NLP Assignment 3 – Report

Nir Rahav – 316275437

Yael Berkovich – 324879501

המשימה והמודלים

נתונים

המטלה שלנו היא לבצע קלסיפיקציה על ציוצים ב-Twitter. נרצה לתייג האם הציוץ נכתב על ידי דונלד טראמפ או לא. באמצעות עיבוד מקדים של הטקסט ובחינת מודלים שונים נרצה לבצע את תיוג הדאטה. לצורך כך ישנן הנחה מקדימה כי חלק גדול מהציוצים שנכתבו על ידי דונלד טראמפ לא נכתבו על ידו אלא על ידי הצוות שלו. הדאטה שלנו מורכב מהעמודות הבאות:

1. Tweet id
2. User handle
3. Tweet text
4. Time stamp
5. Device

את המחלקות נגדיר בצורה הבאה: ציוצים שנכתבו לפני אפריל 2017, מהיוזר "realDonaldTrump", דרך מכשיר אנדרואיד יוגדרו כדונלד טראמפ. נגדיר עמוד חדשה בשם "author" כאשר לפי היוריסטיקה המוזכרת במידה וציוץ נכתב על ידי טרמפ מוגדר 0 ובמידה והצוות שלו כותב את הציוץ יוגדר 1 כאשר ניתן לראות את החלוקה שלהם בצורה הבאה:

```
0    2255
1    1250
```

נרצה לבחון האם מודלים של למידת מכונה מסוגלים לבצע את ההבחנה הזו בין התיוגים השונים בצורה טובה. לצורך כך ישנן שתי הנחות שאנו מסתמכים עליהן

1. לאחר אפריל 2017 לא ניתן לקבוע את כותב הציוץ מכיוון שטראמפ עבר לאייפון
2. אין לנו חוסרים משמעותיים בדאטה שלנו

המודלים

לטובת המשימה נבדקו 5 מודלים:

1. מודל LogisticRegression - חלק מדרישת הבסיס - מודל פשוט, קל לפירוש הסתברותי, ומהיר לחישוב.
2. מודל SVM - חלק מדרישת הבסיס - מודל יותר מורכב חישובית אך בעל ביצועים טובים באופן אמפירי לבעיות קלסיפיקציה פשוטות וגם מורכבות.
3. מודל XGBoost - מודל רובסטי בעל ביצועים טובים.
4. מודל FFNN - חלק מדרישת הבסיס - נבחר בזכות גמישותו ויכולת הלמידה שלו דפוסים לא לינאריים ומורכבים.
5. מודל BERT

Feature Engineering

למודלים השונים הוזנו פיצ׳רים בצורה של וקטור אשר הורכבו ממחלקה עצמאית (FeatureVectorizer). מטרת המחלקה הייתה ליצור וקטור פיצ׳רים שישלב לפי קלט המשתמש בין TF-IDF, Word2Vec וקטור שנבנה בצורה עצמאית עם הפיצ׳רים הללו:

מספר המילים, מספר האותיות הגדולות, מספר סימני השאלה (״?״), מספר סימני הקריאה (״!״), אורך הממוצע של מילה, האם כל המילה באותיות גדולות, האם נכתב בחג או לא, שעת הציוץ, היום בשבוע של הציוץ, החודש שבו נכתב הציוץ, השנה שבה נכתב הציוץ

הערכת המודלים

לטובת הבחינה הדאטה סט חולק ל80% עבור הtrain set ו20% עבור הtest set, לכלל המודלים בוצע שימוש באופטונה על מנת לבצע אופטימזציה להייפר פרמטרים הטובים ביותר, בנוסף במודל הSVC בוצע תנאי עצירה מקדים על פונקציית הloss במידה ולא יבוצע שיפור עד כדי אפסילון.

Precision	Recall	F1	Accuracy	היפר פרמטרים אופטימליים	וקטור	מודל
0.87	0.73	0.79	0.86	'C': 9.864948922299297, 'penalty': 'l1'	tfidf	Logistic Regression
0.79	0.69	0.73	0.82	'kernel': 'rbf', 'C': 4.31289165277942, 'gamma': 0.030396135757067042	tfidf + additional info	SVM
0.88	0.78	0.83	0.89	n_estimators': 477, 'max_depth': 9, 'learning_rate': 0.016548545657454578, 'subsample': 0.8444703426347393, 'colsample_bytree': 0.6238505376857482, 'reg_lambda': 0.4156047528735853	tfidf + additional info	XGBoost
0.75	0.69	0.72	0.80	‘hidden_dim = 256, n_epoch = 15, dropout = 0.4, batch_size = 512 lr = 1e-3	tfidf	FFNN
0.94	0.71	0.81	0.88	epoch = 3, batch_size = 8	Plain text	BERT

ניתוח

תהליך preprocess

הטקסט והתאריך עברו תהליך מקדים אשר כלל את הצעדים הבאים:

- הסרת תווים מיוחדים והחלפתם בתווים תקניים.
- הרחבת קיצורים באנגלית (לדוגמה: "do not" → "don't").
- החלפת HTML, URL, תאריכים, מספרים וערכים עשרוניים בתוויות מתאימות, וניקוי תווים פגומים.
- מחלק את הטקסט ל־tokens וכולל אפשרות ל:
  - המרת אותיות לקטנות עבור tokens שאינם placeholders.
  - הסרת stopwords (למעט placeholders).
  - ביצוע lemmatization על tokens (למעט placeholders).

ההחלטה להמרה של האותיות לקטנות עבור tokens שאינם placeholders וביצוע של הסרת stopwords נתונה להחלטת היוזר ואינה מחייבת. המרה של טקסט לאותיות קטנות זוהי שיטה מקובלת אך ההחלטה על כך היא לשמר את ה"טון הדיבור" בציוץ שכן זהו יכול להיות מאפיין חשוב לכותב הציוץ.

הגדרות, פיצ'רים ופרמטרים

הפרמטרים הטובים ביותר נקבעו באופטימיזציה בשיטת Cross Validation באמצעות ספריית Optuna, שבדקה טווחים רבים של הייפר פרמטרים ובחרה את המוצלחים ביותר בזמן ריצה קצר, האופטימיזציה כללה פרמטרים כמו penalties, גודל וכמות ה-hidden layers, כמות epochs, עומק מקסימלי ומספר מעריכים. חשוב לציין שבהינתן יותר זמן ומשאבים היה ניתן להגדיל את מספר הניסויים. המודל של BERT מיובא מ Hugging Face בצורה של pretraining לטובת משימת הסיווג הבינארי של מחבר הציוץ, הוא מקודד את הטקסטים באמצעות טוקניזציה, מחלק את הנתונים לסט אימון וסט ולידציה, ומאמן את המודל תוך שימוש ב־Hugging Face Trainer שכולל מעקב אחרי ביצועים, עצירה מוקדמת ושמירה של המודל הטוב ביותר. לאחר האימון, המודל מסוגל לחזות את מחבר הציוץ החדש על סמך טקסט בלבד.

מסקנות ותובנות

המודל הטוב ביותר - מודל XGBoost ומודל BERT, שני המודלים הוציאו תוצאות גבוהות ברוב המדדים אם כי מודל BERT היה עם זמן חישוב גבוה משמעותי, ועל כן אם צריך לבחור את המודל הטוב ביותר מודל XGBoost עדיף בגלל זמני הריצה.

וקטוריזציה - TF-IDF היה יעיל ביותר במודלים שהשתמשו בו, הוא הצליח לשפר ביצועים כאשר שולב עם פיצ'רים מהונדסים, מכיוון שגודל הדאטה קטן יחסית Word2Vec לא היה טוב במיוחד.

היפר-פרמטרים: שימוש ב-Optuna אפשר למצוא היפר-פרמטרים אופטימליים שהובילו לשיפור משמעותי בביצועים.

תובנות מהדאטה: הסרת מילות עצירה ולמטיזציה תרמו לשיפור הביצועים על ידי הפחתת רעשים והכנסת משמעות לשפה הטבעית. עיבוד הנתונים והניקוי היו חיוניים להשגת תוצאות מדויקות ואמינות. הפיצ'רים הנוספים הוסיפו שיפור מסוים, אך לא הצליחו לספק תוצאה חיזוי טובה בפני עצמם. אנו מאמינים שבהינתן יותר דאטה, מודלים גדולים כמו BERT (ואולי גם FFNN) היו מבצעים טוב יותר.