

# List of Figures

1.1	Structure of Thesis with Proposed Algorithms . . . . .	11
1.2	Organization of the Thesis . . . . .	12
2.1	Taxonomy of closed high utility itemsets mining according to mining approaches . . . . .	27
2.2	Taxonomy of high utility itemsets mining with negative utility according to mining approaches . . . . .	37
3.1	Set-enumeration tree for $I = \{A, B, C, D, E\}$ . . . . .	49
3.2	Execution times on chess dataset with different <i>max_length</i> constraints . .	60
3.3	Execution times on mushroom dataset with different <i>max_length</i> constraints	61
3.4	Memory usage on chess dataset with different <i>max_length</i> constraints . .	62
3.5	Memory usage on mushroom dataset with different <i>max_length</i> constraints	63
3.6	Scalability Runtime Comparison on chess dataset with different <i>max_length</i> constraints . . . . .	65
3.7	Scalability Runtime Comparison on mushroom dataset with different <i>max_length</i> constraints . . . . .	66
3.8	Scalability Memory Comparison on chess dataset with different <i>max_length</i> constraints . . . . .	67
3.9	Scalability Memory Comparison on mushroom dataset with different <i>max_length</i> constraints . . . . .	68
4.1	Runtime evaluation on accident dataset . . . . .	84
4.2	Runtime evaluation on chess dataset . . . . .	84
4.3	Runtime evaluation on mushroom dataset . . . . .	85
4.4	Runtime evaluation on foodmart dataset . . . . .	87
4.5	Runtime evaluation on retail dataset . . . . .	87
4.6	Memory consumption on accident dataset . . . . .	87
4.7	Memory consumption on chess dataset . . . . .	87
4.8	Memory consumption on mushroom dataset . . . . .	88
4.9	Memory consumption on foodmart dataset . . . . .	88
4.10	Memory consumption on retail dataset . . . . .	88
4.11	Runtime scalability of the algorithms on accidents dataset . . . . .	89
4.12	Runtime scalability of the algorithms on foodmart dataset . . . . .	89

4.13	Memory scalability of the algorithms on accidents dataset . . . . .	89
4.14	Memory scalability of the algorithms on foodmart dataset . . . . .	89
5.1	Set-enumeration tree for $I = \{a, b, c, d, e\}$ . . . . .	96
5.2	Comparison of RSU (left) and REU (right) upper bounds . . . . .	101
5.3	Execution time on dense datasets . . . . .	109
5.4	Execution time on sparse datasets . . . . .	110
5.5	Memory usage on dense datasets . . . . .	112
5.6	Memory usage on sparse datasets . . . . .	113
5.7	Scalability Runtime Comparison on various datasets . . . . .	116
5.8	Scalability Memory Comparison on various datasets . . . . .	117
6.1	Calculate <i>RTWU</i> using utility-array . . . . .	126
6.2	Runtime performance on accidents dataset . . . . .	133
6.3	Runtime performance on chess dataset . . . . .	134
6.4	Runtime performance on mushroom dataset . . . . .	135
6.5	Runtime performance on T10I4D100K dataset . . . . .	136
6.6	Runtime performance on T40I10D100K dataset . . . . .	137
6.7	Memory consumption on accidents dataset . . . . .	138
6.8	Memory consumption on chess dataset . . . . .	139
6.9	Memory consumption on mushroom dataset . . . . .	140
6.10	Memory consumption on T10I4D100K dataset . . . . .	141
6.11	Memory consumption on T40I10D100K dataset . . . . .	142
6.12	Runtime scalability on mushroom dataset . . . . .	143
6.13	Runtime scalability on T10I4D100K dataset . . . . .	144
6.14	Memory scalability on mushroom dataset . . . . .	145
6.15	Memory scalability on T10I4D100K dataset . . . . .	146
7.1	Calculate <i>RTWU</i> using utility-array . . . . .	153
7.2	Execution time on dense datasets . . . . .	162
7.3	Execution time on sparse datasets . . . . .	163
7.4	Memory usage on dense datasets . . . . .	165
7.5	Memory usage on sparse datasets . . . . .	167
7.6	Scalability Runtime Comparison on accidents and T40I10D100K datasets	169
7.7	Scalability Memory Comparison on accidents and T40I10D100K datasets	169

# List of Tables

2.1	A transaction dataset . . . . .	19
2.2	External utility value . . . . .	19
2.3	Transaction Utility (TU) . . . . .	20
2.4	TWU values of items . . . . .	21
2.5	An overview of two-phase top-k High utility itemset mining algorithms . . . . .	24
2.6	An overview of one-phase top-k High utility itemset mining algorithms . . . . .	25
2.7	An overview of closed high utility itemsets mining algorithms . . . . .	30
2.8	Transactional dataset . . . . .	32
2.9	External utility of items . . . . .	32
2.10	Redefined Transaction Utility . . . . .	34
2.11	<i>RTWU</i> value of each item . . . . .	35
2.12	Sorted items according to $\succ$ total order . . . . .	35
2.13	HUIs of the running example . . . . .	36
2.14	An overview of High utility itemset mining algorithms with negative utility values . . . . .	40
3.1	Revised <i>TU</i> values in the example . . . . .	46
3.2	<i>TWU</i> values of items . . . . .	46
3.3	<i>RTWU</i> values of items . . . . .	46
3.4	Transaction Merging . . . . .	51
3.5	Final HUIs of the running example . . . . .	58
3.6	Statistical information about datasets . . . . .	59
3.7	Relative runtime improvement analysis on chess dataset . . . . .	63
3.8	Relative runtime improvement analysis on mushroom dataset . . . . .	64
3.9	Relative memory consumption analysis on chess dataset . . . . .	64
3.10	Relative memory consumption analysis on mushroom dataset . . . . .	64
3.11	Relative runtime best, average and minimum comparison . . . . .	64
3.12	Relative memory consumption best, average and minimum comparison . . . . .	65
4.1	A transaction dataset . . . . .	72
4.2	External utility value . . . . .	72
4.3	Transaction Utility . . . . .	73
4.4	Transaction Weighted Utility . . . . .	73

4.5	Top-k HUIs for $k = 10$ . . . . .	73
4.6	$TWU$ values of items as $\succ$ order. . . . .	75
4.7	$EUCS$ map for transaction $T_1$ . . . . .	75
4.8	$EUCS$ map up-to transaction $T_2$ . . . . .	75
4.9	$EUCS$ Map with all $TWU$ . . . . .	76
4.10	Final $EUCS$ Map. . . . .	76
4.11	Statistical information about datasets . . . . .	84
4.12	Improvements of TKEH over kHMC at the highest value of $k$ ( $k = 1000$ ) . . . . .	89
5.1	Comparison of the characteristics of the existing algorithms with our proposed algorithm EHIN . . . . .	103
5.2	Transaction Utility (TU) . . . . .	106
5.3	Redefined Transaction Utility . . . . .	106
5.4	$RTWU$ values of items based on redefined $TU$ . . . . .	106
5.5	Final HUIs of the running example . . . . .	107
5.6	Statistical information about datasets . . . . .	108
5.7	Relative runtime improvement analysis on dense datasets . . . . .	114
5.8	Relative runtime improvement analysis on sparse datasets . . . . .	114
5.9	Relative memory usage analysis on dense datasets . . . . .	115
5.10	Relative memory usage analysis on sparse datasets . . . . .	115
5.11	Relative runtime analysis on best, average and minimum case . . . . .	115
5.12	Relative memory analysis on best, average and minimum case . . . . .	116
6.1	Redefined Transaction Utility . . . . .	120
6.2	Sorted items according to $\succ$ total order with their $RTWU$ value . . . . .	131
6.3	Final HUIs of the running example . . . . .	131
6.4	Dataset characteristics. . . . .	132
6.5	Number of candidates on accidents dataset . . . . .	133
6.6	Number of candidates on chess dataset . . . . .	133
6.7	Number of candidates on mushroom dataset . . . . .	134
6.8	Number of candidates on T10I4D100K dataset . . . . .	135
6.9	Number of candidates on T40I10D100K dataset . . . . .	136
6.10	Number of HUIs mined on accidents and chess dataset . . . . .	137
6.11	Number of HUIs mined on mushroom dataset . . . . .	138
6.12	Number of HUIs mined on T10I4D100K and T40I10D100K dataset . . . . .	139
6.13	Runtime improvements of EHNL over EHNL(RSUP) and EHNL(TM) . . . . .	146
6.14	Memory improvements of EHNL over EHNL(RSUP) and EHNL(TM) . . . . .	147
7.1	HUIs of the running example . . . . .	160
7.2	CHUIs of the running example . . . . .	160
7.3	Statistical information about datasets . . . . .	160

7.4 Runtime and memory improvements of CHN over CHN(RSU-Prune),  
CHN(TM) and FHN . . . . . 168