CSE 601: Data Mining and Bioinformatics

Project 1

part2

Association Analysis

Jingsong Li (50322345)

Kaige Gao (50320916)

Junyang Li (50320301)

Brief description of Apriori algorithm and the flow of the association rule generation algorithm.

a. Steps to implement Apriori algorithm:
    1. Import data from the given file, then add description prefix of each data as required.
    2. Set the minimum support and confidence.
    3. Generate candidate itemsets of length=1 and name it as d1.
    4. Prune d1, delete unfrequent items and generate frequent length=1 itemsets.
    5. Repeat followed steps until there is no new frequent itemsets:
        1) generate length=k candidate itemsets based on previous length=(k-1) frequent itemsets, compare the first k-2 attributes of two length=(k-1) itemsets, if they are the same, merge them to a new length=k candidate itemset
        2) count the support of each new length=k candidate itemset
        3) eliminate infrequent candidate itemsets, generate length=k frequent itemsets and name it as dk

b. Flow of the association rule generation algorithm:
    1. Start from d2, compare each item in dk with previous generated items
    2. If item in dk contains the frequent set generated, we can say it can form a rule
    3. Use the item in dk as rule, and item generated as body, the part not in item generated is head
    4. Store rule, head and body into object
    5. Store rules into a list

Result (This is the statistical result of the data. See anwser.txt file generated during program execution for detailed data.) :

Support = 30%

number of length-1 frequent itemsets: 196
number of length-2 frequent itemsets: 5323
number of length-3 frequent itemsets: 5251
number of length-4 frequent itemsets: 1463
number of length-5 frequent itemsets: 388
number of length-6 frequent itemsets: 61
number of length-7 frequent itemsets: 3
Total number of frequent itemsets: 12685

Support = 40%

number of length-1 frequent itemsets: 167
number of length-2 frequent itemsets: 753
number of length-3 frequent itemsets: 149
number of length-4 frequent itemsets: 7
number of length-5 frequent itemsets: 1
Total number of frequent itemsets: 1077

Support = 50%

number of length-1 frequent itemsets: 109
number of length-2 frequent itemsets: 63
number of length-3 frequent itemsets: 2
Total number of frequent itemsets: 174

Support = 60%

number of length-1 frequent itemsets: 34
number of length-2 frequent itemsets: 2
Total number of frequent itemsets: 36

Support = 70%
number of length-1 frequent itemsets: 7
Total number of frequent itemsets: 7

Support = 50% and Confidence = 70%

number of length-1 frequent itemsets: 109
number of length-2 frequent itemsets: 63
number of length-3 frequent itemsets: 2
Total number of frequent itemsets: 174
number of rules is 117

Template1

asso_rule.template1("RULE", "ANY", ['G59_UP'])	26
asso_rule.template1("RULE", "NONE", ['G59_UP'])	91
asso_rule.template1("RULE", 1, ['G59_UP', 'G10_Down'])	40
asso_rule.template1("HEAD", "ANY", ['G59_UP'])	9
asso_rule.template1("HEAD", "NONE", ['G59_UP'])	108
asso_rule.template1("HEAD", 1, ['G59_UP', 'G10_Down'])	17
asso_rule.template1("BODY", "ANY", ['G59_UP'])	17
asso_rule.template1("BODY", "NONE", ['G59_UP'])	100
asso_rule.template1("BODY", 1, ['G59_UP', 'G10_Down'])	24

Template2

asso_rule.template2("RULE", 3)	9
asso_rule.template2("HEAD", 2)	6
asso_rule.template2("BODY", 1)	117

Template3

asso_rule.template3("1or1", "HEAD", "ANY", ['G10_Down'], "BODY", 1, ['G59_UP'])	24
asso_rule.template3("1and1", "HEAD", "ANY", ['G10_Down'], "BODY", 1, ['G59_UP'])	1
asso_rule.template3("1or2", "HEAD", "ANY", ['G10_Down'], "BODY", 2)	11
asso_rule.template3("1and2", "HEAD", "ANY", ['G10_Down'], "BODY", 2)	0
asso_rule.template3("2or2", "HEAD", 1, "BODY", 2)	117
asso_rule.template3("2and2", "HEAD", 1, "BODY", 2)	3