

מפת החיפוש בקטלוג

איך שאילתה של לקוח הופכת לרשימת מוצרים, היום. הקטלוג מכיל כ-4,000 ספרים, אבל המערכת אמורה לשרת בעתיד כל סוג מוצר: כלי עבודה, בדים, חומרי בניין. הערוץ שמוצג כאן הוא הטלפוני, שבו הלקוח מדבר והמודל מחפש בשבילו.

הזרימה

Gemini Live המודל הקולי שומע את הלקוח ומנסח שאילתה כאן נוצרים רוב השיבושים: "ש"ס חתנים" נשמע כ"שש חתנים", "שך חתנים" ו"שוקאמין".	
voice/facts/book-search.js הכנת השאילתה - טבלה קבועה של תיקוני שיבושים ("מילתך" ← "בלכתך") שלב 0: השאילתה כמו שהיא, ובנוסף גרסה עם מילים נרדפות (ש"ס ↔ סט) שלב 1, רק אם שלב 0 לא מצא: בלי מילים תיאוריות ("חומש") ובלי שם הוצאה	
catalog/lib/search.js core/hebrew.js שרת הקטלוג: סינון ודירוג - נרמול: ניקוד, גרשיים ואות שימוש בתחילת מילה (ה/ו/ב/כ/ל/מ/ש) מוסרים כל מילה חייבת להתאים לאחד מהשדות: שם, קטגוריה, ספק או הערות - סובלנות לשגיאות לפי אורך המילה: עד 3 אותיות בלי שגיאה, 4-6 אותיות שגיאה אחת, 7 ומעלה שתיים - דירוג לפי שם המוצר בלבד. אין אינדקס, כל חיפוש עובר על כל הרשומות	
coversTitle() שער כיסוי: עדיף "לא מצאתי" על מוצר שגוי כל מילות התוכן בשאילתה חייבות להופיע בשם המוצר. מילה קצרה דורשת התאמה מדויקת. אם שום מועמד לא עובר, התשובה היא 0 תוצאות, בכוונה: בטלפון, מחיר בטוח של המוצר הלא נכון גרוע מ"לא מצאתי".	
collapseSeries() סידור וכיווץ - קודם התאמות מדויקות ואחר כך התאמות עם שגיאה. אם הלקוח ביקש סט, הסטים ראשונים 4 פריטים ומעלה עם אותו שם בסיס ובאותו מחיר מוצגים כשורה אחת. אם המחיר שונה, אלה מוצרים שונים - חיתוך ל-5 תוצאות לכל היותר, שחוזרות למודל	

ערוץ המייל עוקף את רוב הזרימה. הוא שולח לשרת את שם המוצר כמו שהמודל חילץ אותו מהמייל, בלי שער כיסוי ובלי מילים נרדפות. כלומר, לאותו קטלוג יש שתי מדיניות חיפוש שונות.

מה כללי ומה ספציפי לספרים

ידע על ספרים, כתוב בקוד

- ש"ס ↔ סט, וזיהוי "סט"
- מילים תיאוריות ("חומש", "מהדורת")
- מילים מבניות ("כרך", "חוברת", "מנוקד")
- טבלת תיקוני השיבושים ורשימת ההוצאות

מנגנון כללי, מתאים לכל תחום

- נרמול עברית
- סובלנות לשגיאות לפי אורך המילה
- שער הכיסוי, והחזרת 0 כשאין התאמה אמינה
- כיווץ לפי גודל הקבוצה ומחיר אחיד

מה שהמבנה לא יודע לעשות: מאפיינים מובנים (צבע, מידה, מק"ט), כללי שגיאות שונים לכל סוג שדה (בדגם DCD796, אות אחת שונה היא מוצר אחר), והתאמה לפי צליל.

מידה: 15 שאלות

11 שאלות משיחות אמיתיות ו-4 שיבושים שתועדו בעבר. כל אחת הורצה דרך הזרימה המלאה מול עותק של הקטלוג, פעם עם המנוע הנוכחי ופעם עם ספריית חיפוש מוכנה (MiniSearch, ברישיון MIT).

מנוע	נכון ראשון	לא נמצא	מוצר שגוי
המנוע הנוכחי	9	6	0
MiniSearch, כל המילים חייבות להתאים	9	6	0
MiniSearch, מספיק שחלק מהמילים יתאימו	11	4	0

שתי השאלות שההגדרה האחרונה הוסיפה נמצאו כי המנוע השמיט את המילה השגויה. שאר ההחטאות לא קשורות למנוע: קיצור בקטלוג (ע"מ = עדות המזרח), שער הכיסוי שפסל תוצאה נכונה בגלל מילה עודפת ("על התורה"), ושיבוש צליל ("שוקאמין"). בקורפוס יש רק שאלתה אחת על מוצר שלא קיים בקטלוג, ולכן הוא קטן מדי כדי למדוד את הסיכון לתשובה שגויה.

על מה אשמח לשמוע את דעתך

- האם להחליף את המנוע הביתי במנוע מוכן (Meiliseach, Typesense, MiniSearch), או לחכות עד שיהיו בקטלוג מאפיינים מובנים?
- שיבוש בתמלול קורה לפני החיפוש. מה עדיף: שדה "שם מדובר" לכל מוצר, מפתח צליל, הטיית המתמלל לאוצר המילים של הקטלוג, או שילוב שלהם?
- האם להשמיט מילה שלא נמצאה ולנסות שוב? הצעד הזה מוצא יותר, אבל מגדיל את הסיכון למוצר שגוי עם מחיר.