CS839 - Project Stage 3

Chakshu Ahuja, cahuja2@wisc.edu
Geetika, geetika@wisc.edu
Swati Anand, sanand@cs.wisc.edu

After downloading Table A, Table B, Candidate Set and Prediction List from CloudMatcher, we experimented:

Candidate Set Size (C)	7120
Prediction List Size (P)	595
Amazon (Table A)	4753
Barnes & Noble (Table B)	5469

Since the Candidate Set obtained was > 500, we moved to the **Density Calculation** step.

Iteration 1 (Without any blocking rules)

We randomly selected a batch of 50 examples from C
Density obtained: 6/50 = **0.12**
Since the density was less than 0.2, we moved to apply some blocking rules.

Iteration 2

Blocking Rule 1: Remove candidates with missing values in the *title* column

After analyzing data from Table A and Table B, we realized that there were missing values in the title column. Since ***title*** is the main field for the comparison of books, we decided to block all such cases with the missing ***title***

Reduced Candidate Set (C1) after this rule: 4464

We again randomly selected a batch of 50 examples and,
Density obtained: 8/50 = **0.16**
As density was again < 0.2, we decided to apply some more blocking rules

---

Iteration 3

Blocking Rule 2: Remove candidates with Non-Matching book format

During the Active Learning steps of Blocking and Matching on Cloud Matcher, we had assumed that book pairs with different book format would be considered as Non-Match. So we applied this blocking rule to remove all pairs with non-matching book format.

Reduced Candidate Set (C2) after this rule: 3809

We again randomly selected a batch of 50 examples and,
Density obtained: 9/50 = **0.18**
As density was again < 0.2, we moved to apply yet another blocking rule.

Iteration 4

Blocking Rule 3: Block candidates with similarity measure less than 0.1 on the *title* column

We did similarity measure using **one-word gram** and computed the fraction of common words (*intersection*) by total words (*union*) in the *title* column value of the pair.
We blocked all pairs with a similarity measure less than 0.10.

Reduced Candidate Set (C3) after this rule: 1520
We again randomly selected a batch of 50 examples and,
Density obtained: 23/50 = **0.46**

As density obtained is now > 0.2, we labeled in total 400 examples (can be seen labeled_pairs.csv) to estimate the precision-recall.

All the blocking rules code can be found in main.ipynb.

Estimating Precision and Recall

Using the jupyter notebook as provided, we calculated the Precision and Recall range as -

Recall = [0.96213812423464351 - 1.0019337320527815]
Precision = [0.95664452530551392 - 0.99573642707543841]