Anand Prashar

INF 558 – Homework 8

Task 1:

Analyze the given data and choose string similarities that you think are appropriate for each field. Explain your choices in the report.

Answer:

Zagats and Fodors have 3 similar columns each: Address, Phone, Cuisine

➔ Address is string, where, until we see 1st digit, we have restaurant name.
Chosen String similarity: Levenshtein Ratio
Reason: In string comparison where overall no. of edits affect similarity,
Levenshtein method are good at matching. Among Levenshtein methods, I found Levenshtein Ratio as more accurate representation of similarity.
ratio = (Levenshtein Distance)/(Alignment length)

➔ Phone has typical format:
In Zagats: ddd-ddd-dddd
In fodors: ddd/ddd-dddd
some phone number overlap into right column of cuisines (when there are 2 or more)
Chosen String similarity: Custom
Reason:
* Best case would be exact match of all 10 digits;
meaning with strong probability, they belong to same restaurant
* 2nd best would be area_code-subzone match
meaning, likely restaurants are placed very closely in a geographic area
* 3rd best would be area_code match
meaning, we can say they are not as closer.
* Last, completely different phone numbers.
meaning with strong probability, restaurants are different.

Why not use typical string similarity-
because, we have extra knowledge here !
for 2 phones: 123-456-789 and 999-456-789
string similarity will match them close, but geographically they are very different !

➔ Cuisine has typical format of-
C1 , or C1/C2 in case of multiple cuisines
Chosen String similarity: Jaro Winkler
Reason: Types of cuisines do not have many variations; Jaro has a bonus for same prefix.

Task 2: Design a scoring function to combine your field similarities. Explain your choices of weights in the scoring function in the report.

Answer:
overall_record_sim = 0.40 * match_address + 0.50 * match_phone+ 0.10 * match_cuisine

Phone is an unique property. Also, it has no string messiness of meaning same thing as Address. In real world we can match based on phone with highest confidence among all three.

Address, while not perfect due to string representation, can identify same/close entities with high degree of confidence.

Cuisine is not a unique identifier for any restaurants. Cuisines are limited, while restaurants are not. It helps if Cuisines match, but only when address and phone also agree with match. Hence, least weight.

Task 3: Design a rule-based blocking approach to reduce the number of comparisons you need to perform. Report the Reduction Ratio of your blocking rules.

Answer:

(token, phone) as blocking key
Reduction Ratio = 1 - #matchingTuples / (#Table1 x #Table2)
1 - 114/(533*331) = 99%