ased Frequent Pattern Mining with Multiple Item Support

Applied Sciences 9(10):2075 · May 2019

Vang

Jui-Yen Chang

ı support cannot comprehensively address the complexity of items in large datasets. In this study, we propose a
ıework (named Multiple Item Support Frequent Patterns, MISFP-growth algorithm) that uses Hadoop-based
achieve high-efficiency mining of itemsets with multiple item supports (MIS). The proposed architecture consists
ı the counting support phase, a Hadoop MapReduce architecture is employed to determine the support for each
ytics phase, sub-transaction blocks are generated according to MIS and the MISFP-growth algorithm identifies
rns. To facilitate decision makers in setting MIS, we also propose the concept of classification of item (COI),
of higher homogeneity into the same class, by which the items inherit class support as their item support. Three
plemented to validate the proposed Hadoop-based MISFP-growth algorithm. The experimental results show
duction in the execution time on parallel architectures. The proposed MISFP-growth algorithm can be
istributed computing framework. Furthermore, according to the experimental results, the enhanced performance
thm indicates that it could have big data analytics applications.

:, actual
item…

Experimental design, dataset attributes and descriptions.

Support value of each product class.

Results of Experiment 1.

applied sciences

icle

ISFP-Growth: Hadoop-Based Frequent Pattern Iining with Multiple Item Support

:n-Shu Wang [1,*] and Jui-Yen Chang [2]

Department of Information and Finance Management, National Taipei University of Technology, Taipei 10608, Taiwan
Department of Management Information System, National Chengchi University, Taipei 11605, Taiwan; ketrelo0225@gmail.com
Correspondence: wangcs@ntut.edu.tw; Tel.: +886-2-2771-2171 (ext. 2355)

ceived: 9 April 2019; Accepted: 8 May 2019; Published: 20 May 2019

chec
upd

ostract: In practice, single item support cannot comprehensively address the complexity of item
:ge datasets. In this study, we propose a big data analytics framework (named Multiple Item Sup
equent Patterns, MISFP-growth algorithm) that uses Hadoop-based parallel computing to ach
gh-efficiency mining of itemsets with multiple item supports (MIS). The proposed architec
nsists of two phases. First, in the counting support phase, a Hadoop MapReduce architect
employed to determine the support for each item. Next, in the analytics phase, sub-transact
ocks are generated according to MIS and the MISFP-growth algorithm identifies the frequenc
itterns. To facilitate decision makers in setting MIS, we also propose the concept of classifica
item (COI), which classifies items of higher homogeneity into the same class, by which the it
herit class support as their item support. Three experiments were implemented to validate
oposed Hadoop-based MISFP-growth algorithm. The experimental results show approxima
% reduction in the execution time on parallel architectures. The proposed MISFP-growth algori
n be implemented on the distributed computing framework. Furthermore, according to
perimental results, the enhanced performance of the proposed algorithm indicates that it co
ve big data analytics applications.

eywords: big data analytics; Hadoop MapReduce parallel computing; frequent pattern discov
ultiple item support

sor technology and ubiquitous use of various mobile devices generate increasing amounts of c
ıch is expected to continue increasing by 35% annually [1]. Interest in data-driven decision makir
wing in response to escalating data generation, and a large variety of big data analytics applicat
e been emerging accordingly. Developing models to find critical information and analyzing va
n big data have become the subject of deep exploration and intense discussion [2–4].

Big data analytics and applications are interesting and important prominent topics [5]. Identify
quent itemsets in large transactional databases, known as frequent patterns (FP), and the F
ociation rules can be valuable to decision makers for setting up strategies [6,7]. Among th
ımonly-applied FP-mining algorithms, the Apriori algorithm is regarded as a classic method [
Apriori algorithm uses a bottom–up approach to generate candidate itemsets. As the quantit
a is extremely large, the bottom–up approach becoming lower processing efficiency. To overc
limitations of the traditional Apriori algorithm, Han et al. proposed an FP-growth algorit
t included two phases: constructing an FP-tree and mining the FP-tree that would be m

cient [10,11]. For both Apriori and FP-growth algorithms, it is important to set reasonable minim
port values [12,13].

However, obviously all items in the database vary widely from many perspectives such as t
e or profit. Generally, if only one minimum support is used for FP mining, those high-profit it
stly with lower selling frequencies, such as refrigerators and smartphones, would not be consid
requent patterns [14]. Furthermore, lower minimum support would lead to the generation of m
aningless association rules, which would complicate the decision-making process [15]. There
ng a single minimum support for all items in the database is insufficient for FP analysis. Sir
n support cannot comprehensively address the complexity of items in large datasets. Thus,
portant and necessary to consider multiple item supports while big data analysis is applie
ctice. Recently, several studies have proposed the concept of multiple item supports (MIS
ress the trade-offs required by decision makers [16,17]. For FP mining research, applying MIS to
ing warrants further analysis. In addition, a large amount of research has attempted to impr
algorithmic efficiency of FP mining when working with big data. Apache's Hadoop system app
pReduce programming to solve these problems encountered in processing big data [18]. Us
doop MapReduce to implement the parallelization of traditional association rule mining approac
h as the Apriori and FP-growth algorithms, was propose to improve the overall performanc
quency pattern mining [19]. For illustration, the k-phase parallel Apriori algorithm was propose
ntify k-frequency items in k-scans using Hadoop MapReduce [20].

To optimize FP mining, this study proposes a multiple item support frequent patt
SFP)-growth algorithm, which mines association rules from FP by using multiple item sup
thermore, to improve the efficiency of the analysis, the proposed algorithm is deployed
doop MapReduce architecture. Section 2 summarizes and discusses the related research
tion 3 details the proposed algorithm of Hadoop-based MISFP-growth algorithm, and Secti
nonstrates the proposed MISFP-growth algorithm with an example implementation. Finally, t
eriments are implemented to validate the accuracy and performance of the proposed Hadoop-ba
SFP-growth algorithm. According to the experiment results, the proposed algorithm achie

This section summarizes association rule mining algorithms with single support and multi supp
the Hadoop MapReduce architecture is explained.

Association analysis attempts to determine the frequency patterns consisting of items
aset. The most well-known association mining algorithm is the Apriori algorithm [8,9], whic
eloped to scan transactional databases iteratively and generate candidate frequent itemsets. T
reshold known as the support filters items to form the candidate itemsets based on their frequ
occurrence. When no additional candidate itemsets can be generated, the frequency patterns
associated rules are constructed for decision makers' reference. To improve the efficiency o
ing, many researchers have attempted to reduce the effort for repeated database (DB) scann
s, the FP-growth algorithm [10] emerged to facilitate FP mining by scanning the DB twice
hout generating candidate itemsets. Both the Apriori and FP-growth algorithms have been sho
uccessfully mine frequent patterns.

However, single item support cannot comprehensively address the complexity of items in la
asets. To apply the same single support to all items is unreasonable and insufficient. For t
son, since 1999, a considerable amount of research has focused on the MS–Apriori algori
MS–Apriori algorithm applies the concept of multiple item support values to the traditi
riori algorithm. Setting different values of support, the multiple item support (MIS) can rea
ision makers' demands for highly detailed analysis [17]. The first step in MISFP mining is
-formulate the MIS for each item and sort all items in ascending order of magnitude. When u
Apriori algorithm in FP mining, the dataset is scanned to obtain candidate itemsets.

Mining association rules with multiple item support (MIS) has become an imperative resea
ic. Hu and Chen (2006) proposed the Complete Frequent Pattern (CFP)-growth algorithm
prove the FP-growth calculation mechanism [21]. The CFP-growth algorithm creates a MIS-
t stores key messages with frequent item patterns. To improve the efficiency of the CFP-gro
orithm, the MISFP-growth algorithm discards two stages of the FP calculation, post-pruning
onstruction [22]. The MISFP-growth algorithm finds the minimum value of the multiple thresh
ues (MIN-MIS) and then drops the items that with less item support than the MIN-MIS. Thus,
rch space is reduced. A comparison of FP mining algorithms is given in Table 1.

Table 1. Comparison of frequent patterns mining algorithm.

Algorithm	Criteria	Algorithm Description
Apriori [8]	Single support	Generates candidate itemsets by scanning the database repeatedly to successfully mine the frequent patterns.
FP-growth [10]	Single support	Scans the database only twice and without generating the candidate itemsets. Creates an FP-tree from which to mine t frequent patterns. However, when the MIN updates, the FP-tree must be reconstructed.
MSApriori [17]	Multi-support	Sets different thresholds of support to identify more rare ite However, candidate itemsets are repeatedly created and thi increases memory requirements and reduces performance.
CFP-growth [21]	Multi-support	The arrangement of the position and order of items in the MIS-tree can be repeatedly adjusted by tuning the MIS. Thi

Apache Hadoop is a cluster system of open source software framework composed of a serie ction modules. The cloud service platform can store and manage big data, and is characteri scalability, reliability, flexibility, and cost-effectiveness. The platform implements MapRed :ributed parallel computing architecture on the Hadoop Distributed File System (HDFS), wl ciently analyzes and stores large datasets [23,24]. The primary Hadoop operation applies cept of "divide and conquer." Through the map and reduce functions, the problem data from le is decomposed into several smaller blocks for calculation. Then the calculation results fron les are transferred, collected, and arranged. One application of MapReduce incorporates the Apı orithm to understand the purchase requirements of customers and attract clients from compe ommerce websites [25].

In summary, this study proposes advantages of using the Hadoop's parallel computing framew vercome the drawbacks of existing single support association rule mining methods, and expre characteristics of each item. Using the MISFP-growth calculation method, we realize a two-st ution that efficiently mines frequent patterns and association rules.

Iadoop-Based Frequent Pattern Mining Architecture

To enhance the efficiency of FP mining with multiple item support, we propose an architec med the MISFP-growth algorithm) based on a distributed computing environment, know doop MapReduce. The main idea of the MISFP-growth algorithm is to split the transact asets into sub-transaction DBs according to item support, and then to analyze them respecti re specifically, transactions that contained items with identical item support would be grou orm a sub-transaction DB. Then, the FP-growth algorithm constructed an FP-tree from tl -transaction DB and then mine the frequency patterns. To further improve efficiency, we implen MISFP-growth algorithm on distributed architecture, which enables individual analysis of -transaction DBs. Finally, the analysis results from each reducer were further aggregated to gene ociation rules for the entire dataset. The proposed Hadoop MapReduce-based MISFP-grov

orithm consists of two phases, a counting support phase and an analytics phase, executed in se os, as depicted in Figure 1, and detailed below.

Step 1: data preprocessing. Step 1 cleans the original transactional DB and enables the decis
ker to set multiple item supports. As shown in ① of Figure 1, the concept of "divide and conq
pplied to the original transactional DB, which is divided into several blocks. Then, each data b
ssigned to different mapper nodes for further item support counting.

Step 2: item support counting by mapper nodes. Step 2 calculates the actual frequency for e
n. As ② of Figure 1 illustrates, each mapper node scans the sub-transaction DB for item freque
istics. In the traditional FP-growth algorithm, this operation is exceedingly time consum
expected, the proposed Hadoop-based architecture can release such loading efficiency beca
item support counting occurs in parallel. Then, the counting results from the mapper nodes
ffled to the reducer nodes in Step 3 for further aggregation.

Step 3: item support aggregation by reducer nodes. Step 3 scans the support of all items from
ppers to find the minimum of the multiple item supports (MIN-MIS). As shown in ③ of Figu
h item support is compared with the MIN-MIS, if the item support is less than MIN-MIS, ther
n is discarded.

Step 4: item sorting. The transaction is sorted in ascending order according to the support va
l formed into new data groups. As shown in ④ of Figure 1, Step 4 generates new sub-transac
s based on item supports, assigning transactions from itemsets with identical support to the s
-transaction DB and corresponding mapper nodes.

Step 5: construction of MISFP-trees by mappers. Step 5 builds a conditional MISFP-tree
h suffix item using the MISFP-growth algorithm. For each sub-transaction DB, the mappers i
put the frequency patterns that are greater than MIN-MIS, and send them to the reducer node
her aggregation.

Step 6: the aggregation of frequency pattern results by reducers. In Step 6, the reducer no
ck the merged results for duplicates and remove the repeated items from the output results.

Step 7: obtain the useful association rules. As shown in ⑥ of Figure 1, the MISFP-growth algori
n discovers association rules by the mining FP-tree.

Item support exerts a substantial influence on the results of FP mining. It is difficult but cru
et appropriate item support thresholds. Often, item support is either determined subjectively
ision makers or it is established by trial and error through recurring adjustments. However, as
aset is large, such a readjustment process becomes time-consuming with lots of loading.

To resolve the constraints and complications of multiple item support setting, we propose
cept of classification of items (COI), which categorizes items of higher homogeneity into the s
s. The main idea of COI is that the support of a specific item is equal to the support of a partic

s that the item belongs to. The concept of COI enables decision makers to set support for diffe
ns by setting class support individually. The algorithm and parameter definition of COI are lis
igure 2 and Table 2. The support of jth item is obtained by: $\sum X_{ij} \in \{1, n+1\}, \forall_j = 1, 2, \ldots, m$.

Table 2. Parameter definition of "classification of items (COI)".

Parameter	Description
C_i	C_i represents the ith product class, where $i = 1, 2, \ldots, n$.

Figure 2. Pseudo code of "classification of items (COI)".

The pseudo code of COI is shown in Figure 2. For illustration, assume that ten items (includ ss cleaner, shower gel, chocolate, whiskey, cleaning rags, cleansing milk, chewing gum, brc mpoo, and popcorn) are represented by $I_1, I_2, \ldots, I_{10}$, as shown in Table 3. There are three sup els for the product classes: C_1: home cleaning = 0.1, C_2: beauty cosmetic = 0.15, and C_3: sna .2. Take I_1 as an example: as $X_{11} = 1$ represents I_1 is categorized as the C_1 class, then the i port of I_1 is equal to 0.1. Conversely, looking at I_4 as an example: as $X_{14} = 4$, from n + 1, this m t I_4 does not belong to any product class. Therefore, decision makers must manually define i port, in this example, the item support value of $I_4 = 0.05$. Thus, COI can quickly set the mult n support thresholds. In this example, COI categorized items as follows: {I_1, I_5, I_8: $C_1 = 0.1$}, {I $C_2 = 0.15$}, {I_3, I_7, I_{10}: $C_3 = 0.2$}, and {I_4: $C_4 = 0.05$}.

Table 3. Demonstration of COI.

C_i \ I_j	I_1	I_2	I_3	I_4	I_5	I_6	I_7	I_8	I_9	I_{10}
C_1	1	0	0	n+1	1	0	0	1	0	0
C_2	0	1	0	0	0	1	0	0	1	0
C_3	0	0	1	0	0	0	1	0	0	1
$\sum X_{ij}$	1	1	1	4	1	1	1	1	1	1
Result	C_1	C_2	C_3	*[1]	C_1	C_2	C_3	C_1	C_2	C_3

[1] Note: * assigned manual.

Finally, confidence is used as an indicator to validate the meaning and reference values of ning results [26,27]. However, higher confidence does not necessarily indicate that the rule repres rong correlation. To this end, we used the *lift* indicator to evaluate the correlation and accu ween items and association rules [28].

The explanation of *lift* is as follows: the statement *lift(X,Y)* > 1 means item *X* has a posit

(X)×P(Y) indicate that item X and item Y are independent of each other. If the previous statem
ot true, then item X is related to item Y. The *lift* correlation indicator is detailed in Equation (

$$lift(X,Y) = \frac{P(X \cup Y)}{P(X)P(Y)} = \frac{Sup.(X \cup Y)}{Sup.(X)Sup.(Y)} = \frac{Confidence(X \rightarrow Y)}{Sup.(Y)}$$

Iadoop MapReduce-Based MISFP Growth Algorithm

This section demonstrates the proposed MISFP-growth algorithm with an example implementa
ıtinuing the example given in the previous section, the transactional database as shown in Table
; the actual support values and MIS of each item.

Table 4. Transaction database, actual support and multiple item supports (MIS) of each item.

(a) **(b)** **(c)**

Transaction ID. Transaction Items. Support MIS TID. Trans. (ordered)

1 B, A, C A 6 4 1 A, B, C
2 E, D, H, G, I B 6 4 2 E, G, H, I
3 C, A, B, J C 5 4 3 A, B, C, J
4 E, C, G D 1 5 4 C, E, G
5 B, A, J, I E 4 3 5 A, B, I, J
6 A, B F 1 4 6 A, B
7 G, C, F, E, J G 4 3 7 C, E, G, J
8 B, A, G H 2 2 8 A, B, G
9 H, A, I I 3 2 9 A, H, I
10 C, E, B J 3 2 10 B, C, E

As illustrated in Figure 3, first, the MISFP-growth algorithm scans the transactional datal
e to find the support value and determine the MIN-MIS = *MIN{MIS(A), MIS(B), . . . , MIS(J)}* =
wn in Table 4b. Next, the MISFP-growth algorithm sorts item {A} to item {Z} in ascending o
ording to item support to satisfy MIN-MIS. Therefore, items {D} and {F} are removed because t
port is less than MIN_MIS. The results are presented in Table 4c. The MISFP-growth algorit
ts the transaction dataset into subgroups according to MIS values and assigns the subgrouj
erent mapper nodes. Each mapper node then scans the assigned sub-transaction DB to obtain
port value of each item and construct an MISFP-tree. This process resembles that of the freque
tern tree constructed by the FP-growth algorithm.

For example, a given mapper node handles the sub-transaction DB named Block "A" for wl
n support is two. The first root is created and labeled with a null node of the MISFP-tree. Tl
litional nodes are inserted for the leading itemsets {E, G, H, I} of Block A. Figure 4 displays al
le links of each item.

Each mapper node constructs an MISFP-tree and conducts frequency pattern mining. Using
le links of item {J} as an example, we establish a conditional pattern subtree with the follov
ditional pattern base: {A, B, I :1}, {A, B, C :1}, and {C, E, G :1}. Nonetheless, the MIS of item
als two. Hence, frequency patterns involve items {A} and {B}, thus we mine (<A :2, B :2 > |J).
ılt, we mine all the frequency patterns of item {J}, which are {B, J :2}, {A, B, J :2}, {A, J:2}.

Figure 3. Demonstration of the proposed model.

Figure 4. MISFP-tree with new data groups in different mapper nodes.

Next, we combine the mined frequent patterns results obtained by all the different mapper nc plicate items are merged to obtain an aggregate output. According to the data mining out erated by this example, the results of the mining process are shown in Table 5. The most interes put of the proposed algorithm was its detection of rare patterns. General association rules usi gle item support, would also have identify itemsets like {plastic gloves, toothbrush}, {polym ss detergent, plastic gloves, toothbrush}, and {polymeric glass detergent, toothbrush}, namely it , {B}, and {J}. However, we also discovered the rare patterns of items {A} and {I}, which repre polymeric glass detergent and baking soda. This might indicate that baking soda is used to cl ss or other household surfaces.

As demonstrated in the Figure 4, the proposed model was implemented in Hadoop-ba ironment. In addition, the proposed model enables decision makers set up multiple it port. Therefore, the MISFP-growth can generate valuable patterns of rare item without nume aningless patterns.

Table 5. The results of frequent patterns.

Mapper	Suffix Item	MIN-SUP	TID [1]	Conditional Pattern Base	Conditional MISFP-Tree	Frequent Patterns
A	J	2	2, 3, 5, 7, 9	{A, B, I :1}, {A, B, C :1}, {C, E, G :1}	{A, B :2}	{B, J :2}, {A, B, J :2}, {A, J :2}
	I	2		{E, G, H :1}, {A, H :1}, {A, B :1}	{A :2}	{A, I :2}
	H	2		{E, G :1}, {A :1}	—	—
	G	3		{E :1}, {C, E :1}	—	—
	E	3		{C :1}	—	—
	C	4		{A, B :1}	—	—
	B	4		{A :2}	—	—
	A	4		—	—	—
B	J	2	2, 4, 7, 8, 10	{C, E, G :1}	—	—
	I	2		{E, G, H :1}	—	—
	H	2		{E, G :1}	—	—
	G	3		{E :1}, {A, B :1}, {C, E :2}	—	—
	E	3		{C :2}, {B, C :1}	—	—
	C	4		{B :1}	—	—
	B	4		{A :1}	—	—
	A	4		—	—	—
C	J	2	1, 3, 4, 5, 6, 7, 8, 10	{A, B, C :1}, {A, B, J :1}, {C, E, G :1}	{A, B :2}	{B, J :2}, {A, B, J :2}, {A, J :2}
	I	2		{A, B :1}	—	—
	H	2		—	—	—
	G	3		{C, E :2}	—	—
	E	3		{C :2}, {B, C :1}	—	—
	C	4		{A, B :2}, {B :1}	—	—
	B	4		{A :5}	{A :5}	{A, B :5}
	A	4		—	—	—

[1] Note: TID is new data group.

xperiments Results and Analysis

To validate the proposed architecture, three experiments were executed, which tested ameters, feasibility, and efficiency accordingly. Appropriate datasets were selected as show

sistency of mining results from stand-alone computing. The final experiment used the datase
ıil market basket dataset [30] to evaluate execution time. The results of experiments prove that

posed MISFP-growth algorithm can be implemented on the distributed computing framewoı
ɟoop MapReduce.

Table 6. Experimental design, dataset attributes and descriptions.

No.	Dataset	Volume	Items	Description
1	Groceries	9835	169	The Groceries dataset consists of transaction data froı grocery store in 2006. Each transaction represents the purchased items.
2	RIA Report Records	43,545	25	The RIA dataset consists of hospital case of radioimmunoassay (RIA) in Taiwan from 2009 to 201
3	Retail Market Basket	88,163	16,470	The dataset covers three non-consecutive periods of supermarket from 1999 to 2000 in Belgian.

The verification process was based on the virtual operating environment of Oracle VM Vir
: for parallel architectures, with three versions of the Ubuntu 12.04 LTS operating system, Apa
ɟoop 2.2.0 Cluster, and Apache Mahout 0.8. One of the nodes was set as the master node, .
› nodes were set as data nodes. Each node had a 3.0 GHz quad-core processor and 8 GB mem
acity. Through the experimental design, we quickly found frequent patterns as well as rare item

Experiment 1: COI Parameters Test

In Experiment 1 (Exp. 1), we attempted to establish the parameters of COI for the groceı
aset [29]. As shown in Table 7, we set levels of support values by product class and thus redu
time required to calculate the minimum support for each item. Each itemset belongs to a spe
duct class, and the support value of any item is equal to that of its product class. For example,
ı, "bottled beer", was classified as an "alcoholic beverage (Class 6)" at the tenth level and t
erited the 4% support value of the category.

Table 7. Support value of each product class.

ɛvels	Class 1	Class 2	Class 3	Class 4	Class 5	Class 6	Class 7	Class 8	Class 9	Class 10
10	7% 34	8% 35	9% 7	6% 30	5% 21	4% 14	2% 4	10% 15	3% 5	1% 4
9	7% 34	8% 35	9% 7	6% 30	5% 21	4% 14	10% 15	2.5% 9	1% 4	
8	7% 34	8% 35	9% 7	6% 30	5% 21	4% 14	10% 15	2% 13		
7	7% 34	8% 35	6.5% 21	6% 30	5% 21	10% 15	2% 13			
6	7% 55	8% 35	9.5% 22	6% 30	4% 14	2% 13				
5	7% [illegible]	2% [illegible]	8% [illegible]	6% [illegible]	5% [illegible]					

2	7%	3%
	122	47

To validate the proposed COI concept in this study, we evaluated the association rules produ
Exp. 1 and found them to be most representative in the third level. Apart from the rare ite
quent patterns at different levels can be found more easily. Furthermore, the average indicato
was calculated to evaluate the correlations of the association rules. Table 8 counts the associa
es mined from the tenth to second levels in Exp. 1 and presents their average *lift*. According to
ults of Exp. 1 shown in Table 8, the third level is the optimum parameter for COI because this l

duced more meaningful FP rules from the groceries dataset. Therefore COI = 3 was adopted
periments 2 and 3.

Table 8. Results of Experiment 1.

Level	Rules	Lift(avg.)	Level	Rules	Lift(avg.)
10	139	1.670	5	51	1.632
9	22	1.574	4	39	1.608
8	23	1.490	3	30	1.762
7	23	1.619	2	19	1.643
6	19	1.597			

Experiment 2: Feasibility Test

In the Experiment 2 (Exp. 2), using the simple dataset mentioned previously [21] and show
le 9a, the MIS thresholds in Table 9b and the RIA dataset were adopted to verify the feasib
he proposed architecture. The verification process was performed on a virtual machine wi
GHz quad-core processor and 8 GB memory capacity. One node was set as the master node,
remaining two nodes were set as data nodes.

Table 9. Transaction database, actual support, and MIS of each item for Experiment 2.

(a)

TID.	Transaction
1	A, C, D, F
2	A, C, E, F, G
3	A, B, C, F, H
4	B, F, G
5	B, C

(b)

Items.	Support	MIS
A	3	4(80%)
B	3	4(80%)

G	2	2(40%)
H	1	2(40%)

In the first part of Exp. 2, the mined association rules {G, F}, {F, C}, {F, C, A}, {F, B}, {F, A} w sistent for both the CFP-growth algorithm and the MISFP-growth algorithm. In the second Exp. 2, 25 items (medical orders) were attributed to the third level of COI, including the horm , the hepatitis test, and the tumor marker test. Supports for three COIs were suggested by ional Health Administration and based on the occurrence of cancer in Taiwan. The weighti ankings were set according to the actual frequency of occurrences in the dataset. Similar it e assigned to the same class, and thus inherited the support value of their COI class, as show ure 5. In the second part of Exp. 2, the medical orders {ABC, ABE, ACHR, ACIGM, B2M, CA 153, FBHCG, FPSA, HAIGM, HAV, HBE, SCC, T3UP, TG, and THY} were abandoned becaus y did not satisfy the MIN threshold. The meaningful association rules {TSH, FT4}, {CEA, CA e consistent with Wang et al.'s mining association rules [31]. The average lift value was 3.32 ociation rules exhibited a positive correlation. These results prove that the proposed model ca lied to the mine association rules with multiple item support thresholds.

Figure 5. Support threshold value of stand-alone operation.

Experiment 3: Efficiency Test

In Experiment 3 (Exp. 3), we tested the efficiency of the proposed algorithm on the retail ma ket dataset [30]. The verification process was implemented in five iterations. The experim ironment consisted of a virtual machine with an Ubuntu 12.04 LTS operating system, Apa

uction of approximately 38% compared to the average execution time of 19,586 ms attained by
ıd-alone operation.

Figure 6. Efficiency test of Experiment 3.

These three experiments validated the proposed model, successfully applying the concep
ltiple item support thresholds to solve the problems of previous methods that use a single sup
ıe. In Exp. 1, the COI concept enabled rapid determination of support thresholds for each it
. the indicator of *lift* showed that the meaningful association rules had a positive correlation or
d level. Experiment 2 verified the feasibility of the proposed model as it identified meanin
ɔciation rules that were consistent with the association rules mined in the past research. In Ex

average execution time of the parallel architecture demonstrated the improved efficiency of
posed model compared to the conventional stand-alone architecture.

)iscussion and Conclusions

Big data analytics is changing our daily lives, and increased application of data-driven deci
king is inevitable, as the rapid growth in data generation produces new opportunities to ext
aningful information from big data. However, the traditional method of association rule mi
n frequent patterns using a single support threshold is not sufficient for today's complex probl
l decision making processes. In addition, the efficiency of data analysis must increase to adap
rapid growth in data generation.

In this study, a MISFP-growth algorithm consisting of two phases, a counting support phase
ining phase, is proposed to realize high-efficiency mining of frequency patterns with multiple
ports. To assist decision makers in setting multiple support values for items, we also propo
concept of the classification of items (COI), which categorizes items of higher homogeneity
same product class from which the items then inherit their data support threshold. Finall
relation indicator, named *lift*, is adopted to evaluate the meaning and accuracy between items
ɔciation rules. Furthermore, the MISFP-growth algorithm was implemented without the prun
l reconstruction steps on the distributed computing framework of Hadoop MapReduce.

t the same analysis results were obtained from both the stand-alone and parallel architectu
eriment 3 demonstrated an approximately 38% reduction in the execution time of MISFP-gro
orithm on parallel architectures. Thus, the results of these experiments confirm that the propo
orithm achieves high-efficiency big data analytics for frequency pattern mining with multiple i
ports. In future work, we expect to perform cross-validation by implementing the proposed m
arious fields and applications.

hor Contributions: C.S. Wang conceived and designed the experiments; J.Y. Change performed
eriments; C.S. and J.Y. analyzed the data and wrote the paper."

ding: This research received no external funding.

nowledgments: This research was partially supported by the laboratory of enterprise resource planning
experiments.

flicts of Interest: The authors declare no conflict of interest.

erences

eavitt, N. Storage challenge: Where will all that big data go? *Computer* **2013**, *9*, 22–25. [CrossRef]
Comuzzi, M.; Patel, A. How organisations leverage big data: A maturity model. *Ind. Manag. Data Syst.* 2
116, 1468–1492. [CrossRef]
Wu, X.; Zhu, X.; Wu, G.Q.; Ding, W. Data mining with big data. *IEEE Trans. Knowl. Data Eng.* **2014**
97–107.
Fournier-Viger, P.; Lin, J.C.W.; Kiran, R.U.; Koh, Y.S.; Thomas, R. A survey of sequential pattern mining. *I*
Science and Pattern Recognition. **2017**, *1*, 54–77.
Choi, T.M.; Chan, H.K.; Yue, X. Recent development in big data analytics for business operations and
management. *IEEE Trans. Cybern.* **2017**, *47*, 81–92. [CrossRef]
Le, T.; Vo, B. An N-list-based algorithm for mining frequent closed patterns. *Expert Syst. Appl.* **2015**
6648–6657. [CrossRef]
Guil, F.; Marín, R. A theory of evidence-based method for assessing frequent patterns. *Expert Syst. A*
2013, *40*, 3121–3127. [CrossRef]
grawal, R.; Srikant, R. Fast algorithms for mining association rules. In Proceeding of the Very Large Dat
Bases, Santiago de Chile, Chile, 12–15 September 1994; Volume 1215, pp. 487–499.

Agrawal, R.; Mannila, H.; Srikant, R.; Toivonen, H.; Verkamo, A.I. Fast discovery of association ru
Adv. Knowl. Discov. Data Min. **1996**, *12*, 307–328.
Han, J.; Pei, J.; Yin, Y. Mining frequent patterns without candidate generation. *ACM Sigmod Rec.* **200**
1–12. [CrossRef]
Han, J.; Pei, J.; Yin, Y.; Mao, R. Mining frequent patterns without candidate generation: A frequent-pat
tree approach. *Data Min. Knowl. Discov.* **2004**, *8*, 53–87. [CrossRef]
Hu, Y.H.; Wu, F.; Liao, Y.J. An efficient tree-based algorithm for mining sequential patterns with mult
minimum supports. *J. Syst. Softw.* **2013**, *86*, 1224–1238. [CrossRef]
Chen, S.S.; Huang, T.C.K.; Lin, Z.M. New and efficient knowledge discovery of partial periodic patterns
multiple minimum supports. *J. Syst. Softw.* **2011**, *84*, 1638–1651. [CrossRef]
Lee, Y.C.; Hong, T.P.; Lin, W.Y. Mining association rules with multiple minimum supports using maxin
constraints. *Int. J. Approx. Reason.* **2005**, *40*, 44–54. [CrossRef]

Syst. **2000**, *15*, 47–55.
Liu, B.; Hsu, W.; Ma, Y. Mining association rules with multiple minimum supports. In proceeding
Knowledge discovery and data mining, San Diego, CA, USA, 15–18 August 1999; pp. 337–341.
Fadnavis, R.A.; Tabhane, S. Big data processing using Hadoop. *Int. J. Comput. Sci. Inf. Technol.* **2015**, *6*, 443
Saabith, A.S.; Sundararajan, E.; Bakar, A.A. Parallel implementation of Apriori algorithms on Hadoop-MapReduce platform-an evaluation of literature. *J. Theor. Appl. Inf. Technol.* **2016**, *85*, 321–351.
Li, N.; Zeng, L.; He, Q.; Shi, Z. Parallel implementation of apriori algorithm based on mapred
In Proceedings of the International Conference on Software Engineering, Artificial Intelligence, Networ
and Parallel/Distributed Computing, Kyoto, Japan, 8–10 August 2012; pp. 236–241.
Hu, Y.H.; Chen, Y.L. Mining association rules with multiple minimum supports: A new mining algori
and a support tuning mechanism. *Decis. Support Syst.* **2006**, *42*, 1–24. [CrossRef]
Darrab, S.; Ergenç, B. Frequent pattern mining under multiple support thresholds. *Wseas Trans. Comput.* **2016**, *4*, 1–10.
Gan, W.; Lin, J.C.W.; Chao, H.C.; Zhan, J. Data mining in distributed environment: a survey. *Wiley Interdi
Rev. Data Min. Knowl. Discovery* **2017**, *7*, 1–19. [CrossRef]
Dean, J.; Ghemawat, S. MapReduce: Simplified data processing on large clusters. *Commun. ACM* **2008**
107–113. [CrossRef]
Verma, N.; Singh, J. An intelligent approach to Big Data analytics for sustainable retail environment u
Apriori-MapReduce framework. *Ind. Manag. Data Syst.* **2017**, *117*, 1503–1520. [CrossRef]
Agrawal, R.; Imieli ński, T.; Swami, A. Mining association rules between sets of items in large datab
ACM Sigmod Rec. **1993**, *22*, 207–216. [CrossRef]
Wu, Y.H.; Chang, M.Y.C.; Chen, A.L. Discovering phenomena-correlations among association rules. *J. Inte
Technol.* **2006**, *7*, 1–11.
Brin, S.; Motwani, R.; Silverstein, C. Beyond market baskets: Generalizing association rules to correlati
ACM Sigmod Rec. **1997**, *26*, 265–276. [CrossRef]
Hahsler, M.; Hornik, K.; Reutterer, T. Implications of probabilistic data modeling for mining association r
In *From Data and Information Analysis to Knowledge Engineering, Studies in Classification, Data Analysis,
Knowledge Organization*; Spiliopoulou, M., Kruse, R., Borgelt, C., Nuernberger, A., Gaul, W., Eds.; Sprir
Berlin, Germany, 2006; pp. 598–605.
Brijs, T. Retail market basket data set. In Proceedings of the Frequent Itemset Mining Implementati
(FIMI), Melbourne, FL, USA, 19 November 2003; pp. 1–4.
Wang, C.S.; Lin, S.L.; Chiu, H.C.; Juan, C.J.; He, X.Y. Is a medical examination necessary? Analysis of mec
examination transactions through association mining using multiple minimum supports. *J. Med. Imag
Health Inform.* **2017**, *7*, 1399–1408. [CrossRef]

Recommended publications

Article Full-text available

The method and application of big data mining for mobile trajectory of taxi based on MapReduce

January 2018 · Cluster Computing

Fansheng Kong · Xiaola Lin

With the rapid development of urbanization, the imbalance between supply and demand of taxi is becoming more and more serious. Therefore, in this paper, the method and application of large data mining of taxi moving trajectory based on MapReduce were proposed. Firstly, the important role of the large data analysis and excavation of the taxi mobile trajectory in various fields and the related ... [Show full abstract]

Article

Mining large-scale repetitive sequences in a MapReduce setting

January 2016 · International Journal of Data Mining and Bioinformatics

Hongfei Cao · Michael Phinney · Devin Petersohn · [...] · Chi-Ren Shyu

Recent research suggests DNA repeats play critical roles in cellular regulatory functions and disease development. The challenge associated with identifying repeats across a collection of genomes arises from the amount of data stored within DNA, and intermediate data generated by alignment- and hash-based approaches are substantial. We present a MapReduce-based method for repeat identification ... [Show full abstract]

October 2017 · Journal of Medical Systems

Chao-Tung Yang · jung-chun Liu · Shuo-Tsung Chen · Hsin-Wen Lu

Big Data analysis has become a key factor of being innovative and competitive. Along with population growth worldwide and the trend aging of population in developed countries, the rate of the national medical care usage has been increasing. Due to the fact that individual medical data are usually scattered in different institutions and their data formats are varied, to integrate those data that ... [Show full abstract]

Chapter

An Innovative Framework for Supporting Frequent Pattern Mining Problems in IoT Environments

July 2018

Peter Braun · Alfredo Cuzzocrea · Carson Leung · [...] · Giorgio Mario Grasso

In the current era of big data, high volumes of a wide variety of data of different veracity can be easily generated or collected at a high velocity from rich sources of data include devices from the Internet of Things (IoT). Embedded in these big data are useful information and valuable knowledge. Hence, frequent pattern mining and its related research problem of association rule mining, which ... [Show full abstract]