

Indian Institute of Technology Jodhpur

CSL7110 Machine Learning with Big Data

Assignment 1: Map-Reduce and Similar Itemsets Mining

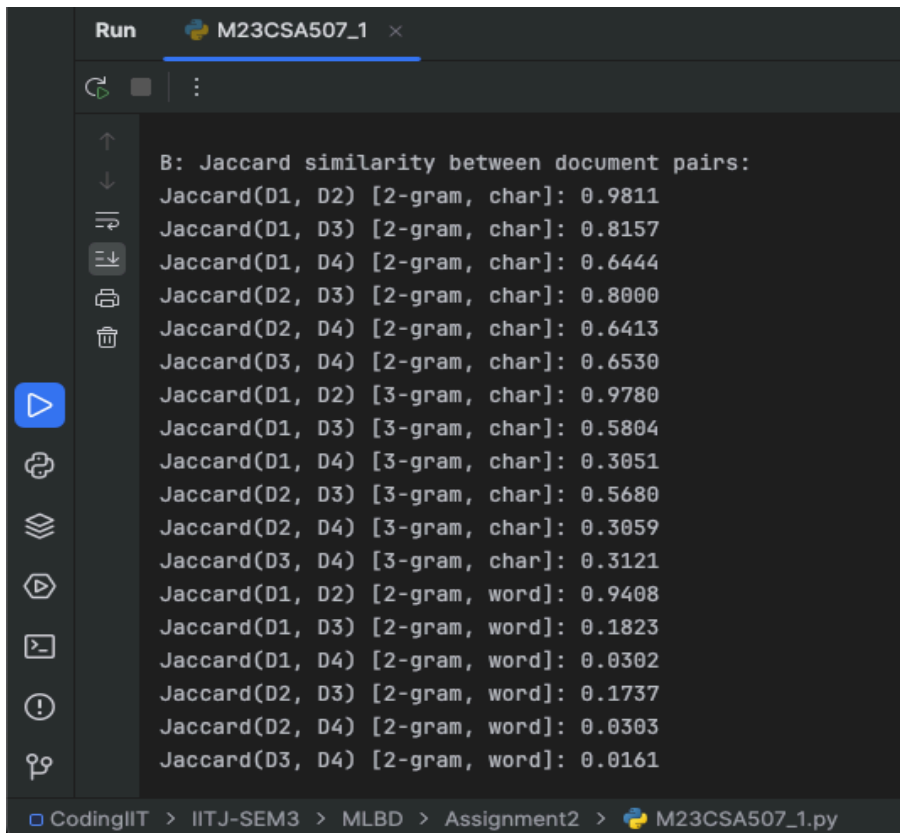
Anchit Mulye

m23csa507@iitj.ac.in

February 12, 2025

1. Create k-Grams

A.



```
Run M23CSA507_1 x
B: Jaccard similarity between document pairs:
Jaccard(D1, D2) [2-gram, char]: 0.9811
Jaccard(D1, D3) [2-gram, char]: 0.8157
Jaccard(D1, D4) [2-gram, char]: 0.6444
Jaccard(D2, D3) [2-gram, char]: 0.8000
Jaccard(D2, D4) [2-gram, char]: 0.6413
Jaccard(D3, D4) [2-gram, char]: 0.6530
Jaccard(D1, D2) [3-gram, char]: 0.9780
Jaccard(D1, D3) [3-gram, char]: 0.5804
Jaccard(D1, D4) [3-gram, char]: 0.3051
Jaccard(D2, D3) [3-gram, char]: 0.5680
Jaccard(D2, D4) [3-gram, char]: 0.3059
Jaccard(D3, D4) [3-gram, char]: 0.3121
Jaccard(D1, D2) [2-gram, word]: 0.9408
Jaccard(D1, D3) [2-gram, word]: 0.1823
Jaccard(D1, D4) [2-gram, word]: 0.0302
Jaccard(D2, D3) [2-gram, word]: 0.1737
Jaccard(D2, D4) [2-gram, word]: 0.0303
Jaccard(D3, D4) [2-gram, word]: 0.0161
CodingIIT > IITJ-SEM3 > MLBD > Assignment2 > M23CSA507_1.py
```

B.

```
Run M23CSA507_1 x
A: Distinct k-grams for each document:
D1 (2-gram, char): 263 unique k-grams
D2 (2-gram, char): 262 unique k-grams
D3 (2-gram, char): 269 unique k-grams
D4 (2-gram, char): 255 unique k-grams
D1 (3-gram, char): 765 unique k-grams
D2 (3-gram, char): 762 unique k-grams
D3 (3-gram, char): 828 unique k-grams
D4 (3-gram, char): 698 unique k-grams
D1 (2-gram, word): 279 unique k-grams
D2 (2-gram, word): 278 unique k-grams
D3 (2-gram, word): 337 unique k-grams
D4 (2-gram, word): 232 unique k-grams
CodingIIT > IITJ-SEM3 > MLBD > Assignment2 > M23CSA507_1.py
```

2. Min-Hashing

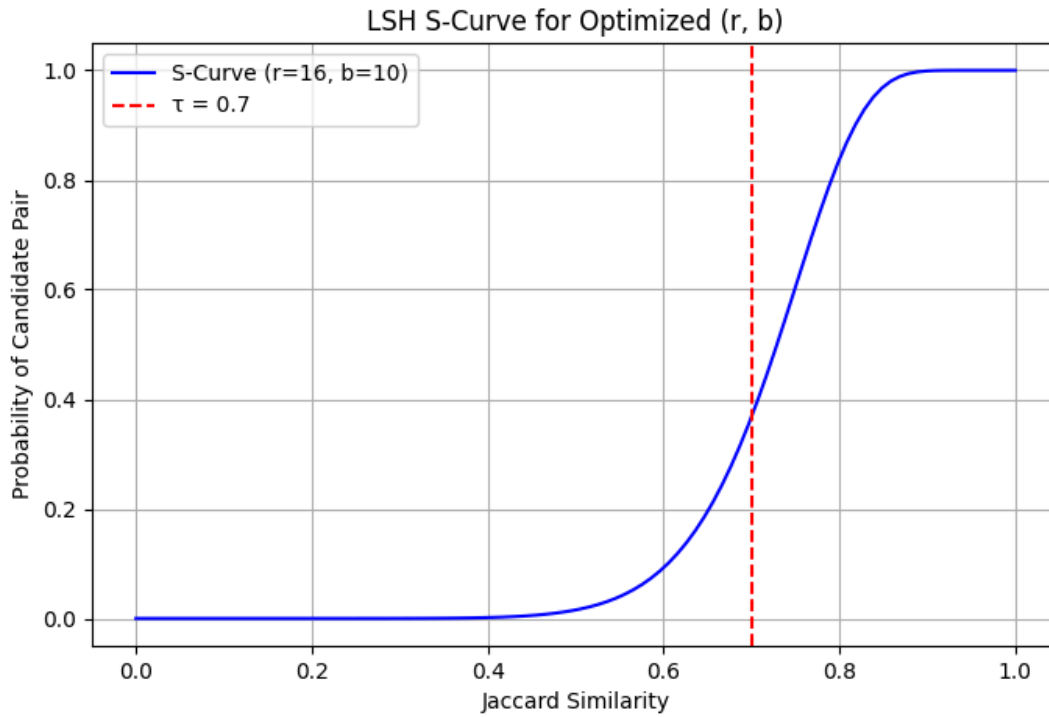
```
Run M23CSA507_2 x
/Users/anchitmulve/Documents/IITJ/CodingIIT/.venv/bin/pyt
A: Approximate jaccard similarity:
MinHash Jaccard(D1, D2) with t=20: 1.0000
MinHash Jaccard(D1, D2) with t=60: 0.9167
MinHash Jaccard(D1, D2) with t=150: 0.9667
MinHash Jaccard(D1, D2) with t=300: 0.9733
MinHash Jaccard(D1, D2) with t=600: 0.9600

B: Best t value based on accuracy vs. time tradeoff:
t=20: Estimated Jaccard = 1.0000
t=60: Estimated Jaccard = 0.9167
t=150: Estimated Jaccard = 0.9667
t=300: Estimated Jaccard = 0.9733
t=600: Estimated Jaccard = 0.9600

Process finished with exit code 0
CodingIIT > IITJ-SEM3 > MLBD > Assignment2 > M23CSA507_2.py
```

3. LSH:

A.



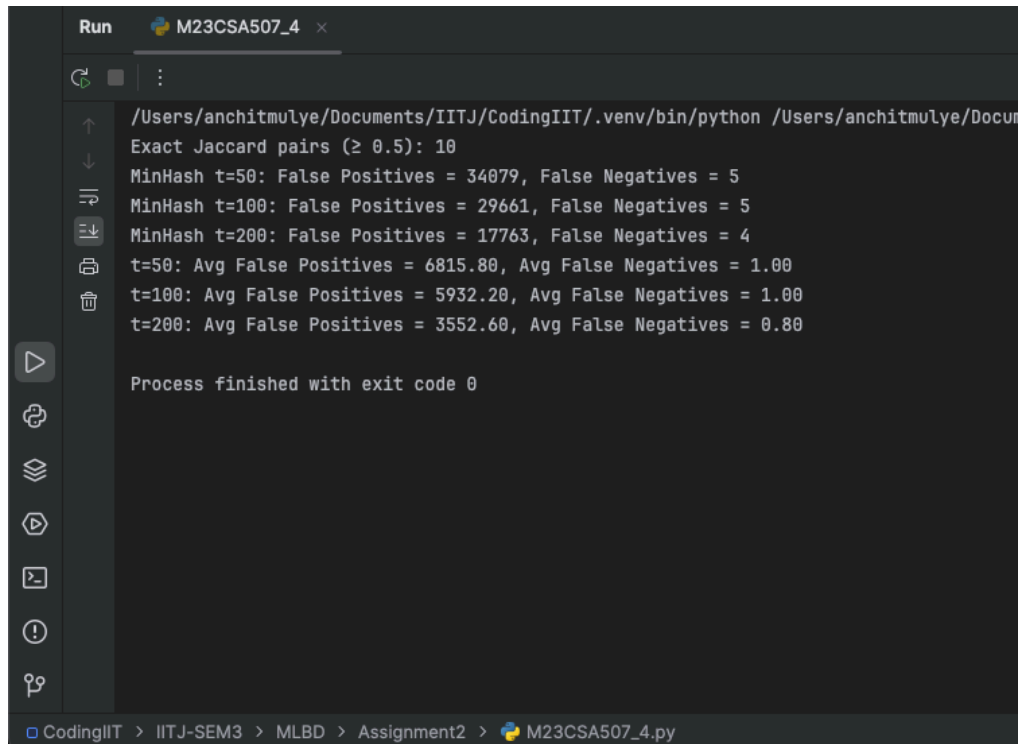
B.

```
Run M23CSA507_3 x
/Users/anchitmulye/Documents/IITJ/CodingIIT/.venv/bin/python /Users/anchitmulye/Docum

B: Best t value based on accuracy vs. time tradeoff:
t=20: Estimated Jaccard = 1.0000
t=60: Estimated Jaccard = 1.0000
t=150: Estimated Jaccard = 1.0000
t=300: Estimated Jaccard = 1.0000
t=600: Estimated Jaccard = 1.0000
Optimal LSH parameters: r = 16, b = 10
LSH Probability(D1, D2) > 0.7: 1.0000
LSH Probability(D1, D3) > 0.7: 0.0671
LSH Probability(D1, D4) > 0.7: 0.0001
LSH Probability(D2, D3) > 0.7: 0.0545
LSH Probability(D2, D4) > 0.7: 0.0001
LSH Probability(D3, D4) > 0.7: 0.0001

CodingIIT > IITJ-SEM3 > MLBD > Assignment2 > M23CSA507_3.py
```

4. Min-Hashing on MovieLens dataset

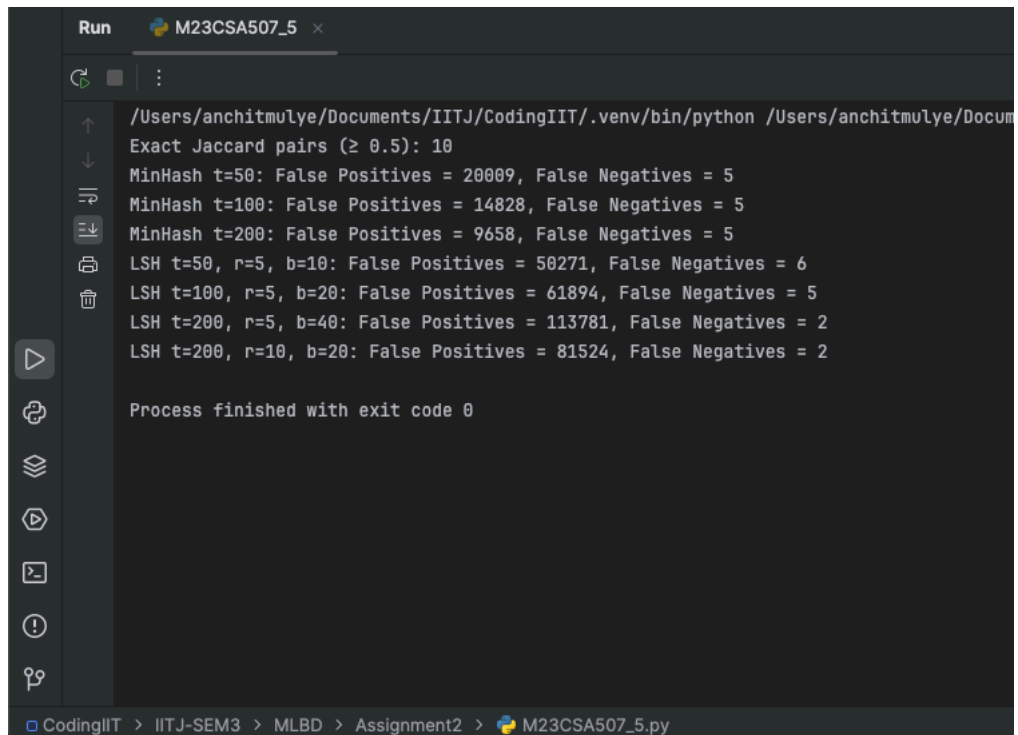


A terminal window titled 'Run M23CSA507_4' showing the output of a Python script. The script calculates Jaccard pairs and MinHash results for different values of t (50, 100, 200). The output includes the number of exact Jaccard pairs (≥ 0.5) and the number of false positives and false negatives for each t value. It also shows the average false positives and false negatives for each t value. The process finished with exit code 0.

```
Run M23CSA507_4 x
/Users/anchitmulye/Documents/IITJ/CodingIIT/.venv/bin/python /Users/anchitmulye/Docum
Exact Jaccard pairs (≥ 0.5): 10
MinHash t=50: False Positives = 34079, False Negatives = 5
MinHash t=100: False Positives = 29661, False Negatives = 5
MinHash t=200: False Positives = 17763, False Negatives = 4
t=50: Avg False Positives = 6815.80, Avg False Negatives = 1.00
t=100: Avg False Positives = 5932.20, Avg False Negatives = 1.00
t=200: Avg False Positives = 3552.60, Avg False Negatives = 0.80

Process finished with exit code 0
CodingIIT > IITJ-SEM3 > MLBD > Assignment2 > M23CSA507_4.py
```

5. LSH on MovieLens dataset



A terminal window titled 'Run M23CSA507_5' showing the output of a Python script. The script calculates LSH results for different values of t, r, and b. The output includes the number of false positives and false negatives for each combination of t, r, and b. The process finished with exit code 0.

```
Run M23CSA507_5 x
/Users/anchitmulye/Documents/IITJ/CodingIIT/.venv/bin/python /Users/anchitmulye/Docum
Exact Jaccard pairs (≥ 0.5): 10
MinHash t=50: False Positives = 20009, False Negatives = 5
MinHash t=100: False Positives = 14828, False Negatives = 5
MinHash t=200: False Positives = 9658, False Negatives = 5
LSH t=50, r=5, b=10: False Positives = 50271, False Negatives = 6
LSH t=100, r=5, b=20: False Positives = 61894, False Negatives = 5
LSH t=200, r=5, b=40: False Positives = 113781, False Negatives = 2
LSH t=200, r=10, b=20: False Positives = 81524, False Negatives = 2

Process finished with exit code 0
CodingIIT > IITJ-SEM3 > MLBD > Assignment2 > M23CSA507_5.py
```