

1 המהפכה השקטה - מבוא למודלי שפה גדולים בעולם העסקים

תקציר

בעידן שבו טכנולוגיה משנה את עולם העסקים בקצב מסחרר, כמה מהפכה חדשה - השקטה אך עמוקה. מודלי שפה גדולים (Large Language Models, LLMs) מושנים את האופן שבו ארגונים מתקשרים, מנתחים מידע ומקבלים החלטות. פרק זה מציג את היסודות המנהליים להבנת הטכנולוגיה, פוטנציאלה ומוגבלותיה, ומספק כלים מעשיים להערכת התועלת העסקית שלה.

מטרות הלמידה

בסיום פרק זה, הקורא יוכל:

- להבין את המהות והפוטנציאל של מודלי שפה גדולים עבור ארגונים
- להיות את שני התפקידים המרכזיים: אינטראקציה בשפה טבעית ועיבוד לוגיקה מורכבת
- להכיר את נקודות החזק והחולשה של LLMs לצורך קבלת החלטות מושכלות
- לחשב ולהעריך את התשואה על ההשקעה (ROI) בישום כל AI
- להיות תרחיishi שימוש מתאים ובלתי מתאימים למודלים אלו

2 פרולוג: בוקר רגיל בעולם חדש

שבע בבוקר. שרה, מנהלת שיווק בחברת SaaS בינונית, מתיישבת מול המחשב עם כוס קפה. לפניה משימה מוכרת: כתיבת חמישה פוסטים לרשותם חברותיות לקמפיין החדש. בעבר, זה היה לוκח לה שעתיים לפחות. היום, היא פותחת את ChatGPT, מקלידה הначיה קצרה עם ההקשר והסגנון המבוקש, וכעבור דקה וחצי - חמישת הפостиים מוכנים. היא משקיעה עוד עשר דקות בעריכה ועיצוב, ועובדת למשימה הבאה.

באוטו זמו, דן, סמנכ'ל הכספיים, מעלה לממשק Claude דוח כספי בן 05 עמודים וմבקש סיכום של המגוון העיקריות ונקודות החירגה. כעבור שתי דקות, הוא מקבל ניתוח מובנה שב吃过 היה דורש מבקר פיננסי שעיה שלמה. הוא לא מסתפק בכך - הוא ממשיך לשאול שאלות המשך, ו-Claude עונה בהקשר מלא, כאילו הוא עמית שקרה את הדוח בעצמו.

בקומה השלישית, רונית ממשאבי אנווש מעבירה ראיון ראשוני עם מועמד. היא לא לבד - לידיה פועל סוכן AI שמקליט, מתמלל ומנתח בזמן אמת תשובות המועמד מול פרופיל התפקיד. כשהשיחה מסתיימת, רונית כבר רואה דוח מסכם עם המלצה ראשונית.

זהו לא מדע בדיוני. זה לא עתיד רחוק. זה היום, oczywiście, בעשרות אלפי ארגונים ברחבי העולם.

אבל מה באמת קורה כאן? מה הופך את הטכנולוגיה זו לשונה מכל אוטומציה שראינו עד כה? ואיך מנהלים אמורים להבין, להעריך ולהטמייע אותה בארגון שלהם?

3 מהם מודלי שפה גדולים? הסבר אינטואיטיבי

1.3 המטאפורה: מכונת השלמת דפוסים

דמיינו לרוגע יلد שגדל בסביבה שבה הוא שומע מיליון שיחות, קורא מיליארדי משפטים, וחושף למעט כל נושא אנושי אפשרי - היסטוריה, מדע, ספרות, עסקים, פילוסופיה. הילד הזה לא מבין בהכרח את העולם כמו שאנו מבינים אותו, אבל הוא מפתח יכולת מדהימה לזהות דפוסים: איך משפטיים בניוים, איך רעיונות מתקשרים, איך בעיות נפתרות, איך אנשים מתקשרים בהקשרים שונים.

כאתם שואלים את הילד הזה שאלה, הוא לא מփש תשובה במאגר מידע. במקום זאת, הוא משתמש בכל הדפוסים שהוא למד כדי **להשלים** את המשפט היכי הגיוני, היכי סביר, היכי מתאים להקשר. אם שאלתם על אסטרטגיית שיווק, הוא יזכיר מיליון שיחות על שיווק שהוא "שמע", וישלים את התשובה בצורה שימושתית את הדפוסים האלה.

זו, בקצרה, המהות של Large Language Model (LLM).

LLM הוא מודל מתמטי ענק שאמן על כמויות אדירות של טקסט - ספרים, מאמרים, אתרי אינטרנט, קוד תוכנה, ועוד. בתהליך האימון, המודל למד דפוסים סטטיסטיים מורכבים:

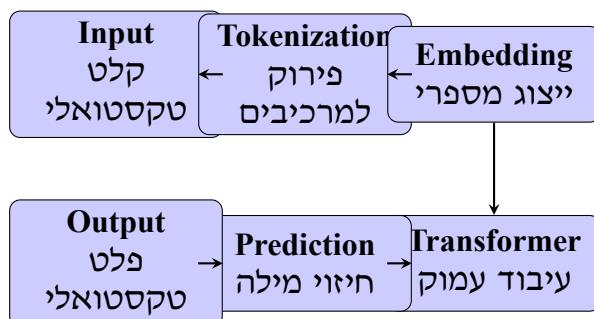
- איך מילים מופיעות לצד אילו מילים
- איך משפטיים בניוים בהקשרים שונים
- איך רעיונות מתקשרים זה לזה
- איך בעיות נפתרות בתחוםים שונים

כאתם כותבים prompt (הנחיה) ל-LLM, המודל "רוואה" את הטקסט שלכם ומשתמש בכל הדפוסים שהוא למד כדי לחזות את השימוש היכי סביר. הוא עושה זאת מילה אחר מילה, תוך התחשבות בכל ההקשר שלפניו.

נקודה קריטית למנהלים: LLM לא "יודע" דברים במובן האנושי. הוא לא מփש מידע בסיס נתונים. הוא יוצר טקסט חדש על בסיס דפוסים סטטיסטיים. זו גם החזקה (יצירתיות, גמישות) וגם החולשה (אפשרות להזיז) שלו.

2.3 מתחת למכסה המנוע: ארכיטקטורה בסיסית

בלי להכנס פרטים טכניים עמוקים מדי, חשוב להבין את המבנה הבסיסי של LLM:



איור 1: ארכיטקטורה בסיסית של LLM - מקלט לפלט

1. **Input (קלט)**: המשמש מאין טקסט - שאלה, בקשה, או הקשר.
 2. **Tokenization (טוקניזציה)**: הטקסט מפורק ל"טוקנים" - יחידות בסיסיות שהמודל מבין. טוקן יכול להיות מילה, חלק ממילה, או סימן פיסוק. למשל, המשפט "שלום עולם" עשוי להיות 2-3 טוקנים.
 3. **Embedding (התמעה)**: כל טוקן הופך לייצוג מתמטי - וקטור של מספרים שמייצג את משמעותו ביחס לטוקנים אחרים.
 4. **Transformer (טראנספורמר)**: זהה הלב של המודל [1]. רשת נוירונים عمוקה שמנתחת את היחסים בין הtokנים, מבינה הקשר, ומחלצת משמעותם. כאן קורה "הקסם" - המודל " מבין" מה שאל וכייך לענות.
 5. **Prediction (חיזוי)**: המודל מנסה את הtokן הבא הכיוון, מוסיף אותו לרצף, וחוזר על התהילה עד שהתשובה שלמה.
 6. **Output (פלט)**: הtokנים והופכים בחזרה לטקסט קריא שהמשתמש רואה.
- למה זה חשוב למנהליס?**
- **טוקנים = עולות**: רוב שירותים ה-API גובים תשלום לפי מספר הtokנים שנשלחים (קלט) ומתקבלים (פלט). הבנת tokנים חיונית לתכנון התקציב.
 - **הקשר מוגבל**: לכל מודל יש "חלון הקשר" (context window) - מספר מקסימלי של tokנים שהוא יכול לעבוד בבאת אחת. למשל, GPT-4 Turbo תומך ב-128,000 tokנים (96,000 מילימ), בעוד GPT-3.5 תומך רק ב-16,000.
 - **מהירות תלולה בגודל**: מודלים גדולים יותר (יותר פרמטרים) בדרך כלל מדויקים יותר, אך איטיים ויקרים יותר.

4 שני הכוחות העד של LLM

1.4 כוח ראשוני: תקשורת טבעית עם מכונה

במשך עשרות שנים, האינטראקציה שלנו עם מחשבים הייתה מוגבלת. רצינו שהמחשב יעשה משהו? היינו צריכים ללמד את שפתו: לחצים, תפריטים, שורותפקודה, שפות תכנות. המחשב לא הבין אותנו - אנחנו היינו צריכים להתאים את עצמנו אליו. LLMs הופכים את המשווה. לראשונה בתולדות המחשב, אנחנו יכולים לתקשר עם מכונה בשפה שלנו, לבדוק כפי שהיינו מדברים עם עמית.

דוגמה מעולם העסקים:

בעבר (אינטראקציה מסורתית עם תוכנה):

1. פתח תוכנת Excel
2. בחר טווח נתונים
3. לחץ על "נתונים" → "סינון" → "סינון מתקדם"
4. הגדר קритריונים מורכבים
5. בחר "עותק למיקום אחר"
6. בחר תא יעד

7. לחץ "אישור"

היום (איןטראקטיב עם LLM):

"תנסן את הטבלה הזו ותראה לי רק לckoחות מאזור המרכז שרכשו מעל 10,000 ש"ח בשבועון האחרון"

המודל מבין את הכוונה, מזהה את הנתונים הרלוונטיים, ומבצע את הפעולה - או אפילו כותב לכם את הנוסחה המתאימה.

ההשלכות העסקיות:

- **הפחתת מחסום הבנייה:** עובדים לא צריכים להיות מומחי תוכנה כדי לבצע משימות מורכבות.
- **מהירות:** מה שלוקח 10 דקות במשק מסורתי, לוקח 10 שניות בשפה טבעית.
- **גמישות:** אפשר לשאול שאלות המשך, לשנות דרישות, לחקור כיוונים שונים - בדיקות כמו בשיחה אנושית.

2.4 כוח שני: עיבוד לוגיקה מורכבת

אבל LLMs הם הרבה יותר מסתם משק נוח. הם מסוגלים לבצע **חשיבה מורכבת** על נתונים וריעונות.

בואו נבחן כמה יכולות מרכזיות:

2.4.1 סיכום והפקת תובנות

ניתן להזין ל-LLM דוח של 100 עמודים ולבקש:

- "סיכום את הנקודות העיקריות ב-5 משפטים"
- "מה הטרנדים המרכזיים שמופיעים בה?"
- "איזה נושאים חווירים על עצם?"

המודל קורא, מזהה דפוסים, ומפיק תובנות - עבודה שעדי לפני כמה שנים דרצה אנליסט אנושי.

2.2.4 השוואת ניתות

"השוואה בין שלוש הוצאות המחיר האלה מבחינת עלות, זמן אספקה ושירות, והמלץ על הספק המתאים ביותר לארגון שלנו".

המודל לא רק משווה - הוא **מנמק** את המליצה שלו על בסיס הקритריונים שהגדתם.

3.2.4 פתרון בעיות רב-שלבי

LLMs מודרניים יכולים לפתור בעיות שדורשות מספר שלבי חשיבה:

1. הבנת הבעיה
2. פירוק לחת-בעיות
3. פתרון כל חת-בעיה
4. שילוב התוצאות לפתרון כולל

דוגמיה:

"יש לנו 5 נציגי מכירות, 120 לינדים חדשים החודש, וכל נציג יכול לטפל בממוצע ב-25 לינדים בחודש. איך כדאי לחלק את הלינדים בהתחשב בכך שנציג א' מתמחה בלקוחות ארגוניים, ב' ו-ג' בעסקים קטנים, ד' בסטארטאפים, וה' חדש ועודין מתאמן?"

המודל יבנה תוכנית חלוקה מפורטת, יסביר את ההיגיון מאחורי, ואפיו יתריע על בעיות פוטנציאליות.

4.2.4 תכנות וכתיבת קוד

כמו GPT-4 [2] ו-Claude [3] מסוגלים לכתוב קוד תוכנה באיכות גבוהה. מנהל בלי רקו תכונתי יכול לבקש:

"תכתב סקריפט Python שקוראקובץ Excel, מחשב את סכום המכירות לכל מוצר, ויוצר גרף עמודות"

המודל יכתב קוד מלא, מתווד, ומוכן להרצה.

השלבה עסקית: זה מורייד את המחסום לאוטומציה. שימושות שב עבר דרשו מתכנת, היום יכולות להתבצע על ידי כל מנהל עם רעיון ברור.

3.4 המשמעות המשולבת: עובד דיגיטלי

כשהשלבים את שני הכוחות האלה - תקשורת טבעית ועיבוד לוגיקה מורכבת - מקבלים משחו חדש לחלוtin בעולם העסקיים: **עובד דיגיטלי** שאפשר להדריך, לשאול שאלות, לתקון, ולשפר - בדיקות כמו עובד אנושי.

זה לא עוד כלי שמבצע משימה אחת קבועה. זה ישות דיגיטלית שיכולה:

- להבין הוראות מורכבות
- להתאים את עצמה למצבים שונים
- לשאול שאלות הברהה
- ללמידה מדוגמאות שאתם נתונים
- להציג שיפורים

זו המהפהכה האמיתית של LLMs.

5 נקודות החזק: מה LLMs עושים טוב במיוחד

1.5 יצירתיות והפקת רעיונות

LLMs מצטיינים ביצירת תוכן חדש ו מגוון:

- **תוכן שיווקי:** מודעות, פוסטים, דפי נחיתה, מיילים
- **תוכן טכני:** מסמכי דרישות, הצעות מחיר, תיעוד
- **רעיונות:** סיור מוחות אוטומטי, זויות חדשות לביעות קיימות

דוגמה מעשית:

מנהל שיווק בחברת SaaS B2B צrica 10 כוורות שונות לкомפין LinkedIn Ads שמקדם כל ניהול פרויקטים לצוותי פיתוח.

היא כתבת ל-ChatGPT:

"צור 10 כוורות לкомפין לנקדאין למוצר ניהול פרויקטים לצוותי פיתוח תוכנה.

המוצר חוסך 20% זמן הנהלה ומשפר שיתוף פעולה. קהל יעד: VP R&D וראשי צוותים. טוון: מڪצועי אך לא יבש, דגש על תוצאות עסקיות."

תוך שניות, היא מקבלת 10 אפשרויות, כל אחת בזווית שונה - ROI, פריוון, איכות חיים, תחרותיות. היא יכולה לבחור, לשלב, או לבקש עוד וריאציות.

זמן שנחץ: שעה של סיעור מוחות ומחשבה → 5 דקות.

2.5 הבנת הקשר ונוונים

בניגוד לכלי אוטומציה מסורתיים שפועלים לפי כללי קשיחים, LLMs מבינים הקשר:

- הם "זוכרים" את כל השיחה ומתייחסים אליה
- הם מבינים כוונות, לא רק מילים
- הם מתאימים את התשובה לטון ולסגנון המבוקש

דוגמה מתמיכת לקוחות:

לקוח כותב: "המערכת שוב קרסה לי באמצעות פעולה חשובה"

Chatbot מסורתי (מבוסס כלליים) יזהה את המילה "קרים" ויחזר:

"אני נסה לأتחל את המערכת ולנקות את הזיכרון הזמן."

טוכן LLM מבין את הקשר הרגשי והעסקני:

"אני מבין את הטעול - זה בוודאי מעצבן כשהזה קורה בעיצומה של עבודה חשובה. בואו נפתרו את זה מהר. האם הבעיה קרתה בפעולה ספציפית? זה יעוז לי לאבחן את הגורם ולודא שהזה לא יקרה שוב."

ההבדל? אמפתיה, הקשר עסקי, וכוונה לפתרון ארוך טווח - לא רק טיפול תסמייני.

3.5 גמישות והתאמאה

LLMs לא דורשים תכנות מראש לכל תרחיש. אפשר "ללמד" אותם "בדרכך":

- **Few-Shot Learning** [4]: תן כמה דוגמאות, והמודל יבין את הדפוס
- **התאמאה לסגנון:** "כתב בסגנון פורמלי/חברי/טכני"
- **שינויי ביוון באמצעות:** "לא, תנסה את הגישה ל..."

דוגמה - קטגוריזציה של פניות:

חברה מקבלת מאות פניות ביום לתמיכה לקוחות. היא רוצה לקטגורизות אותן אוטומטית ל-5 קטגוריות.

גישה מסורתית: שכירת מתכנת שיבנה מודל ML מותאם אישית, יאוסף נתונים אימון, ויקח שבועיים.

גישה LLM:

"קטגר את הפניות הבאות לאחת מחמש הקטגוריות: טכני, חיוב, שאלת מכירה, תלונה, בקשה פיצ'ר. הנה שלוש דוגמאות:

[דוגמה 1...] [דוגמה 2...] [דוגמה 3...]

עכשו, קטגר את הפניות האלה: [רשימת פניות...]"

המודל יבין את הדפוס מהדוגמאות ויקטרג נכון 59-60% מהפניות.

זמן "ישום": שבועיים → 30 דקות.

4.5 עבודה עם שפות מרובות

LLMs מודרניים כמו GPT-4, Claude, Gemini ודברים عشرות שפות בצורה שוטפת. זה פותח אפשרויות:

- תרגום אוטומטי איכותי (מעבר לתרגום מילה במילה - הבנת הקשר תרבותי)
- תמיכת לקוחות רב-לשונית בלי צוות עצום
- יצירת תוכן בשפות מרובות באופן מיידי

דוגמה: חברת ישראלית שמוכרת לאירופה צריכה לתרגם מסמך טכני מעברית לגרמנית, צרפתית, וספרדית. במקום שלושה מתרגמים מקצועיים ומספר ימים, Claude מתרגם את שלושת הגרסאות תוך דקות, עם שמירה על טרמינולוגיה טכנית עקבית.

6 נקודות החולשה: מה LLMs לא עושים טוב

1.6 היזות (Hallucinations)

זהו אוליה החולשה הクリיטית ביותר של LLMs: הנטייה "להמציא" מידע [5]. זכרו - LLM הוא מכונת השלמת דפוסים. הוא לא מחפש מידע בסיס נתונים; הוא מחזה את המשך הסביר ביותר. לעיתים, אם המידע הנוכחי לא קיים בזיכרון הסטטיסטי שלו, המודל ייצור עובדות שנשמעות מהימנות - אבל שגיאות לחלוין.

דוגמה מסוכנת:

עורך דין ביקש מ-ChatGPT לספק תקדים משפטיים לתמיכה בתביעה. המודל מסר רשימה של שישה תקדים, כולל שמות תיקים, מספרי תיק, ותאריךיכים. הכל נראה לגיטימי. הבעה? **אף אחד מהתקדים לא היה אמיתי.** ChatGPT המציא אותם, כי הם נשמעו הגיוניים בהקשר.

המשפט הסתויים בסנקציות חמורות על עורך הדין.

למה זה קורה?

LLM נועד "להישמע" מהימן ורהור. אין לו מגנון פנימי שאומר "אני לא יודע". במקרה זאת, הוא ממשיך לייצר את המשך הכי סביר, גם אם זה לא מבוסס עובדות.

השלכות עסקיות:

- אין לסמוך על LLM לעובדות ללא אימומות. תמיד יש לבדוק מיידע קריטי.
- מתאים ליצירת רעיונות וטיוטות ראשוניות, פחות מתאימים לדוחות עובדתיים סופיים.
- חובה לשלב בקרה אנוית במערכות קריטיות.

2.6 חוסר עדכניות (Knowledge Cutoff)

כל LLM נאמן עד תאריך מסוים - ה"knowledge cutoff" שלו. למשל:

- GPT-4 (גרסה מ-4202): נתונים עד אפריל 2023

- Claude 3.5 Sonnet: נתונים עד אפריל 2024

המשמעות: המודל לא יודע כלום על אירועים, מוצרים, טכנולוגיות, או שינויים שהתרחשו אחרי התאריך הזה.

דוגמה:

GPT-4 שואל את OFC:

"מה שער הדולר מול השקל היום?"

התשובה תהיה מבוססת על נתונים ישנים, או החלופין - זהה.

פתרונות:

- שימוש במודלים עם גישה לאינטרנט (כמו ChatGPT Plus עם browsing mode)
- שילוב (RAG) (Retrieval-Augmented Generation) - הזרקת מידע עדכני למודול
- שימוש ב-APIs חיצוניים שהסוכן יכול לקרוא להם

הנפקות עסקיות: בתחוםים דינמיים (פיננסים, חדשות, נתונים תפעוליים), אין להסתמך על LLM לבדו. יש לספק לו מידע עדכני או לשלב אותו עם מקורות מידע חיים.

3.6 עלויות - לא זניח

שימוש ב-LLMs דרך API עולה כסף, וההוצאות יכולה להפצע ארגונים שלא תכנו לכך.

מודלים מתומחרים לפי טוקנים:

Model	Input (\$/1M tokens)	Output (\$/1M tokens)
GPT-4 Turbo	10.00	30.00
GPT-3.5 Turbo	0.50	1.50
Claude 3.5 Sonnet	3.00	15.00
Claude 3 Haiku	0.25	1.25
Gemini 1.5 Pro	1.25	5.00

טבלה 1: מחירי מודלים נפוצים (נכון מרץ 2024)

דוגמה חישוב:

נניח שחברה משתמשת ב-GPT-4 Turbo לתמיכה לקוחות. כל שיחה:

- קלט ממוצע: 2,000 טוקנים (הקשר + שאלת הלקוח)
- פلت ממוצע: 800 טוקנים (תשובה)

עלות לשיחה:

$$\begin{aligned} \text{Cost} &= \left(\frac{2,000}{1,000,000} \times 10 \right) + \left(\frac{800}{1,000,000} \times 30 \right) \\ &= 0.02 + 0.024 \\ &= \$0.044 \end{aligned} \quad (1)$$

זה נשמע זניח, נכון? אבל אם יש 10,000 שיחות בחודש:

$$(2) \quad \text{Monthly Cost} = 10,000 \times 0.044 = \$440$$

ואם זה גדול ל-100,000 שיחות בחודש (חברה גדולה):

$$(3) \quad \text{Monthly Cost} = 100,000 \times 0.044 = \$4,400/\text{month} = \$52,800/\text{year}$$

אסטרטגיות חיסכון:

- שימוש במודלים זולים יותר למשימות פשוטות (GPT-3.5 GPT-4 במקומם)
- אופטימיזציה של prompts להיות קצרים ומוקדים
- Caching - שמירת תשובות לשאלות נפוצות
- שימוש במודלים self-hosted (למשל Llama 3) לנפחים גדולים

4.6 חוסר שקיפות (Black Box)

LLMs הם "קופסה שחורה". כשהם נותנים תשובה, אי אפשר לדעת במדויק **למה** הם הגיעו למסקנה זו. אין "שרשרת הוכחה".

בעיה עסקית:

- **רגולציה:** בתחוםים מוסדרים (בנקאות, בריאות), לעיתים נדרש להסביר החלטות. "כפי AI אמר" לא מספיק.
- **אמון:** מנהלים מתקשים לסמוך על מערכת שלא יכולה להסביר את ההיגיון שלה.
- **Bias:** קשה לזהות ולתקן הטיות כשלא רואים את תהליכי החשיבה.

פתרונות חלקיים:

- שימוש ב-chain-of-thought prompting [7] - לבקש מהמודל להסביר את הצעדים שלו
- שילוב מערכות explainable AI משלימות
- בקרה אנושית במקרים קרייטיים

7 נסחאות מנהליות להערכת LLMs

1.7 ממד ROI של יישום AI

לפני השקעה בכלים AI, חשוב לחשב את התשואה הצפוייה על ההשקעה [8], [9].

נוסחת ROI בסיסית:

$$(4) \quad ROI = \frac{\text{הוצאות מנוי AI} - (\text{הוצאות שעת עבודה} \times \text{שעות נחסכות})}{\text{הוצאות מנוי AI}} \times 100\%$$

דוגמה מעשית:

וצאות תמיית ל��וחות בן 5 אנשים משתמש ב-Claude לטיפול בפניות שגרתיות.

נתונים:

- כל נציג מטפל ב-30 פניות ביום
- לפני AI: זמן ממוצע לפניה = 15 דקות
- עם AI (סיווע בכתביה, חיפוש מידע, תבניות): זמן ממוצע = 10 דקות
- חיסכון לפניה: 5 דקות
- ימי עבודה בחודש: 22
- עלות שעת עבודה ממוצעת: 100 ש"ח
- עלות מנוי Claude Pro לכל נציג: \$20/\$חודש = 75 ש"ח

чисוב:

שעות נחסכות בחודש:

$$\begin{aligned} & \text{ימיים} \times \text{שעות} \times \frac{5}{60} \times \text{פניות/יום} \times \text{נציגים} = \text{שעות נחסכות} \\ & = 5 \times 30 \times 0.0833 \times 22 \\ & = 275 \text{ שעות} \end{aligned} \tag{5}$$

ערך החיסכון:

$$(6) \quad \text{ש"ח} 275 \times 100 = 27,500 = \text{ערך}$$

הוצאות מנוי:

$$(7) \quad \text{ש"ח} 75 \times 5 = 375 = \text{הוצאות}$$

$$(8) \quad ROI = \frac{27,500 - 375}{375} \times 100\% = 7,233\% \quad : ROI$$

משמעות: על כל שקל שהושקע ב-AI, החברה חוסכת 72 שקלים. זו תשואה עצומה.

נוסחה מורחבת (כוללת עלויות נוספות):

$$(9) \quad ROI = \frac{\text{הוצאות כוללת} - \text{רווחת כוללת}}{\text{הוצאות כוללת}} \times 100\%$$

כאשר:

- **רווחת כוללת** = חיסכון בשעות + הגדלת מכירות + שיפור שביעות רצון (בערך כ-7%)
- **הוצאות כוללת** = מנויים + הטמעה + הדרכה + תחזוקה

2.7 נוסחת עלות טוקנים

להבנת העלות התפעולית של שימוש ב-API:

$$(10) \quad \text{עלות שיחה} = \left(\frac{\text{טוקני קלט}}{1,000,000} \times \text{מחיר קלט} \right) + \left(\frac{\text{טוקני פלט}}{1,000,000} \times \text{מחיר פלט} \right)$$

דוגמה:

סוכן AI לנתח משוב ל��וחות משתמש ב-GPT-4 Turbo :

- מחיר קלט: \$10 למיליאון טוקנים
- מחיר פלט: \$30 למיליאון טוקנים
- כל ניתוח כולל: 3,000 טוקני קלט + 1,200 טוקני פלט
- נפח: 5,000 ניתוחים בחודש

עלות לנתח בודד:

$$\begin{aligned} \text{Cost}_{\text{single}} &= \left(\frac{3,000}{1,000,000} \times 10 \right) + \left(\frac{1,200}{1,000,000} \times 30 \right) \\ &= 0.03 + 0.036 \\ &= \$0.066 \end{aligned} \tag{11}$$

עלות חודשית:

$$(12) \quad \text{Cost}_{\text{monthly}} = 5,000 \times 0.066 = \$330$$

שימוש מעשי: נוסחה זו מאפשרת למנהלים לחזות עלויות על בסיס נפח העסקאות הצפוי ולהחליט בין מודלים שונים.

Break-Even Analysis 3.7

מתי כדאי לעבור משירותים cloud לפתרון self-hosted ?

$$(13) \quad \frac{\text{עלות הקמה חד-פעמיות}}{\text{חיסכון חודשי}} = \text{נקודת איזון (חודשיים)}$$

דוגמה:

חברה משתמשת ב-API GPT-4 ומשלמת \$5,000/חודש. היא שוקלת להקים שרת פנימי עם Llama 3 70B.

עלויות self-hosted :

- שרת + GPU : \$15,000
- הקמה ותוכנה: \$5,000
- עלות חודשית (חישול, תחזקה, כ"א): \$2,000

חיסכון חודשי:

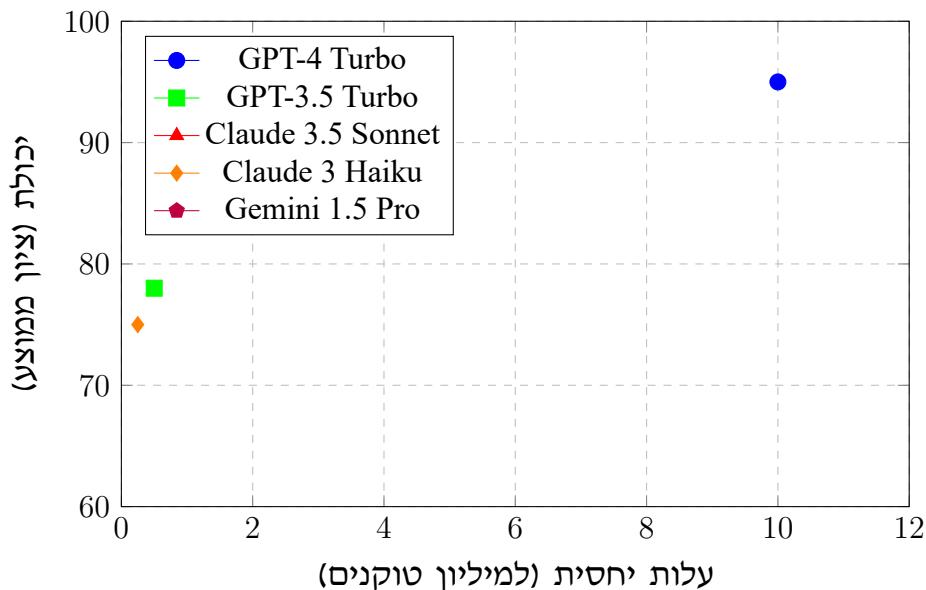
$$(14) \quad \text{חיסכון} = 5,000 - 2,000 = \$3,000$$

נקודת אייזון:

$$(15) \quad \text{Break-even} = \frac{15,000 + 5,000}{3,000} = 6.67 \text{ חודשים}$$

משמעות: תוך כ-7 חודשים ההשקעה תשתלם, ומשם והלאה החברה תחשוך \$3,000/חודש.

8 השוואת מודלים: תרשימים עלות מול יכולות



איור 2: השוואת מודלים מוביילים - עלות מול יכולות

תרשים 2 מציג את הטריד-אוף בין עלות ליכולות:

- **GPT-4 Turbo:** היקר והמסוגל ביותר - מתאים למשימות קריטיות ומורכבות
- **Claude 3.5 Sonnet:** איזון מצוין - 29% מהיכולת ב-30% מהמחיר
- **Gemini 1.5 Pro:** חלופה חזקה במחיר נוח
- **GPT-3.5 Haiku:** למשימות פשוטות בנפח גביה

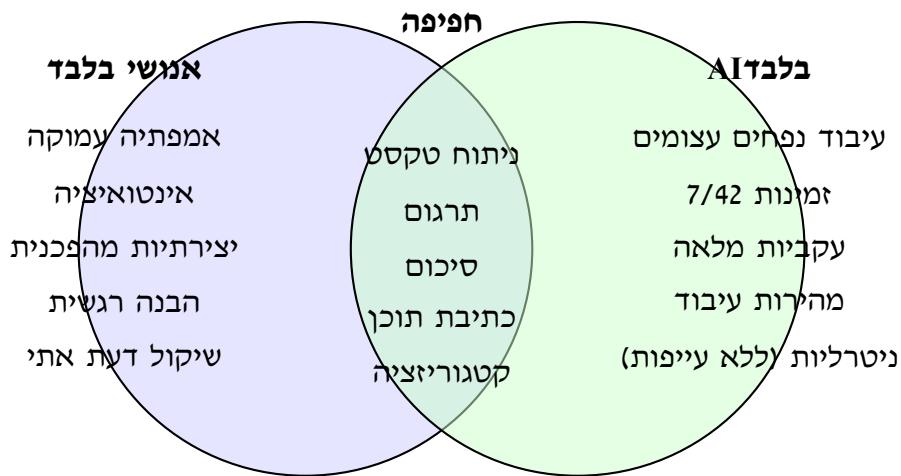
סטרטגיה מומלצת: שימוש במודלים שונים לצרכים שונים. משימות פשוטות (סיכום, עריכה) במודלים זולים; משימות מורכבות (ניתוח, החלטות) במודלים מתקדמים.

9 תרשים Venn: חפיפה בין יכולות אנושיות ו-AI

תרשים 3 ממחיש היכן כדאי להשתמש ב-AI, והיכן האדם עדין בלתי תחלפי:

אזור החפיפה - משימות שהן AI יכול לסייע או אפילו להחליף אדם:

- עיבוד ואנליזת טקסט
- תרגום והתאמה לשונית



איור 3: חפיפה בין יכולות אנושיות ויכולות AI

- סיכום מסמכים ארוכים
- כתיבת תוכן סטנדרטי
- מילון וקטגוריזציה

אזור אנושי בלבד - שימושים שבהן האדם עדין חיוני:

- החלטות אסטרטגיות מורכבות
- מצלבים הדורשים אמפתיה אמיתי
- יצירתיות פורצת דרך (לא הרכבה של דפוסים)
- שיקול דעת אישי ומוסרי
- הבנה عمוקה של הקשר ארגוני ותרבותי

אזור AI בלבד - שימושים שבהן AI עדיף על אדם:

- שימוש נACHI מיידע עצומים
- זמינות מתמדת ללא הפסקה
- עקיבות מוחלטת ביצוע
- מהירות ותגובה מיידית
- ניטרליות (אין השפעת עיביות, רגשות חולפים)

המסקנה המנהלית: הגישה האופטימלית היא **שיתוף פעולה**, לא תחרות. AI כעזר שמשחרר את האדם ממשימות שגרתיות, ומאפשר לו להתמקד ב יתרונות הייחודיים שלו.

10 דוגמאות מעשיות: LLMs בעבודה

1.01 ניהול שיווק: תכנון ויצירת קמפיין

תרחיש:

רינה, מנהלת שיווק בחברת SaaS, צריכה להשיק קמפיין ל מוצר חדש. לפני:

- בניית אסטרטגיה

- כתיבת תוכן לעורוצים שונים
- יצירת וריאציות ל מבחני A/B

השימוש ב-ChatGPT:

שלב 1 - סיעור מוחות אסטרטגי:

"אני משיקה מוצר SaaS לניהול פרויקטים לצוותי שיווק. קהל יעד: מנהלי שיווק B2B. הערך המרכזי: חיסכון של 10 שעות שבועיות אוטומטית. תן לי 5 זויות שונות לקמפיין."

מציע: ChatGPT

1. "עשה יותר עם פחות" - דגש על עילות
2. "זמן לאסטרטגיה" - שחרור מדיניות לחשיבה
3. "צוות קטן, תוכאות גדולות" - יתרון לסטארטאפים
4. "ROI ברור מיום ראשון"
5. "כל הכלים במקום אחד" - פשוטות

שלב 2 - פיתוח זווית נבחרת:

רינה בחרה בזווית 2. היא ממשיכה:

"תפתח את זווית 'זמן לאסטרטגיה'. כתוב לי 3 כותרות ל-LinkedIn Ads, פוסטים קצרים, ודף נחיתה בסגנון חברי-מקצועי."

תוך דקה וחצי, רינה מקבלת תוכן מלא. היא משקיעה עוד 20 דקות בעריכה ובהתאמה למוגן.

שלב 3 - וריאציות ל-A/B Testing:

"צורך עוד 5 וריאציות של הכותרת הראשונה - אחת עם מספרים, אחת עם שאלה, אחת עם אמוג', אחת קצרה מאוד, ואחת ארוכה ומפורטת."

תוצאה:

- זמן שנחסך: 4-5 שעות של כתיבה וחשיבה
- איכות: טובה מאוד (לאחר עריכה קלה)
- יצירתיות: זויות שרינה לא חשבה עליהם

2.01 סמנכ"ל כספי: ניתוח דוחות מורכבים

תרחיש:

יוסי, CFO של חברת ציבורית, מקבל דוח רבוני של 80עמודים מהחברת ביקורת. הוא צריד:

- להבין מגמות עיקריות

- להזות חריגות
- להכין סיכום להנלה

השימוש ב-Claude (יכולת הulat קבצים):
יוסי מעלה את ה-PDF ל-Claude וושאל:

"נתח את הדוח הזה וספק: 1. המגמות החיוביות העיקריות 2. 3 נקודות הדאגה העיקריות 3. חריגות משמעותיות מול רביעון קודם 4. המלצות ראשוניות"

Claude קורא את כל הדוח (למרות אורכו) ותוך 2 דקות מספק:
מגמות חיוביות:

1. צמיחה של 21% בהכנסות חוזרות (ARR)
2. שיפור של 5% ב-Gross Margin
3. ירידה של 81% בעליות רכישת לקוח (CAC)

נקודות דאגה:

1. עלייה של 22% ב-churn rate בסגמנט BMS
2. הארכת מחזור המכירה מ-45 ל-63 ימים
3. עלייה בהוצאות R&D ללא השקעה מקבילה

חריגות:

- סעיף "הוצאות שונות" קופץ פי 3 - דורש>b>הבהרה
- נכס בלתי מוחשי חדש בסכום חריג

המלצות:

- לחקור את עליית churn - אפשרי שינוי בתחרות או בטיב השירות
- לברר את הארכת מחזור המכירה - האם בעיה במכירות או בכלכלת הכללית
- לדרש פירוט על "הוצאות שונות"

יוסי עובר על הניתוח, משווה למסמכים נוספים, ומעמיק בנקודות הקritisיות. אבל הסיכון הראשוני חסך לו שעה וחצי של קריאה צפופה.

שימוש נוספים - שאלות המשך:
יוסי ממשיך לשאול:

"על בסיס הדוח, האם נוכל לעמוד ביעדי ה-EBITDA לסוף השנה?"

Claude מנתח את הנתונים ומספק תשובה מנומקת עם הנתונות.

3.01 ניהול משאבי אנוש: סינון קורות חיים

תרחיש:

מיכל, מנהלת גיוס, קיבלה 150 קורות חיים לתפקיד Product Manager. דרישות התפקיד:

- ניסיון של 3-5 שנים בניהול מוצר B2B SaaS
- רקע טכני - יתרונותomi שעובד כمفחת
- ניסיון בעבודה עם צוותים מבזוריים
- אנגלית שוטפת

גישה מסורתית: מיכל הייתה צריכה לקרוא כל קו"ח ידנית - זמן: 6-5 שעות.

גישה AI:

מיכל כתבת סקריפט Python פשוט (בעזרת ChatGPT) שקורא את כל הקבצים ושולח כל קו"ח ל-GPT-4 עם ה-prompt:

"דרג קורות חיים אלה לתפקיד reganaM tcudorP B2B SaaS. דרישות: - 3-5 שנים ניסיון PM - רקע טכני (יתרונות) - עבודה עם צוותים מבזוריים - אנגלית שוטפת

דרוג: TIF ON / EBYAM / TIF DOOG / TIF GNORTS

המערכת מעבדת את 150 הקורות חיים תוך 10 דקות ומחזירה:

- 12 מועדים STRONG FIT
- 28 מועדים GOOD FIT
- 45 מועדים MAYBE
- 65 מועדים NO FIT

מיכל קוראת רק את 12 ה-STRONG FIT ואת 28 ה-GOOD FIT - סך הכל 40 קורות חיים - זמן: שעה.

תוצאה: חישוכן של 4-5 שעות, תוך שמירה על איזות הסינון.

הערה חשובה: מיכל לא סומכת באופן עיוור על ה-AI. היא עדין קוראת בעצמה את הקורות חיים הרלוונטיים. ה-AI משמש ככלי סינוון ראשון, לא כקובע סופי.

11 תרגילים

1.11 תרגיליםシアורתיים

תרגיל 1.11. חישוב ROI של הטמעת כל AI

הנץ מנהל/ת מחלקת תוכן בחברת e-commerce. הוצאות שלך (8 כתובים) מייצר 40 מאמרים בחודש. כל מאמר לוקח כיום 4 שעות עבודה.

אתה שוקל להטמע ChatGPT Plus (\$20/חודש לכל כתוב) שลดעתך יקצר את הזמן ל-2.5 שעות למאמר.

נתונים:

- עלות שעת עבודה ממוצעת: 120 ש"ח
- ימי עבודה בחודש: 22

שאלות:

- א) חשב את ה-ROI החודשי
- ב) כמה זמן יעבור עד שה השקעה תשתלם (בהנחה שיש גם עלות הדרכה חד-פעמית של 5,000 ש"ח)?
- ג) האם תמליץ על ההטמעה? נמק.

תרגיל 11.2. זיהוי MERCHANTABILITYS מתאיימות ובלתי מתאיימות ל-LLM

בחן את MERCHANTABILITYS הבאות בארגון שלך וסוג כל אחת:

- א. מתאיימת מאוד ל-LLM (יכול להחליף אדם לחלווטין)
- ב. מתאיימת חלקית (יכול לסייע לאדם)
- ג. לא מתאיימת (האדם חיוני)

Merchantises:

1. כתיבת מדיניות פרטיות לאתר
2. קבלת החלטה על פיטורי עובד
3. תרגום חוזה מאנגלית לעברית
4. בניית אסטרטגיית עסקית ל-3 שנים
5. סיכום פרוטוקול ישיבה
6. ניהול משא ומתן עם לקוחות כועס
7. ניתוח חשוב לקוחות מסקר (NPS)
8. בחירת ספק אסטרטגי לטוווח ארוך
9. יצירה של מבנים מייל לתמיכה לקוחות
10. ראיון عمוק עם מועמד לתפקיד בכיר

נמק את הבחירה שלך.

תרגיל 11.3. השוואת GPT-4 ל- Claude לצלבי הארגון

הນך CTO של חברת FinTech. אתה צריך לבחור מודל LLM מרכזי לארגון.

תרחישי שימוש מרכזים:

- ניתוח מסמכים משפטיים ופיננסיים (דוחות, חוזים)
- תמיכת לקוחות אוטומטית
- סיוע למפתחים בכתיבת קוד
- יצירה תוכן שיווקי

კრიტერიונים:

- עלות (נפח גבוה - 50 מיליון טוקנים/חודש)
- דיקט במסמכים ארוכים
- יכולת קוד
- תמיכה בעברית

- אבטחה ופרטיות
- **בנה טבלת השוואה ובחר מודל. נמק.**

תרגיל 4.11. ניתוח כישלון AI - מה השتبש?

חברת ביוטה הטמיה סוכן AI ל证实keit ל��וחות. אחרי חודש, היא גילתה שהסוכן:

- נתן מידע שגוי על CISCO ביטוחי ב-51% מהמקרים

- "המציא" פוליסות שלא קיימות

- לא הצליח להבין שאלות בעברית עם מונחים ביטוחיים

שאלות:

א) מה הסיבות האפשריות לכישלון?

ב) איזה חולשות של LLMs באו לידי ביטוי?

ג) איך הייתה ממליצה לתקן את המערכת?

ד) האם הייתה ממליצה להפסיק את הפרויקט לחלוות? למה?

תרגיל 5.5. בניית Business Case ליישום LLM

הנץ מנהל/ת מחלקת מכירות עם 20 נציגי מכירות. אתה רוצה להטמי סוכן AI שיסייע להם במשימות הבאות:

- כתיבת הצעות מחיר מותאמות אישית

- מענה על שאלות טכניות של לקוחות (באמצעות RAG על מסמכיו המוצר)

- סיוכום שיחות עם לקוחות ומעקב אוטומטי

כתב Business Case שפولל:

א) **בעיה עסקית:** מה הכאב הנוכחי?

ב) **פתרון מוצע:** מה בדיקת תטמייע?

ג) **תועלות:** מה יהיו היתרונות המדידים?

ד) **עלויות:** כמה זה יעלה? (כולל טכנולוגיה, הטעמה, הדרכה)

ה) **ROI:** תוך כמה זמן ההשקעה תשתלם?

ו) **סיכוןים:** מה עלול להשتبש?

ז) **המלצה:** האם כדאי להתקדם?

2.11 תרגילי קוד Python

תרגיל 6.6. חישוב עלות שימוש חודשית ב-API

כתב תוכנית Python שמקבלת:

- מספר שיחות/פעולות חודשיות

- ממוצע טוקני קלט לפעולה

- ממוצע טוקני פלט לפעולה

- מחיר קלט למיילון טוקנים

- מחיר פלט למיילון טוקנים

התוכנית צריכה לחשב ולהדפיס:

א) עלות לפעולה בודדת

ב) עלות חודשית כוללת

ג) עלות שנתית

ד) השוואה: כמה היה עולה באותו נפח עם מודל אחר (שהמשתמש מזמן מחירים שלו)

דוגמת הרצה:

```
Enter monthly operations: 50000
```

```
Enter avg input tokens: 2000
```

```
Enter avg output tokens: 800
```

```
Enter input price ($/1M tokens): 10
```

```
Enter output price ($/1M tokens): 30
```

```
==== Cost Analysis ====
```

```
Cost per operation: $0.044
```

```
Monthly cost: $2,200.00
```

```
Annual cost: $26,400.00
```

```
Compare with another model? (y/n): y
```

```
Enter input price ($/1M tokens): 0.5
```

```
Enter output price ($/1M tokens): 1.5
```

```
Alternative model monthly cost: $112.00
```

```
You would save: $2,088.00/month (94.9%)
```

21 סיכום הפרק

במהלך פרק זה עברנו מסע מקיף בעולם מודלי השפה הגדולים:

למدى מהם LLMs:

- מכונות השלמת דפוסים שאומנו על כמויות עצומות של טקסט

- לא מסדי נתונים, אלא מודלים סטטיסטיים מורכבים שיוצרים תוכן חדש

הכרנו את שני הכוחות העל:

- **תקשרות טבעית:** סוף סוף, מכונות שمبינות אותנו

- **עיבוד לוגיקה מורכבת:** לא רק משקל נוח, אלא חשיבה אמיתית

זהינו נקודות חזק:

- יצירתיות והפקת רעיונות
- הבנת הקשר ונואנסים
- גמישות והתאמה
- יכולת רב-לשונית

למדנו על נקודות חולשה קרייטיות:

- האיזות - המצאת מידע שנשמע מהימן
- חוסר עדכנות - מוגבל לתאריך אימון
- עליות - לא זנich בקנה מידה
- חוסר שקייפות - קופסה שחורה

רכשו כלים מנהליים:

- נוסחת ROI להערכת השקעה
- נוסחת עלות טוקנים לתוכנו תקציב
- ניתוח break-even להשוואת אלטרנטיבות

ראיינו דוגמאות מהשיטה:

- מנהלת שיווק שchosכת שעות ביצירת תוכן
- CFO שמנתח דוחות מורכבים במהירות
- מנהלת HR שמסננת קורות חיים ביעילות

המסר המרכז:

LLMs הם לא קסם, והם לא מושלמים. אבל כশובינים את היכולות והמוגבלות שלהם, ומישמים אותם בצורה מושכלת - הם כלי עסקי עצמאי שמשנה משחק. ההצלחה לא תליה בטכנולוגיה בלבד, אלא באופן שבו מנהלים מבנים, מתכננים ומישמים אותה. פרק זה סיפק לכם את היסודות - בפרק הבאם נצלול עמוק יותר לאקויסיטים, לארכיטקטורה, וליישום מעשי.

מקורות והמלצות לкриיאת נוספה

- 1. Transformer - מאמר היסוד על ארכיטקטורת Attention Is All You Need
- 2. GPT-4 Technical Report - תיעוד רשמי של OpenAI GPT-4 Technical Report
- 3. Anthropic Claude Documentation - מדריך מקיף למודלי Claude
- 4. The Economics of Large Language Models - ניתוח עלויות ו-ROI
- 5. Prompt Engineering Guide - מדריך מעמיק לכתיבות prompts

פתרונות מלאים לתרגילים

פתרון תרגיל 1: חישוב ROI

נתונים:

- 8 כותבים
- 40 מאמרים/חודש
- זמן נוכחי: 4 שעות/מאמר
- זמן עתידי: 2.5 שעות/מאמר
- עלות שעה: 120 ש"ח
- עלות חודש: $75 \text{ ש"ח} = \$20 \text{ ChatGPT Plus}$

חישוב:

סך שעות נוכחי:

$$(16) \quad \text{שעות/חודש} = 40 \times 4 = 160$$

סך שעות עתידי:

$$(17) \quad \text{שעות/חודש} = 40 \times 2.5 = 100$$

שעות נחיסכות:

$$(18) \quad \text{שעות/חודש} = 160 - 100 = 60$$

ערך החיסכון:

$$(19) \quad \text{ש"ח/חודש} = 60 \times 120 = 7,200$$

הוצאות מנוי:

$$(20) \quad \text{ש"ח/חודש} = 8 \times 75 = 600$$

ROI חודשי:

$$(21) \quad \frac{7,200 - 600}{600} \times 100\% = 1,100\%$$

זמן החזר השקעה:

חיסכון נטו חודשים:

$$(22) \quad \text{ש"ח/חודש} = 7,200 - 600 = 6,600$$

הוצאות הדרכה:

$$(23) \quad \text{ש"ח/חודש} = 5,000$$

זמן החזר:

$$(24) \quad \frac{5,000}{6,600} \approx 0.76 \text{ חודשים}$$

המלצת: בהחלטת כדאי ROI עצום, וההשקעה מתשלמת תוך פחות מחודש.

פתרונות תרגיל 2: זיהוי משימות

- כתיבת מדיניות פרטיות:** ב' (מתאים חלקית) - AI יכול לכתוב טיוותה מצוינת, אבל עורך דין צריך לאשר.
- החלטה על פיטורים:** ג' (לא מתאים) - דורש שיקול דעת אנושי, אמפתיה, הבנה של הקשר ארגוני.

3. **תרגומ חזה:** ב' (מתאים חלקית) - תרגום ראשוני מצוין, אבל חזה דורש עריכה משפטית אנושית.
4. **אסטרטגייה ל-3 שנים:** ב'-ג' - AI יכול לסייע בניתוח וברענוןת, אבל החלטה סופית דורשת אדם.
5. **סיכום פרוטוקול:** א' (מתאים מאוד) - מシימה מובנית שה-AI עשויה מצוין.
6. **ושא ומتن עם ליקוי כועס:** ג' (לא מתאים) - דורש אמפתיה אמיתית ושיקול דעת דינמי.
7. **ניטוח סקר NPS:** א'-ב' (מתאים מאוד) - AI מצוין בזיהוי דפוסים ומכמות.
8. **בחירה ספק אסטרטגי:** ב'-ג' - AI יכול לנתח, אבל החלטה כזו דורשת שיקולים רבים שאדם מבין טוב יותר.
9. **tabניות מייל:** א' (מתאים מאוד) - מシימה שה-AI עשויה מצוין.
10. **ראיון לתפקיד בכיר:** ג' (לא מתאים) - דורש הבנה عمוקה של תרבויות ארגוניות ושיקול דעת אנושי.

פתרונות תרגיל 6: קוד nohty לחישוב עלויות

להלן קוד מלא ומתווד:

Listing 1: LLM ילודם לתיולן ובשחם API

```

1 """
2 LLM API Cost Calculator
3 Calculates monthly and annual costs for LLM API usage
4 """
5
6 def calculate_cost(operations, input_tokens, output_tokens,
7                     input_price, output_price):
8     """
9     Calculate cost per operation and total costs
10
11     Args:
12         operations: Number of monthly operations
13         input_tokens: Average input tokens per operation
14         output_tokens: Average output tokens per operation
15         input_price: Price per 1M input tokens ($)
16         output_price: Price per 1M output tokens ($)
17
18     Returns:
19         dict: Cost breakdown
20     """
21
22     # Cost per operation
23     cost_per_op = (

```

```

23     (input_tokens / 1_000_000) * input_price +
24     (output_tokens / 1_000_000) * output_price
25 )
26
27 # Total costs
28 monthly_cost = operations * cost_per_op
29 annual_cost = monthly_cost * 12
30
31 return {
32     'cost_per_operation': cost_per_op,
33     'monthly_cost': monthly_cost,
34     'annual_cost': annual_cost
35 }
36
37 def compare_models(operations, input_tokens, output_tokens,
38                     model1_prices, model2_prices):
39 """
40 Compare costs between two models
41
42 Args:
43     operations: Number of monthly operations
44     input_tokens: Average input tokens
45     output_tokens: Average output tokens
46     model1_prices: (input_price, output_price) for model 1
47     model2_prices: (input_price, output_price) for model 2
48
49 Returns:
50     dict: Comparison results
51 """
52     cost1 = calculate_cost(operations, input_tokens, output_tokens,
53                            model1_prices[0], model1_prices[1])
54     cost2 = calculate_cost(operations, input_tokens, output_tokens,
55                            model2_prices[0], model2_prices[1])
56
57     savings = cost1['monthly_cost'] - cost2['monthly_cost']
58     savings_pct = (savings / cost1['monthly_cost']) * 100
59
60     return {
61         'model1_monthly': cost1['monthly_cost'],
62         'model2_monthly': cost2['monthly_cost'],
63         'monthly_savings': savings,

```

```

64     'savings_percentage': savings_pct
65 }
66
67 def main():
68     """Main interactive calculator"""
69     print("== LLM API Cost Calculator ==\n")
70
71     # Get user input
72     operations = int(input("Enter monthly operations: "))
73     input_tokens = int(input("Enter avg input tokens: "))
74     output_tokens = int(input("Enter avg output tokens: "))
75     input_price = float(input("Enter input price ($/1M tokens): "))
76     output_price = float(input("Enter output price ($/1M tokens): "))
77
78     # Calculate costs
79     costs = calculate_cost(operations, input_tokens, output_tokens,
80                           input_price, output_price)
81
82     # Display results
83     print("\n== Cost Analysis ==")
84     print(f"Cost per operation: ${costs['cost_per_operation']:.4f}")
85     print(f"Monthly cost: ${costs['monthly_cost']:.2f}")
86     print(f"Annual cost: ${costs['annual_cost']:.2f}")
87
88     # Compare with another model?
89     compare = input("\nCompare with another model? (y/n): ")
90     if compare.lower() == 'y':
91         alt_input_price = float(
92             input("Enter input price ($/1M tokens): "))
93
94         alt_output_price = float(
95             input("Enter output price ($/1M tokens): "))
96
97
98         comparison = compare_models(
99             operations, input_tokens, output_tokens,
100             (input_price, output_price),
101             (alt_input_price, alt_output_price))
102
103
104     print(f"\nAlternative model monthly cost: "

```

```

105         f"${comparison['model2_monthly']:.2f}")
106     print(f"You would save: "
107           f"${comparison['monthly_savings']:.2f}/month "
108           f"({comparison['savings_percentage']:.1f}%)")
109
110 if __name__ == "__main__":
111     main()

```

הרכבת הקוד:

- שמור את הקוד בקובץ `yp.rotaluclac_tsoc_mll`
- הרץ: `yp.rotaluclac_tsoc_mll nohtyp`
- עקוב אחר הנסיבות האינטראקטיביות

תרגיל מוחחב: הרחב את הקוד כך שייתמוך ב:

1. שימירת תוצאות לקובץ CSV
2. ויזואלייזציה של השוואת עלויות (גרף)
3. חישוב break-even עבור מעבר בין מודלים
4. תמיינה בהשוואה של יותר מושני מודלים

סוף הפרק הראשון

בפרק הבא נצלול לאקויסיטטם הבינה המלאכותית - מפת הכלים, הספקים והטכנולוגיות
שמנהל מודרני צריך להכיר.

מקורות

- 1 A. Vaswani et al., “Attention is all you need,” *Advances in Neural Information Processing Systems*, vol. 30, 2017.
- 2 OpenAI, “Gpt-4 technical report,” *arXiv preprint arXiv:2303.08774*, 2023.
- 3 Anthropic, “Claude 3 model card,” 2024. [Online]. Available: <https://www.anthropic.com/clause>
- 4 T. Brown et al., “Language models are few-shot learners,” *Advances in Neural Information Processing Systems*, vol. 33, 1877–1901, 2020.
- 5 L. Huang et al., “A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions,” *ACM Transactions on Information Systems*, 2024, arXiv:2311.05232. doi: [10.1145/3703155](https://doi.org/10.1145/3703155)

- 6 P. Lewis et al., “Retrieval-augmented generation for knowledge-intensive nlp tasks,” *Advances in Neural Information Processing Systems*, vol. 33, 9459–9474, 2020.
- 7 J. Wei et al., “Chain-of-thought prompting elicits reasoning in large language models,” *Advances in Neural Information Processing Systems*, vol. 35, 24824–24837, 2022.
- 8 E. Brynjolfsson, D. Li, and L. R. Raymond, “Generative ai at work,” *National Bureau of Economic Research*, 2023.
- 9 G. Almousa, R. Lovelace, V. L. Ziegler, et al., “The economic implications of large language model selection on earnings and return on investment: A decision theoretic model,” *arXiv preprint arXiv:2405.17637*, 2024.