

English to Hebrew Movie Subtitles Generator

מגישים: ליאור אליסברג, גיא דהן

מנחה: ד"ר תמר שרוט

המחלקה להנדסת תוכנה – שנה"ל תשפ"ב



SCE
המכללה האקדמית להנדסה ע"ש סמי שמעון

מהנדסים לעולם טוב יותר!
PROJECT ORIENTED בסביבת

תוכן עניינים

3	תוכן איורים
3	קישור לפרויקט
4	מבוא
5	סקירת המצב הקיים בשוק
5	1. תרגום מבוסס אדם
5	2. שלבים בתרגום סרטים
6	בינה מלאכותית בתעשיית התרגום
6	בלשנות חישובית
6	גישות ותחומי מחקר
7	1. דקדוקים פורמליים
7	2. גישות סטטיסטיות ומבוססות קורפוס
8	אתגרים בתרגום למידת מכונה
8	1. תורת ההגה
8	2. מורפולוגיה
9	3. תחביר
9	4. סמנטיקה
9	מכונת תרגום - מודלים קיימים
9	1. Statistical Machine Translation (SMT)
10	2. Rule-based Machine Translation (RBMT)
10	3. Hybrid Machine Translation (HMT)
11	4. Neural Machine Translation (NMT)
12	Attention-Based Neural Machine Translation
12	1. Sentence Embeddings
13	2. RNN Encoder/Decoder
14	3. Attention Mechanism
15	4. Attention-based NMT
16	תיאור המערכת
16	ימון המודל
20	יצירת הקורפוס
22	כלים טכנולוגיים
24	תוצאות
26	דרישות מערכת
29	דרישות GUI
30	מסע לקוח
30	1. העלאת קישור מ-Youtube
33	2. העלאת קובץ וידאו מהמחשב
35	3. פונקציונליות נוספות במערכת
37	System Overview - UML Diagram
38	מסמך ניהול סיכונים
38	קהל יעד

39	קשיים ואתגרים
39	1. אתגר מחקרי
39	2. יצירת המודל
39	3. אימון המודל
39	4. יצירת הקורפוס
40	סיכום
40	1. דיון בתוצאות
40	2. מחקר עתידי
41	ביבליוגרפיה

תוכן איורים

Figure 1 - Neural machine translation [11].	11
Figure 2 - Differences between Hebrew and English in grammar and pronunciation [17]	12
Figure 3 - The encoder-decoder architecture [21]	13
Figure 4 - Recurrent Neural Network [20]	13
Figure 5 - Encoder/Decoder architecture [23]	13
Figure 6 - Attentional model [11]	16
Figure 7 - Elaborated scheme of the Attention-based NMT [18]	17
Figure 8 - UML diagram for Attention-based NMT	19
Figure 9 - Generated random rows from our dataset	20
Figure 10 - matching HONDA and HYUNDAI with Levenshtein distance algorithm [16]	21
Figure 11 - Levenshtein distance relations between two Hebrew words [22].	21
Figure 12 - Accuracy convergence throughout the training	24
Figure 13 - Loss convergence throughout the training	24
Figure 14 - plotting the Attention	25
Figure 15 - plotting the Attention	25
Figure 16 - System Overview - UML diagram	37

קישור לפרויקט

Project on GitHub [Link](#) / Model on Gist [Link](#)



מבוא

בעת המודרנית, עולם הקולנוע, הוידאו והסטרימינג גודל בצעדי ענק עם טכנולוגיות חדישות בסאונד ועריכה - כמו CG, GFX, שבלעדיהם, חווית הקולנוע שלנו כיום לא הייתה נראית אותו הדבר.

אוטומציה, מחשבים וכלים חכמים כבר נמצאים בשימוש עשרות שנים, אך למרות הקידמה, עולם תרגום הסרטים נשאר מאחור ולא התפתח רבות. בימינו אנו, בניית תרגום לסרט מתבצעת ברובה על-ידי מתורגמנים. פעולת התרגום הינה יקרה ועורכת מספר שלבים עד לאישור סופי של תרגום הסרט, כפי שמתואר בכתב עת [1]. איך יתכן שבעולם כה מתקדם עדיין יש צורך בעבודת כפיים כזו?

כיום, ענקיות טכנולוגיה, כדוגמת Netflix [2] ו-Google [3], עמלות על פיתוח תרגום מכונה מבוסס מודלי NLP. השאיפה הרווחת היא לבצע תרגום מלאכותי לסרטים, ואף גם סרטונים - בלחיצת כפתור. זאת אומרת - ללא התערבות אדם.

תרגום אודיו לטקסט מניב קשיים רבים, כפי שמתאר החוקר במאמרו [4], - בעיה זו נובעת מסיבות רבות בגלל צירופי לשון, חוקי דקדוק שונים בין שפות, סלנג, ביטויים, משפטים משולבי-שפות ומילים שאינם חד-ערכיים ועלולים להשתמע לשתי פנים.

מטרת העל במיזם שלנו היא ליצור מערכת לתרגום לקבצי וידאו, כאשר שפת המקור היא אנגלית, ושפת היעד היא עברית.

האתגר הראשון עבורנו, יהיה יצירת בסיס נתונים המכיל תרגומים שלמים של משפטים
2b השפות = שפת המקור (אנגלית) ושפת היעד (עברית).

למשימה זו קיים ביג דאטה גולמי, עצום וגניש שמתרגמים אנושיים עמלו עליו תוך דגש על לוקליזציה (סלנג, משלב-לשוני וקונטציות תרבותיות).
מקור מידע גולמי זה נוצר עבור אינספור סרטים וסדרות לאורך השנים.

השאיפה העיקרית שלנו היא להשתמש בנתון זה לטובתנו, לרתום את התרגומים המוכנים ליצירת Data-set ולאמן אותו בהתאמת משפטים שלמים, וכתוצאה נצפה לקבל תרגומים מדויקים יותר.
נרצה להוסיף שכבה נוספת של תרגום עם מודל קיים וזאת בכדי לשפר את ביצועי התרגום ובנוסף ללמד את המערכת שלנו תרגומים נוספים ולהרחיב את יכולותיה במרוצת הזמן.

שימוש בכתוביות מסרטים מתורגמים כמקור ליצירת Data-set נותן מענה לבעיות רבות הנובעות מתרגום מילה במילה כמו ריבוי משמעות בתרגום ניבים, פתגמים וסלנג, וכפל משמעויות למילים ומשפטים.

הפרויקט שלנו מתחלק לשני חלקים -

החלק היישומי יכלול מערכת אפליקטיבית שתקבל כקלט קבצי וידאו/אודיו ובעזרת למידת מכונה תמיר את תוכן האודיו המופק לטקסט (Speech-to-Text) לאחר מכן תבצע מניפולציות על קובץ הטקסט (הטמעת חותמות זמן, אינדקסים וכו'). בהמשך תבצע תרגום לקובץ הטקסט בעזרת מערכת התרגום (Machine Translation) שניצור, ולבסוף הטמעת הטקסט המתורגם בכתוביות בסרטון שיהיה זמין להורדה למשתמש.

בחלק התיאורטי, נגדיר גישה חדשנית שאנו מציעים לבעיה - ננסה ליצור מערכת תרגום חדשה בעלת שתי שכבות, הצפויה לשפר ביצועים בהשוואה למערכות הקיימות כיום וזאת מכיוון שאנו נשענים בשכבה הראשונה על מודלים קיימים, ובשכבה השנייה נטמיע מודל חדש בהתבסס על הביג דאטה אשר נבצע עליו את אימון המודל.

סקירת המצב הקיים בשוק

תרגום כתוביות לסרטים הוא ניהול עסקית חשובה בתעשיית התרגום ומהווה מקור חיוני להפצתם של תכני אומנות ותרבות בעולם ממדינת המקור לשאר העולם. למעשה, שירותי תרגום כתוביות לסרטוני וידאו הופכים לנכס שיווקי שמשמעותו גוברת בעולם בו מתרחשים תהליכי הגלובליזציה.

כתוביות לסרטים הופיעו לראשונה בסרט "Uncle Tom's Cabin" שהיה סרט עילם, בשנת 1903 באותם זמנים, כתוביות נקראו גם "intertitles".

לאחר שגם נוסף קול לסרטים, המופע השני של כתוביות בסרט היה בשנת 1929 בסרט "The Jazz Singer", אשר הכיל כתוביות בצרפתית, ופתח דלת לקהל נוסף בתעשיית הסרטים [5].

כיום, מרבית התכנים שגולשים מחפשים באינטרנט הם סרטוני וידאו והמגמה ברורה. ככל שקצב העולם העסקי נעשה מהיר יותר, בשל הגלובליזציה, ובמקביל כושר הריכוז של האדם הממוצע צונח, תרגום סרטונים הופך למצרך חיוני וחשוב ביותר, כפי שמוסבר בכתב העת [1].

1. תרגום מבוסס אדם

תרגום סרטים מקצועי נע בין הפקת כתוביות מחויבת למקור, שהוא בעיקרו עניין טכני, ועד יצירת גרסה חדשה לגמרי ליצירה המקורית המתואמת לקהל יעד חדש. לעיתים, גרסה של תרגום המיועדת לקהל יעד שונה, אפילו לא נחשבת כתרגום. היא נחשבת כ-לוקליזציה מלאה בפני עצמה ולכן ישנן דרישות שיש לעמוד בהן על מנת להפיק תרגום כתוביות מוצלח לסרט:

- i. שימוש במתרגמים הבקיאיים ביותר בשפה -
זאת אומרת שהמתרגמים מחויבים לשלוט הן בשפות היעד והן בשפת המקור כשפת אם או קרוב מאוד לכך.
- ii. היכרות עמוקה עם התרבויות של קהל היעד -
על מנת להנגיש ניואנסים עדינים וחיוניים להבנת העלילה של הסרט. לעיתים על מנת לזהות מבטאים וקונוטציות תרבותיות, המתייגות אדם וכובלות אותו למעמד חברתי וכלכלי ספציפיים, יכולים להתפסס מתרגום שפת המקור לשפת היעד. היכולת להעביר את הרעיונות המקוריים של הסרט עלולים להיפגע באופן משמעותי.

2. שלבים בתרגום סרטים

- i. עשיית תמלול מקצועי -
תמלול כל מילה ורעש רקע בסרט וציון חותמות זמן.
מתי מתחיל ומסתיים כל קטע ומה נמצא בתוכו בשפת המקור ותרגום מקריאה לשפת היעד.
- ii. תרגום כתוביות משפת המקור לשפת היעד תוך עשיית לוקליזציה -
במהלך התרגום יש להתחשב באורך השונה של מילים בין שפה לשפה ולהתאים את הסלנג, משלב-לשוני והקונוטציות התרבותיות לקהל היעד.
- iii. עריכה סופית והטמעת הכתוביות בסרט -
זהו השלב הסופי של התרגום, מדובר בשלב טכני במהותו.
העריכה הסופית מבטיחה את איכות התכנים ותיקון שגיאות שעולות במהלך העבודה.
לאחר מכן, בהתאם לחותמות הזמן מפרידות בין סצנה לסצנה, משתילים בתזמון מדויק את הטקסט המתורגם. התזמון מהותי כי פספוס אפילו בשנייה אחת בין הסצנה המוצגת לתרגום עלול לקלקל משמעותית את חווית הצפייה.

בינה מלאכותית בתעשיית התרגום

כמו כל תחומי החיים, ובעיקר בעקבות הפוטנציאל העסקי והכלכלי בתרגום סרטים, גם בינה מלאכותית מתפתחת בצעדי ענק בתעשיית התרגום.

חברות פרסום מדיה גלובליות צריכות להפוך את התוכן שלהן למתאים לצריכה של צופים מאזורים שונים. על מנת לספק תוכן וידאו עם כתוביות במספר שפות, בתי הייצור יכולים להשתמש בטכנולוגיות מבוססות בינה מלאכותית כמו יצירת שפה טבעית ועיבוד שפה טבעית (NLP) [6].

אחד היתרונות הגדולים ביותר הוא היכולת לתרגם דוגמיות טקסט גדולות בזמן קצר מאוד, עם זאת קיימים פערים רבים ואתגרים בשימוש של תרגום מכונה. ברמה הבסיסית תרגום מכונה מבצעת החלפה ישירה של מילים בשפת המקור לשפת היעד והרכבת המשפט מחדש.

לעיתים ישנם מילים רבי-משמעויות, סלנגים ומשלבים-לשוניים שהופכים תרגום טקסטים רבי מלל למשימה קשה. לא די במציאת התרגומים האפשריים של כל מילה, אלא נדרשת הבנה של המשמעות הנכונה בהקשר של אותה שפה, ודגש על הדקדוק התקין לאותו משפט [4].

כיום, רוב מערכות תרגום המכונה מייצרות "תרגום גרעיני", כלומר תרגום שנותן את עיקרו של טקסט המקור, אך לבד מזה אינו שמיש [7].

בלשנות חישובית

ענף מחקר רב-תחומי המקשר בין בלשנות לשונית ובין מדעי המחשב. כפי שמסביר החוקר במאמרו [4], ישנן שתי דרכים שונות להתבונן בבלשנות חישובית:

1. בלשנות חישובית תיאורטית - תחום המיישם שיטות ותוצאות של מדעי המחשב בבלשנות, על מנת לחקור שאלות יסוד של הבלשנות כגון מהי שפה וכיצד בני האדם משתמשים בה ולומדים אותה.
2. הבלשנות חישובית מעשית - יישום ידע ושיטות של הבלשנות במדעי המחשב על מנת לייצר יישומי מחשב מבינות דיבור אנושי, מתרגמות בין שפות טבעיות ובאופן כללי מתקשרות באופן מילולי עם בני אנוש בדרכים המתאימות לאנשים ולא למחשבים, לעיתים משתמשים במונח עיבוד שפה טבעי.

גישות ותחומי מחקר

על מנת להסביר את הגישות הקיימות, נסביר תחילה מספר מונחים הכרחיים:

שפה טבעית –

שפה טבעית (Natural language) היא שפה שמדברים בה בני אדם. בניגוד ל"שפות מלאכותיות" כמו שפות המחשב שבעזרתן מתקשרים עם מחשבים, או שפות לוגיות ואחרות, השפה הטבעית נוצרה באופן טבעי, על ידי בני אדם וכדי לתקשר אחד עם השני. אנגלית או עברית למשל, הן שפות טבעיות. יש בעולם כ-6000 שפות טבעיות ועד לפני מספר שנים לא ניתן היה לתקשר עם מחשב באף אחת מהן.

במתמטיקה, לוגיקה ומדעי המחשב, שפה פורמלית היא קבוצה כלשהי של רצפים סופיים של סימנים מקבוצה סופית.

אוסף מאמרים, ספרים וכדומה שנכתבו בנושא מסוים או על ידי אותו מחבר.

נוסחה אמפירית המתארת את התפלגות שכיחות מילים בטקסט בשפה טבעית. החוק התגלה ונוסח בשנות ה-30 של המאה ה-20, על ידי הבלשן האמריקאי ג'ורג' קינגסלי זיף, אבי הבלשנות החישובית. התפלגות זו נקראת "התפלגות זיף". בעקבותיו בדקו חוקרים אחרים תופעות טבעיות ואנושיות אחרות וגילו תופעות נוספות המתפלגות לפי התפלגות זיף.

1. דקדוקים פורמליים

בשנת 1956 הציע הבלשן נועם חומסקי את האפשרות ליצור קבוצת חוקים דקדוקיים, שעל ידי הפעלתם, ניתן לייצר את כל המשפטים החוקיים בשפה, ורק אותם (תחילתה של השפות הפורמליות).

חומסקי חילק את השפות הפורמליות לארבע רמות, הנבדלות ביניהן בכוח ההבעה שלהן, כלומר במידת המורכבות של המשפטים שניתן להביע באמצעותן.

ארבעת הרמות הן: שפות בלתי-מוגבלות, שפות תלויות-הקשר, שפות חסרות הקשר ושפות רגולריות. השפות מסודרות ברמת סיבוכיות עולה, כלומר ככול שהסיבוכיות של שפה עולה, כך נדרש כוח חישובי רב יותר לפרש את השפה.

הוא טען שהשפות הטבעיות הן שפות חסרות הקשר, כלומר הן שייכות לרמה השלישית בהיררכיה שלו. הניסיון ליצור מודלים פורמליים לתחביר של שפות טבעיות הפך למטרה של הבלשנות החישובית התיאורטית.

2. גישות סטטיסטיות ומבוססות קורפוס

ניתן לבצע תרגומים מסובכים, תוך התחשבות בטיפול טוב יותר בניגודים, טיפולוגיה, הכרה מפורשת ותרגומים של ניבים וביטויים [10].

לפי חוק זיף, באוסף גדול של מופעים משפה טבעית, יש מספר קטן מאוד של מילים שמופיעות מספר רב של פעמים. הרוב הגדול של המילים מופיעות מעט מאוד פעמים.

חוק זיף מדגים את העובדה שיש בשפה תופעות שניתן לגלות ולחקור בכלים סטטיסטיים. כלומר, אם ניקח קורפוס גדול מספיק של השפה הטבעית, נוכל סווג בו מאפיינים סטטיסטיים.

אתגרים בתרגום למידת מכונה

האתגרים בתרגום מבוסס למידת מכונה רבים וזאת מכיוון ששפות טבעיות הן קשות מטבען, התהליכים הקוגניטיביים הכרוכים בהבנה וביצירה של משפטים בשפה טבעית מורכבים מאין כמותם ומהוות מכשול עיקרי להתפתחות הבינה המלאכותית בכלל.

במאמר [4], מציין החוקר כי הקושי הרב ביצירת מכונת תרגום אמינה נובע מבעיה עיקרית אחת - ריבוי משמעות (ambiguity), זאת אומרת שמילים ומשפטים רבים ניתנים לניתוח במספר אופנים שונים, כך שניתן לייחס יותר מייצוג אחד.

ניתן לסווג בעיה זו ל 4 קטגוריות עיקריות:

1. תורת ההגה

תורת ההגה היא תחום המחקר הבלשני העוסק בהיגוי. הן פונטיקה - העוסק בחקר הקולות שמופקים בעת הדיבור, הצלילים, דרכי היווצרותם, הגייתם וקליטתם, והן בפרנולוגיה, החוקרת את הצלילים המופקים בעת דיבור ברמה מופשטת, מגדירה אותם ואת צירופיהם ובוחנת את תפקודם במערכת הלשונית.

ריבוי משמעות ברמה הפרנולוגית מתבטא באופנים שונים. הנפוצה ביותר היא הומופונים, או מילים שונות הנהגות באופן זהה, כגון צמד המילים **שלו** (שייך לו, קשור אליו) ו-**שלא** (מילת שלילה) שנשמעים בדיוק אותו הדבר, אך שונים לחלוטין במשמעותם.

2. מורפולוגיה

המורפולוגיה חוקרת את מבנה המילים הקיימות בשפה לצירופים בעלי משמעות, המכונים "מורפות" או צורנים. לא קיים יישום ממוחשב של עיבוד שפות טבעיות שלא יצריך ידע מורפולוגי (מילון, מערכת חיפוש אינטרנטית חכמות).

הדבר נכון גם לגבי מנתח מורפולוגי, וזאת מכיוון שעליו למצות את צורת הבסיס מתוך צורות נטויות של מילים. כאן בולט במיוחד ריבוי המשמעות, והוא נובע מתהליכי גזירה ונטייה.

למשל, צורן הסיום - 'י' בעברית משמש הן לציון שייכות (גוף ראשון יחיד) והן להפיכת שם עצם לתואר. לפיכך, למילה **ביתי** שתי משמעויות - הרגשה של בית והבית שלי.

3. תחביר

אחד מתחומי המחקר העיקריים בלשנות הוא התחביר (syntax). התחביר עוסק בשאלות של הצטרפות מילים לצירופים והצטרפות הצירופים למשפטים, ומעמיד שאלות כבדות של ריבוי משמעויות.

לדוגמא, עבור המשפט **לקחתי חולצה מאחי** צירוף היחס **מאחי**, מתקיים כתיאור לפועל **קיבלתי**, ובונה את הקשר המשפט היחסי.

לעומת זאת, המשפט **לקחתי חולצה מכותנה** צירוף היחס **מכותנה** משמש כמתלווה לשם עצם שקדם לו. לכן למרות שהמבנה התחבירי דומה, סביר שיהיה שונה, משום שכל ניסיון של מנתח תחבירי לנתח את המשפט **קיבלתי חולצה** נדרש להתמודד עם שני המבנים האפשריים של המשפט.

4. סמנטיקה

סמנטיקה עוסקת במשמעות של מילים, צירופים ומשפטים בשפה. ריבוי המשמעות מתחיל כבר במילון ויוצר סתירה בין ביטויים ומשמעותם.

לצורך הדוגמא, בעבור המילים באנגלית **hot-dog**, מהווה גם תרגום של נקניקיה וגם ביטוי צרוף אך שבור של **כלב-חם**. אדם אשר דובר אנגלית ידע להבדיל בין המשמעויות השונות, אך מכונה עלולה להתקשות בתרגום עם צמד מילים פשוט שכזה.

מכונת תרגום - מודלים קיימים

תרגום מכונה הוא ענף בבלשנות חישובית המוגדר כתהליך אוטומטי על ידי מערכת ממוחשבת הממיר קטע טקסט משפה טבעית אחת לשפה טבעית אחרת עם התערבות אנושית או לא, במטרה לשחזר את המשמעות של הטקסט המקורי בטקסט המתורגם [13].

1. Statistical Machine Translation (SMT)

הרעיון של תרגום מכונה סטטיסטי מרמז על שימוש בסטטיסטיקה.

זוהי גישה מבוססת נתונים שבה נדרש קורפוס עצום כדי ללמוד כיצד היחידות הדקדוקיות של שפה זרה מתאימות לשפת היעד תוך כדי תרגום.

SMT משתמש בכללי ההחלטה של חוק בייס (Bayes) ובתורת ההחלטות הסטטיסטית כדי למזער טעויות החלטה, בנוסף לרעיונות כמו אנטרופיה מתורת המידע [8].

2. Rule-based Machine Translation (RBMT)

מערכות MT מבוססות-כללים פועלות על סמך מפרט הכללים למורפולוגיה, תחביר, פונולוגיה וסמנטיקה. אוסף חוקים ולקסיקונים (דו-לשוניים או רב-לשוניים) הם המשאבים שבהם נעשה שימוש ב-RBMT.

מודל התרגום מורכב ממספר שלבים:

במהלך שלב הניתוח מתבצע ניתוח לשוני על משפט מקור הקלט על מנת לחלץ מידע במונחים של מורפולוגיה, חלקי דיבור, ביטויים, יישויות ומשמעות המילים הכוללות דו-משמעות.

במהלך שלב ההעברה הלקסיקלי, ישנם שני שלבים - תרגום מילים ותרגום דקדוק. בתרגום מילים, מילת השורש של שפת המקור מוחלפת במילת השורש של שפת היעד בעזרת מילון דו-לשוני ובתרגום דקדוק, ולאחר מכן מתבצע תרגום לסימונות המילים.

בשלב השלישי, הנקרא גם שלב הפונולוגיה, המגדרים של המילים המתורגמות מתוקנות. אלה מבטיחים את המגדר, המספר והאדם של קבוצות מקומיות בביטויים המתוקנים, שכן גם מגדר הפעלים של האובייקטים הנבדקים משקפים את תקינות הביטוי [7].

3. Hybrid Machine Translation (HMT)

תרגום מכונה היברידי היא שיטה לתרגום מכונה המשלבת מאפיינים של מספר גישות תרגום בתוך מערכת תרגום מכונה אחת.

המוטיבציה לפיתוח מערכות תרגום מכונה היברידיות נובעת מכישלון של כל טכניקה בודדת להשיג רמת דיוק מספקת [9].

למרות שישנן מספר צורות של תרגום מכונה היברידי כאלה, הצורות הנפוצות ביותר הן:

- Rule-Based to Statistical

במודל זה, התרגומים מבוצעים באמצעות מנוע מבוסס כללים. לאחר מכן נעשה שימוש בסטטיסטיקה בניסיון להתאים/לתקן את הפלט ממנוע הכללים. ידוע גם "החלקה סטטיסטית".

- Statistical to Rule-Based

כללים משמשים לעיבוד מוקדם של נתונים בניסיון להנחות טוב יותר את המנוע הסטטיסטי. כללים משמשים גם לאחר עיבוד הפלט הסטטיסטי לביצוע פונקציות כגון נורמליזציה. לגישה הזו יש הרבה יותר כוח, גמישות ושליטה בעת התרגום. ניתן לטפל בבעיות רבות בשורשיהן באמצעות כללים החורגים מהיכולות בגישה סטטיסטית בלבד.

4. Neural Machine Translation (NMT)

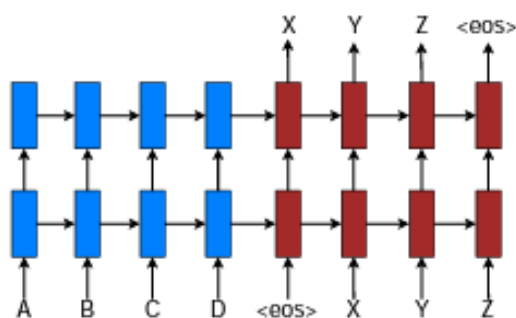


Figure 1 - Neural machine translation – recurrent network architecture for translating a source sequence [11].

NMT הופיעה בתור הגישה המבטיחה ביותר לתרגום עם פוטנציאל של טיפול בחסרונות רבים של מערכות תרגום מכונה מסורתיות, החוזק של הגישה טמונה ביכולת ללמוד ישירות מקצה לקצה את המיפוי מטקסט המקור לטקסט היעד המקושר.

ארכיטקטורת המכונה מורכבת ממפענח ומקודד, שבנויים משכבות של רשתות עצביות. המקודד והמפענח ממירים את רצפי המקור והיעד, בהתאמה, לזוקטורים ממשיים, שבעזרתם ניתן לבצע חיזוי סטטיסטי מוצלח לשפת היעד.

בפרויקט שלנו, בחרנו לממש מכונת תרגום מסוג NMT בשילוב עם מנגנון Attention, עליו נרחיב בהמשך [3].

Attention-Based Neural Machine Translation

כאשר בונים מודל לשפה, יש צורך להתחשב בחוקים דקדוקיים ולשוניים, שכמובן שונים בין שפות.

השפה העברית הינה שפה שמית, שנכתבת מימין לשמאל. (RTL)
השפה האנגלית הינה שפה הינדו-אירופאית, אשר נכתבת משמאל לימין. (LTR)

"The fork, **it** was here!" In Hebrew will be "The fork, **he** was here!"

המזלג, הוא הֵיָה כָאן! – (Humalog, hu haya kan)

קיימים הבדלים מובהקים בין השפות:

אוצר מילים, חוקי שייכות, חוקי דקדוק, מגדר, מרחב האותיות, עיצורים, ועוד [12].
כל אלה מהווים מכשול שמכונת התרגום נדרשת להתמודד איתם.

Verb form	Person	Singular		Plural	
		Masculine	Feminine	Masculine	Feminine
Present tense / Participle		מְדַבֵּר ~ מדובר medubar I am / you m. sg. are / he / it is spoken	מְדַבֶּרֶת ~ מדוברת meduberet I am / you f. sg. are / she / it is spoken	מְדַבְּרִים ~ מדוברים medubarim we / you m. pl. / they m. are spoken	מְדַבְּרוֹת ~ מדוברות medubarot we / you f. pl. / they f. are spoken
	1st	דִּבַּרְתִּי ~ דוברתי dubarti I was spoken		דִּבַּרְנוּ ~ דוברנו dubarnu we were spoken	
Past tense	2nd	דִּבַּרְתָּ ~ דוברת dubarta you m. sg. were spoken	דִּבַּרְתְּ ~ דוברת dubart you f. sg. were spoken	דִּבַּרְתֶּם ~ דוברתם* dubartem you m. pl. were spoken	דִּבַּרְתֶּן ~ דוברתן* dubarten you f. pl. were spoken
	3rd	דִּבֵּר ~ דובר dubar he / it was spoken	דִּבְּרָה ~ דוברת dubra she / it was spoken	דִּבְּרוּ ~ דוברו dubru they were spoken	
Future tense	1st	אֶדְבֹּר ~ אדובר adubar I will be spoken		נִדְבֹּר ~ נדובר nedubar we will be spoken	
	2nd	תִּדְבֹּר ~ תדובר tedubar you m. sg. will be spoken	תִּדְבְּרִי ~ תדוברי tedubri you f. sg. will be spoken	תִּדְבְּרוּ ~ תדוברו tedubru you m. pl. will be spoken	תִּדְבְּרֶנָּה ~ תדוברנה* tedubarna you f. pl. will be spoken
	3rd	יִדְבֹּר ~ ידובר yedubar he / it will be spoken	תִּדְבֹּר ~ תדובר tedubar she / it will be spoken	יִדְבְּרוּ ~ ידוברו yedubru they m. will be spoken	תִּדְבְּרֶנָּה ~ תדוברנה* tedubarna they f. will be spoken

Figure 2 - Differences between Hebrew and English in grammar and pronunciation [17]

נסביר מונחי יסוד על מנת להציג את המודל

Sentence Embeddings 1.

הרעיון העיקרי הוא לייצג מילה $x \in \Sigma$ כווקטור בעל ממדים של מספרים ממשיים.
הגודל של שכבת ההטמעה נבחר בדרך כלל להיות קטן בהרבה מגודל אוצר המילים $|\Sigma|$. $d \ll |\Sigma|$.
ניתן לייצג את המיפוי מהמילה לייצוג המבוזר שלה על ידי מטריצה להטמעה מהצורה $E \in \mathbb{R}^{d \times |\Sigma|}$,
כאשר העמודה ה- x^{th} (מסומנת כ- $E_{:x}$) מכילה את ייצוג הממדים של המילה x . זאת כדי שלמודל יהיה ייצוג מתמשך
לדקדוק של משפט המקור וליחס בין הטוקנים במשפט [15].

ה-Encoder קורא ומקודד משפט מקור לווקטור באורך קבוע. לאחר מכן ה-Decoder מוציא תרגום מהווקטור המקודד. כל מערכת המקודד-מפענח, המורכבת מהמקודד והמפענח עבור צמד שפות, מאומנת במשותף כדי למקסם את ההסתברות לתרגום נכון בהינתן משפט מקור.

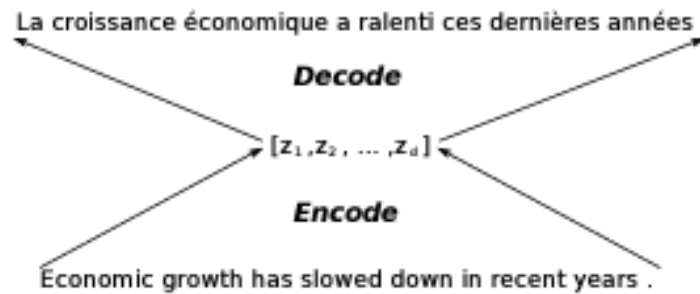


Figure 3 - The encoder-decoder architecture, demonstrated in French [21]

המקודד מקבל כקלט רצף ווקטורים $x = (x_1, \dots, x_{T_x})$ לתוך וקטור c . הגישה הרווחת בשימוש ב-RNN הוא מהצורה שכל שכבה חבויה ברשת, מחושבת ע"י הפונקציה $h_t = f(x_t, h_{t-1})$, המועברת מהשכבה שקדמה לה בזמן t .

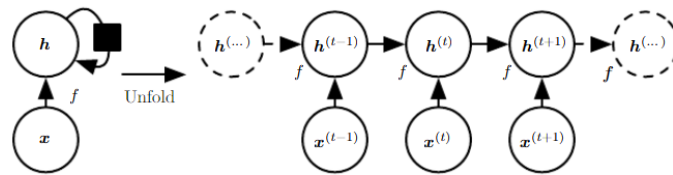


Figure 4 - Recurrent Neural Network, each state h_t , receives information from the previous state, h_{t-1} [20]

ו- $c = q(\{h_1, \dots, h_{T_x}\})$ אוסף ווקטורים מקודדים שנבנו מהשכבות הקודמות. ניתן לראות כי לרשת אין פלט, הרשת מקבלת קלט מווקטור x , ומטמיע אותו ל h .

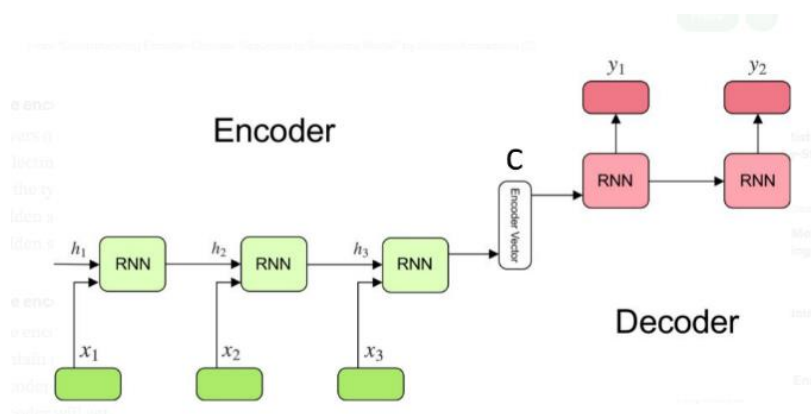


Figure 5 - Encoder/Decoder architecture [23]

המפענח מחפש לנחש את המילה הבאה y_t מווקטור c וכל המילים שקדמו לו $\{y_1, \dots, y_{t-1}\}$. במילים אחרות, המפענח מגדיר הסתברות על פני התרגום y על ידי פירוק ההסתברות המשותפת לתנאים המסודרים מהצורה:

$$p(y) = \prod_{t=1}^T p(y_t | \{y_1, \dots, y_{t-1}\}, c) \quad \text{כאשר } y = (y_1, \dots, y_{T_y})$$

בעבור שימוש ב-RNN במקודד/מפענח, כל הסתברות מותנית מחושבת לפי הנוסחה

$$p(y_t | \{y_1, \dots, y_{t-1}\}, c) = g(y_{t-1}, s_t)$$

כעת, ניתן לחשב את הניחוש של המפענח. פונקציות לא לינאריות, ויכולות לשמש כשכבות נוספות (כגון LSTM) [14].

3. Attention Mechanism

במאמרו [11], הציג הכותב את מגנון ה-Attention, כפתרון לווקטור קידוד באורך קבוע, כך למקודד הייתה גישה מוגבלת למידע המתקבל מהקלט. אורך ווקטור קידוד קבוע גרם לפגיעה ביכולת המודל לחזות ולפענח רצפים באורך גדול. על-מנת להבין את המנגנונים, נגדיר פעולות.

• Alignment scores

מקבל רצף מצבים חבויים מהמקודד h_i , ומספק למפענח פלט נוסף, s_{t-1} , על מנת לחשב את $e_{t,i}$, שמעיד עד כמה הרכיבים של רצף הקלט מתיישרים עם הפלט הנוכחי במקום t . המודל מיוצג על ידי פונקציה, $a()$, שניתן ליישם על ידי רשת עצבית מהצורה

$$e_{t,i} = a(s_{t-1}, h_i)$$

• Weights

המשקולות $a_{t,i}$ מחושבות ע"י פונקציית SoftMax על Alignment scores הקודם:

$$a_{t,i} = \text{softmax}(e_{t,i})$$

• Context vector

ווקטור שמוזן למפענח בכל צעד, הוא מחושב על ידי שקלול כל המשקולות של המצבים החבויים של המקודד, T

$$c_t = \sum_{i=1}^T a_{t,i} h_i$$

המודל מחשב:

1. כל ווקטור, $q = s_{t-1}$, מותאם לרשימת מפתחות על מנת לחשב את scores. פעולת ההתאמה מחושבת כמכפלה של השאילתה הספציפית הנבחרת עם כל ווקטור מפתח, k_i

$$e_{q,k_i} = q \cdot k_i$$

2. scores מועברים דרך פעולת SoftMax ליצירת המשקולות:

$$a_{q,k_i} = \text{softmax}(e_{q,k_i})$$

3. הattention הכללי מחושב לאחר מכן על ידי סכום של ווקטורי הערך, V_{k_i} , כאשר כל וקטור ערך מצוות עם מפתח מתאים:

$$\text{attention}(q, K, V) = \sum_i a_{q,k_i} V_{k_i}$$

שכבת הattention מאפשרת למפענח לגשת למידע שחולץ על ידי המקודד. הוא מחשב וקטור מכלל ה Context vectors, ומוסיף את זה לפלט של המפענח.

4. Attention-based NMT

במאמרו [11], מסביר החוקר כיצד רשתות עצביות המסורתיות מתקשות לעבד רצפי מידע ארוכים, מכאן מגיע הצורך להשתמש במנגנון שפותר את הבעיה. הרעיון מאחורי מנגנון זה הוא להיות שכבה מגשרת בין המקודד והמפענח, ולאפשר למפענח לגשת ולנצל את החלקים הרלוונטיים ביותר של רצף הקלט בצורה גמישה, על ידי שילוב משוקלל של כל ווקטורי הקלט המקודדים, כאשר הווקטורים הרלוונטיים ביותר מיוחסים למשקל הגבוה ביותר.

כפי שהוסבר במודלים הקיימים, NMT מחפש את ההסתברות המותנת $p(y|x)$ עבור משפט המקור x_1, \dots, x_n לתרגום מיטבי עבור משפט יעד y_1, \dots, y_m .

לכן, ניתן לחשב את ההסתברות

$$\log p(y|x) = \sum_{j=1}^m \log p(y_j | y_{<j}, s)$$

ניתן גם לחשב את ההסתברות לכל מילה $y(j)$ במשפט היעד באופן הבא.

$$h_j = f(h_{j-1}, s) \text{ , כר ש } p(y_j | y_{<j}, s) = \text{softmax}(g(h_j))$$

נגדיר

g : פונקציה טרנספורמציה שפולטת ווקטור כגודל המילון.

h : נירון חבוי של RNN.

f : פונקציה המחשבת את הנירון החבוי הנוכחי בהינתן הנירון שקדם לו.

ניתן להגדיר את מטרת האימון לתהליך התרגום בצורה $J_t = \sum_{(x,y) \in D} -\log p(y|x)$

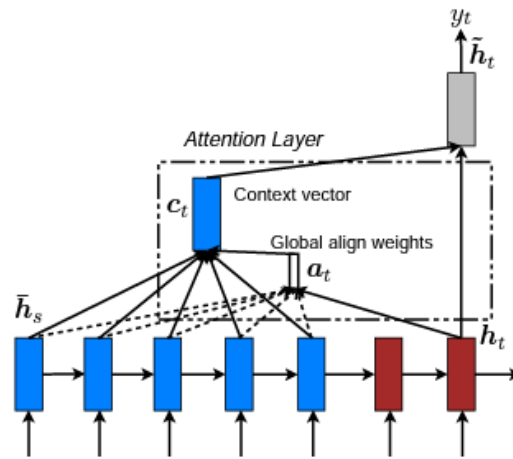


Figure 6 - Attentional model – at each time step t , the model creates an alignment weight vector, a_t , based on the current target state, h_t , and all source states \bar{h}_s . A context vector c_t is then computed as the weighted average, according to a_t , over all the source states [11].

תיאור המערכת

אימון המודל

לאחר שבנינו את הדאטה-סט שלנו, שמונה מעל 1.3 מיליון התאמות של צמדי משפטים באנגלית ותרגומם בעברית, נרצה לאמן את המודל.

בעקבות מחקר נרחב בסוגי מודלי תרגום מכונה, בחרנו להשתמש במודל מסוג Neural Machine Translation, שהוא מודל מסוג מכונת תרגום, בעולם של עיבוד שפה טבעית, שתפס תאוצה בתחום המחקרים בשנים האחרונות.

המודל מממש 3 רכיבים מרכזיים: מקודד, מפענח ושכבת "רגישות" (Attention), וכל אחד שכבות והיפר-פרמטרים שמגדירים אותו.

– Encoder

המקודד בנוי מ-2 שכבות:

- השכבה הראשונה היא שכבת הקלט, והיא בנויה מ-256 ניוונים, ותפקיד להמיר את הטוקנים המתקבלים לטוקטורים.
- השכבה השנייה Gate recurrent network (GRU), היא סוג של RNN. שכבה זו מוגדרת להיות Bidirectional, וזאת מכיוון שגודל ה Context vector קבוע, והינה ניזונה ממצבי עבר וגם עתיד של שכבות חבויות. זאת אומרת, הקלט מוזן משני הכיוונים לתוך הניוונים, על מנת לנתח את הרצף טוב יותר.

המקודד:

- לוקח רשימה של טוקנים (מ-Context vector).
- מחפש וקטור הטמעה עבור כל טוקן (באמצעות שכבת הקלט).
- מעבד את הטוקנים לרצף חדש (באמצעות ה-BI-RNN).
- מחזירה את הרצף המעובד. זה יועבר לראש ה-Attention.

המפענח בנוי מ-3 שכבות:

- השכבה הראשונה היא שכבת הפלט והיא בנויה מ-256 נוירונים, ותפקיד להמיר את הטוקנים המתקבלים לטוקטורים.
- השכבה השנייה Gate recurrent network (GRU), היא סוג של RNN. שכבה זו אינה יכולה להיות דו-כיוונית מכיוון שבכל צעד, היא מעבדת מילה אחת שאותה היא "מנחשת" עבור הקלט. הפלט של ה-RNN הוא הקלט של שכבת Attention.
- שכבת Dense הפלט, תפקידה לנתח את הווקטור מהשכבה הקודמת, ולהמיר אותו לטוקן הבא עבור כל מילה ברצף המקור.

המפענח:

- הוא מחפש הטמעות עבור כל טוקן ברצף היעד.
- הוא משתמש ב-RNN כדי לעבד את רצף היעד, ולעקוב אחר מה שהוא יצר עד כה.
- הוא משתמש בפלט RNN כ"שאלתה" לשכבת Attention, כאשר הוא מטפל בפלט של המקודד.
- בכל מיקום בפלט הוא חוזה את האסימון הבא.

– Attention

שכבת Attention מאפשרת למפענח לגשת למידע שחולץ על ידי המקודד. הוא מחשב ווקטור מכל Context vector, ומוסיף את זה לפלט של המפענח.

ה-Attention מכיל שכבת נוירונים אחת בגודל 256 (במקרה שלנו), שמתחברים לנוירון יחיד שנקרא ראש השכבה.

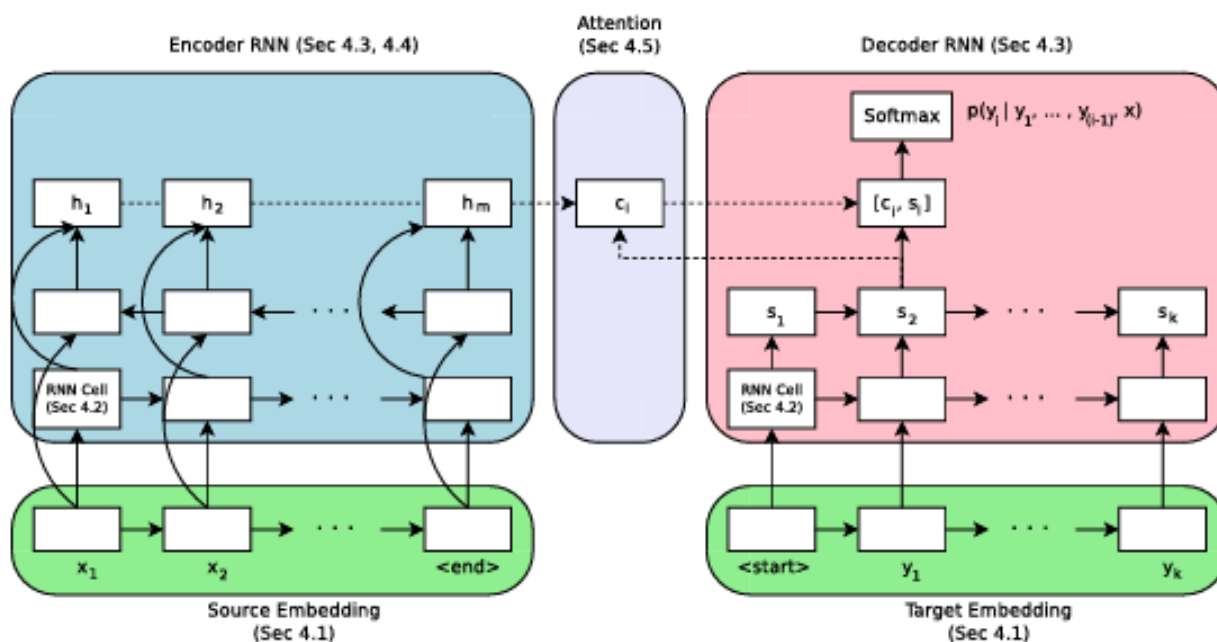


Figure 7 - Elaborated scheme of the Attention-based NMT [18]

– Optimizer

מטרתנו להפחית את הloss הכולל ולשפר את הלמידה על-ידי תיקון טעויות. ישנם מספר Optimizers פופולריים, אנחנו בחרנו להשתמש ב-Adam.

– Vocabulary size

כמה מילים המודל יכול ללמוד. הגישה הנאיבית אומרת שכמה שיותר, אבל כמות גדול מידי של מילים, ביחס לפרמטרים, גורמת לפגיעה באיכות המודל. הערך ההתחלתי שבחרנו היה 5,000 מילים, אך מהר מאוד המודל מילא את המילון ולכן לא יכל ללמוד מילים חדשות.

ניסיון נוסף היה 50,000 מילים, אך גם זה השפיע לרעה על תוצאות המודל, כי ראינו שנוצרים פערים גדולים מידי בין גודל המילון בעברית לעומת המילון באנגלית.

לבסוף, לאור תוצאות טובות יותר, בחרנו במיון בגודל 10,000.

– Layers

למקודד בחרנו להשתמש ב-2 שכבות, כאשר אחת היא דו-כיוונית, למקודד בחרנו להשתמש ב-3 שכבות. Attention השתמשנו בשכבה אחת.

– Epochs

אומנם במודל בחרנו להשתמש ב-100 Epochs, בחרנו להוסיף callback שמגדיר $patience=5$, זאת אומרת, שאם המודל מזהה שלא חל שיפור על ה-loss יחסי ב-5 Epochs האחרונים, הוא עוצר את הלמידה. וזאת, בכדי לחסוך בזמן חישוב.

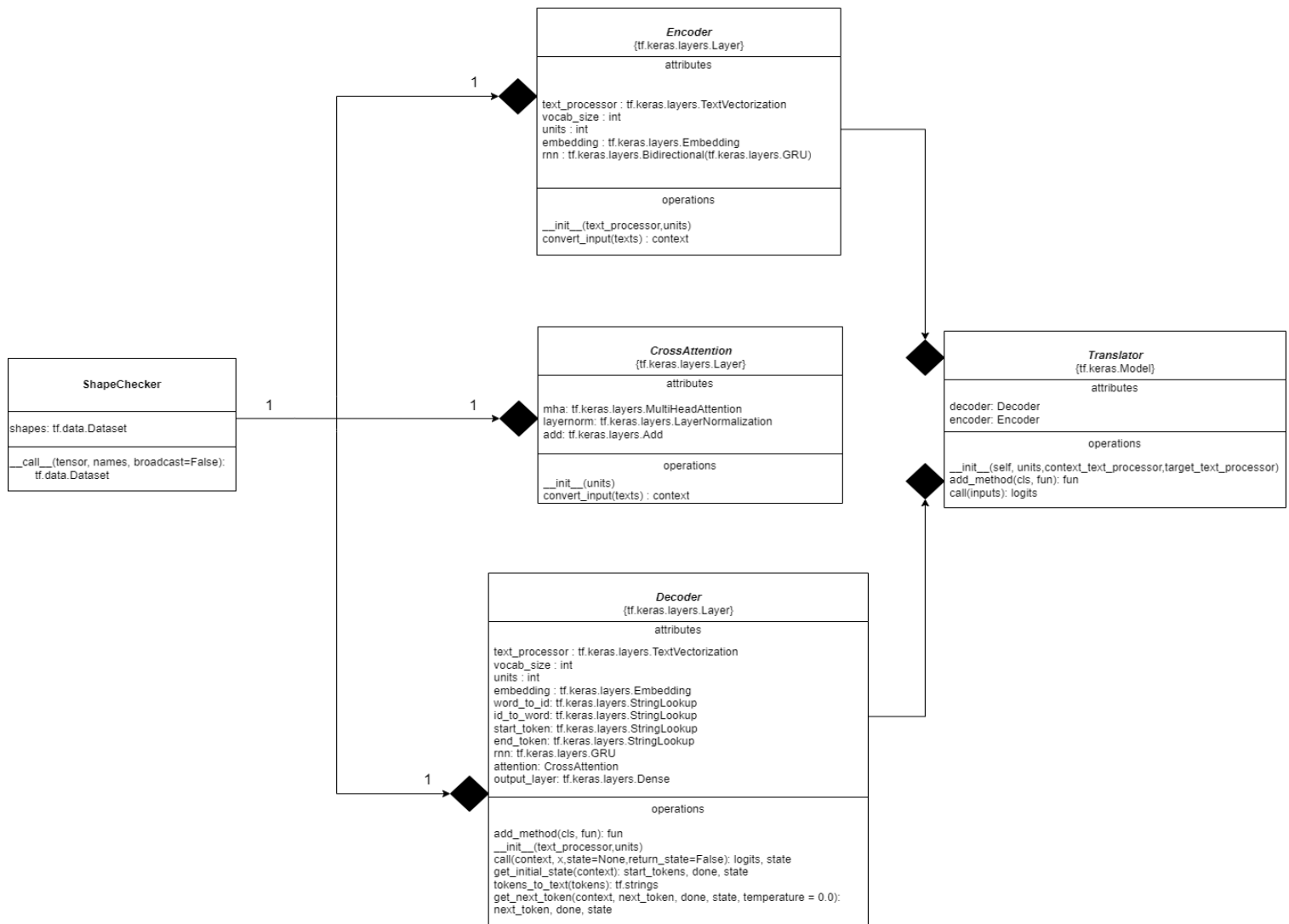


Figure 8 - UML diagram for Attention-based NMT

על מנת להציג את ארכיטקטורת המודל, בחרנו ליצור דיאגרמת UML, שמתארת את הקשרים בין המקודד, המפענח, וה-Attention, ואת המתודות שכל מחלקת מממשת.

תהליך יצירת הדאטה סט היווה מכשולים רבים.

הרעיון הראשוני והנאיבי הגיע מתוך הבנה שקבצי SRT (קבצי תרגום) בנויים מאינדקסים, לכן יהיה ניתן להשוות חותמות זמן ואינדקסים בין שני קבצי תרגום (אחד באנגלית והשני בעברית) בעבור אותו סרט.

כאשר ניסינו לבצע את תהליך ההתאמה בצורה הנ"ל נחשפנו לפערים עצומים בין תרגומי הסרטים – קיימים פערים גורפים בין 2 קבצי תרגום בעבור אותו סרט. נרחיב ונאמר, שלא מצאנו אפילו 2 קבצי כתוביות שתואמים באותה שפה. הדבר קורה כתוצאה ישירה מהעובדה שתרגום סרטים נעשה על-ידי בני אדם.

מתרגמים שונים בונים את קבצי התרגום בדרכים שונות, לדוגמא ייתכן שבמהלך הסרט, אחת הדמויות מתכתבת בטלפון. לא מחייב שהמתרגם יבחר להכניס את תוכן ההודעה לכתוביות.

תרגומי מכתבים, מקומות ותארכים, קרדיטים ופתיחים, כל אלה נתונים לשיקול המתרגם.

הפתרון שעלה על מנת לעקוף את המכשול היה לתרגם ידנית, אך קבצי SRT של סרט מכילים בדרך כלל אלפי משפטים בממוצע. בכדי ליצור מודל שבאמת יענה על הציפיות שלנו, הדרישה הראשונית הייתה ליצור כמה עשרות אלפי משפטים בעלי התאמה מרבית.

הפתרון המיטבי שבחרנו היה ביצירת אלגוריתם התאמה.

אספנו אלפי קבצי תרגום בשפת היעד ושפת המקור דרך ממשק אינטרנטי. בהתאמה חיפשנו קבצי תרגום באנגלית, לאחר מכן ניקינו את חותמות הזמן והאינדקסים, מיינו את תרגומי הסרטים לפי שם קובץ בהתאמה לעברית ואנגלית עם הפרדה של תו מיוחד בין משפטים שלמים. לבסוף קיבלנו קורפוס המונה 13.5 מיליון התאמות בין שפת המקור ושפת היעד.

63	This beer will not hurt you	That beer won't do you no good	הבירה הזאת לא תזיק לך
68	If you're still impoverished I have another dollar	If you're still broke I think I got another dollar	אם אתה עדיין מרושש יש לי עוד דולר
71	- Chance you let him do it	-Chance you going to let him do that	צ'אנס אתה נותן לו לעשות את זה-
69	Give him the key to your cell whenever he wants	I'll let him have the key to your cell anytime he wants it	נותן לו את המפתח של התא שלך כל פעם שהוא רוצה
69	- He's too calm and has no one to back him up	It'd be too easy He's got nobody to back him up	הוא יותר מדי רגוע אין לו אף אחד לגיבוי-
67	If he speaks without a permit, pour a bucket of water on him	If he talks out of turn throw a bucket of water on him	אם הוא ידבר בלי היתר שפוך עליו דלי מים
92	- I'll pour him in the middle of the bed and leave him to sleep in it	I'll throw one in the middle of his bed and leave him to sleep in it	אני אשפוך לו באמצע המיטה ואשאיר אותו לישון בה-

Figure 9 - Generated random rows from our dataset

פיתחנו אלגוריתם התאמה שרץ על מערך המשפטים כקלט, לוקח משפט בעברית, מתרגם בעזרת מכונת תרגום קיימת. לאחר מכן, בעזרת אלגוריתם "Levenshtein Distance" משופר ואלגוריתמי התאמה נוספים, ביצענו חיפוש יעיל למשפט באנגלית בקובץ.

בכדי לשפר תוצאות, הרצנו על כל התאמה ריצה חוזרת עם למדא של משפטים קודמים ועוקבים.

לאחר מכן שמצאנו התאמה, שמרנו את השדות הבאים:

המשפט בעברית מתרגומי הסרטים, המשפט המתורגם, המשפט באנגלית מתרגומי הסרטים ואחוזי התאמה בין המשפטים באנגלית.

ניתן לראות את הבדלי התרגום, איך תרגום הסרטים כן מצליח לתפוס סלנגים וביטויים יותר מתרגום של מכונה רגילה.

נסקר את האלגוריתמים שהחוקר מציע בעזרת המאמר [16].

$$\text{lev}_{a,b}(i,j) = \begin{cases} \max(i,j) & \text{if } \min(i,j) = 0, \\ \min \begin{cases} \text{lev}_{a,b}(i-1,j) + 1 \\ \text{lev}_{a,b}(i,j-1) + 1 \\ \text{lev}_{a,b}(i-1,j-1) + 1_{(a_i \neq b_j)} \end{cases} & \text{otherwise.} \end{cases}$$

הפונקציה מגדירה מרחק בין שתי מחרוזות על ידי מספר פעולות העריכה שצריכים לבצע כדי להגיע ממחרוזת אחת לשנייה. פעולת עריכה אפשרויות - הוספה של תו, מחיקה של תו או החלפה של תו.

למשל, המרחק בין המילה **שלו** למילה **חלום** הוא 1, משום שכדי לעבור מהמילה הראשונה לשנייה אנו צריכים לבצע פעולת עריכה אחת - להחליף **ש** ב-**ח**. המרחק בין המילה **שלו** למילה **שלו** גם הוא 1, משום שגם כאן כדי לעבור מהמחרוזת הראשונה לשנייה צריכים לבצע פעולת עריכה אחת - להוריד את האות האחרונה **ם**.

		ח	י	פ	א	י	ם
	0	1	2	3	4	5	6
ח	1	0	1	2	3	4	5
י	2	1	0	1	2	3	4
פ	3	2	1	0	1	2	3
א	4	3	2	1	1	2	3
י	5	4	3	2	2	1	2
ו	6	5	4	3	3	2	2
ת	7	6	5	4	4	3	3

Figure 11 – Levenshtein distance relations between two Hebrew words [22].

		H	Y	U	N	D	A	I
	0	1	2	3	4	5	6	7
H	1	0	1	2	3	4	5	6
O	2	1	1	2	3	4	5	6
N	3	2	2	2	2	3	4	5
D	4	3	3	3	3	2	3	4
A	5	4	4	4	4	3	2	3

Figure 10 - matching HONDA and HYUNDAI with Levenshtein distance algorithm [16]

Enhanced Levenshtein Distance

הבעיה העיקרית עם מרחק לוינסטיין הוא חוסר התחשבות באורך המחרוזות המתקבלת. בדוגמה של חיפוש הפלפלים, נראה כי המונח הדומה ביותר ל"פלפלים" הוא "פלפל" (במרחק 2), אחריו "אבטיח" (במרחק 5) ובמקום האחרון "פלפל ירוק חריף" (במרחק 9), וזאת למרות שהיינו רוצים ש"פלפל ירוק חריף" יהיה קרוב יותר מאשר "אבטיח".

נתגבר על הבעיה הזאת אם נשתמש במרחק לוינסטיין מנורמל: נחלק את מרחק לוינסטיין באורך המחרוזת הארוכה מבין שני המונחים אותם אנחנו משווים. בצורה זו נקבל מספר בין 0 ל-1 שיהווה חיווי של מידת הקרבה, כאשר 0 הוא הקרוב ביותר ו-1 הוא הרחוק ביותר.

בשיטה זו, המרחק בין "פלפלים" ל"אבטיח" הוא 0.83 והוא אכן גדול יותר מהמרחק בין "פלפלים" ל"פלפל ירוק חריף", שהוא 0.64. שיטה זו מביאה לידי ביטוי את העובדה שבמרחק לוינסטיין הרגיל ל"פלפל ירוק חריף" יש מחיר גבוה, בעיקר בגלל האורך שלו. למרות הדמיון ל"פלפלים" בתחילת המחרוזת, ואילו המרחק הגבוה של "אבטיח" נובע מכך שהוא שונה לחלוטין מ"פלפלים".

אלגוריתם ההתאמה שלנו נדרש לאמוד מרחק בין מחרוזות שלמות ולא רק בין תווים, תוך השמת דגש על שלל חוקים דקדוקיים מאנגלית (רבים, שיוך, זמנים אותיות גדולות וכו').

כיוון שהפרויקט שבחרנו מגוון ודורש מחקר ופיתוח, במהלך פיתוח הפרויקט התנסו והשתמשנו בכלים רבים ונרחבים. ננסה לסקור את המרבית הכלים הנדרשים כדי לממש את המערכת.

פיתוח הקורפוס ואלגוריתם ההתאמה

• Pandas –



ספריית Python המשמשת לעבודה עם מבני נתונים, בעלת פונקציות לניתוח, ניקוי, עיבוד ומניפולציה בנתונים. השם "Pandas" מתייחס ל"Python Data Analysis" והוא נוצר על ידי Wes McKinney ב-2008. Panda מאפשרת לנו לנתח ביג דאטה ולהסיק מסקנות על המידע.

• GoogleTrans Api –

ספריית שמממשת את ה-API של Google Translate. משתמשת ב-Google Translate Ajax API כדי לבצע קריאות לשיטות כמו זיהוי ותרגום. השימוש בספרייה עבורנו נועד על מנת לכמת את היחס בין משפט המקור ומשפט היעד, כאשר נרצה לשמור לקורפוס רק צמדי משפטים בעלי יחס תרגום גבוה (במטרת לצמצם רעשים ולחזק אמינות הקורפוס).

• Num2words –

ספרייה לעיבוד נתונים מספריים לטקסט ותומכת במספר שפות, אנחנו נעזרו בעיבוד מאנגלית לעברית.

• FuzzyWuzzy –

ספרייה להתאמת מחרוזות ותתי-מחרוזות, אשר משתמשת ב-"Levenshtein Distance" כדי לחשב את ההבדלים בין רצפים.

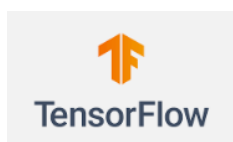
פיתוח הדאטה-סט ואימון המודל

• Google Collaboratory –



או בקיצור "Colab", הוא מוצר מ-Google Research המאפשר למפתחים לכתוב ולהפעיל קוד פייטון דרך הדפדפן, והוא מתאים במיוחד ללמידת מכונה, ניתוח נתונים ואימון מודלים. מבחינה טכנית יותר, Colab הוא שירות מחברת Jupyter, תוך מתן גישה ללא תשלום למשאבי מחשוב כולל GPUs אשר רצים על-גבי שרתים מרוחקים. על-מנת לשפר את יכולות החישוב של המודלים שלנו, וייעול תהליכים, בחרנו להרחיב לחשבון עסקי בתשלום.

• TensorFlow –



ספריית קוד פתוח לפייטון לביצוע ניתוחים וחישובים נומריים, ומאגדת שלל מודלים ואלגוריתמים של למידת מכונה ולמידה עמוקה (המכונה גם רשתות עצביות). TensorFlow משתמש ב-Python או ב-JavaScript כדי לספק ממשק API נוח לבניית יישומים. בפרויקט שלנו, אנחנו משתמשים בכלים רבים של הספרייה, כגון בניית המודל, ניתוח והערכה לביצועי המודל, עיבוד המידע ועוד.



- NumPy –

ספריית קוד פתוח ב-Python המשמשת לעבודה עם מבני נתונים בקנה מידה גדול, יש לו גם פונקציות לעבודה בתחום של אלגברה לינארית, טרנספורמציה לינארית ומטריצות. NumPy נוצר בשנת 2005 על ידי Travis Oliphant, וראשי תיבות של NumPy הם Numerical Python. בפרויקט שלנו השתמשנו רבות בספרייה זו, על מנת לנרמל את המידע לאימון המודל.



- Matplotlib –

ספריית קוד פתוח Cross-Platform, הדמיית נתונים, ויזואליזציה וציור גרפי עבור Python. מפתחים יכולים גם להשתמש בממשקי ה-API של matplotlib (ממשקי תכנות יישומים) כדי להטמיע ייצוג גרפי ביישומי GUI. בפרויקט, השתמשנו בספרייה זו על-מנת להציג ויזואלית את איכות המודל והמדדים שהוכרעו.

פיתוח ממשק המשתמש

- Kivy –

ספריית פיתוח GUI מרובת פלטפורמות פתוחות עבור Python ויכולה מערכות הפעלה רבות. הרעיון הבסיסי מאחורי Kivy הוא לאפשר למפתח לבנות אפליקציה פעם אחת ולהשתמש בה בכל המכשירים, מה שהופך את הקוד לריוזאבילי וקל לפריסה, מה שמאפשר עיצוב אינטראקציה מהיר וקל ויצירת ממשק מהיר.



סביבת הפיתוח

שפת הפיתוח בה בחרנו להשתמש היא Python, וזאת משום ישנם כלים רבים הקשורים בתחום של למידת מכונה ככלל, ועיבוד שפה טבעית (NLP) בפרט. בנוסף, עבור Python קיימים ממשקי משתמש רבים ונוחים לשימוש.

על מנת לפתח מודל מכונת תרגום, השתמשנו ב-Google Colab, וזאת מכיוון שגוגל מספקת מכונה וירטואלית חזקה שעל גבה ניתן לפתח ולאמן את המודל לא צורך בכוח עיבוד מצידנו. המודל אומן על גבי "המחברת" ולא דרש מאיתנו להריץ את התוכנית על מחשב.

את ממשק המשתמש בנינו על גבי visual studio code ו-PyCharm עם Python, מטעמי נוחות והיכרות קודמת עם הכלים, בנוסף, השתמשנו ב-Kivy, שהיא ספריית GUI מודולרית שמאפשרת פיתוח חלונות בצורה נוחה.

תוצאות

נבחן במודל המאומן מספר פרמטרים, אך תחילה נגדיר אותם, וכיצד הם בודקים את איכות המודל:

Loss – הוא "קנס" על חיזוי רע. במילים אחרות, Loss הוא ערך שמציין עד כמה גרוע חיזוי המודל היה בעבור דוגמה אחת.

Accuracy – מידת הדיוק היא מדד אחד להערכת מודל. במילים פשוטות, רמת הדיוק היא אותו חלק מהחיזוי שהמודל שלנו פעל נכון.

Confusion matrix – מטריצת $N \times N$, המשמשת להערכת הביצועים של המודל, כאשר N הוא מספר מחלקות היעד. המטריצה משווה את ערכי היעד בפועל לאלו שמודל חזה. המטריצה נותנת לנו ראייה הוליסטית של ביצועי מודל הסיווג שלנו ואילו סוגי שגיאות הוא עושה.

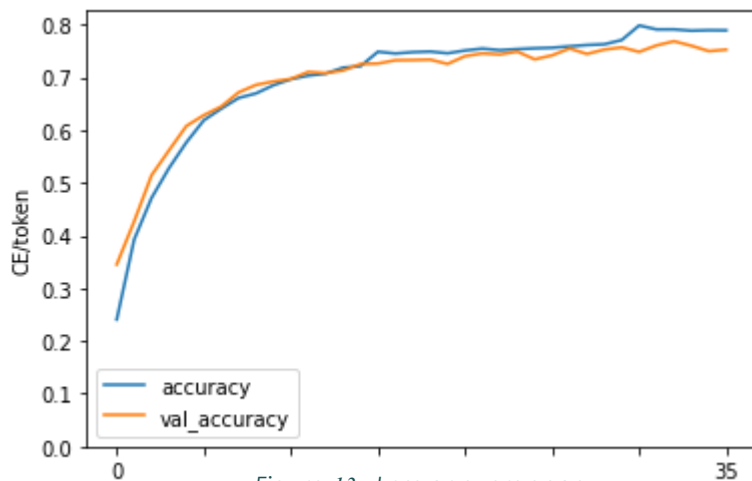


Figure 13 - Loss convergence throughout the training

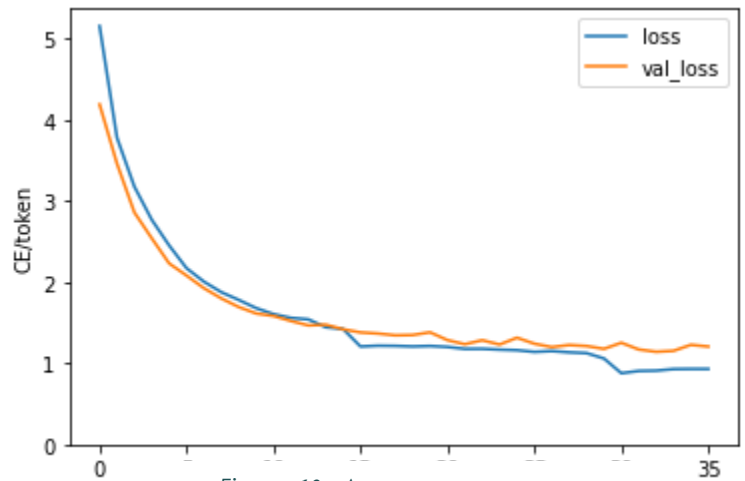


Figure 12 - Accuracy convergence throughout the training

ניתן לראות את התכנסות ה-Loss function כלפי מטה, לאחר האימון, עד להגעה לטווח רצוי, ומנגד, ניתן לראות את התכנסות כלפי מעלה ה-Accuracy, לאחר האימון, עד להגעה לטווח רצוי.

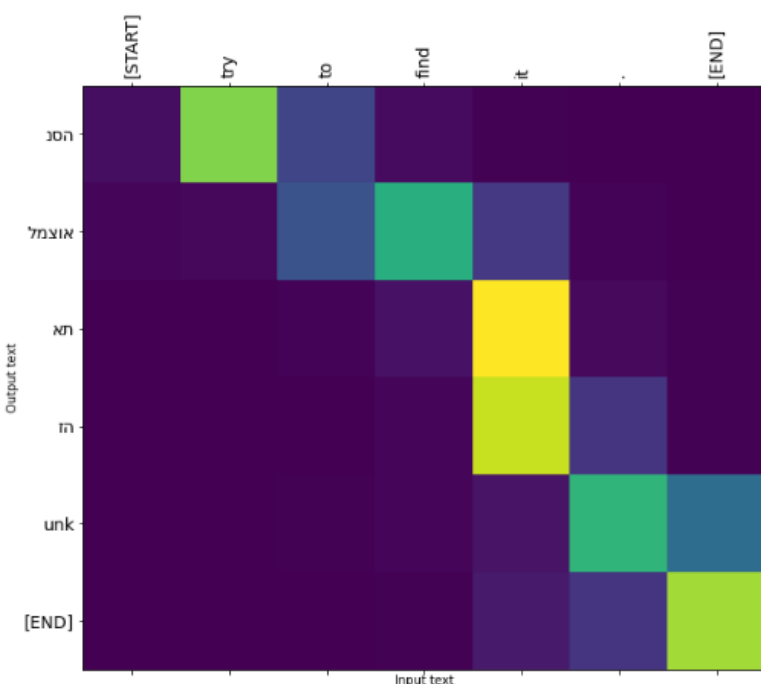


Figure 14 - plotting the Attention for source sentence "Try to find it.", and target sentence "נסה למצוא את זה."

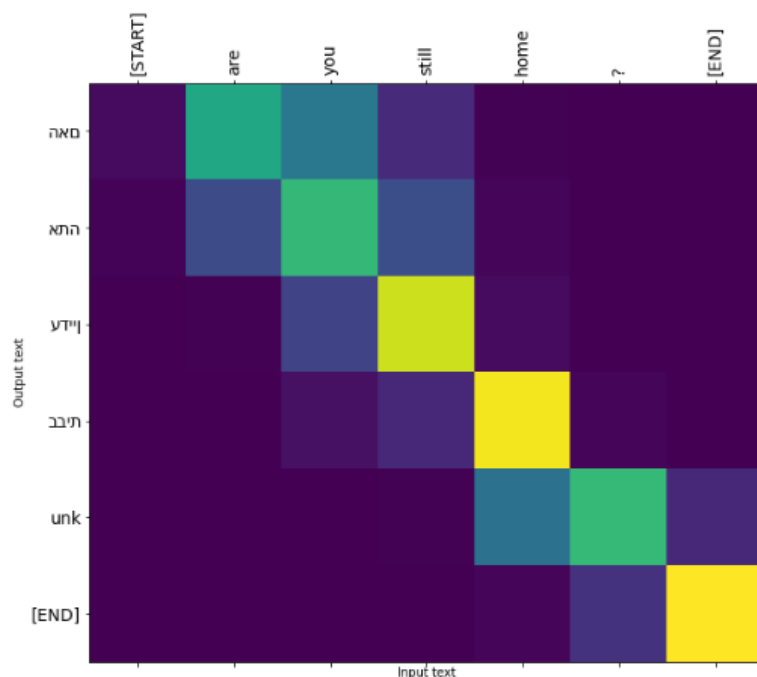


Figure 15 - plotting the Attention for source sentence "Are you still home?", and target sentence "האם אתה עדיין בבית?"

באיוורים, ניתן לראות את confusion matrixes של שכבת הAttention עבור משפטי המקור "are you still home?" ו-"try to find it.", ומשפטי היעד "האם אתה עדיין בבית?" ו-"נסה למצוא את זה.", בהתאמה.

ניתן לראות על אילו מילים הAttention למד לשים דגש, ולמה.

באיוור הימני, יש דגש על מילת המקור **home** אל מול מילת היעד **בבית**. שהיא שם-עצם באנגלית, אך גם תואר הפועל וגם תואר (כפי שנענה בבלוג של מילון בריטניקה [19]). אך, המילה **בבית** מורכבת מאות שימוש + שם עצם, **ב+בית**, (בלבד- בצורתה "בית").

באיוור השמאלי, ניתן לראות שצירוף המילים **to find** תורגם ל-**למצוא**, זאת אומרת ששכבת הAttention, ידעה לתת משקול גבוה יותר למילה המקור **to** ביחס למילת היעד **למצוא**. לפי צבעי המשבצות, ניתן להבין המילה **find** היא המילה שממנה המילה **למצוא** תורגמה, אך ניתן לראות את הקשר בין המילים.

בנוסף, מילת המקור **it** תורגמה לצירוף המילים בעברית **את זה**. על-פי צבעי המשבצות, ניתן להבין שהמילה **את** קיבלה יחס יותר גבוה, אך עדיין, משקול דיי קרוב. המילה **את**, גם היא בעלת משמעות נוספת שלא יוחס כאן, **את – you**, כנקבה. מסקירת המודל, נראה האימון היה מספק, וניתן לראות את היכולת להתמודד עם חלק מאותם אתגרים בתרגום למידת מכוונה, שהצגנו.

דרישות מערכת

1. כמשתמש, ארצה לקבל כתוביות תרגום בעברית לסרט או סרטון, כדי שאוכל לקרוא את הכתוביות ולהבין את הסרט/סרטון. (FR)
 - 1.1. כמערכת, אצפה לקבל קובץ וידאו מהמשתמש, עליו אצור את הכתוביות. (FR)
 - 1.2. כמערכת, אצפה לקבל קובץ אודיו מהמשתמש, עליו אצור את הכתוביות. (FR)
 - 1.3. כמערכת, אצפה לקבל קישור לאתר YouTube, ממנו אצור את הכתוביות. (FR)
 - 1.4. כמשתמש, אצפה שתיהנה יכולת שליטה של הזזת חותמות הזמן בקובץ התרגום על מנת שאוכל להטמיע את הכתוביות בתזמון מדויק. (FR)
 - 1.5. כמשתמש, ארצה שתיהנה יכולת הזזת הכתוביות למקום שאבחר על סרטון הוידאו. (FR)
 - 1.6. כמשתמש, ארצה שתיהנה היכולת לבחור את סוג וגודל פונט התרגום. (FR)
 - 1.7. כמשתמש, אצפה שתיהנה יכולת להעלות קובץ וידאו / אודיו / כתובת URL של סרטון YouTube. (FR)
 - 1.8. כמערכת אצטרך את היכולת להפוך קובץ וידאו לקובץ אודיו באיכות גבוהה. (FR)
2. כמערכת, ארצה שיצירת הכתוביות לקובץ לא יערך מעבר לזמן סביר ומוגדר מראש ביחס ליכולות המערכת. (NFR)
 - 2.1. כמשתמש אצפה להמתנה לסרט הפלט בזמן סביר. (NFR)
3. כמערכת, ארצה לא לפגוע באיכות הסרטון אותו קיבלתי. (NFR)
 - 3.1. כמשתמש, אצפה שקובץ הפלט יהיה באיכות זהה (או טובה יותר) לזו שהכנסתי. (NFR)
 - 3.2. כמשתמש ארצה שגודל קובץ הפלט יהיה שקול לקובץ הקלט. (NFR)
4. כמערכת, ארצה לתמוך במספר רב של פורמטי וידאו. (NFR)
 - 4.1. כמערכת ארצה לתמוך בפורמטים הבאים mp4, wmv, avi, mov, swf הן לקלט והן לפלט. (NFR)

5. כמערכת אצטרך להחזיק ביכולות תרגום מתקדמות ביותר על מנת שאיכות הצפייה והבנת קובץ הוידאו/אודיו יהיו טובים יותר. (FR)
- 5.1. כמערכת אצטרך יכולת למידת מכונה ובינה מלאכותית לתרגום איכותי. (NFR)
- 5.2. כמערכת ארצה אחוז התאמה גבוה בין התרגום משפת המקור לשפת היעד, ותרגום חוזר של הטקסט בשפת היעד לשפת המקור. (FR)
- 5.3. כמערכת תרגום מכונה ארצה שתי שכבות בדיקה אחת עם מודל קיים, והשנייה עם המודל הנוכחי. (FR)
6. כמערכת, אצטרך מערכת בינה מלאכותית שתבצע speech to text על קובץ אודיו. (FR)
- 6.1. כמערכת אצטרך את היכולת להטמיע חותמות זמן בקובץ הפלט. (FR)
- 6.2. כמערכת אצטרך לקבוע את פורמט קובץ הפלט כקובץ (NFR). (SRT)
7. כמערכת, ארצה לשמור בענן את ניתוח הקובץ שביצעתי עבור המשתמש, על מנת שאוכל ללמוד ממנו. (NFR)
8. כמערכת, ארצה להחזיק בסיס נתונים המכיל את תרגומי המשפטים. (NFR)
- 8.1. כמערכת, ארצה לשייך את הסרט/סרטון לקטגוריה, על מנת שאוכל לקטלג אותו ומצבים דומים לו בצורה טובה יותר. (FR)
9. כמערכת, ארצה שיהיה לי גישה לרשת בכל רגע נתון בו המערכת בשימוש. (NFR)
10. כמערכת, ארצה להיות מאובטחת נגד תקיפות, כדי לא לאפשר פגיעה במערכת. (NFR)
- 10.1. כמערכת, ארצה להיות מאובטחת נגד תקיפות (NFR). (cross-site-scripting)
- 10.2. כמערכת, ארצה להיות מאובטחת נגד תקיפות (NFR). (SQL Injection)
11. כמערכת, ארצה להחזיק במאגר מידע המכיל מחרוזות של כתוביות (בעברית ואנגלית) על מנת שאוכל ללמוד בעזרתה. (NFR)
- 11.1. כמפתח, ארצה את היכולת למשוך מהרשת זוגות של מסמכי כתוביות עבור כל סרט שנכנס למאגר. (עבור כל סרט - זוג תרגומים). (NFR)
- 11.2. כמפתח, ארצה את היכולת להתאים בין האינדקסים של הקבצים השונים, על מנת להגיע להתאמה מרבית בין התרגומים. (NFR)
- 11.3. כמערכת, אצטרך יכולת תרגום ותמיכה בשפה העברית ובשפה האנגלית למשפטים. (NFR)
- 11.4. כמפתח, אצטרך יכולות אוטומציה לתהליכים הקשורים ביצירת מאגר המידע ואסיפת הנתונים. (NFR)
- 11.4.1. כמפתח, אצטרך יכולות אוטומציה להורדת מסמכי כתוביות בשתי השפות. (NFR)

11.4.2. כמפתח, אצטרך יכולות אוטומציה להתאמה וריצה בין 2 האינדקסים בקבצי שפות שונים

עבור אותו סרט. (NFR)

12. כמפתח, ארצה לקבל דוח אודות שיעור התקינות של המידע היוצא מהמערכת. (FR)

12.1. כמפתח, ארצה לקבל דוח אודות שיעור הסטיות של המערכת עבור כל קובץ תרגום. (FR)

12.2. כמפתח, ארצה לקבל דוח אודות שיעור המילים הלא מוכרות למערכת. (FR)

12.3. כמפתח, ארצה לקבל דוח אודות כמות המילים בשפות שאינן תואמות את המערכת. (FR)

13. כמערכת, ארצה לתמוך במספר סוגי מסכים, כדי לפנות לקהל רחב יותר. (NFR)

13.1. כמערכת, ארצה לתמוך במסכי מחשב. (NFR)

13.2. כמערכת, ארצה לתמוך במוניטורים. (NFR)

13.3. כמערכת, ארצה לתמוך בטאבלטים. (NFR)

13.4. כמערכת, ארצה לתמוך במובייל. (NFR)

14. כמערכת, ארצה לתמוך במספר דפדפנים שונים, כדי לפנות לקהל רחב יותר. (NFR)

14.1. כמערכת, ארצה לתמוך בדפדפן (FR. Chrome)

14.2. כמערכת, ארצה לתמוך בדפדפן (FR. Edge)

14.3. כמערכת, ארצה לתמוך בדפדפן (FR. Firefox)

15. כמערכת, ארצה שכל המשתמשים יהיו חשופים לאותם נתונים. (NFR)

16. כמערכת, ארצה להתממשק לבסיס נתונים. (NFR)

17. כמערכת, ארצה להתממשק לשרת. (NFR)

18. כמערכת, ארצה גישה לממשקים חיצוניים, על מנת שאוכל להגדיל את יכולותיי. (NFR)

18.1. כמערכת, ארצה גישה לממשק (NFR. Speech-TO-Text)

18.2. כמערכת, ארצה גישה לממשק תרגום אנגלית-עברית. (NFR)

דרישות GUI

19. כמשתמש, ארצה שיהיה אזור במערכת בו אכניס את הקובץ שלו ארצה ליצור כתוביות. (FR)
- 19.1. כמשתמש, ארצה שיהיה כפתור עליו אלחץ כדי לבחור קובץ וידאו מהמחשב. (FR)
- 19.2. כמשתמש, ארצה שיהיה כפתור עליו אלחץ כדי לבחור קובץ אודיו מהמחשב. (FR)
- 19.3. כמשתמש, ארצה שיהיה אזור טקסט בו אכניס את קישור URL לסרטון ה-Youtube שבחרתי. (FR)
- 19.4. כמשתמש, ארצה שיהיה כפתור אישור, לאחר שבחרתי את הסרט שלו ארצה ליצור כתוביות. (FR)
20. כמשתמש, לאחר שהעליתי את הקובץ שלו ארצה ליצור כתוביות ואישרתי על-ידי הכפתור המתאים, יופיע משבצת המורה על טעינת הקובץ. (FR)
21. כמשתמש, לאחר שהעליתי את הקובץ שלו ארצה ליצור כתוביות ואישרתי על-ידי הכפתור המתאים, ארצה לקבל את קובץ הפלט, ע"י לחיצה את כפתור ההורדה. (FR)
- 21.1. כמשתמש, לאחר שהעליתי את קובץ הוידאו שלו ארצה ליצור כתוביות ואישרתי על-ידי הכפתור המתאים, ארצה לקבל את קובץ וידאו בחזרה המכיל את הכתוביות, ע"י לחיצה את כפתור ההורדה. (FR)
- 21.2. כמשתמש, לאחר שהעליתי את קובץ האודיו שלו ארצה ליצור כתוביות ואישרתי על-ידי הכפתור המתאים, ארצה לקבל את קובץ וידאו בחזרה המכיל את הכתוביות, ע"י לחיצה את כפתור ההורדה. (FR)
- 21.3. כמשתמש, לאחר שהעליתי את הקישור ל-Youtube שעבורו ארצה ליצור כתוביות ואישרתי על-ידי הכפתור המתאים, ארצה לקבל קובץ כתוביות מתאים עבור אותו סרטון, המכיל את הכתוביות, ע"י לחיצה את כפתור ההורדה. (FR)

מסע לקוח

בחלק זה נתאר את חווית המשתמש UI/UX ואת אופן השימוש באפליקציה.

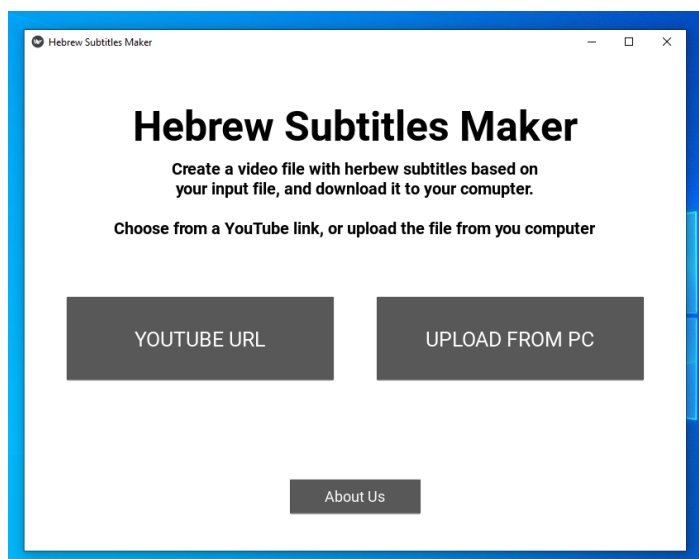
1. העלאת קישור מ-YouTube

המשתמש יזין את הלינק לקישור ה- YouTube שעל גבו ירצה להוסיף כתוביות בעברית. ברגע שבחר, המערכת תדע לפרק את קובץ הוידאו לקובץ כתוביות באנגלית, ולקובץ וידאו. לאחר מכן, המערכת תבצע תרגום מאנגלית לעברית לכל משפט בקובץ, ותכתוב מחדש את הטקסט בקובץ הכתוביות.

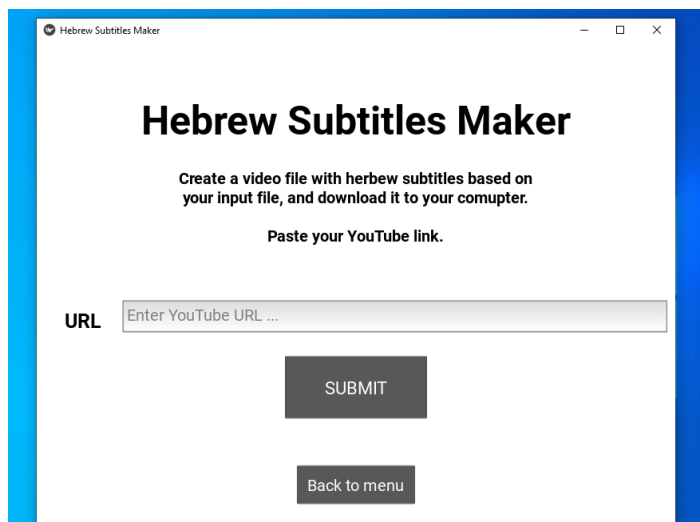
המערכת תטמיע את קובץ הכתוביות המתורגם בקובץ הוידאו שנשמר, ותאפשר למשתמש לבצע הורדה של הקובץ הוידאו בתוספת הכתוביות למחשבו של המשתמש.

כעת, יוכל המשתמש לצפות בסרטון עם כתוביות בשפה העברית.

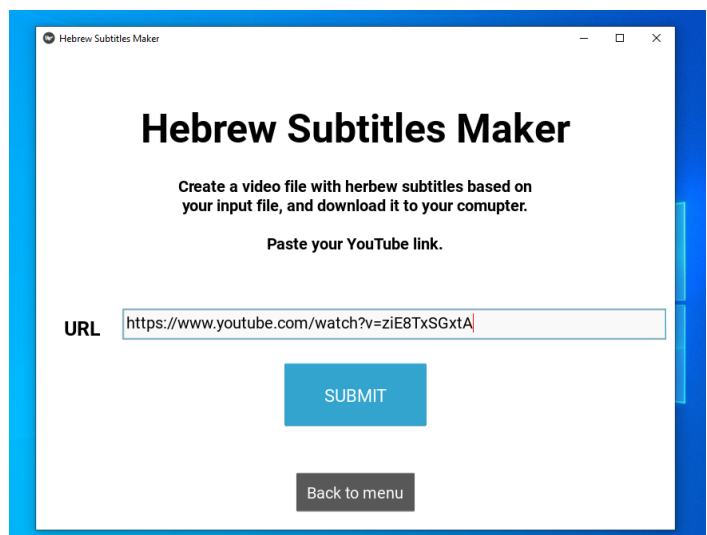
1. המשתמש יכנס ליישום, מסך הראשי יפתח.



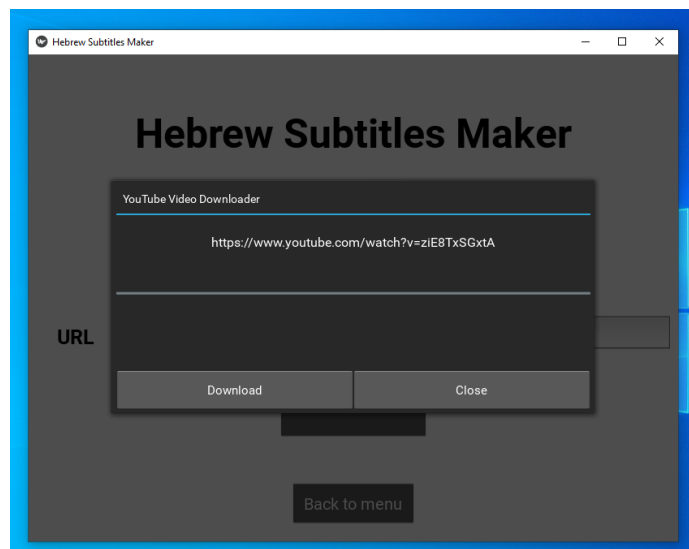
2. המשתמש ילחץ על כפתור ה- "YOUTUBE URL", ויעבור למסך הבא.



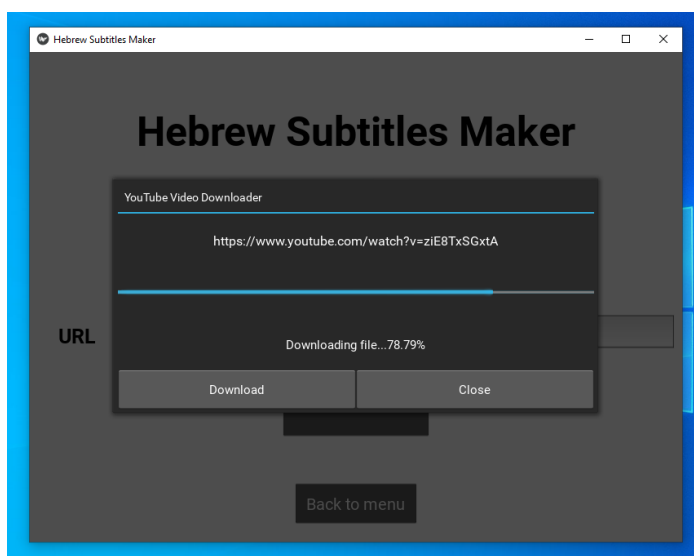
3. המשתמש יזין את קישור הסרטון, שעל גביו הוא מעוניין ביצירת תרגום. לאחר מכן, ילחץ על כפתור ה- "SUBMIT".



4. חלון לניהול ההורדה יפתח, למשתמש יוצג הלינק הנבחר ויהיה ביכולתו להזין אותו למערכת על-ידי לחיצה על כפתור ה-
"DOWNLOAD".



5. המערכת תציג את טעינת הקישור ואת התקדמות ההורדה.



6. בסיום ההורדה, המשתמש יוכל לבחור לייצר את קובץ הוידאו המכיל את התרגום לעברית, ולהורידו למחשבו האישי.

7. למשתמש יוצג חלון ניהול הורדה, שם יבחר את המיקום הרצוי להורדת הקובץ.

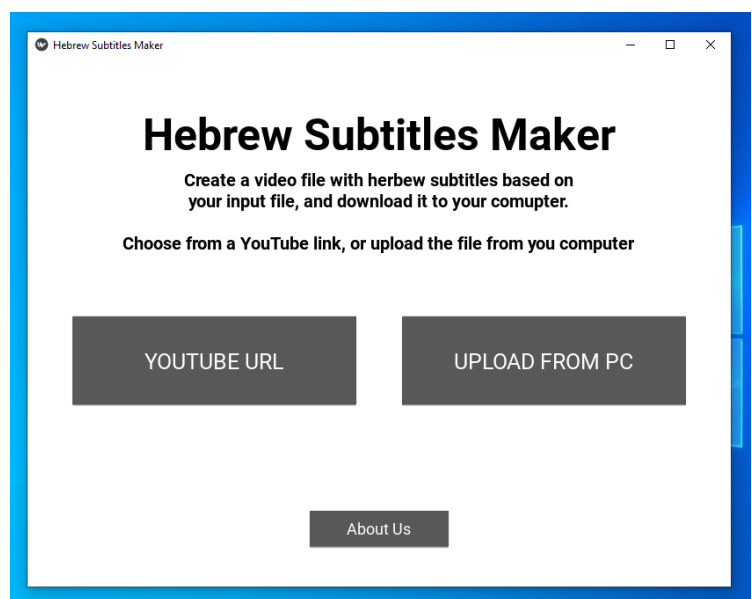
8. בסיום ההורדה, תוצג הודעת הצלחה לסיום ההורדה.

9. המשתמש יוכל לפתוח את קובץ הוידאו המכיל את הכתוביות בעברית

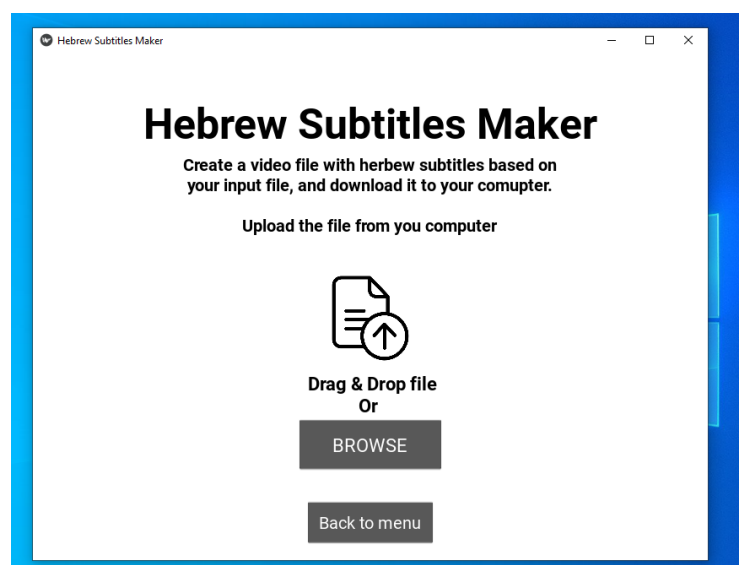
2. העלאת קובץ וידאו מהמחשב

למשתמש יפתח חלון לבחירת קובץ וידאו ממחשבו האישי שעל גבו ירצה להוסיף כתוביות בעברית. לאחר שבחר, המערכת תדע לפרק את קובץ הווידאו לקובץ אודיו ולקובץ וידאו. המערכת תפעיל ממשק speech-to-text על מנת לחלץ מקובץ האודיו את הטקסט, לאחר מכן, המערכת תרכיב קובץ SRT מחתימות זמן שניבנו על ידה. קובץ הטקסט יתורגם על-ידי המודל המאומן, ויוזן לתוך כותב הכתוביות בעברית. המערכת תטמיע את קובץ הכתוביות המתורגם בקובץ הווידאו שנשמר, ותאפשר למשתמש לבצע הורדה של הקובץ הווידאו בתוספת הכתוביות למחשבו של המשתמש. כעת, יוכל המשתמש לצפות בסרטון עם כתוביות בשפה העברית.

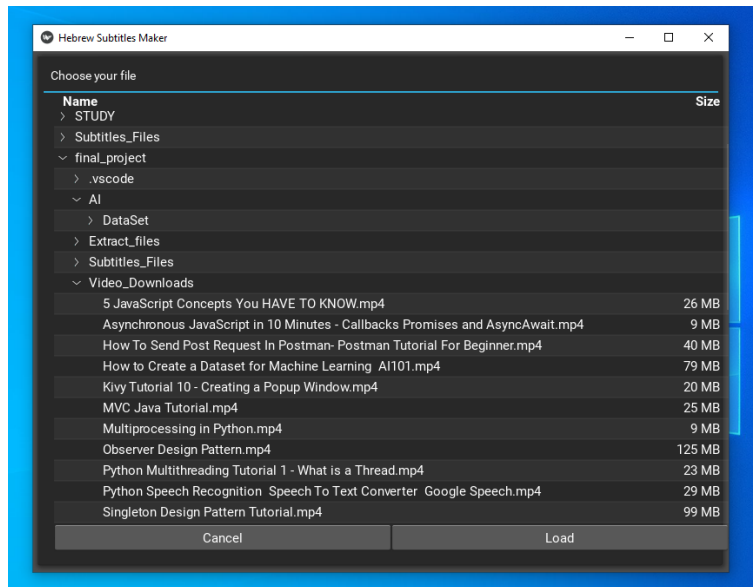
1. המשתמש יכנס ליישום, מסך הראשי יפתח.



2. המשתמש ילחץ על כפתור ה- "UPLOAD FROM PC", ויעבור למסך הבא.



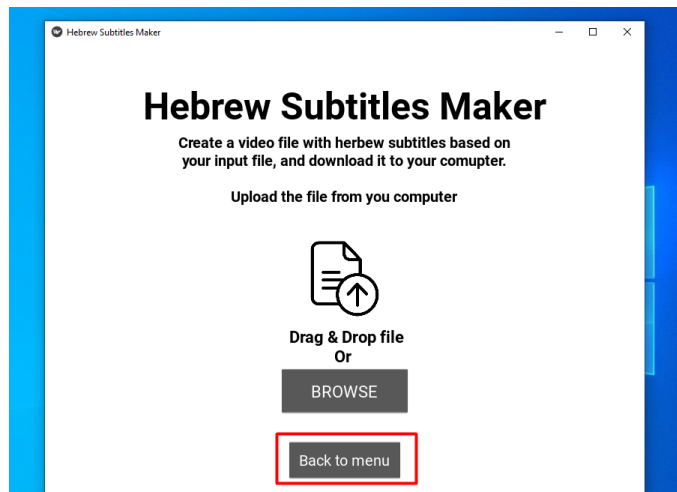
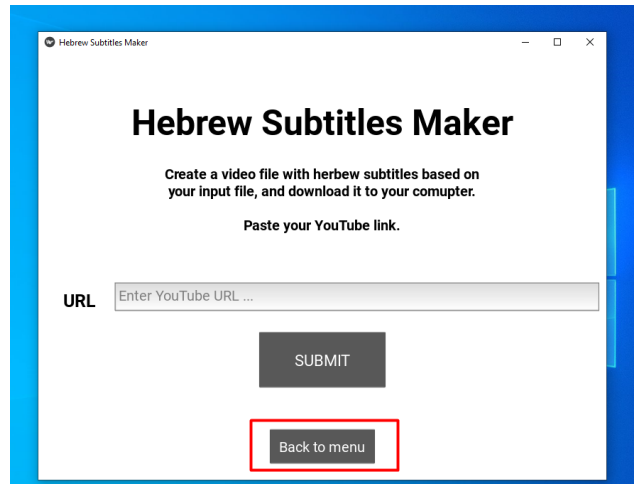
3. המשתמש יוכל לבחור "לזרוק" קובץ למסך, או לבחור את הקובץ הרצוי דרך ניהול חלונות.



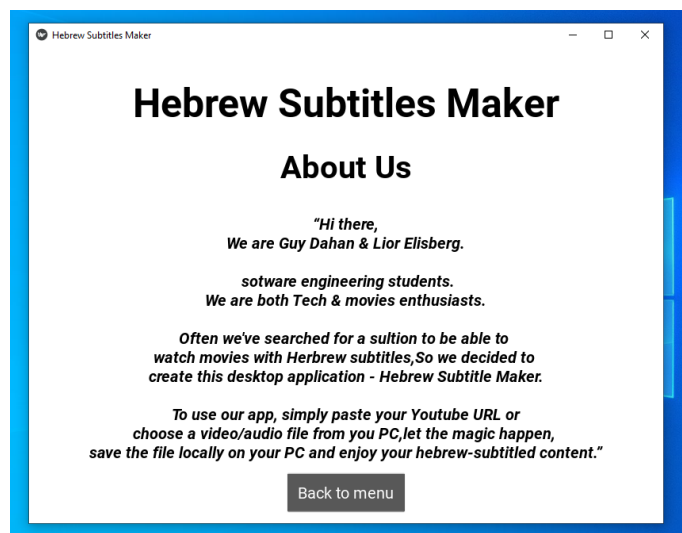
4. המשתמש יבחר את הנתבי הרצוי עבור קובץ הוידאו, ולבחור את הקובץ על-ידי כפתור ה-"LOAD". לאחר מכן יופיע חלון לניהול הורדת קובץ הוידאו הנבחר, כאשר הוא מכיל את התרגום לעברית.
5. למשתמש יוצג חלון ניהול הורדה, שם יבחר את המיקום הרצוי להורדת הקובץ.
6. בסיום ההורדה, תוצג הודעת הצלחה לסיום ההורדה.
7. המשתמש יוכל לפתוח את קובץ הוידאו המכיל את הכתוביות בעברית.

3. פונקציונליות נוספות במערכת

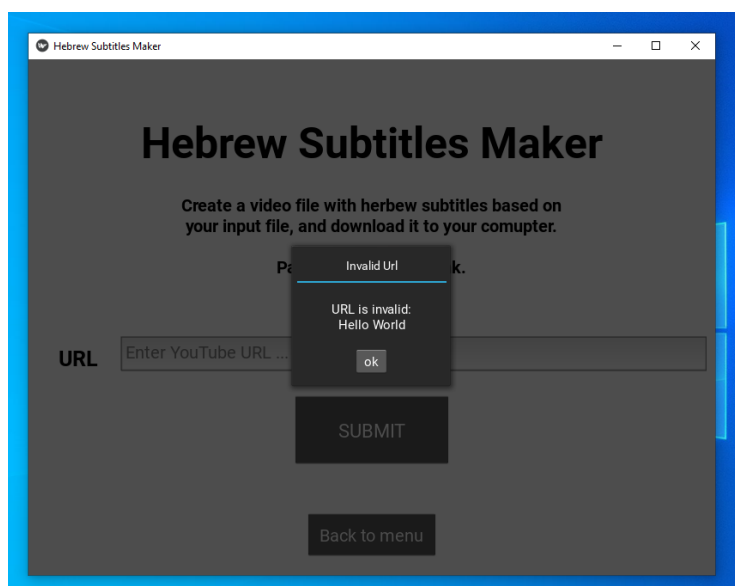
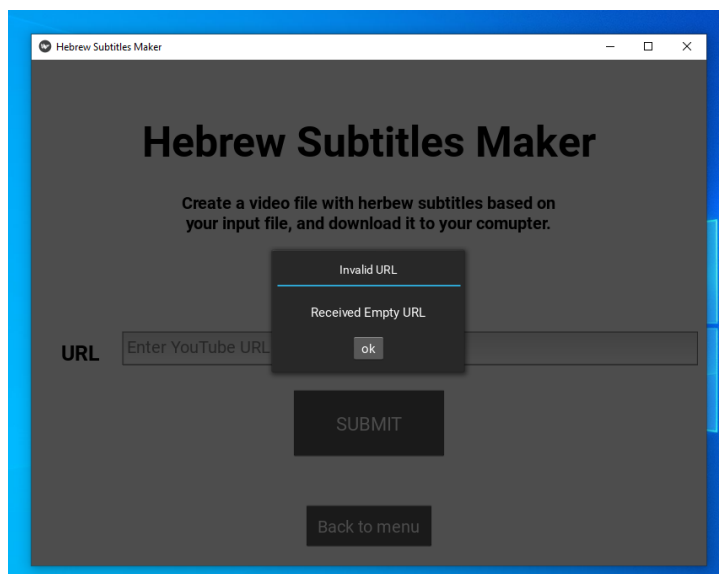
1. מכל חלון פתוח, יוכל המשתמש לחזור לחלון הראשי.



2. המשתמש יוכל לבחור בכפתור ה-"About Us".



3. במהלך השימוש ביישום, למשתמש יופיע חלונות התראה, במידה והתגלתה התנהגות בלתי-רצויה, בגון קישור לא תקין.



System Overview - UML Diagram

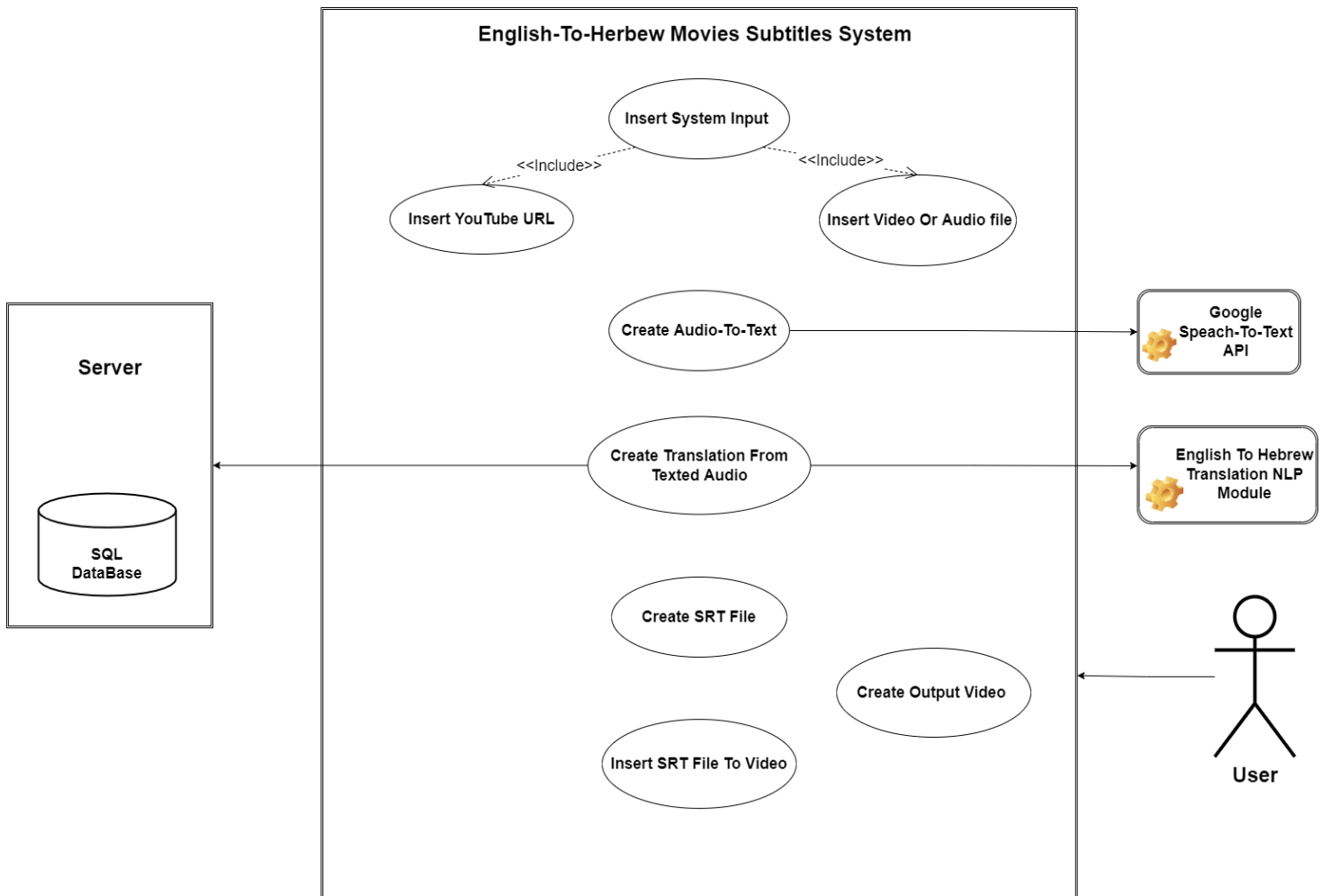


Figure 16 - System Overview - UML diagram

מסמך ניהול סיכונים

גורמי סיכון	רמת פגיעה	הסתברות	ציון משוקלל	דרך פתרון
חוסר ניסיון בפיתוח מערכת מורכבת	5	3	15	למידה עצמאית של המערכות הדרושות לבניית המערכת.
אי-עמידה בזמנים	5	3	15	ביצוע המשימות על-פי לוח זמנים קבוע.
חוסר תאימות בעבודת צוות	5	2	10	הגדרת משימות הן עצמאיות, והן קבוצתיות, תוך מתן דוח התקדמות שבועי בין חברי הצוות
היעדר פילוח הנתונים לקבוצות הומוגניות	5	3	15	יצירת מבנה נתונים עמיד ומגוון, על-מנת למקסם את דיוק המערכת.
איכות נתונים ורמת עדכניות נמוכה	4	3	12	טיוב נתונים קריטיים טרם הפיתוח

קהל יעד

מכונת התרגום מיועדת לכל שכבות האוכלוסייה, לצורכי לימוד, בידור תקשורת וכל העולה על דמיון המשתמש, ככל שהמכונה תגיע לאחוזי תרגום גבוהים יותר ככה השימוש בה יהיה רלוונטי יותר. האפליקציה מיועדת לתעשיית הסרטים לצורך שיטת תרגום חדשנית מהירה יותר וזולה יותר ובמובן גם למשתמש בקצה שיוכל להשתמש באפליקציה לתרגום סרטים, סרטונים וסדרות עם תרגום מיועד לסרטים, שאמור לשים דגש ולהתחשב באורך השונה של מילים בין שפה לשפה ולהתאים את הסלנג, משלב-לשוני והקונטציות התרבותיות בין השפות.

קשיים ואתגרים

1. אתגר מחקרי

האתגרים בתרגום מבוסס למידת מכונה רבים וזאת מכיוון ששפות טבעיות הן קשות מטבען, התהליכים הקוגניטיביים הכרוכים בהבנה וביצירה של משפטים בשפה טבעית מורכבים מאין כמותם ומהוות מכשול עיקרי להתפתחות הבינה המלאכותית בכלל.

2. יצירת המודל

מציאת המודל המתאים היא פעולה מחקר שדורשת זמן רב. ההבנה שלכל אלגוריתם או היוריסטיקה קיימים את החסרונות והיתרונות שלהם, ועלינו למצוא את הכלים שמטיבים עם המידע שלנו והתוצאה שאליה אנו שואפים. בנוסף, על מנת לשפר ביצועים, יש צורך בהוספת ארכיטקטורות קוד שונות כמו הוספת Attention layer למודל, או להשתמש ב, Bidirectional LSTM כלים להערכת ביצועים הרצת הדגימות על גבי המול המשופר ובדיקה שאכן קיים שיפור ולא פגיעה באיכות המודל.

3. אימון המודל

אימון היא פעולה כבדה שדורשת זמן, הרצת מאות-אלפים ועד מיליוני דגימות דורש כוח חישוב וזמן. לכל הרצה יש צורך להמתין מספר רב של ימים, בתקווה שהמודל המאומן יהיה איכותי מספיק, אך הדבר לא קורה בפעם הראשונה ולכן יש צורך במספר רב של הרצות עד קבלת התוצאה הרצויה.

4. יצירת הקורפוס

על מנת ליצור את הדאטה-סט שלנו, נדרשו לאסוף מאות עד אלפי קבצי כתוביות. את הקבצי פירקנו למשפטים ועל גבי אותם משפטים הרצנו את אלגוריתם ההתאמה שבנינו על מנת ליצר מערך נתונים שמכיל משפטים בעברית ותרגומם באנגלית, שגם זה דרש פרק זמן ממושך על מנת לנקות ולנרמל מיליוני שורות לקורפוס.

סיכום

על מנת לשפר את חווית הלקוח, להגיע לקבוצות אוכלוסייה רבות ורחבות יותר, וחיזוק תהליכי גלובליזציה בחברה המודרנית, קיימים צרכים תרבותיים, חברתיים, כלכליים ועסקיים לייעל את תהליך תרגום הסרטים ולצמצם את "עבודת הכפיים" הנדרשת כיום על מנת לתרגם סרט.

כיום, חברות, ארגונים ותאגידים גדולים עומלים על מנת לפתח ולשפר את תרגום המכונה בעזרת מודלים חדשניים בעיבוד השפה הטבעית ולמידת המכונה. תקציבי ענק מושקעים במחקר ופיתוח של מודלים אלו בעיקר בשפה האנגלית, מסיבות ברורות, אך בשפה העברית המחקר נותר מאחור באופן יחסי, ומכיל קשיים רבים כתוצאה מעצם היותה שפה שמית ומורכבת, בעלת רבדים ועומקים רבים. בחרנו לעסוק בנושא זה, מכיוון ששנינו נהנים ממנו רבות, לשלב בין עולם הפיתוח, לבין עולם הקולנוע והסרטים, ואנחנו מרוצים מהתוצר הסופי שהגענו אליו.

במהלך הפרויקט, למדנו רבות על בלשנות ספרותית וחישובית, עיבוד שפה טבעית ומכונות תרגום, כלים מתקדמים בפיתוח מודלים בבינה מלאכותית, בדיקות ו-וולידציה, פיתוח אפליקציית מחשב.

1. דיון בתוצאות

באמצעות אלגוריתם התאמה שכתבנו, הצלחנו ליצור קורפוס אנגלית-עברית חדש לגמרי, שמבוסס על תרגומי סרטים בלבד. פיתוח האלגוריתם דרש מחקר מקדים, בשיטות התאמה sequence-to-sequence ואופטימיזציה של חיפוש התאמות במרחוזות באמצעות Levenshtein Distance.

לאחר מכן, פיתחנו מודל בהמשך למחקר רב שנעשה על מכונות תרגום שקיימות בתחום עיבוד השפה הטבעית, ושיפורים אפשריים שמאוד תלויים בסוגי השפות ובגודלו של הדאטה-סט, כמו מנגנון Attention. נוכחנו להכיר בצורך למצוא מכונה וירטואלית על מנת להריץ גם את אלגוריתם ההתאמה, שמבוסס מכונת תרגום קיימת, וגם את המודל שלנו, שכן זמני לימוד דורשים כוח חישובי, ובעיקר זמן. כל שינוי של hyper-parameter כזה או אחר, דורש הרצה נוספת. עד שלבסוף עברנו להשתמש ב-Google Colab, שמאפשר לבנות קטעי קוד קצרים ולהשתמש בהם מספר פעמים ללא צורך בהרצה של כל התוכנית.

הצלחנו להגיע למודל שעונה על הדרישות הראשוניות שלנו, ומעבר להם, ואפשר לנו לעבור לשלב הבא, שהוא פיתוח ממשק משתמש נוח וקל, ובשאיפה להיות מהיר, נגיש וחינמי.

לכן, בחרנו לסכם את פיתוח הפרויקט ניתן לחלק לשלושה חלקים עיקריים:

- איסוף הדגימות, נרמול המידע על ידי אלגוריתם התאמה שיצרנו בעבור המידע.
- יצירת מודל למידת מכונה איכותי שמאפשר תרגום משפת מקור (אנגלית) לשפת היעד (עברית).
- פיתוח יישום-מחשב קל ונוח לשימוש, שמקבל כקלט קובץ אודיו/וידאו באנגלית, מייצר קובץ כתוביות מתורגם בעברית ומאפשר למשתמש להוריד אל המחשב שלו את הקובץ כאשר הכתוביות מוטמעות על גביו.

2. מחקר עתידי

המערכת שלנו מהווה פתרון לבעיה, שהיא מחסור בתרגום עבור סרטים חדשים שיוצאים, או סרטונים ב-Youtube, שלדוגמה מלמדים למידת מכונה, אך מאוד קשה להבין את המדריך כתוצאה ממבט לא מוכר.

אך, בכל זאת ניתן לשפר את המערכת שלנו לפחות במספר אופנים:

- הגדלת הקורפוס ע"י בניית מערכת שמורידה סרטים וסרטונים מהרשת באופן אוטונומי, ומחלצת את הכתוביות מתוך קטעי הוידאו. ניתן גם לבצע הורדה של קבצי תרגום, שקיימים גם הם במספרים אסטרונומיים ברחבי הרשת.
- ניתן לשפר את המודל על ידי מחקר מעמיק ונוסף, על הארכיטקטורה טובה אפילו יותר לתרגום מאנגלית לעברית, שכן הם שפות שונות במהותן. ניתן לאמן את המודל המשופר עם הדאטה-סט הרחב עוד יותר.
- ניתן לבנות רכיב, ש"מתדרנר" על גבי הדפדפן, ומזהה את מיקום הסרטון בחלום. בלחיצת כפתור, יהיה ניתן "להלביש" חלון חבוי המכיל כתוביות לסרטון.

1. Kimor, G. (2020, December 29). *All you need to know about the subtitled industry.* Tomedes.
2. Alarcon, N., & Alarcon, V. A. P. B. N. (2020, July 15). *Netflix Builds Proof-of-Concept AI Model to Simplify Subtitles for Translation.* NVIDIA Developer Blog.
3. Wu Y, Schuster M, Chen Z, Le Q, Norouzi M. (2016). *Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation.* E-print arXiv:1609.08144.
4. Wintner S. (2004). *Hebrew Computational Linguistics: Past and Future.* Department of Computer Science University of Haifa.
5. Saraf, M. (2020, February 19). *An Introduction to Subtitling.* Women in Localization.
6. Insight, A. (2021, September 9). *Artificial Intelligence in Film Industry is Sophisticating Production.* Analytics Insight.
7. Sreelekha S. (2018). *Statistical Vs Rule Based Machine Translation, A Case Study on Indian Language Perspective.* Dept. of Computer Science & Engineering, Indian Institute of Technology Bombay, India.
8. Babhulgaonkar, A Bharad S. V (2017). *Statistical machine translation.* IEEE Conference Publication | IEEE Xplore
9. Dhariya O, Malviya S, Tiwary U (2017). *A Hybrid Approach for Hindi-English Machine Translation.* IEEE, 16822894
10. Wikipedia contributors. (2021, November 26). *בלשנות חישובית.* Wikipedia.
11. D.Manning, C., Pham, H. and Luong, M.-T. (2015) *Effective approaches to attention-based neural machine translation: Minh-thang luong.*
12. *Hebrew For Beginners* (2021) *Hebrew vs English: 9 main differences, Hebrew for Beginners.* Hebrew For Beginners.
13. Kituku, Benson, Lawrence Muchemi, and Wanjiku Nganga. "A review on machine translation approaches." *Indonesian Journal of Electrical Engineering and Computer Science* 1.1 (2016): 182-190.
14. Bahdanau, D., Cho, K., & Bengio, Y. (n.d.). *Neural machine translation by jointly learning to align and translate.* NASA/ADS.
15. Stahlberg, Felix. "Neural machine translation: A review." *Journal of Artificial Intelligence Research* 69 (2020): 343-418.
16. Insights, C. (2021) *The Levenshtein algorithm, Cuelogic an LTI Company.* Cuelogic Technologies

17. Cho, Kyunghyun, et al. "On the properties of neural machine translation: Encoder-decoder approaches." *arXiv preprint arXiv:1409.1259* (2014).
18. Britz, Denny, et al. "Massive exploration of neural machine translation architectures." *arXiv preprint arXiv:1703.03906* (2017).
19. "House" and "Home", *Ask the Editor, The Britannica Dictionary*.
20. Goodfellow, Ian, Yoshua Bengio, and Aaron Courville. "Sequence modeling: recurrent and recursive nets." *Deep learning* (2016): 367-415.
21. Martínek, J. (2019). Deep Neural Networks for Selected Natural Language Processing Tasks.
22. *Wikipedia contributors. (2021, July 24). מרחק לוינסטיין Wikipedia.*
23. *Reddivarimadhusudhan (2021, Jun 5). Sentence Correction using RNN's (Deep learning). medium*

www.sce.ac.il

קמפוס באר שבע

ביאליק 56, באר שבע 84100

קמפוס אשדוד

ז'בוטינסקי 84, אשדוד 77245

