DJILLALI LIABES UNIVERSITY OF SIDI BEL ABBES
FACULTY OF EXACT SCIENCES
DEPARTMENT OF COMPUTER SCIENCES

Module : Apprentissage Automatique
1ST YEAR OF MASTER'S DEGEREE IN
NETWORKS,SYSTEMS & INFORMATION SECURITY(RSSI)
2021/2022

Règles d'associations Algorithme Apriori

Student:
HADJAZI Mohammed Hisham
Group: 01 / RSSI

Module Instructor:
Pr.ELBERRICHI Zakaria
TD Instructor:
Dr.FAHSI.Mahmoud

A paper submitted in fulfilment of the requirements for the
Apprentissage Automatique TD-06

December 29, 2021

Contents

List of Figures

Chapter 1

Fiche TD-06 Solutions

1.1 Questions de cours.

1.1.1 1. Donner le principe de l'algorithme Apriori ainsi que ses avantages et inconvénients ?

In 1994 the Apriori algorithm was developed by R. Srikant and R. Agrawal. The term 'Apriori' means with prior knowledge and is relevant to what the algorithm does. The algorithm's Apriori principle approach is to do level-wise searches. Where k is the number of itemsets with a specific property, by repetitive iterations the algorithm which is familiar with the specific property finds the k1 frequent dataset in the most popular use of the Apriori algorithm.

How it works ?

Apriori algorithm is a sequence of steps to be followed to find the most frequent itemset in the given database. This data mining technique follows the join and the prune steps iteratively until the most frequent itemset is achieved. A minimum support threshold is given in the problem or it is assumed by the user.

1. In the first iteration of the algorithm, each item is taken as a 1-itemsets candidate. The algorithm will count the occurrences of each item.

2. Let there be some minimum support, minsup . The set of 1-itemsets whose occurrence is satisfying the min sup are determined. Only those candidates which count more than or equal to minsup, are taken ahead for the next iteration and the others are pruned.

3. Next, 2-itemset frequent items with minsup are discovered. For this in the join step, the 2-itemset is generated by forming a group of 2 by combining items with itself.

4. The 2-itemset candidates are pruned using min-sup threshold value. Now the table will have 2-itemsets with min-sup only.

5. The next iteration will form 3-itemsets using join and prune step. This iteration will follow antimonotone property where the subsets of 3-itemsets, that is the 2-itemset subsets of each group fall in minsup. If all 2-itemset subsets are frequent then the superset will be frequent otherwise it is pruned.

6. Next step will follow making 4-itemset by joining 3-itemset with itself and pruning if its subset does not meet the minsup criteria. The algorithm is stopped when the most frequent itemset is achieved.

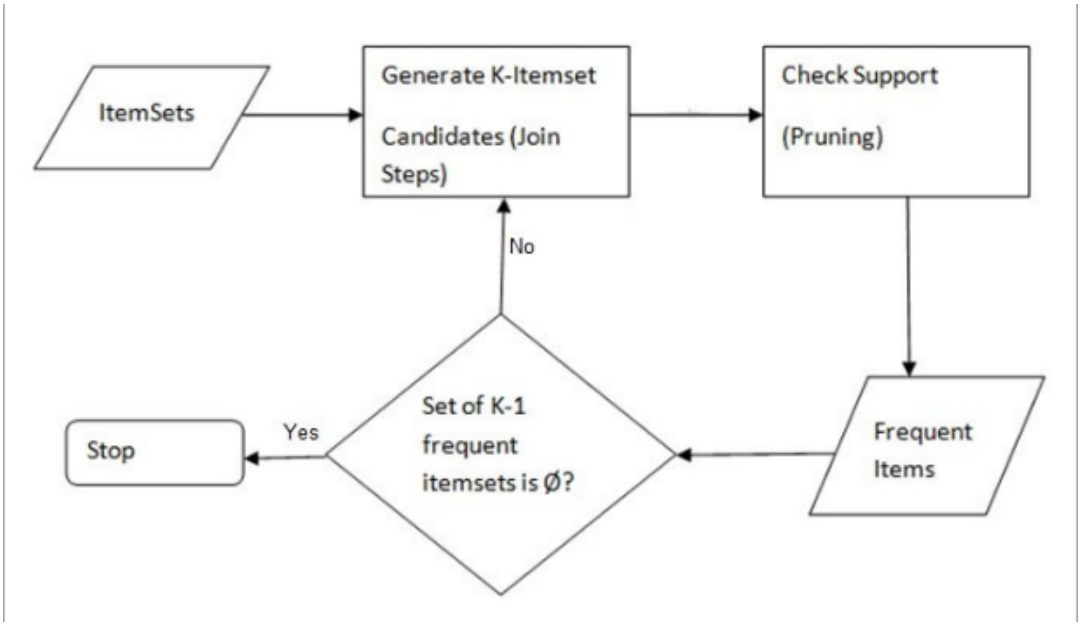

FIGURE 1.1: Apriori algorithm

avantages

1. This is the most simple and easy-to-understand algorithm among association rule learning algorithms
2. The resulting rules are intuitive and easy to communicate to an end user
3. It doesn't require labeled data as it is fully unsupervised; as a result, you can use it in many different situations because unlabeled data is often more accessible
4. Many extensions were proposed for different use cases based on this implementation for example, there are association learning algorithms that take into account the ordering of items, their number, and associated timestamps
5. The algorithm is exhaustive, so it finds all the rules with the specified support and confidence

inconvénients

One of the biggest limitations of the Apriori Algorithm is that it is slow. This is so because of the bare decided by the :

1. A large number of itemsets in the Apriori algorithm dataset.
2. Low minimum support in the data set for the Apriori algorithm.
3. The time needed to hold a large number of candidate-sets with many frequent itemsets.
4. Thus it is inefficient when used with large volumes of datasets.

1.1.2 2. Donner quelques domaines d'applications de l'algorithme.

1. **In Education Field**: Extracting association rules in data mining of admitted students through characteristics and specialties.
2. **In the Medical field**: For example Analysis of the patient's database.
3. **In Forestry**: Analysis of probability and intensity of forest fire with the forest fire data.
4. Apriori is used by many companies like Amazon **in the Recommender System** and by Google for the auto-complete feature.

1.1.3 3. Nous avons 2 ensembles de règles A et B :

A : Si X et Y Alors Z et T B : Si X et Y Alors Z , Si X et Y Alors T

Qui implique l'autre ? Vous supposez A interessant (Sup > Suiel, conf > suiel), est ce que automatiquement B est interessant, ou l'inverse, ou equivalence ?

I think **A** can be either include **B** or be equal to it. as it shows that Z and T are together and not separated.

1.2 Exercice 2: En utilisant l'algorithme Apriori, trouver les regles d'associations de seuils support = 3 et confiance 80%

N° Ticket	Items
1	F M B E
2	O F E
3	O A H S
4	D B F E
5	D A H S
6	O M E
7	O A D H S

C1	
ItemSet	Sup Count
A	3
B	2
D	3
E	4
F	2
H	3
O	3
S	3
M	2

L1	
ItemSet	Sup Count
A	3
D	3
E	4
H	3
O	3
S	3

C2	
ItemSet	Sup Count
AD	2
AE	0
AH	3
AO	2
AS	3
DE	1
DH	2
DO	0
DS	2
EH	0
EO	2
ES	0
HO	2
HS	3
OS	2

L2	
ItemSet	Sup Count
AH	3
AS	3
HS	3

C3	
ItemSet	Sup Count
AHS	3

Association Rules:			
A = (HS)	3 / 3	100.00%	Accepted
H = (AS)	3 / 3	100.00%	Accepted
S = (HS)	3 / 3	100.00%	Accepted
AH = (S)	3 / 3	100.00%	Accepted
AS = (H)	3 / 3	100.00%	Accepted
HS = (A)	3 / 3	100.00%	Accepted

1.3 Exercice 3 : Utilisez l'algorithme Apriori, pour trouver les règles d'associations de min support = 3 et de (min) confiance = 100%, à partir du tableau suivant :

TransId	Items
T1	ABCFH
T2	CF
T3	BCDFG
T4	AFH
T5	BCEFG
T6	ADEFH

C1	
ItemSet	Sup Count
A	3
B	3
C	4
D	2
E	2
F	6
G	2
H	3

L1	
ItemSet	Sup Count
C	4
F	6
H	3

C2	
ItemSet	Sup Count
AF	3
AH	3
BC	3
BF	3
CF	4
FH	3

L2	
ItemSet	Sup Count
AF	3
AH	3
BC	3
BF	3
CF	4
FH	3

C3	
ItemSet	Sup Count
AFH	3
BCF	3

Association Rules:			
A = (FH)	3 / 3	100.00%	Accepted
F = (AH)	3 / 6	50.00%	Rejected
H = (AF)	3 / 3	100.00%	Accepted
AF = (H)	3 / 3	100.00%	Accepted
AH = (F)	3 / 3	100.00%	Accepted
FH = (A)	3 / 3	100.00%	Accepted
B = (CF)	3 / 3	100.00%	Accepted
C = (BF)	3 / 4	75.00%	Rejected
F = (BC)	3 / 6	50.00%	Rejected
BC = (F)	3 / 3	100.00%	Accepted
BF = (C)	3 / 3	100.00%	Accepted
CF = (B)	3 / 4	75.00%	Rejected

1.4 Exercice 4 : En utilisant l'algorithme Apriori, trouver les règles d'associations de min support = 3 et de (min) confiance = 100%, à partir du tableau suivant :

TransId	Items
T1	A B C F
T2	C F
T3	B C D F G
T4	A F
T5	B C E F G

C1	
ItemSet	Sup Count
B	3
C	4
F	5

L1	
ItemSet	Sup Count
B	3
C	4
F	5

C2	
ItemSet	Sup Count
BC	3
BF	3
CF	4

L2	
ItemSet	Sup Count
BC	3
BF	3
CF	4

C3	
ItemSet	Sup Count
BCF	3

Association Rules:			
B = (CF)	3 / 3	100.00%	Accepted
C = (BF)	3 / 4	75.00%	Rejected
F = (BC)	3 / 5	60.00%	Rejected
BC = (F)	3 / 3	100.00%	Accepted
BF = (C)	3 / 3	100.00%	Accepted
CF = (B)	3 / 4	75.00%	Rejected

1.5 Exercice 5 : En utilisant l'algorithme Apriori, trouver les règles d'associations de min support = 3 et de (min) confiance = 100%, à partir du tableau suivant :

TransId	Items
T1	C F G
T2	A B C F G
T3	D F G
T4	A B C F
T5	B C E F

C1	
ItemSet	Sup Count
A	2
B	3
C	4
D	1
E	1
F	5
G	3

L1	
ItemSet	Sup Count
B	3
C	4
F	5
G	3

C2	
ItemSet	Sup Count
BC	3
BF	3
BG	1
CF	3
CG	2
FG	3

L2	
ItemSet	Sup Count
BC	3
BF	3
CF	3
FG	3

C3	
ItemSet	Sup Count
BCF	3
BCFG	1
BFG	1
CFG	2

L3	
ItemSet	Sup Count
BCF	3

Association Rules:			
B = (CF)	3 / 3	100.00%	Accepted
C = (BF)	3 / 4	75.00%	Rejected
F = (BC)	3 / 5	60.00%	Rejected
BC = (F)	3 / 3	100.00%	Accepted
BF = (C)	3 / 3	100.00%	Accepted
CF = (B)	3 / 3	100.00%	Accepted

Bibliography

[1] URL: https://www.jigsawacademy.com/blogs/data-science/apriori-algorithm/.

[2] URL: https://www.softwaretestinghelp.com/apriori-algorithm/.

[3] URL: https://www.engati.com/glossary/apriori-algorithm.