CSE 5243 Autumn'22 Assignment #5 – Intr Data Mining

Neng Shi

2022/11/15

1 Problem 1.

First, we compute hash functions for the four different documents and obtain the following table.

Row	S_1	S_2	S_3	S_4	$(2x+4)\ mod\ 5$	$(3x+1)\ mod\ 5$
0	1	0	0	1	4	1
1	0	0	1	0	1	4
2	0	1	0	1	3	2
3	1	0	1	1	0	0
4	0	0	1	0	2	3

Now, let us simulate the algorithm for computing the signature matrix. Initially, the matrix consists of all ∞'s:

	S_1	S_2	S_3	S_4
h1	∞	∞	∞	∞
h2	∞	∞	∞	∞

After considering row 0, the updated signature matrix is:

	S_1	S_2	S_3	S_4
h1	4	∞	∞	4
h2	1	∞	∞	1

After considering row 1, the updated signature matrix is:

	S_1	S_2	S_3	S_4
h1	4	∞	1	4
h2	1	∞	4	1

After considering row 2, the updated signature matrix is:

	S_1	S_2	S_3	S_4
h1	4	3	1	3
h2	1	2	4	1

After considering row 3, the updated signature matrix is:

	S_1	S_2	S_3	S_4
h1	0	3	0	0
h2	0	2	0	0

After considering row 4, the final updated signature matrix is:

	S_1	S_2	S_3	S_4
h1	0	3	0	0
h2	0	2	0	0

2 Problem 2.

Initial candidates: All four singleton sequences: A, T, C, G.

Cand.	sup
A	4
T	4
~~C~~	3
G	4

Then we generate length-2 candidate sequences.

Cand.	sup
AA	4
AT	4
AG	4
TA	4
~~TT~~	2
TG	4
GA	4
~~GT~~	2
~~GG~~	3

Then we generate length-3 candidate sequences.

Cand.	sup
~~AAA~~	3
AAT	4
~~AAG~~	3
ATA	4
ATG	4
AGA	4
TAA	4
~~TAT~~	1
~~TAG~~	2
TGA	4
GAA	4
~~GAT~~	1
~~GAG~~	1

Then we generate length-4 candidate sequences.

Cand.	sup
~~AATA~~	3
~~AATG~~	3
~~ATAA~~	3
ATGA	4
~~AGAA~~	3
~~TAAT~~	1
~~TGAA~~	3
~~GAAT~~	1

So, finally, the frequent sequences are A, G, T, AA, AT, AG, TA, TG, GA, AAT, ATA, ATG, AGA, TAA, TGA, GAA, and ATGA.

3 Problem 3.

Initial candidates: All three singleton sequences: A, B, C.

Cand.	sup
A	4
B	4
C	4

Then we generate length-2 candidate sequences, scan the database once, count support for each candidate, and find length-2 frequent sequences.

Cand.	sup
AA	4
AB	4
AC	4
BA	4
BB	4
BC	4
~~CA~~	2
~~CB~~	3
CC	4
~~(AA)~~	0
(AB)	3
~~(AC)~~	2
~~(BB)~~	0
~~(BC)~~	1
~~(CC)~~	0

Up to this point, we find length-1 frequent sequences: A, B, and C, and length-2 frequent sequences: AA, AB, AC, BA, BB, BC, CC, and (AB). Then we repeat the process, i.e., once we have length-k frequent sequences, we can generate length-(k+1) candidate sequences from length-k frequent using Apriori, scan the database, and find length-(k+1) frequent sequences until no frequent sequence or no candidate can be found.