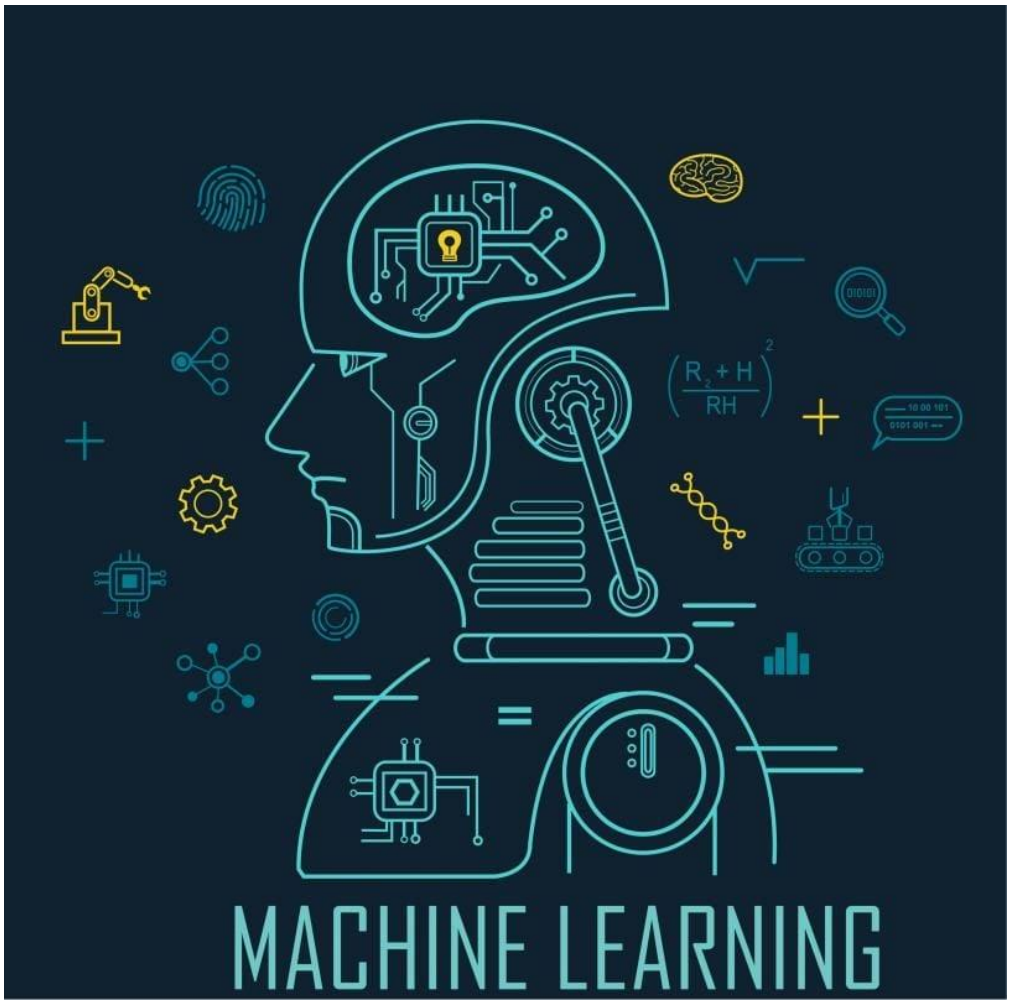
פרויקט לימוד מכונה חלק א'



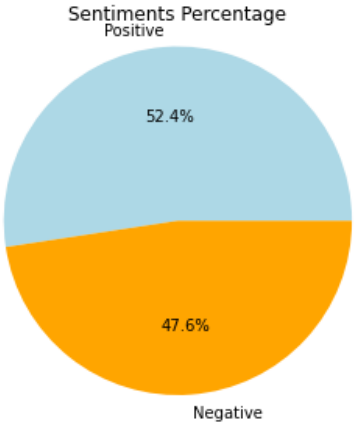
מגיש:
יאיר גזית- 314720517
ירין מור יוסף- 315811828

תוכן עניינים

Data Collection and Sensing

1. Data collection זהו תהליך של איסוף נתונים ממקורות שונים. תהליך זה בנוי מאוסף של סמפלים שנאספו מאותו התחום והוא מייצג את העולם האמיתי אותו אנחנו רוצים ללמוד, במקרה שלנו איסוף דגימות ההודעות מהרשתות החברתיות. הSensing שבוצע על הדאטה הוא גם Static וגם Dynamic. נבין כי הישות שלנו היא המשתמשים ברשתות החברתיות, ולכן תהליך החישה הדינאמי כולל נתונים כמו תאריכי הודעות קודמות, תאריכי עקיבה/עוקבים חדשים, נתונים אשר משתנים לאורך זמן. הרכיב הסטטי כולל נתונים כמו תוכן ההודעה, מגדר המשתמש וכדומה, אשר לא משתנים לאורך זמן.
2. מאחר ובוצע גם תהליך חישה סטטי וגם דינאמי, לא נציע סוג Sensing חדש.
3. קטגורית משימת הלמידה היא Supervised Classification. מאחר ואנו יודעים מהם הLabels שלנו (positive/negative) אנו יודעים כי משימת הלמידה הינה מונחית. בנוסף, המשימה הינה משימת סיווג מאחר ואנחנו רוצים לחלק את הסמפלים ל-2 Classes בלבד. מאחר ואנחנו מקבלים את הסמפלים כאשר הם מסווגים ל"positive" ו"negative" משימת הלמידה תהיה מסוג Binary Classification. מכיוון שמטרתנו היא לייצר מודל- שיודע להבין בין 2 הקלאסים האלה.

Dataset Creation



Exploratory data analysis

התפלגות המאפיינים בנתונים וקשר למשתנה המטרה:

Sentiment- משתנה המטרה:

משתנה קטגוריאלי המציין אם ההודעה היא בעלת אופי שלילי או חיובי.
סט הנתונים כמעט ומאוזן, אם אנחנו משערים כי בעולם 50% מההודעות הן בעלות אופי שלילי ו50% הנותרים בעלי אופי חיובי.

משתנים מסבירים:

TextID: משתנה ייחודי המציין את מזהה ההודעה. מכיוון שאין חזרתיות במשתנה זה ואין ערכים חסרים, לא מצאנו קשר בינו לבין משתנה המטרה, ולכן לא נציג מידע הקשור למשתנה זה.

Text: משתנה טקסטואלי המתאר את גוף ההודעה הנשלחת. ישנם מילים המופיעות בתדירות גבוהה במידע. נציג את ה15 מילים שמופיעות בתדירות הגבוהה ביותר בהודעות שמסומנות כחיוביות ובהודעות שמסומנות כשליליות ואת האחוז היחסי שלהם מכלל המילים שנכתבו בהודעות.

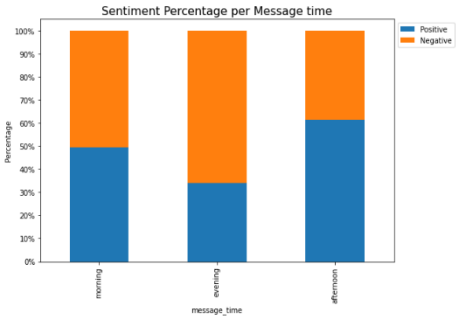
חיוביות

```
Top 15 words in positive sentiment:
have: 0.86%
good: 0.82%
love: 0.81%
day: 0.68%
all: 0.63%
your: 0.59%
just: 0.57%
Happy: 0.52%
like: 0.45%
get: 0.45%
out: 0.43%
great: 0.41%
from: 0.37%
had: 0.35%
happy: 0.35%
```

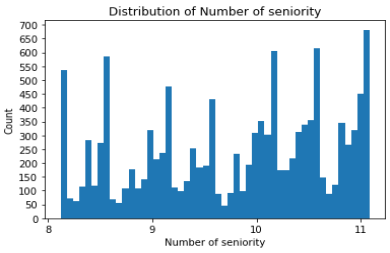
שליליות

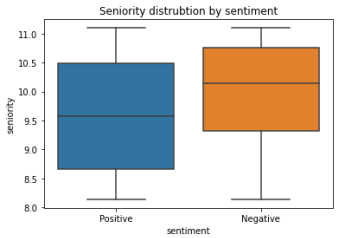
```
Top 15 words in negative sentiment:
have: 0.89%
just: 0.63%
not: 0.56%
all: 0.54%
good: 0.53%
like: 0.51%
day: 0.49%
get: 0.47%
out: 0.46%
love: 0.44%
your: 0.43%
up: 0.40%
go: 0.38%
got: 0.36%
****: 0.36%
```

Message Date: משתנה המייצג את תאריך שליחת ההודעה במקור משתנה זה מסוג DATETIME אך בחרנו להשתמש במשתנה בעזרת דיסקרטיזציה על שעות היום ויצירת 3 קבוצות קטגוריה על מנת שנוכל לראות את ההבדלים באופן מובהק בין קבוצות השונות (message_time). החלוקה שלנו התבצעה כך: Morning= 0-8, Afternoon= 8-20, evening= 21-7.

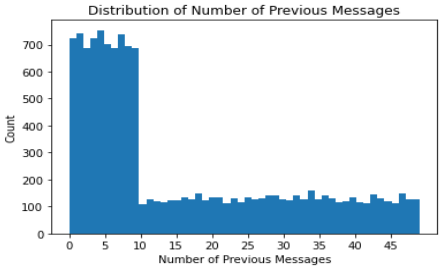


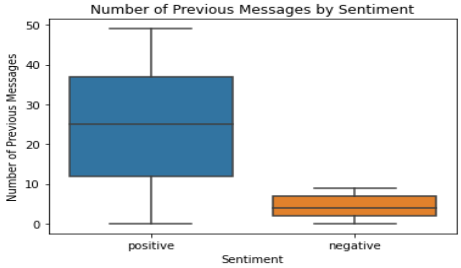
Account creation date: משתנה המייצג את תאריך יצירת המשתמש ממנו נשלחה ההודעה, אנחנו בחרנו להסתכל על משתנה זה בעזרת טרנספורמציה אשר מציגה את וותק המשתמש. מכיוון שתאריכים לא נתנו לנו יכולת החלטה לגבי חיזוי הסנטימנט.
ניתן לראות לפי הגרף השמאלי את ההתפלגות של הנתונים באופן כללי בנתונים, ובגרף הימני את ההבדלים בהתפלגויות לפי סנטימנט המתבטאים בגודל הרבעונים ובחציון.



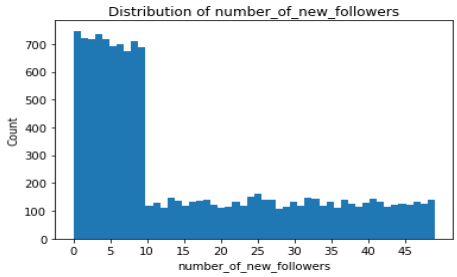


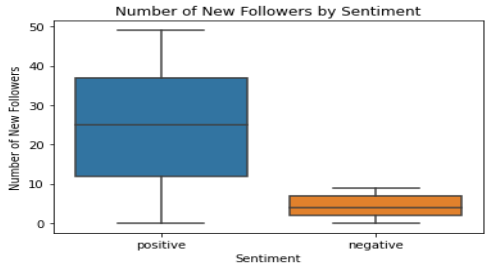
Pervious message dates: משתנה המייצג את התאריכים הקודמים של ההודעות שנשלחו מאותו המשתמש. על מנת לנתח את משתנה זה ביצענו מניפולציה על הנתונים וסכמנו את כלל התאריכים על מנת לקבל את כמות ההודעות שנשלחו מאותו משתמש. בגרף השמאלי ניתן לראות את התפלגות משתנה זה, ניתן לראות כי מסת משתמשים רבה ביצעה עד 10 הודעות, ושאר ההתפלגות מ10-50 הודעות, היו במספרים קטנים יותר, מה שעוד גילינו בעזרת התרשים הימני כי ההתפלגויות שונות מאוד ביחס לסנטימנט וכי ההתפלגות של סנטימנט שלילי היא בין 0 ל10 הודעות בלבד.



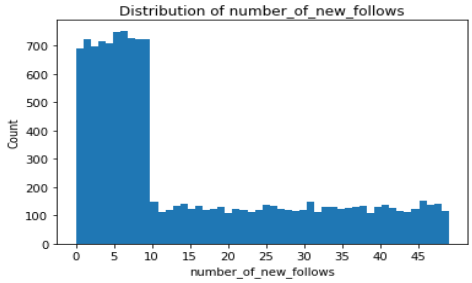


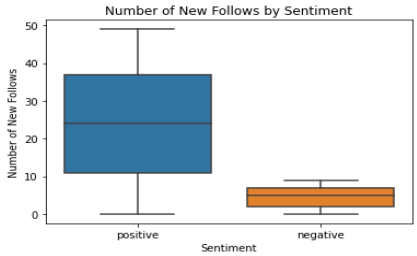
Date of new follower: משתנה המייצג את התאריכים בהם המשתמש קיבל עוקבים.
ביצענו ניתוח על משתנה זה בעזרת מניפולציה שביצענו על הנתונים אשר בעזרתה סכמנו את כלל התאריכים על מנת לבדוק את כמות העוקבים שיש למשתמש.
ניתן לראות מהגרפים, כי קיבלנו מסקנות דומות מאוד כמו למשתנה הקודם.



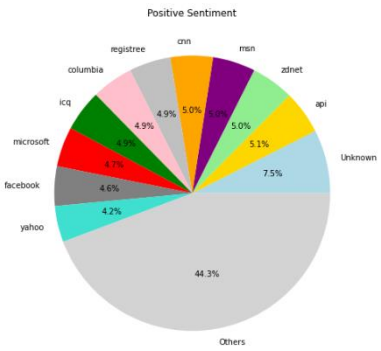


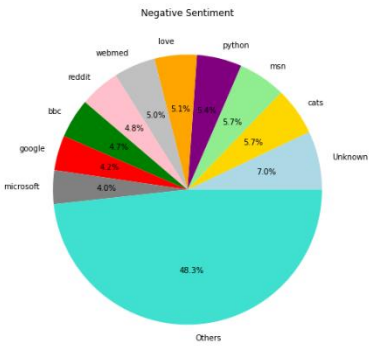
Date of new follow: משתנה המייצג את התאריכים בהם המשתמש עקב אחרי אחרים.
ביצענו ניתוח על משתנה זה בעזרת אותם מניפולציות שעשינו על שני המשתנים הקודמים בעזרתם סכמנו את כלל התאריכים על מנת לבדוק את כמו העקיבות שיש למשתמש וגם כאן, קיבלנו גרפים יחסית דומים ואת אותן המסקנות מקודם. בעקבות כך נבדוק את הקורלציה בין המשתנים הללו ונשקול בסופו של דבר לבחור רק אחד מהם.

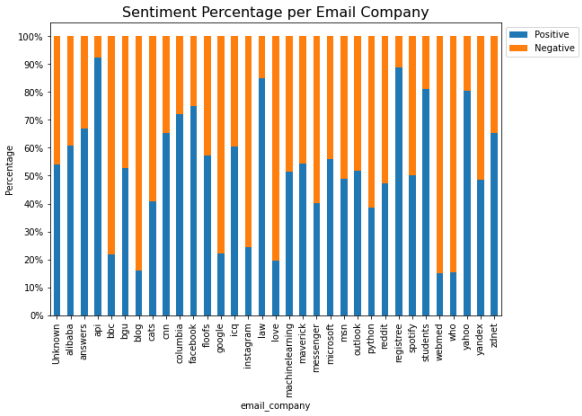




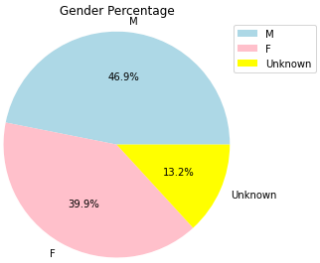
Email: משתנה קטגוריאלי המייצג את כתובת המייל של המשתמש. מאפיין זה מכיל 33 שונות ולכן בחרנו להציג את החברות שהופיעו מעל 4% בנתונים, כל שאר החברות הכנסנו תחת קטגוריית Others. בתרשימים החלטנו לפצל את הנתונים ל2 קבוצות לפי קבוצות הסנטימנט. כך ניתן לראות באופן יחסית ברור את ההבדל בין החברות לסנטימנט ההודעה. לדוגמה cats (בשלילי 5.7% ובחיובי פחות מ4%).

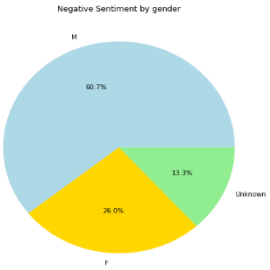


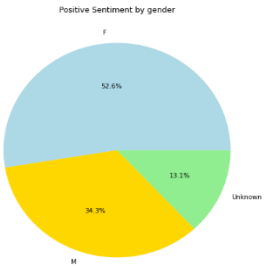


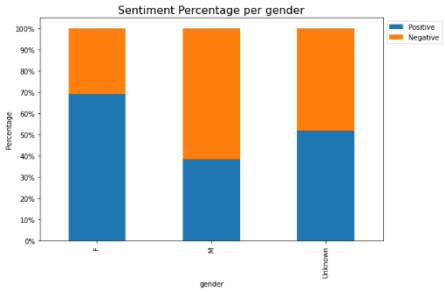


Gender: משתנה קטגוריאלי המייצג את מין המשתמש. ניתן לראות כי יש 13.2% של נתונים חסרים (כ1686 סמפלים).כמו כן, כרגע במידע שלנו יש רוב לגברים. בנוסף נשים לב כי 60% מהגברים נוטים לשלוח הודעות בעלות סנטימנט שלילי יותר מנשים.

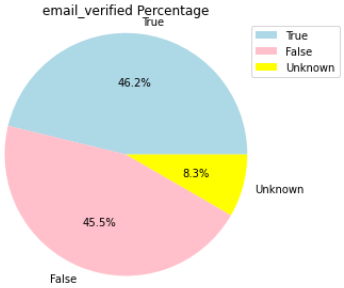


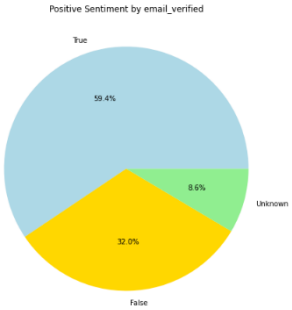


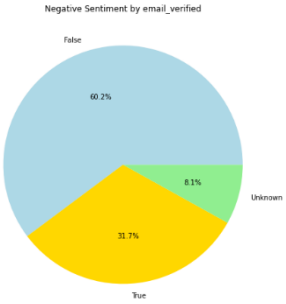


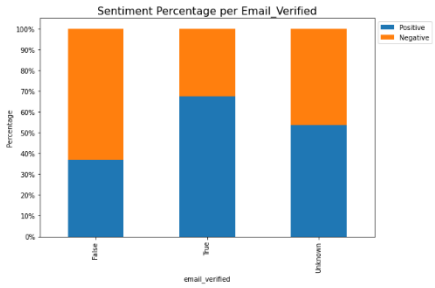


Email Verified: משתנה קטגוריאלי המציין אם האימייל מאומת. ניתן לראות כי ישנם 8.3% נתונים חסרים (כ1060 סמפלים). ניתן לראות כי משתנה זה יחסית מאוזן לנתונים וכי אם המייל לא מאומת 60% שיהיה בהודעה בעלת בסנטימנט שלילי.

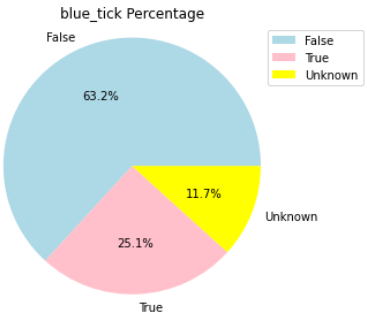


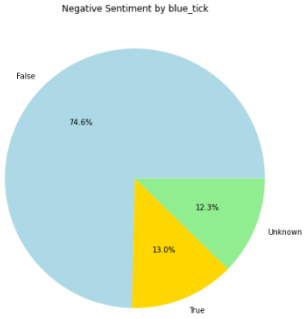


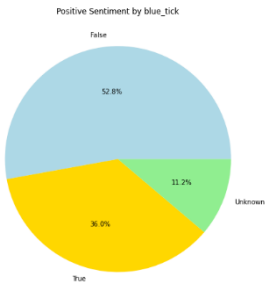


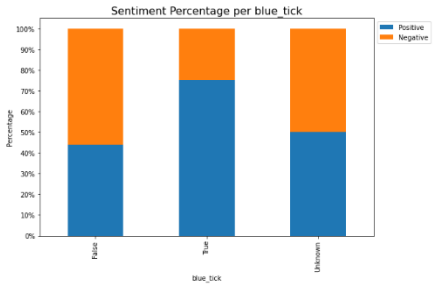


Blue Tick: משתנה קטגוריאלי המייצג האם המשתמש מאומת. ניתן לראות כי ישנם 11.7% נתונים חסרים (כ-1495 סמפלים). משתנה זה אינו מאוזן, רוב המשתמשים אינם מאומתים ולכן שנפלח את כלל ההודעות לפי הסנטימנט עדיין אחוז הלא מאומתים יהיה הגדול ביותר. אך ניתן לראות שאם המשתמש מאומת, 80% הוא ישלח הודעה בעלת סנטימנט חיובי.



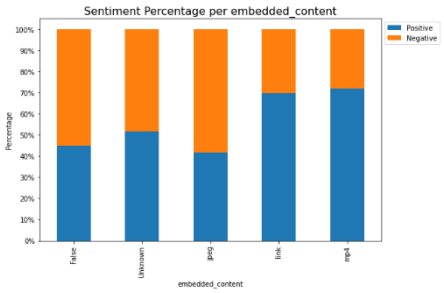


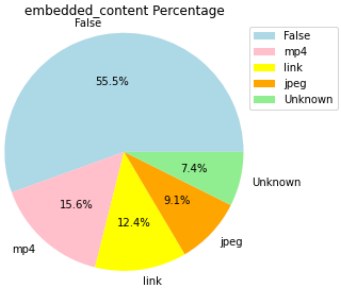




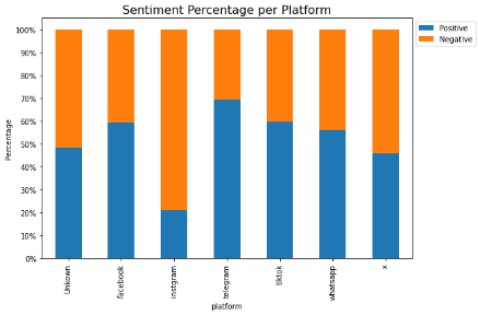
Embedded Content: משתנה קטגוריאלי אשר מציין אם נוסף תוכן בגוף ההודעה. יש לנו 7.4% נתונים חסרים (כ-945 סמפלים), רוב ההודעות מגיעות ללא תוכן נוסף בהודעה והסיכוי שיהיו בסנטימנט שלילי או חיובי יחסית שווה. אך, בהודעות עם תוכן נוסף, ניתן

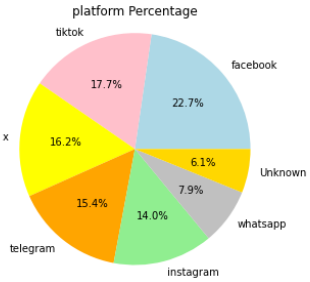
לראות כי האחוזים משתנים (link,mp4 כ70% לסנטימנט חיובי וב-jpeg 40% לסנטימנט חיובי).





Platform: משתנה קטגוריאלי המציין את הרשת החברתית שבה נשלחה ההודעה. ניתן לראות כי יש 6.1% נתונים חסרים (כ-780 סמפלים). ניתן לראות כי לכל רשת חברתית יש סיכוי שונה שההודעה שתשלח ממנה תהיה בעלת סנטימנט חיובי/שלילי.

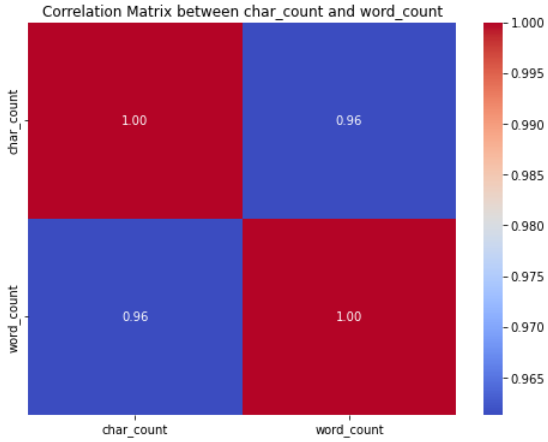




<u>קשרים בין משתנים מסבירים:</u>

עשינו בדיקות קורלציה בין משתנים אשר יצרנו, מצאנו כי יש קורלציה גבוהה בין char_count לword_count המציג קשר חזק בין משתנים אלו (0.96).
לכן נשקול בהמשך להציג למודל הלימוד רק משתנה אחד מהם.

- ניתן לראות את מבחן הקורלציה על ידי הגרף הבא:



ביצוע שלבי הDataset Creation:

Pre-Processing עיבוד מקדים לנתונים כדי שנוכל להשתמש בהם בצורה טובה יותר.

1) התמודדות עם חזרתיות בנתונים- ביצענו בדיקה של כלל המשתנים ללא TextID לכל שורה על מנת לזהות כפילויות בנתונים מכיוון שאינן מוסיפות למשימת הלימוד ולהסירם. בטבלה לא היו דגימות שחזרו על עצמן.
2) התמודדות עם מידע חסר- החלטנו לבצע מילוי של הערכים החסרים לפי ההסתברויות אפריוריות של כל משתנה, בדקנו כיצד הוא מתפלג בנתונים ולמעשה כך הוא קיבל את ההסתברות למילוי הערך. מקרה יוצא דופן אחד שהיה לנו הינו במשתנה Email_company מכיוון שיש קטגוריות רבות של חברות האיימיילים החלטנו לדגום כל פעם אקראית את אחד מהחברות ולהוסיפו לערך חסר, כך שמרנו על פיזור הדומה להתפלגות המקורית מבלי לדעת את ההסתברויות המדויקות.
3) המרות של ערכים- ביצענו שיוך ערכים לקטגוריות, במשתנה message_date אשר הינו רציף יצרנו 3 קטגוריות על מנת לסווג את הנתונים, morning, afternoon, evening כלומר ביצענו דיסקריטיזציה. עשינו זאת על ידי חישוב שעת שליחת ההודעה ולפי השעה חולקו לקטגוריות.
4) ניהול יעיל של הנתונים- בשלב הEDA ראינו כי ישנם משתנים בעלי מספר רב של תאריכים, וכי הפיכתם לכמות התאריכים (כמו number of previous messages) מבדיל את הקלאס בצורה טובה וברורה.
5) איזון הנתונים – המידע הנתון כמעט ומאוזן באופן מושלם (52.4% סנטימנטים חיוביים ו47.6% סנטימנטים שליליים), מכיוון שאנו לא יודעים את המצב האמיתי, ניסינו על ידי קריאה באינטרנט לראות האם זוהי חלוקה מייצגת, מכיוון שלא ניתן לדעת ללא מחקר מעמיק נניח כי הנתונים אכן מייצגים את העולם האמיתי.

Segmentation

מכיוון שהישות שלנו בנתונים הם משתמשים, אנו מניחים כי בוצעו עליהם כבר סגמנטציה, כלומר הסגמנטציה נעשתה על הישות וכך קיבלנו את הנתונים שלנו כעת.

Feature Extraction

המידע שקיבלנו כבר מכיל בתוכו פיצ'רים, ישנם מספר מאפיינים שפחות הצלחנו להפיק מהם ערך למשימת הלמידה ולכן חילצנו מהם מאפיינים למשימה זו, המאפיינים הינם:

- מmessage_date חילצנו את message_time אשר מייצג את קבוצות השיוך קטגוריות של שעת ההודעה.

  בנוסף מmessage_date חילצנו את השנה של כל שליחת הודעה.

- מaccount_creation_date יצרנו את seniority אשר מייצג את הותק של כל משתמש בשנים.
- מprevious_message_date יצרנו את number_of_previous_messages אשר סופר את כמו ההודעות הקודמות אשר שלח כל משתמש
- מdate_of_new_follower יצרנו את number_of_new_followers אשר סופר את כמות העוקבים של המשתמש, בנוסף יצרנו את Latest_ date _of_follower וEarlier_date_of_follower אשר מייצגים את השנה המוקדמת ביותר שעקב אחרי מישהו והשנה המאוחרת ביותר אשר עקב אחרי מישהו
- מdate_of_new_follows יצרנו את number_of_new_follows אשר סופר את כמות העקיבות שבוצעו על ידי המשתמש, בנוסף יצרנו את Latest_date_of_follow וEarlier_date_of_followd אשר מייצגים את השנה המוקדמת ביותר שנעקב על ידי מישהו והשנה המאוחרת ביותר בה נעקב אחרי מישהו בהתאמה.
- מEmail יצרנו את Email_company אשר מייצג את חברת האימייל איתו המשתמש נרשם אל הרשת החברתית.
- מtext יצרנו מספר פיצ'רים טקסטואליים:
  1. Char_count- אשר סופר את כמות התווים שיש בהודעה.
  2. Word_count- אשר סופר את מספר המילים שיש בהודעה.
  3. Sentence_count- אשר סופר את כמות המשפטים בהודעה.
  4. Avg_word_length- אשר מחשב את המספר אותיות הממוצע במילה.

  בנוסף יצרנו בעזרת TF-IDF משקלים ל50 מילים אשר מופיעות פחות בדאטה, כאשר הIDF-TF לוקח בחשבון כמה מהמילים הללו הופיעו גם בסמפלים אחרים. הסרנו את המילים אשר מופיעות ביותר מ10% מהטקסטים על מנת לטפל במילים גנריות כמו is, the וכדומה.

Feature Representation

- בצענו קידוד למשתנים קטגוריאליים בעזרת One hot encoding ו Frequency encoding. עשינו one hot encoding למשתנים הקטגוריאליים platform, embedded_content, message_time. בעקבות כך שלמשתנה Email_company יש מספר רב של קטגוריות, בחרנו בFrequency encoding על מנת לא ליצור כמות גדולה מדי של עמודות בדאטה סט.
- נרמול כלל הערכים לערכים בין 0 ל-1: נרמלנו ערכים מספריים רציפים בעזרת חלוקה בערך המקסימלי באותה עמודה-Min-Max Scaling, כך נוכל להסתכל על כמה מאפיינים שונים באותה הסקאלה, הפכנו משתנים של True/False למשתנים בינאריים.

Feature Selection

בשלב זה, השתמשנו בשיטת ה- Fischer score לחישוב ציון לכל הפיצ'רים. ככל שהציון גבוהה יותר הפיצ'ר מחלק טוב יותר את המחלקות השונות במשתנה המטרה, כאשר בבחירת הפיצ'רים נרצה לבחור את הפיצ'רים בעלי הציון הכי גבוה. בחרנו את הפיצ'רים בעלי ערכים נומריים בלבד. בנוסף הסתכלנו על מפת חום אשר מראה את כלל מבחני הקורלציה (פירסון) בין כלל המשתנים על מנת לגלות האם יש תלות בין המשתנים, החלטנו שאם הקורלציה שגדולה מ0.7 או קטנה מ-0.7 בין שני משתנים, נוציא אחד מהמשתנים.

*ניתן לראות את הגרפים בנספחים בסוף הדו"ח.

הפיצ'רים אשר בחרנו להוריד:

- TextID- עמודה חד חד ערכית, לא רלוונטית לניתוח הנתונים לטובת משימת הלימוד ולכן בחרנו להסירה.
- Text- כלל עמודות הטקסט המקוריות אינן רלוונטיות מכיוון שהפקנו מהם פיצ'רים רלוונטיים ולכן ניתן להסירה.
- Message_date, account_creation_date, previous_messages_dates, date_of_new_follower, date_of_new_follow- כלל משתנים אלה הם משתנים אשר מכילים תאריכים, מכיוון שחילצנו מהם פיצ'רים רלוונטיים בחרנו להסירם.
- Email- כתובת האימייל של שולחי ההודעה לא רלוונטית יותר מכיוון שגם ממשתנה זה חילצנו מאפיינים, ולכן ניתן להסיר את המשתנה.
- Word_count- מכיוון שיש קורלציה גבוהה בינו לבין char_count (0.96) החלטנו להוריד את משתנה זה.
- בנוסף הורדנו את הפיצ'רים עליהם ביצענו קידוד קטגוריאליים- embedded_content, platform, message_time.

סיכום המאפיינים אשר בחרנו להשאיר:

Seniority	Blue_tick	Email_verified	Gender
earlier_date_of_follower	number_of_new_follows	Number_of_new_followers	Number_of_previous_messages
Email_company	Latest_date_of_follow	Latest_date_of_follower	earlier_date_of_follow
Avg_word_count	Sentences_count	Char_count	Message_year
embedded_content_link	embedded_content_jpeg	Embedded_content_False	Avg_word_length
platform_telegram	platform_instagram	platform_facebook	embedded_content_mp4
message_time_afternoon	platform_x	platform_whatsapp	platform_tiktok
	כלל הפיצ'רים שהוצאנו בעזרת TF-IDF	message_time_morning	message_time_evening

Dimensionality reduction

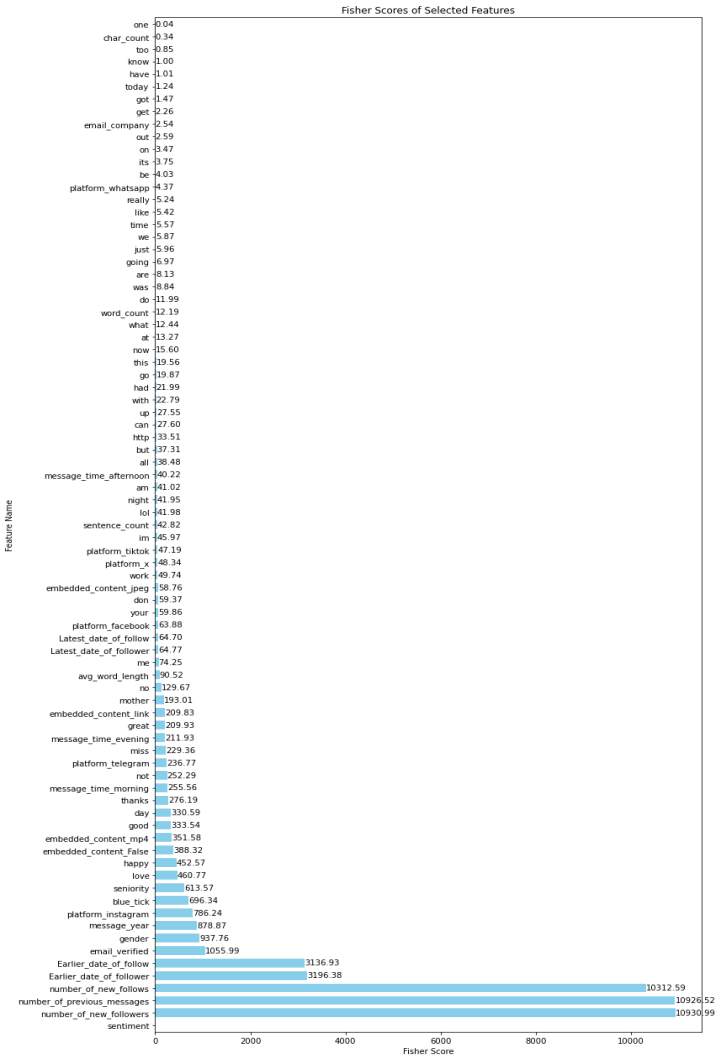
לדעתנו אין צורך בשלב זה, מכיוון שמרבית הפיצ'רים שלנו הם בעלי אינפורמציה, ואם נוותר על חלקם נאבד מידע. חלק גדול מהפיצ'רים הם המרות של המאפיינים שלנו למאפיינים מספריים ולכן כבר בשלב של בחירת הפיצ'רים, החלטנו לבחור את הפיצ'רים בעלי אינפורמציה גבוהה ושאינם תלויים זה בזה. לדעתנו, המודל מכיל מספר מספק של פיצ'רים ועשוי לבצע את משימת הלמידה ללא הורדת המימד ולכן נוותר על שלב זה.

Validation

- נבחר בשיטת הוולידציה- Cross Validation. נחלק את הנתונים שלנו לtraining set וvalidation set לפי שיטת החלוקה של K-fold. בחרנו בשיטה זו כי היא משתמשת בכלל הנתונים ומבצעת את משימת הלמידה באופן יעיל אשר מכסה אופציות רבות של חלוקת הנתונים. שיטה זו עדיף מ- "Leave one out" אשר מבצעת איטרציות על כלל הנתונים מכיוון שיש לנו נתונים רבים וכן עדיפה על "Holdout" מכיוון שחלוקת הנתונים בשיטה זו יכולה לבצע הטיה למודל שלנו.
- נבצע את הוולידציה באמצעות חלוקה ל10 folds (מתוך 12272 רשומות, לכן כל fold יכיל 1227 סמפלים) ונאמן את המודל על k-1 folds (9 יוקצו לאימון) והfold האחרון ישמש לוולידציה. נחזור על התהליך עד שנבחן את כלל הfolds. לבסוף נבצע ממוצע לכלל התוצאות וכך נוכל להסיק האם יש צורך בשינוי/עדכון המודל או לא. נבחר במטריקת confusion matrix אשר משמשת לבעיות של סיווג לקבוצות מובחנות באופן ברור וכן מתאימה כאשר הנתונים יחסית מאוזנים.

Appendices

:Fischer score result



Heat map Correlation