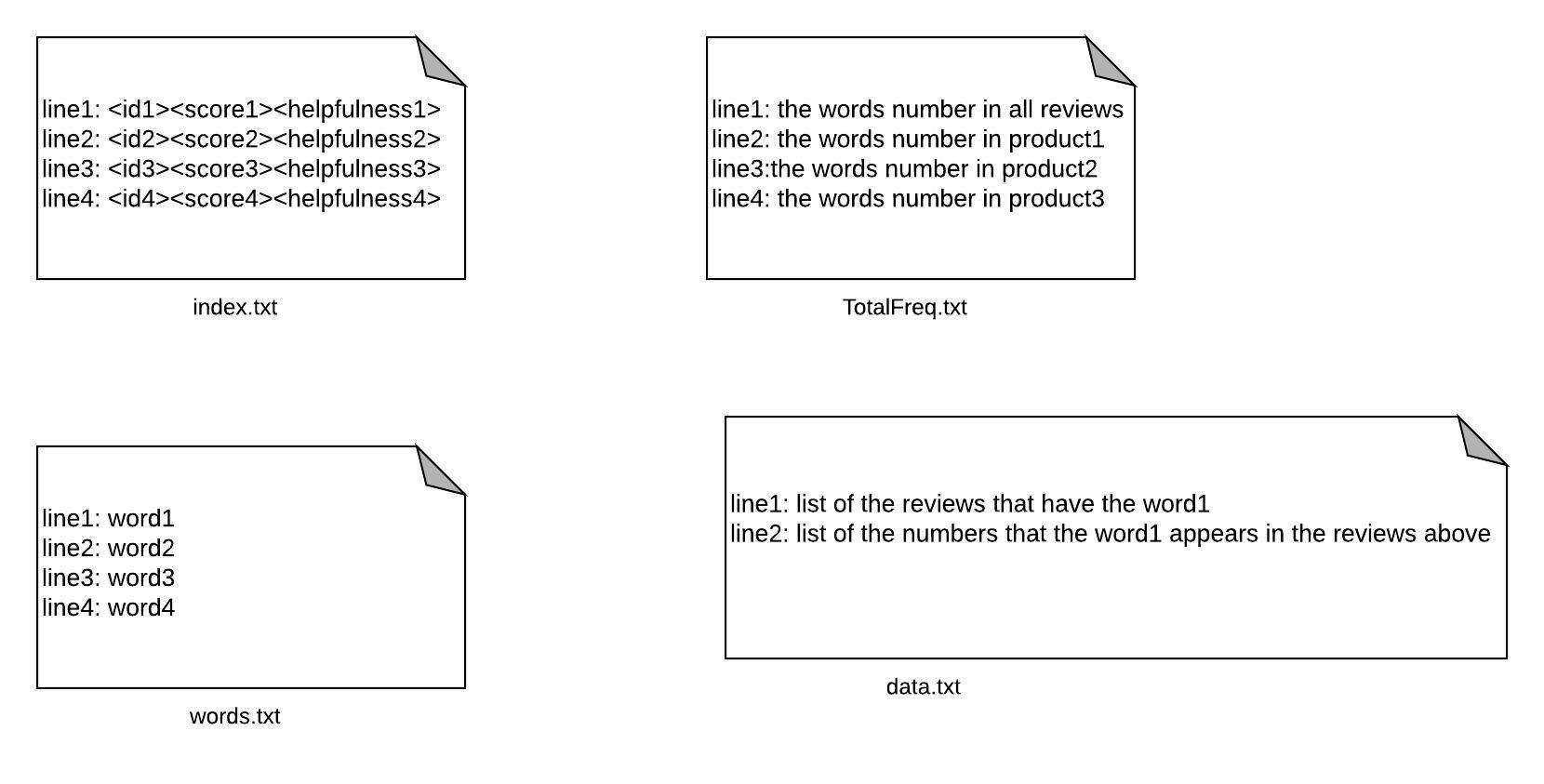
**Analyzes**

**פרטי האינדקס:**

כאשר מריצים את התוכנית מתווצר ארבעה קבצים בינאריים בפורמט הבא:

* קובץ index.txt המכיל בכל שורה ארבעה נתונים(מספר המזהה של כל review הוא מספר השורה בקובץ):
  + productId: האורך שלו קבוע (10 תווים) והוא נשמר בתחילת הקובץ.
  + Score: בגודל קבוע (תו בודד) הוא מספר של 1Byte.
  + Helpfulness: נשמר בצורה של מחרוזת בתוך המחרוזת יש "\" כדי שנוכל להפריד בין המונה והמכנה וברגע שקוראים אחד מהם אנו ממירים אותו לinteger-.
* קובץ TotalFreq.txt שמכיל בשורה הראשונה מספר המילים בכל ה Reviews ושאר השורות אורך של כל Review (מספר המזהה של כל review הוא מספר השורה בקובץ).
* קובצ ה-words.txt מכיל את כל המילים שנמצאים בכל ה-reviews ללא חזרה כך שכל מילה נמצאת בשורה בודדת.
* קובץ ה-data.txt מכיל את הנתונים של המילים כך שכל שתי שורות קשורות למילה בקובץ ה-words לפי הסדר שלהם כך שהשורה הראשונה שומרת מערך באיזה reviews מכילים את המילה והשורה השניה מתארת מערך של מספר הופעתה ב reviews.



**קריאת הנתונים מהזיכרו/הדיסק**

כאשר מריצים את התוכנית שום דבר לא נשמר בזיכרון, וברגע שצריכים לקרוא נתונים מסוימים ניגשים לקובץ המתאים בדיסק, קוראים את מה שצריכים ואז סוגרים את הקובץ שמסימים ממנו.

**הגודל הצפוי של האינדקס**

אפשר לחשב גודל האינדקס דרך הנוסחאות הבאות:

* קובץ ה-index.txt: שומרים productid, score, helpfulness בגדלים קבועים שהם 10, 1, 3-5 בהתאם לכן הנוסחא היא 16 כפול מספר ה-review.
* קובץ TotalFreq.txt בכל שורה שומרים מספר מסוג integer שגודלו 3 byte במומצע אז הנוסחא היא 3 כפול מספר ה-reviews.
* קובץ ה-words: ממוצע האותיות במילה הוא 7 אותיות, גודל השורה היא 2 bytes לכן הנוסחא: (2bytes\* מספר ה-reviews) + (7 \* מספר המילים ללא חזרה)
* קובצ ה-data.txt: עבור כל מילה שומרים שני מערכים כך שכל מערך מכיל מספרים של ה-reviews שהופיעה בהם המילה, הנוסחא היא: 2 \* מספר המילים \* (2 bytes + bytes 3 \* מספר ההופעה למילה)