

הערכה עצמית – מטלה מס' 3 – פיתוח מכונת טיורינג באמצעות סוכני AI פנימיים של CLI

מזהה קבוצה:

30af9fe9-f034-4e99-972f-319b8171fb1d

תעדות זהות:

206591596

205965437

קישור לリフォן:

<https://github.com/Almog16/3997-turing-machine>

הציון העצמי שלי: 94 / 100

♦ **תיעוד פרויקט (PRD + ארכיטקטורה) – 20/20**

- הפרויקט מכיל קובץ [architecture.md](#) ו- [adri](#).
- הפרויקט מכיל קובץ [PRD.md](#) על דרישות הפרויקט.
- המרכז של הפרויקט מכיל הסבר מפורט על התקינה, ובנה הפרויקט. כמו כן הוא מכיל [README.md](#).
- הסבר על [cost models](#).
- תחת [docs](#) תוכל למצוא עוד חומרם רלוונטיים על הפרויקט.
- תחת [assets](#) תוכל למצוא תמונות וסרטוני הפעלה.

♦ **14/15 ותיעוד קוד – README**

- README מקיים ומוכיח עם הוראות הפעלה צעד-אחר-צעד, בדיקות, ניפוי שגיאות וצלומי מסך.
- תמיכה בשני מצבי הריצה, עם הסברים עבור שניהם.
- קוד פיתון מתועדר עם [docstring](#).

♦ **14/15 מבנה פרויקט ואיכות קוד –**

- מבנה הפרויקט פשוט וקל להבנה, במידה הצורך יש תיעוד מפורט על התקינה ושימוש.
- תחת [commands](#) תוכל למצוא את כל המודולים הרלוונטיים עבור [agents](#), כמו כן גם [scripts](#) שנתמכים בעד שימוש ב-[code-claude](#).

♦ **9/10 קונפיגורציה ובטחה –**

- תמיכה בקובץ [env.example](#) ושימוש נכון בקוד.

♦ **בדיקות ואיכות (14/15 – Testing & QA –**

- סקריפט קל לשימוש לבדיקת rates שונים ביחס ל'cosine similarity'. הסקריפט מחשב מרחקים ומיציר גוף השוואתי בין error rates, כך שביציר העיון תוכלו למצוא את cosine similarity וביציר האות error rate.
- הוספה נוספת unit tests עבור test_similarity_and_test_error_sweep עם覆盖率 98%.

♦ מחקר וניתוח – 13/15 ♦

- סקריפט קל לשימוש לבדיקת rates שונים ביחס ל'cosine similarity'. הסקריפט מחשב מרחקים ומיציר גוף השוואתי בין error rates, וכך שביציר העיון תוכלו למצוא את cosine similarity וביציר האות error rate.
- מחברת analysis.py.

♦ משקל משתמש והרחבה (10/10) UI/UX & Extensibility – (10/10) ♦

- תומך בשני ממשקי הרצה –
 - ממשק אינטראקטיבי דרך claude code עם הוראות התקינה ברורות.
 - הרצת של headless option bash ע"י script bash של claude.

♦ נימוק כללי ♦

הפרויקט פותח מטרת מחויבות מלאה לעמידה בכל דרישות דף ההנחיות ובסטנדרטים של פרויקט תעשייתי. לאורך תהליכי העבודה, הושקעו כ-1-2 ימי פיתוח אינטנסיביים שככלו אפיון, תכנון ארכיטקטוני, כתיבת קוד, בניה ובדיקות CL, ניסויים אמפיריים, הפקת גרפים, וכתיבת תיעוד מקיף וברור.

במהלך הפיתוח העמכונו בשימוש ב-Claude Code וביכולות המתקדמיות שלו — סוכנים (Agents), מיומניות (Skills), ופקודות (Commands) — ובנינו תהליכי מלא המדגימים שרשרת תרגום רב-שלביClaude Code המבוססת על סוכנים עצמאיים. בנוסף, פותחו תרחיש הרצה מגוונים: מצב אינטראקטיבי דרך Headless Shell לשילוב אוטומטי, לצד Bash scripts שמאפשרים חווית שימוש מוצעת וგמישה.

העבודה על הפרויקט גם חידדה הבנה עמוקה בעיבוד שפה טבעית, תרגום רב-שלבי, וניתוח מבוסס Embeddings, כולל חישובי Cosine Similarity וניהול ניסויים על רמות שונות של שגיאות כתיב. כל אלו תועדו בצורה סדרה בקבצי README, PRD, ADR וקבצי ארכיטקטורה.

לצד ההצלחה, זוהתה גם נזקודה לשיפור עתידי. שימוש במודל מתקדם ואחד יותר לצורכי חישוב Embeddings עשוי לשפר את עקביות ואיכות הבדיקות.

הצהרת יושר אקדמי (Academic Integrity Declaration)

אני מצהיר/ה בזאת ש:

ההערכה העצמית שלי היא כנה ואמיתית

בדקתי את העבודה מול כל הקритריונים לפני קביעת הציון

אני מודע/ת שציון עצמי גבוה יוביל לבדיקה דקדקנית יותר

אני מקבל/ת את העבודה שהצין הסופי עשוי להיות שונה מהציון העצמי

22.11.25

העבודה היא פרי עבודתי/נו (של הקבוצה) ואני/ו אחראית/ם לכל תוכנה