות הופעת קולקציה מסוימת בתוך המשפט מתוך קבוצת קולקציות**: מאפיין זה של קולקציות חשוב מפני שדובר יכול להיות בעל נטייה להשתמש בקולקציות מסוימות כמו "אני", "אדוני", ו- "גברתי". כמו כן, הקולקציות יכולות לשקף את הנושאים בהם חבר הכנסת יותר מעורב, או אם הוא חלק מהקואליציה או האופוזיציה. למשל בכנסת הנוכחית, השימוש במילה "רפורמה משפטית" יכול להעיד על חבר כנסת מהקואליציה, לעומת חבר כנסת מהאופוזיציה שיעדיף את הביטוי "מהפכה משפטית". כמו כן, הקולקציה יכולה להעיד על התפקיד של הדובר, כמו בדוגמאות שהוזכרו קודם.
בחירת הקולקציות נעשית באופן דינמי ע"י הקריאה לפונקציה 'decisive_collocations'. הפונקציה עוברת על כל זוג דוברים מבין הדוברים הנתונים לה ברשימת הדוברים תוך התחשבות ברצף (כלומר, בהינתן שני הדוברים speaker1, speaker2 קיימים עבורם שני זוגות (speaker1, speaker2) ו- (speaker2, speaker1)), ומוצאת את 6 הקולקציות בעלות ההבדל הגדול ביותר בתדירות ההופעה לטובת הדובר הראשון בזוג. הקולקציות של כל זוג מתווספות למבנה נתונים מסוג set על מנת להימנע מכפילויות, שבתורו מוחזר כפלט.

שתי הפונקציות שנוצרו עבור שלבים 3.1 ו- 3.2 נקראו מה- ()main פעם עבור כל משימה. כאשר עבור משימת הסיווג הבינארית הווקטורים נוצרו בהתבסס על המשפטים של שני הדוברים, ואילו במשימת הסיווג מרובת המחלקות גם המשפטים של 'other' נלקחו בחשבון.

שלב 4

לצורך המשימה, אימנו עבור כל משימה 4 מסווגים: 2 (KNN, Logistic Regression) לכל סוג וקטור מאפיינים. המסווגים הוערכו באמצעות שיטת Cross Validation עם 5 folds כנדרש. התוצאות עבור כל מסווג מצורפות למטה.

<u>למסווגים מסוג KNN</u> כן ראינו לנכון להשתמש בערכים השונים מהערכים של ברירת המחדל. עבור שני המסווגים השתמשנו ב- **n_neighborhood = 8**, מתוך מחשבה שהערך 5 של ברירת המחדל לא נותן מספיק דיוק ועלול להיות רגיש לרעשים במשימה של סיווג משפטים, למשל במקרה שהמשפט מכיל באופן מקרי מלים הייחודיות לדובר האחר. על כן, השאיפה הייתה להגדיל את מספר השכנים באופן שיבטיח יציבות אך באופן זהיר שלא יביא להתאמת יתר. פרמטר אחר שבחרנו לשנות הוא **weights**, בכך שהחלטנו להשתמש בשיטת המשקול של 'distance' עבור שני המסווגים כי חשוב לנו שהמשקל יהיה יחסי לקרבה ושתהיה העדפה לשכנים הקרובים יותר. עבור שיטת חישוב המרחק, בווקטור Td-idf הוחלט להשתמש בערך ברירת המחדל של המרחק האוקלידי, מפני שהשוני בין ערכי תכונות

הווקטורים קריטיים ויש משמעות למרחק היחסי. מצד שני, בווקטור המאפיינים שלנו, רוב התכונות לא היו בעלי ערכים רציפים שהמרחק בהם בעל משמעות, אלא בינאריים המצביעים על קיום התכונה או אי-קיומה. למשל: האם המשפט מכיל ספרה, או האם הוא מפרוטוקול המליאות, ובשל כך הוחלט להשתמש במרחק מנהטן ע"י לתת ערך 1 לפרמטר p.

<u>למסווגים מסוג Logistic Regression</u> הוחלט להשתמש בפרמטר של **מס' האטירציות המקסימלי (max_iter)** שנבחר להיות 1500, מפני שערכים נמוכים יותר, כולל של ברירת המחדל, לא הבטיחו תמיד התכנסות בסיום האטירציות, במיוחד במקרה של Tf-idf, כנראה בשל היותו בעל מספר תכונות גדול. כמו כן, השתמשנו ב- L1 במקום L2 הדיפולטי עבור ווקטור המאפיינים מאותה סיבה שנ"ל שבגללה השתמשנו במרחק מנהטן ב- KNN.

<u>התוצאות שהתקבלו עבור משימת הסיווג הבינארית</u>:

KNN classifier with tf-idf features:

	precision	recall	f1-score	support
speaker1	0.86	0.82	0.84	2159
speaker2	0.83	0.87	0.85	2159
accuracy			0.84	4318
macro avg	0.84	0.84	0.84	4318
weighted avg	0.84	0.84	0.84	4318

Logistic Regression classifier with tf-idf features:

	precision	recall	f1-score	support
speaker1	0.80	0.95	0.87	2159
speaker2	0.94	0.76	0.84	2159
accuracy			0.85	4318
macro avg	0.87	0.85	0.85	4318
weighted avg	0.87	0.85	0.85	4318

KNN classifier with custom features:

	precision	recall	f1-score	support
speaker1	0.91	0.92	0.92	2159
speaker2	0.92	0.91	0.92	2159
accuracy			0.92	4318
macro avg	0.92	0.92	0.92	4318
weighted avg	0.92	0.92	0.92	4318

```
Logistic Regression classifier with custom features:

              precision    recall  f1-score   support

    speaker1       0.92      0.88      0.90      2159
    speaker2       0.88      0.93      0.90      2159

    accuracy                           0.90      4318
   macro avg       0.90      0.90      0.90      4318
weighted avg       0.90      0.90      0.90      4318
```

<u>התוצאות שהתקבלו עבור משימת הסיווג מרובת מחלקות</u>:

```
KNN classifier with tf-idf features:

              precision    recall  f1-score   support

       other       0.60      0.63      0.62      2159
    speaker1       0.64      0.57      0.60      2159
    speaker2       0.77      0.82      0.79      2159

    accuracy                           0.67      6477
   macro avg       0.67      0.67      0.67      6477
weighted avg       0.67      0.67      0.67      6477
```

```
Logistic Regression classifier with tf-idf features:

              precision    recall  f1-score   support

       other       0.61      0.76      0.68      2159
    speaker1       0.68      0.61      0.64      2159
    speaker2       0.90      0.77      0.83      2159

    accuracy                           0.71      6477
   macro avg       0.73      0.71      0.72      6477
weighted avg       0.73      0.71      0.72      6477
```

```
KNN classifier with custom features:

              precision    recall  f1-score   support

       other       0.79      0.68      0.73      2159
    speaker1       0.72      0.79      0.75      2159
    speaker2       0.84      0.86      0.85      2159

    accuracy                           0.78      6477
   macro avg       0.78      0.78      0.78      6477
weighted avg       0.78      0.78      0.78      6477
```

```
Logistic Regression classifier with custom features:

              precision    recall  f1-score   support

       other       0.68      0.57      0.62      2159
    speaker1       0.57      0.63      0.60      2159
    speaker2       0.73      0.78      0.76      2159

    accuracy                           0.66      6477
   macro avg       0.66      0.66      0.66      6477
weighted avg       0.66      0.66      0.66      6477
```

שלב 5

בהתאם לנדרש במשימה, יצרנו פונקציה בשם 'sentences_classify' אשר מקבלת כקלט מודל, וקטורייזר, שמות שני הדוברים שנמצאנו בשלב הראשון, ושתי כתובות אחת לקובץ ממנו יקראו המשפטים, ואחת לתיקייה אליה בה ישמר קובץ התוצאות. הפונקציה קוראת את הקלט, מבצעת את הסיווג לאחד מבין שלושת המחלקות (שני הדוברים, ו- "other"), יוצרת את הקובץ של הפלט וכותבת אליו את התוצאות תוך שהיא מחליפה את שמות שני הדוברים ל- first ו- second.

במשימה הוחלט להשתמש במסווג Logistic Regression בגלל שהציג יכולת דיוק טובה יותר בשלב ההערכה (שלב 4).

התוצאות נמצאות בקובץ classification_results.txt המצורף.

שאלות

1. **מה הם האתגרים שיכולים להיווצר בשימוש במחלקה "אחר" במשימת הסיווג?**
   בניגוד למחלקות של שני הדוברים, המחלקה "אחר" מגוונת ולא ייחודית לאיש מסוים, ולכן יכול להיות שהמודל יתקשה במציאת דפוסים מסוימים בטקסטים של המחלקה או מאפיינים שיהוו בסיס לסיווג. כמו כן, ובשל היותה חסרת דפוס, משפטים הצהרתיים (Declarative sentences) או משפטים חסרי ייצוג רב של מאפייני שני הדוברים (למשל, במשפטים קצרים) עשויים לרוב להיסוג כ- "אחר".

2. **נניח שאתם משתתפים בתחרות מודלים לחיזוי בינארי שבה אם המודל שלכם יחזה נכון את כל הדוגמאות של הדובר הראשון, תקבלו פרס כספי גדול, ואם המודל שלכם יטעה על אפילו דוגמה אחת של הדובר הראשון תקבלו קנס כספי**

גבוה. מבין המדדים המופיעים ב classification report , איזה מדד תרצו למקסם? איזה מהמודלים שאימנתם תבחרו למטרה זו? הסבירו.

לצורך התחרות, נבחר למקסם את המדד של Recall אשר מודד את כמות החיזויים הנכונים שלו לקטגוריה מבין כל הדוגמאות של הקטגוריה, בשאיפה להשגת Recall = 1 אשר יבטיח אפס טעויות. לגבי המודל, נבחר בזה שהביא ל- Recall הגבוה ביותר עבור הדובר הראשון. במקרה שלנו, זה היה תלוי ווקטור מאפיינים, כאשר בשימוש בווקטור Tf-idf ה- Logistic Regression השיג Recall יותר טוב עבור הדובר הראשון, ואילו בשימוש בווקטור המאפיינים שלנו ה- KNN היה היותר מדויק.

3. **ענו שוב על 1 כאשר שינו את החוקים בתחרות וכעת אם המודל שלכם יסווג נכון את כל הדוגמאות של שני הדוברים תקבלו פרס כספי גבוה, אבל אם המודל שלכם יסווג אפילו דוגמה אחת בצורה לא נכונה, תקבלו קנס כספי גבוה.**

במקרה הזה נרצה למקסם את ה- Accuracy, אשר מודד את אחוז הדוגמאות שסווגו נכון מבין כל הדוגמאות, בשאיפה להשגת 100%. באשר למודל, נבחר את המודל בעל ה- Accuracy הגבוה מבין השניים. בהסתמך על התוצאות שלנו, גם כאן בדומה ל- Recall, ה- Logistic Regression היה יותר טוב לעומת ה- KNN כאשר השתמשנו בווקטור ה- Tf-idf, וה- KNN היה יותר טוב במקרה של הוקטור מאפיינים שיצרנו.

4. **הסבירו מה היתרונות והחסרונות של שיטת ה cross validation על פני חלוקה פשוטה למחלקת אימון ובדיקה. איזו משיטות ההערכה אמינה יותר לדעתכם?**

<u>היתרונות</u> של ה- Cross Validation על פני החלוקה למחלקות אימון ובדיקה: נוטה להיות יותר יציב ומדויק בגלל שהוא ממצע כמה חלוקות שונות, במקום חלוקה אחת שיכולה להיות מוטה, במיוחד כאשר כמות הנתונים מעטה או מכילה גיוון רב. בנוסף, השיטה עושה שימוש במספר גדול יותר של נתונים להערכה מפני שהיא משתמשת בכל הדאטה גם לאימון וגם בדיקה, בניגוד לשיטה האחרת אשר בה כל נתון בדאטה יכול לשמש רק לאחד מבין אימון ובדיקה.

<u>החסרונות</u> של ה- Cross Validation לעומת החלוקה למחלקות אימון ובדיקה: סיבוכיות חישוב גבוהה יותר בשל הצורך לבצע חלוקה, אימון והערכה מספר פעמים לעומת פעם אחת בשיטה האחרת. החיסרון הזה משמעותי יותר כאשר יש לנו נפח דאטה גדול המצריך זמן רב של עיבוד.

<u>לדעתנו</u>, ה- Cross Validation אמינה יותר, בשל הסיבות שהוזכרו כגון יציבות ושימוש במספר גדול יותר של נתונים להערכה ולאימון, כמו גם בממוצע של הערכות אשר אמור להפחית את האפקט של הרעש.

5. הסבירו מהם היתרונות והחסרונות של שני סוגי המסווגים KNN, LogisticRegression בהם השתמשתם. האם לדעתכם אחד מהם עדיף על פני השני, עבור משימות הסיווג שבתרגיל?

<u>היתרונות של מודל KNN</u>:

- **הרעיון שלו פשוט ואינטואיטיבי** ומתבסס על חיפוש השכנים הקרובים ביותר, ולכן קל ליישום.
- **יכול להתמודד עם נתונים שהם לא לינאריים**.
- **תומך במאפיינים שהם בינאריים**, ויכול לשלב מאפיינים רציפים ובינאריים, דבר שהשתמשנו בו במטלה שלנו, כאשר כללנו את המאפיין הרציף אורך המשפט, ואת המאפיין הבינארי של אם המשפט מכיל ספרה או לא.

<u>החסרונות של מודל KNN</u>:

- **דורש זמן חישוב גבוה יחסית**.
- **קושי בלבחור את מספר השכנים המתאים** (את ה- k), כמו גם קושי בהערכת הטווח של הערכים אשר לא יביאו ל- Underfitting או Overfitting. דבר שחווינו בעת בחירת ה- k במטלה שלנו, ולמרות שערכי k יותר גבוהים מזה שבחרנו נתנו תוצאות טובות יותר בשלב ההערכה, בחרנו לא להסתכן ולהתרחק יותר מדי מערך ברירת המחדל.
- **רגיש לרעש**, ומספיק ששכן אחד יהיה מוטה, הוא יכול להשפיע לרעה על התוצאה של הסיווג, במיוחד כאשר מספר קטן של שכנים נלקח בחשבון.
- למרות שהשילוב של מאפיינים רציפים ובינאריים הוזכר כיתרון, אולם בלי שימוש בנרמול**, מאפיינים בעלי טווח ערכים גדולים יכולים להיות בעלי השפעה גדולה יותר ממאפיינים בינאריים, או כאלה הרציפים עם טווח ערכים קטן**.

<u>היתרונות של מודל LogisticRegression</u>:

- **זמן החישוב שלו מהיר** יותר בהשוואה ל- KNN.
- **פחות רגיש לרעשים** בהשוואה ל- KNN.

<u>החסרונות של מודל LogisticRegression</u>:

- **מניח קשר לינארי בין הנתונים**, ומתקשה להתמודד עם דאטה שאינה ניתנת להפרדה לינארית.
- **מתקשה להתמודד עם דאטה קטנה**.

- **מתקשה להתמודד עם דאטה לא מאוזנת**, ואם לא ננקטו צעדים של איזון, הסיווג שלו עלול להיות מוטה לטובת המחלקות בעלות מספר הדוגמאות הגדול יותר.

במשימת הסיווג בתרגיל בה סיווגנו את המשפטים לפי ווקטור ה- Tf-idf, נעדיף להשתמש ב- **Logistic Regression** על פני KNN, גם לאור התוצאות שהתקבלו בהערכה, אך גם בשל יעילותו החישובית בהינתן שבמשימות כאלו יכולה להיות לנו כמות דאטה לא קטנה, בנוסף להיותו פחות רגיש לרעש, היות הרעש אופייני במשימת סיווג משפטים.

6. **יחידת הסיווג בתרגיל היא משפט אחד. אם במקום זאת, היינו מחליטים על יחידת סיווג שמאחדת יחד מספר משפטים מאותה מחלקה, כך שכל דוגמה לסיווג הייתה מקבץ של משפטים. מה היו היתרונות והחסרונות בכך? התייחסו בתשובתכם ליחידות סיווג של 2, 5, 10, 100 משפטים.**

<u>היתרונות של יחידת סיווג המורכבת ממספר משפטים</u>:

- **הקשר רחב יותר שיפחית את את מידת השפעת המשפטים הרועשים**.
- **הכללת מספר רב יותר של מאפיינים בתוך היחידה, מה שהופך אותה ליציבה ומייצגת יותר**. למשל חלק גדול מהמאפיינים בווקטור המאפיינים שיצרנו מתבססים על מספר ההופעות של קולקציה מסוימת במשפט, וסביר להניח שרוב המאפיינים עבור יחידה בגודל משפט אחד יקבלו ערך 0 ושנדיר שאחת תקבל יותר מ- 2. לעומת זאת בשימוש ביחידה שמורכבת ממספר משפטים יהיו יותר מאפיינים שהם לא 0, בנוסף לכך שיהיה גיוון רב יותר בערכים, כמו גם יהיה קל יותר לזהות קולקציה המאפיינת דובר מסוים. בדאטה שלנו זה יכול להיות אפקטיבי מפני שהרבה מהמשפטים הם בגודל 4-6 מלים שעלולים לא להכיל ייצוג של מספיק מאפיינים.

<u>החסרונות של יחידת סיווג המורכבת ממספר משפטים</u>:

- **קושי בסיווג משפטים בודדים**, מפני שסקאלת המרחק בין המשפט ליחידות סיווג תגדל, בדרך שתקשה להבדיל בין מרחק קטן לגדול.
- **חוסר עקביות בסיווג**, כאשר הסיווג עלול להיות מושפע מהאופן בו איחדנו את המשפטים.
- **הפחתת מספר הדוגמאות לסיווג**, זהו חסרון במקרה שגודל הדאטה קטן או כאשר המסווג לא מתמודד טוב עם מעט דאטה כמו במקרה של Logistic Regression.

- **משקל פחות למאפיינים יחידים שהופעתם יכולה להכריע את הסיווג**, בשל כך שיותר מאפיינים יופיעו ביחידה כאשר היחידה מורכבת ממספר משפטים, אם לא נלקחו צעדים שאמורים למנוע זאת. למשל, במקרה של המאפיין "הופעת ספרה ביחידת הסיווג", שיהפוך להיות נכון ברוב המקרים, לצד תרומתו שתפחת. צעד שאפשר לקחת על מנת למנוע זאת, הוא החלפתו במאפיין "מספר הספרות ביחידת הסיווג".
- **סיבוכיות נוספת** כתוצאה משלב האיחוד שמתווסף.

<u>השימוש ביחידת סיווג בגודל 2</u> יכולה להוסיף זמן חישוב בשל הזמן הנוסף שהאיחוד יצטרך, בלי להוסיף יתרון משמעות, כך שהייצוג של המאפיינים והפחתת הרעש לא הולכים להשתפר בצורה גדולה.

<u>השימוש ביחידת סיווג בגודל 100</u> אמור להפחית רעש באופן משמעותי, כמו גם להכיל מידע גדול יותר לגבי המאפיינים. בנוסף, ובשל כך שהרבה פעמים יהיה קשה למצוא 100 משפטים של אותו דובר באותו הקשר, נצטרך לשלב בין משפטים מהקשרים שונים, למשל, במטלה שלנו, משפטים של הדובר מפרוטוקולים שונים, מה שהולך להפחית את האפקט של המאפיינים התלויי הקשר לטובת המאפיינים תלויי אישיות. מצד שני, יחידת סיווג גדולה כזאת יכולה להכיל מספר רב של מאפיינים ולמנוע מהמסווג לזהות את הקשר בין המאפיינים ואיזה מהמאפיינים סבירים להופיע עם מאפיינים אחרים. למשל, כאשר משפט של דובר נלקח מפרוטוקול של מליאה נצפה שהמאפיינים של "הופעת הקולקציה אדוני היושב - ראש", "הופעת הקולקציה אני", ו- "אורך קצר של משפט" יופיעו בתדירות גבוהה במשפטים, לעומת כאשר המשפט נלקח מפרוטוקול של ועדה, שם המאפיינים של "הופעת ספרה במשפט" ו- ו- "אורך גדול של משפט" יותר מתלוות. הדבר הזה של איחוד משפטים מפרוטוקולים שונים יכול להימנע ביחידות סיווג קטנות, אך לא יהיה מנוס ביחידות סיווג בגודל 100. חסרון נוסף, שהוזכר לפני כן, הוא בזה שמאפיינים מסוימים שנותנים תרומה רבה לסיווג נכון, ובשל ריבוי המאפיינים, הולכים להיות בעלי משקל נמוך יותר.

השימוש ביחידות סיווג מגודל של 5 או 10, לדעתנו, יכול להיות הכי אידאלי, בשל היותו מפחית רעש ומכיל מידע גדול באופן משמעותי, אשר מפצים על זמן החישוב הנוסף, כמו גם, שבמספר כזה של משפטים נוכל לשלוט בזה שיהיו בהקשר אחד או לכל היותר שני הקשרים כך שהחסרונות הנובעות מאיחוד משפטים מהקשרים שונים שהוזכרו למעלה יקרו רק לעיתים רחוקות.

7. **איזה גודל של יחידת סיווג עדיף לדעתכם (1 , 2 , 5 , 10 , 100) במשימות שלנו? הסבירו.**

נעדיף להשתמש ביחידות סיווג בגדלים של 5 או 10, בשל הסיבות שהוזכרו בתשובה לשאלה הקודמת: שהגודל הזה מפחית רעש הנוצר ממשפטים חריגים, מוסיף ייצוג יותר של מאפיינים ביחידה, וניתן למנוע באופן גדול את הסבירות של איחוד בין משפטים שנלקחים מהקשרים שונים אם נרצה בכך.