תכנות מערכות מבוזרות - עבודה 3
דור שטרקר 204467757 וולדימיר שרגורודסקי 310016621

עיצוב המערכת:

המערכת מחולקת לשני חלקים:

1. חישוב ה-Mi. (Mutual information)
2. חישוב ה-Sim. (Similarity)

<u>חישוב ה-Mi:</u>

בשלב זה, ברצוננו לחשב את הנוסחה הבאה: $mi(p, slot, w) = \log\left(\frac{|p,slot,w| \times |*,slot,*|}{|p,slot,*| \times |*,slot,w|}\right)$.

להלן השלבים שמתבצעים כדי לחשב את הנוסחה:

א. ביצוע סינון לקלט:
קלט: כל השורות מקבצי ה-biarcs.
פלט: כל השלשות מהצורה $x, path, y$ וכמות המופעים של כל שלשה.
הסינון מתבצע כך שכל שלשה תהיה חוקית, כלומר path מכיל פועל ומילות קישור ו-x, y הם שמות עצם.
במקביל לביצוע השלב הנ"ל מתבצע גם החישוב עבור $|*, slot, *|$. (סך כל ההשלמות עבור כל המסלולים, מספר זה נשמר בסביבה)
בשלב זה לא בוצעו הנחות על הזיכרון.

ב. שלב ביניים עבור חישוב הרכיבים עבור הנוסחה:
שלב זה מורכב משלושה צעדי MR:
כל שלב מחשב חלק בנוסחה, למעט החישוב שבוצע בשלב הקודם עבור $|*, slot, *|$.
קלט: הפלט משלב א'.
פלטים:
$|p, slot, w|$ - מספר הפעמים שהמילה w משלימה slot מסוים במסלול p.
$|p, slot, *|$ - סך כל ההשלמות עבור slot מסוים במסלול p.
$|*, slot, w|$ - סך כל הפעמים שהמילה w משלימה את כל המסלולים עבור slot מסוים.
בשלב זה לא בוצעו הנחות על הזיכרון.

ג. שלב חישוב הנוסחה:
שלב זה מורכב משני צעדי MR:
השלב הראשון מבצע את החישוב הבא: $\frac{|p,slot,w| \times |*,slot,*|}{|p,slot,*|}$, השלב השני ידאג לשאר החישוב.
קלטים: עבור השלב הראשון - $|p, slot, w|$, $|p, slot, *|$. ($|*, slot, *|$ ילקח מהסביבה)
עבור השלב השני - הפלט של השלב הראשון ו-$|*, slot, w|$.
פלט: תוצאת ה-Mi עבור כל זוג, לכל Slot.
הנחות על הזיכרון:
בשלב הראשון הנחנו כי ניתן לשמור בזיכרון את כל המילים המשלימות מסלול (שמגיע כמפתח ל-reducer).
בשלב השני הנחנו כי ניתן לשמור בזיכרון את כל המסלולים שמילה (שמגיעה כמפתח ל-Reducer) משלימה.

עבור ריצה של 100% מהקבצים קיבלנו את מספרי זוגות ה מפתח-ערך הבאים:

שלב / פעולה	Mapper input records	Mapper output records	Reducer input records	Reducer output records
א'	184,982,852	10,004,661	10,004,661	10,004,661
ב'	30,013,983	30,013,983	8,828,749	2,907,012
ג'	5,776,743	5,776,743	5,776,743	5,776,743

<u>חישוב ה-Sim:</u>

בשלב זה ברצוננו לחשב את הנוסחות הבאות:

$$sim(slot_1, slot_2) = \frac{\sum_{w \in T(p_1,s) \cap T(p_2,s)} mi(p_1,s,w) + mi(p_2,s,w)}{\sum_{w \in T(p_1,s)} mi(p_1,s,w) + \sum_{w \in T(p_2,s)} mi(p_2,s,w)}$$

$$S(p_1, p_2) = \sqrt{sim(SlotX_1, SlotX_2) \times sim(SlotY_1, SlotY_2)}$$

להלן השלבים שמתבצעים כדי לחשב את הדומות בין המסלולים:

א. סינון ה-Mi:
קלט: הפלט מהשלב הקודם (חישוב Mi), TestSet הנבחר על ידי המשתמש.
פלט: קובץ Mi המכיל את המסלולים מה-TestSet בלבד. (במידה וקיימים כאלו)
הנחה על הזיכרון - ניתן לאחסן את המסלולים מה-TestSet בזיכרון.

ב. שלב ביניים עבור חישוב הנוסחה הראשונה:
שלב זה מורכב משלושה צעדי MR:
צעד א': יצירת "טבלה" של כל המסלולים האפשריים p_i, לכל מילה w, כולל ה-$mi(p_i, slot, w)$.
צעד ב': חישוב סכום ה-$mi(p, s, w_i)$ לכל מסלול p, עבור כל המילים w_i אשר משלימות אותו.
צעד ג': חישוב סכום ה-$mi(p, s, w_i) + mi(p', s, w_i)$ לכל זוג מסלולים $p, p', p \neq p'$ עבור כל המילים w_i אשר משלימות את שניהם. (על ידי הטבלה מהצעד הראשון)
קלט: הפלט מהשלב הקודם.
פלטים: סכום ה-$mi(p, s, w_i)$ לכל מסלול p, סכום ה-$mi(p, s, w_i) + mi(p', s, w_i)$ לכל זוג מסלולים $p, p', p \neq p'$.
הנחה על הזיכרון - ניתן לאחסן את המסלולים מה-TestSet בזיכרון.

ג. חישוב הנוסחה הראשונה:
קלט: שני הפלטים מהשלב הקודם.
פלט: $sim(slot_1, slot_2)$ עבור x, y לכל זוג מסלולים.

הנחה על הזיכרון - ניתן לשמור את כל המסלולים (שקיימים ב-TestSet) בזיכרון. אנחנו עושים זאת כדי ליצור את כל הזוגות מהקלט על מנת שיגיעו אותם מפתחות ל-Reducer.

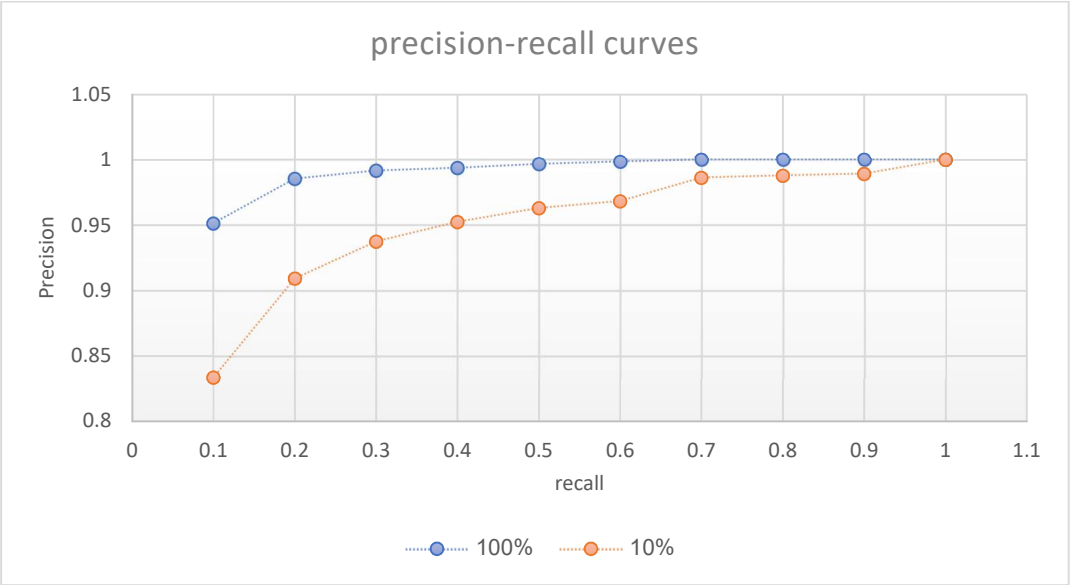
ד. חישוב הנוסחה השניה וביצוע מיון על פי הדומות:
שלב זה מורכב משני צעדי MR:
צעד א': חישוב הנוסחה השניה.
צעד ב': מיון.
קלט: הפלט מהשלב הקודם. (הנוסחה הראשונה)
פלט: הדומות בין המסלולים ב-TestSet, ממוינת בסדר יורד.
בשלב זה לא בוצעו הנחות על הזיכרון.

בשלב חישוב הsim, מספר זוגות ה מפתח – ערך המגיעים ל mapper הראשון הם כמספר השורות בקובץ ה MI החושב בשלב הקודם.

חישוב ה-F1 Measure:

Threshold Input size	0.01	0.001
100%	0.6996	0.9692
10%	0.7239	0.9435

גרף ה-Precision recall:



ניתוח שגיאות:

עבור threshold = 0.01 ו-100% מהקבצים:

True negative	False negative	True positive	False positive
• Associate with,	• Accompany with,	• Accompany, follow	• Be against, produce

distinguish from • Destroy, produce • Resemble, differ from • Contract, kill • Have, kill	associate with • Be for, prescribe for • Be in, enter • Give for, help • Kill, eradicate	• Contain, include • Bring, give • Cause, trigger • Develop, produce	

עבור threshold = 0.01 ו-10% מהקבצים:

True negative	False negative	True positive	False positive
אין	• Contain, have • Have, receive • Include, use for • Contain, include • Help, control	• Contain, consist of • Join with, unite with • Reduce to, convert to • Receive, contain • Control, reduce	• Contract, kill • Have, kill

ניתן לראות כי עבור 100% מהקבצים קיבלנו פחות False positive, בנוסף ניתן לראות כי עבור 10% מהקבצים אין כלל True negative. לכן, ניתן להסיק שעבור 100% מהקבצים התוצאות יהיו מדויקות יותר.

נשים לב כי הזוגות שהתקבלו כ-False positive ב-10% מהקבצים, מזוהים כ-True negative ב-100% מהקבצים.

נשים לב כי למשל הזוג Contain, include התקבל כ-False negative ב-10% מהקבצים, אך ב-100% מהקבצים הוא התקבל כ-True positive.