**סיפריות חיצוניות ותלויות**

Tweepy- זוהי הספרייה שבה השתמשנו כדי להתחבר לטוויטר ולשלוף ממנו את כל המידע של המשתמשים במאגר המידע שלנו. כדי להשתמש בספרייה היינו צריכים להירשם בתור מפתחים לטוויטר ולקבל מהם מפתח להתחברות לממשק. לקחנו את מזהי המשתמשים ממאגר המידע באמצעותם שלפנו מידע של המשתמשים ושמרנו באובייקטים שהגדרנו שמכילים את כל המידע על המשתמש ואת אותם אובייקטים שמרנו בקובץ לשם שליפה מהירה של המידע בזמן ריצה של המודל.

Langdetect- ספרייה שבה השתמשנו בשביל לזהות את שפת הציוצים של המשתמשים במאגרי המידע שלנו זאת כדי לסנן משתמשים שהציוצים שלהם לא באנגלית.

Genism- השתמשנו בספריית genism בשביל השימוש שלנו בword2vec. כדי להשתמש בword2vec היינו צריכים לבצע עיבוד מקדים על הטקסט של הציוצים של המשתמשים כדי שיוכלו לשמש כקלט למודל הנ"ל.

Nltk- השתמשנו בספרייה זו בשביל תהליך הtokenization שהציוצים אמורים לעבור לפני הכנסתם כקלט לword2vec ובנוסף בשביל מאגר של stop words באנגלית כלומר מילות הקשר ותחביר שאין צורך להרחיב בהגדרותיהם בשלב השימוש במקורות מידע חיצוניים ולכן התעלמות מהן בשלב הזה היא באמצעות ספרייה זו.

Pywikibot- זהו בוט של wikidata איתו יכולנו לשלוף מידע על כל מיני מונחים ואישים חשובים ממאגר המידע של wikidata. לשם השימוש בו היינו צריכים להגדיר קבצים עם פרטי התחברות לאתר wikidata שבאמצעותם הממשק שולף מידע מהאתר.

Gdelt- בהתחלה השתמשנו בספרייה gdelt כדי לנסות לשלוף מידע ממאגר המידע הזה אך המהירות בו השאילתות של שליפת המידע חזרו היו איטיות מדי בשביל כמויות המידע והזמן שהיה לנו, ולכן ניסינו להשתמש בgoogle BigQuery ונרשמנו לשירות הנ"ל של גוגל כדי לקבל גישה באמצעות שאילתות יותר מדויקות ומהירות למאגר הנתונים gdelt . לאחר ביצוע כמה שאילתות התברר כי השימוש בשירות עולה כסף ולכן השתמשנו באופצייה האחרונה של הורדת מקטעים של מאגר המידע לוקלית, ולכן כתבנו סקריפט שמוריד את הקבצים המתאימים לתאריכים מהם אנחנו רוצים לשלוף מידע במאגר, סידרנו את המידע ושלפנו ממנו בדיוק את המידע שלו היינו זקוקים לזמן האימון של המודל ושמרנו אותו בקובץ json לשליפה מהירה יותר מאוחר.

Pytorch- השתמשנו בספריית Pytorch בשביל כל המודלים של הלמידה חוץ מודל ה-WordToVec, כלומר בכל מקום שמתואר בלוק או מודל של למידה בתוך המודל שלנו מדובר באחד שנכתב עם החבילה הזאת. בפרט ה-LSTM נמצא ישירות ב-Pytorch בעוד שה-TCN נמצא בספרייה אחרת אך ממומש ב-Pytorch בתוכה.

Astar-TCN- זאת הספרייה שבה ממומש מודל ה-TCN שהשתמשנו בו.  
היא כתובה באמצעות Pytorch.  
הספרייה מבוססת על המאמר <הכנס מאמר> שבו הוגדר המושג TCN במקור.

Matplotlib- השתמשנו בספרייה הזאת בכדי ליצור גרפים מתוצאות האימון (בפרט pyplot).