FOUNDATION

POORNIMA FOUNDATION

LECTURE NOTES

Campus: PCE Course: BTECH in CSE Class/Section: III Yr. Section- A Date: 9/3/21

Name of Faculty: Praveen Kumar Yadav Name of Subject: Machine Learning Code: 6CS4-02

Date (Prep.): 9/3/21 Date (Del.): 13/9/21 Unit No.: II Lect. No: 4+5

OBJECTIVE: To be written before taking the lecture (Pl. write in bullet points the main topics/concepts etc., which will be taught in this lecture)

FP- Growth Algorithm.

IMPORTANT & RELEVANT QUESTIONS:

what is FP-Tree? why FP-Growth algo^m is required.

FEED BACK QUESTIONS (AFTER 20 MINUTES):

what are the Advantages of FP-Growth algo^m

OUTCOME OF THE DELIVERED LECTURE: To be written after taking the lecture (Pl. write in bullet points about students' feedback on this lecture, level of understanding of this lecture by students etc.)

Good

REFERENCES: Text/Ref. Book with Page No. and relevant Internet Websites:

Sikkit with ML.

PAGE NO.

FP-Growth Algorithm:-

- FP stands for Frequent pattern.
- This Algo^m is an improvement to the Apriori method.
- A priori method needs a generation of candidate itemsets. These itemset may be large in number if the itemsets in the DB is large.
- Apriori needs multiple scans of the DB to check the support of each itemset generated and this leads to high costs.

Algo^m - Here a frequent pattern is generated without the need of candidate generation. It represents the database in the form of a tree called a frequent pattern tree or FP Tree.

Purpose of FP Tree is to mine the most frequent pattern. Each node of FP Tree represents an item of the itemset.

Confidence

FP Tree :-

↳ root node represents null while lower node represents the itemset. The association of the nodes with the lower nodes i.e. itemset with other itemset are maintained while forming the tree.

FP-Algo^m steps:-

1) To scan the DB to find the freq of itemset in DB (i.e support count)

2) Construct FP tree. (Here root is represented by NULL).

3). Scan the DB again and examine the Transaction again. Examine the Transaction and find out the itemset in it. The itemset is maximum is taken at top, the next itemset with lower count and so on

4) The next Transaction in DB is examined. The itemset are ordered in descending order of count. If any itemset is already present in another branch, then this transaction would share common prefix of the root

5). count of the itemset is incremented as it occurs in the transactions. (Both common and new node counter increased by one)

6) mine the created FP-Tree.

7). Construct a Conditional FP Tree. formed by a count of itemset in the path. The itemset meeting the threshold support are considered in conditional FP Tree.

PAGE NO.

Eg of FP-growth algo:-

Transaction	List of items
T1	I1, I2, I3
T2	I2, I3, I4
T3	I4, I5
T4	I1, I2, I4
T5	I1, I2, I3, I5
T6	I1, I2, I3, I4

given support threshold = 60%
confidence = 60%
so min support = 3

Step 1 - count of each item

Item	Count
I1	4
I2	5
I3	4
I4	4
I5	2

< support count (eliminated)

Step 2- Sort the itemset in descending order.

Item	count
I2	5
I1	4
I3	4
I4	4

eg-

Transaction	List of items
T1	I1, I2, I5
T2	I2, I3, I4
T3	I4, I5
T4	I1, I2, I4
T5	I1, I2, I3, I5
T6	I1, I2, I3, I4

count of each item

I_1	4
I_2	5
I_3	4
I_4	4
I_5	2

support = 2

save the items in descending order

Item	Count
I2	5
I1	4
I3	4
I4	4

Build FP tree-

T_1 - $I_1, I_2, I_5 \rightarrow [I_2\ I_1, I_3]$

- I_2 - ~~I_2~~ ~~I_2~~ ~~I_2~~ ~~I_2~~ ~~I_2~~ 4 5
  (T_2) (T_4) T_5 T_6

I_1 - ~~I_1~~ 2 ~~2~~ ~~2~~ 4
[T_4] [T_5] T_6

I_3 - ~~I_3~~ 2 3
T_5 T_6

T_2 - $I_2\ I_3\ I_4 \rightarrow [I_2\ I_3\ I_4]$

I_3 - 1

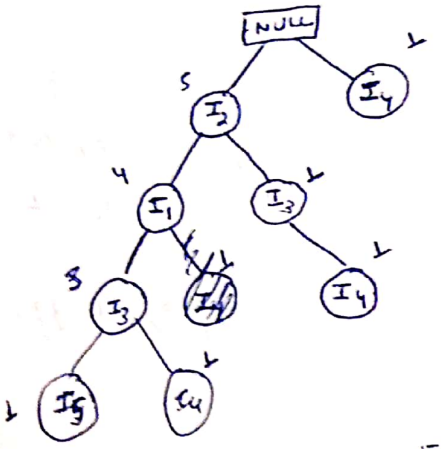
I_4 - 1

T_3 - $I_4\ I_5$

I_4 - 1

T_4 $I_1\ I_2, I_4 \rightarrow [I_2\ I_1\ I_4]$

I_4 - 1



T_5 - $I_1\ I_2\ I_3\ I_5 \rightarrow [I_2\ I_1\ I_3\ I_5]$

I_5 - 1

T_6 - $I_1\ I_2\ I_3\ I_4 \rightarrow [I_2\ I_1\ I_3\ I_4]$

I_4 ← 1

POORNIMA
FOUNDATION

LECTURE NOTES

[illegible]

Name of Subject: Machine Learning

OBJECTIVE: To be written before taking the lecture (P): write in bullet points the main topics/concepts etc., which will be taught in the lecture.

FP-Growth Algorithm (Numerical Implementation)

IMPORTANT & RELEVANT QUESTIONS:

What is FP Tree?

FEED BACK QUESTIONS (AFTER 20 MINUTES):

How to build FP-Tree?

OUTCOME OF THE DELIVERED LECTURE: To be written after taking the lecture (P): write in bullet points about students' feedback on the lecture, level of understanding of the lecture by students etc.)

Good

REFERENCES: Text/Ref. Books with Page No. and relevant Internet Websites:

ML with Python, Towards Data Science

Psuedo Code For FP-Growth Algorithm:-

Algorithm:- FP-Tree Construction

Input - A transactional database, min_support

output - The complete set of frequent patterns.

Steps:- a) scan the transactional database D once. collect the set of frequent items F and their support.

b) sort F in support descending order as L (list), i.e list of frequent items

c) create the root of an FP-tree and label it as "null". For each Transaction in D do the following-

i) select and sort the frequent items in Transaction to the order of L.

(let sorted frequent item list in Trans be [p|P]

where p = first element and P is remaining list.

ii) call insert_tree ([p|P], T) as follows-

If T has a child N such that

If N.item-name = P.item-name, then increment N count by 1.

else create a new node N and its count be 1 and its parent link be linked to T.

iii) its node links to the nodes with same item-name via node-link structure.

iv) if P is non-empty, call insert_tree (P, N) recursively.

Campus: PCE Course: BTECH | Class/Section: III CSE-D | Date: 11/3/21

Name of Faculty: Praveen Kumar Yadav | Name of Subject: ML | Code: 6CS705

FP-GROWTH ALGORITHM :-

Numerical Implementation: Find the frequent itemsets in D data given

Tid	List of item sets in transaction Id
T100	I_1, I_2, I_5
T200	I_2, I_4
T300	I_2, I_3
T400	I_1, I_2, I_4
T500	I_1, I_3
T600	I_2, I_3
T700	I_1, I_3
T800	I_1, I_2, I_3, I_5
T900	I_1, I_2, I_3

Tid	List-items
T100	I_2, I_1, I_5
T200	I_2, I_4
T300	I_2, I_3
T400	I_2, I_1, I_4
T500	I_1, I_3
T600	I_2, I_3
T700	I_1, I_3
T800	I_2, I_1, I_3, I_5
T900	I_2, I_1, I_3

Step 1- Find the support_count of each distinct data items or item set in given set D. i.e distinct items

	Support_count (Support)
I_1	6
I_2	7
I_3	6
I_4	2
I_5	2

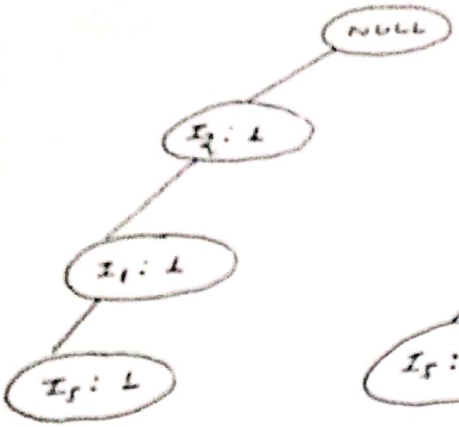
i) Arrange all items in descending order

	Support count (Support)
I_2	7
I_1	6
I_3	6
I_4	2
I_5	2

Step-3 Construction of FP-Tree for each Individual Transaction

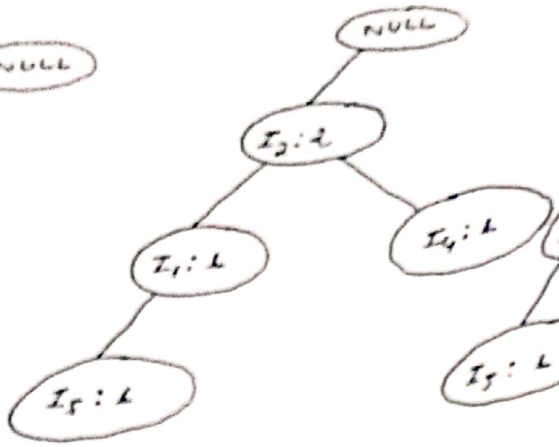
→ add a NULL node as a root of the tree i.e

For T-100



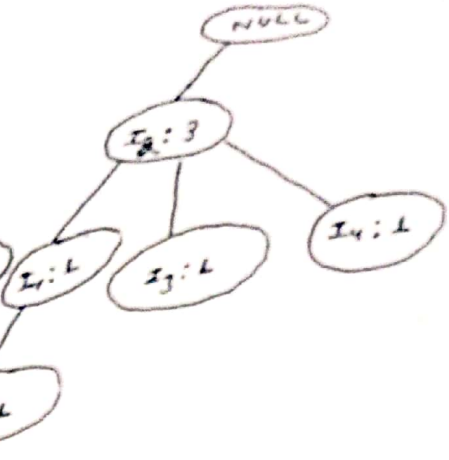
T1

For T-200 -



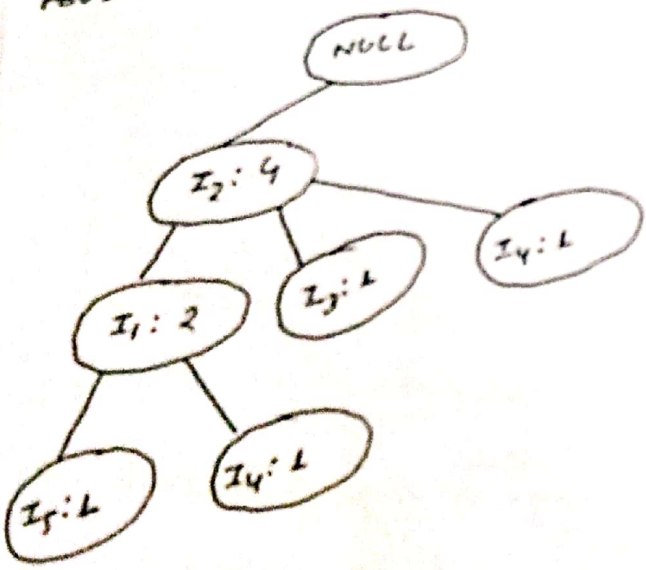
T2

For T-300 -



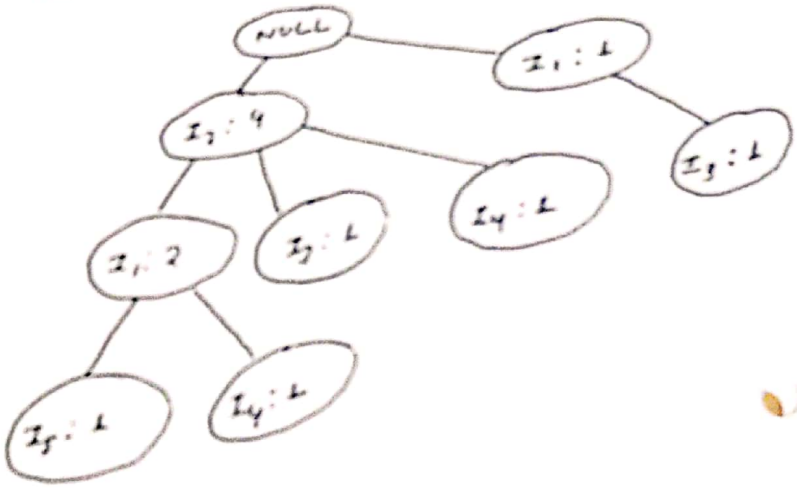
T3

For T-400



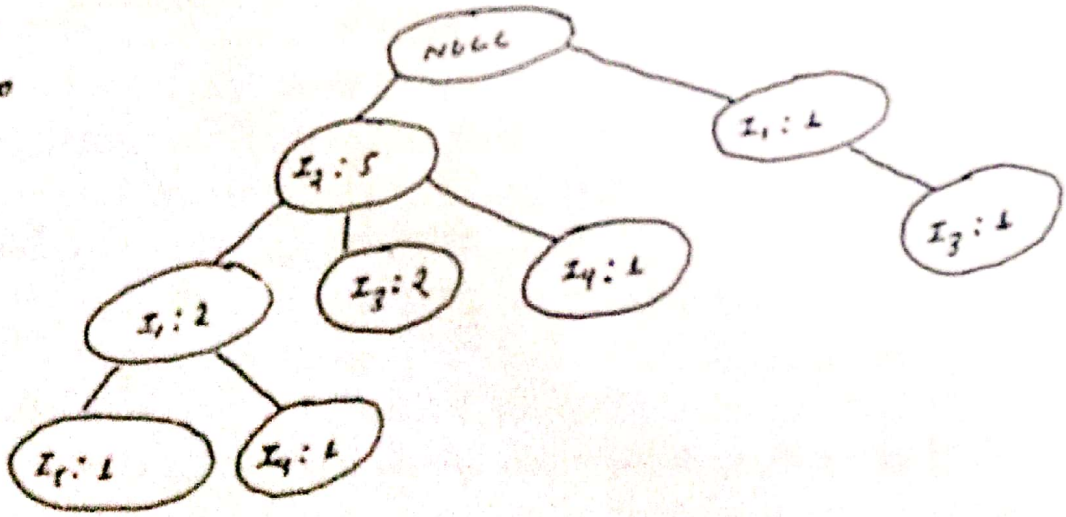
T4

For T-500



T5

For T-600



Campus: PCE. Course: BTECH

Name of Faculty: Praveen Kumar Yadav

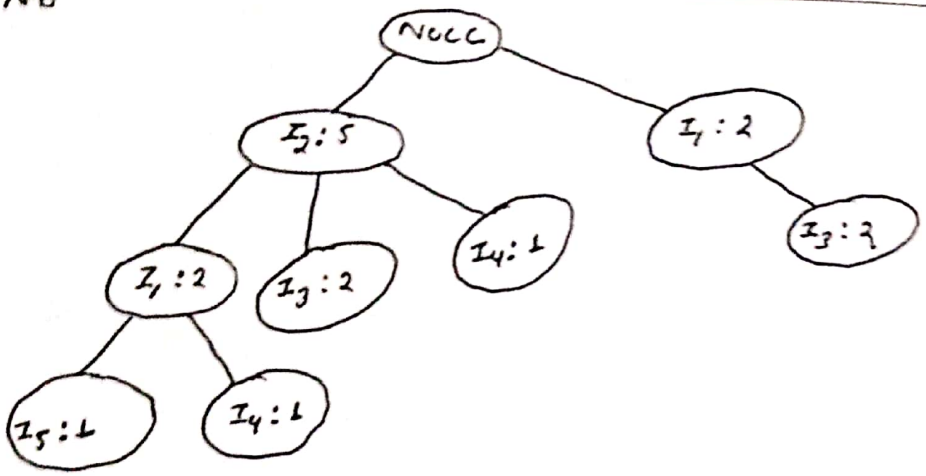
Class/Section: IV 4th Sem A

Name of Subject: ML

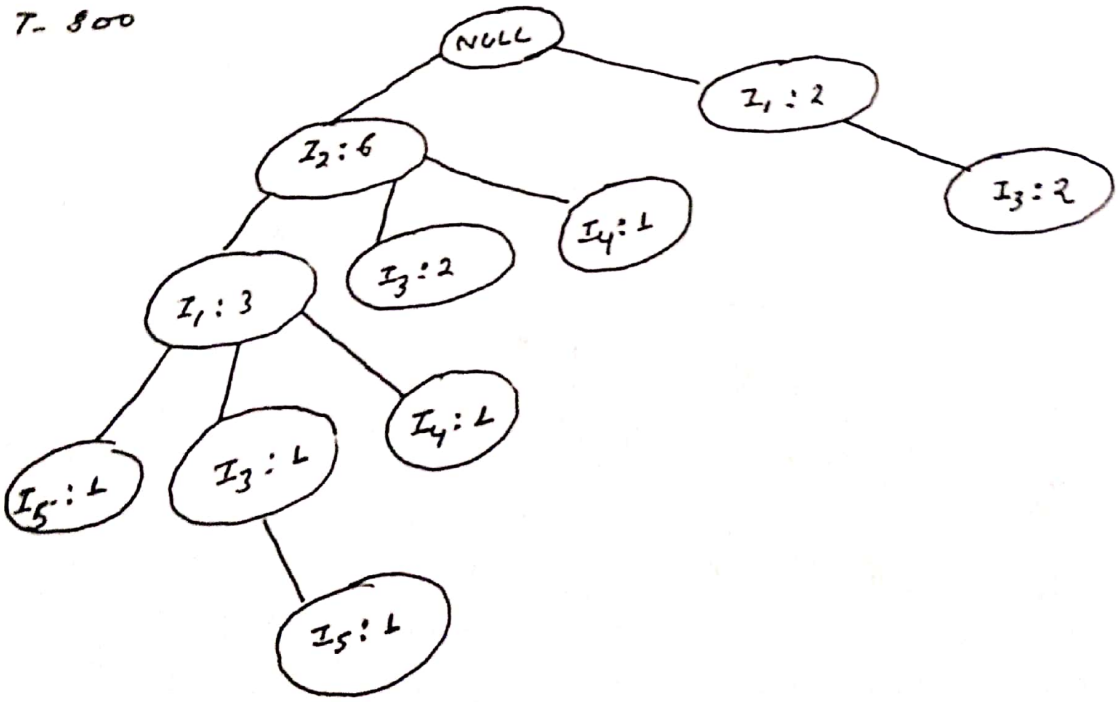
Date: 11/5/21

Code: 5CS4-ML

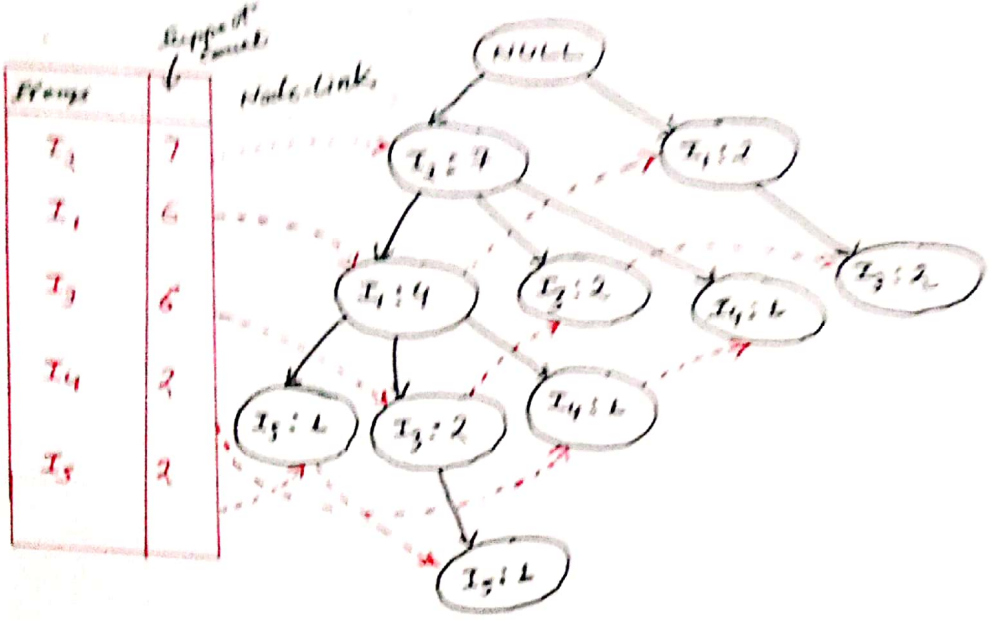
For - T-700



For - T-800



for Transaction Page 1



Final FP-tree that registers compressed, frequent pattern information.

Step-2 Mining the FP-tree by creating conditional (sub) pattern base.

Item	Conditional pattern base
I_5	$\{(I_2\ I_1 : 1), (I_2, I_1, I_3 : 1)\}$
I_4	$\{(I_2\ I_1 : 1), (I_2 : 1)\}$
I_3	$\{(I_2, I_1 : 2), (I_2 : 2), (I_1 : 2)\}$
I_1	$\{(I_2 : 4)\}$

$I_5 : 2$

$I_4 : 2$

$I_3 : 6$

$I_1 : 6$

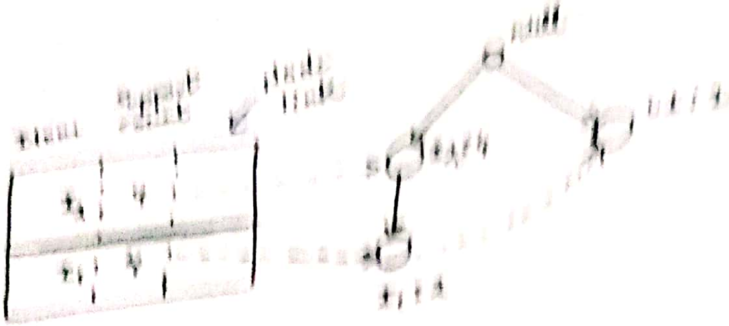
Campus: PCE. Course: BTECH
Name of Faculty: Praveen Kumar Yadav
Class/Section: IV [illegible] Sem A
Name of Subject: ML
Date: 8/12/21
Code: CS4-02

Step-3 Generate Conditional FP-Tree. (Min-Support = 2)

Item	Conditional pattern Base	Conditional FP-Tree
I_5	$\{(I_2\ I_1 : 1), (I_2, I_1, I_3 : 1)\}$	$\{I_2 : 2, I_1 : 2\}$
I_4	$\{(I_2\ I_1 : 1), (I_2 : 1)\}$	$\{I_2 : 2\}$
I_3	$\{(I_2, I_1 : 2), (I_2 : 2), (I_4 : 2)\}$	$\{I_2 : 4, I_1 : 2\}, \{I_1 : 2\}$
I_1	$\{(I_2 : 4)\}$	$\{I_2 : 4\}$

Step 4- Frequent pattern Generation.

	Frequent patterns generated
I_5	$I_2 I_5 : 2,\ I_1 I_5 : 2,\ I_2 I_1 I_5 : 2$
I_4	$I_2 I_4 : 2$
I_3	$I_2 I_3 : 4,\ I_1 I_3 : 4,\ I_2 I_1 I_3 : 2$
I_1	$I_2 I_1 : 4$



Conditional FP-tree associated with conditional node I3.