A-Class AprioriAlgoritham

Operations:

1-set\_trans():This operation find the number of transactions in the given file and return .

The set\_trans() will transactionFile and put it’s data into string and return no\_of\_transaction .When constructor of AprioriAlgo [AprioriAlgo(String file\_name,int Sup,int Con)] is start then by using set\_trans() will give the no\_trans in the given file.

2-remove\_last\_comm(String s): This will remove comma due to generate by excel file.

3-int get\_apriori() :This will generate the first candidate set using the transaction file given by the user.

/\*get\_apriori() is used in PoliceGeneration class at line 27.\*/

In notepad there is comma like

t100, i1, i2, i5, i6,

t101, i2, i4, i8,

t102, i2, i3,

t103, i1, i2, i4,

t104, i1, i3, i5,

t105, i2, i3,

t106, I1, I3,

t107, I1, I2, I3, I5,

There is lots of comma every line show we remove that comma.

This will remove by calling string remove\_last\_comma(string s). Which will remove last comma.

t100,i1,i2,i5,i6

t101,i2,i4,i8

t102,i2,i3

t103, i1, i2, i4

t104, i1, i3, i5

t105, i2, i3

t106, I1, I3

t107, I1, I2, I3, I5

String function str=str.substring(str.indexOf(",")+1) calling within the get\_apriori() will give result like

i1, i2, i5, i6

i2, i4, i8

i2, i3

i1, i2, i4

i1, i3, i5

i2, i3

I1, I3

I1, I2, I3, I5

Complete\_String=Complete\_String.concat(str.toUpperCase() + ","); this function will give the result

i1, i2, i5, i6,i2, i4, i8,i2, i3, i1, i2, i4, i1, i3, i5, i2, i3, I1, I3, I1, I2, I3, I5

Within get\_apriori() we apply three methods

1- remove\_last\_comma(str)

2- str=str.substring(str.indexOf(",")+1);

3- Complete\_String=Complete\_String.concat(str.toUpperCase() + ",");

Will give i1, i2, i5, i6, i2, i4, i8, i2, i3, i1, i2, i4, i1, i3, i5, i2, i3, I1, I3, I1, I2, I3, I5

I1,I2,I5,I6,I2,I4,I8,I2,I3,I1,I2,I4,I1,I3,I5,I2,I3,I1,I3,I1,I2,I3,I5,

Complete\_String=Complete\_String.substring(0, Complete\_String.length()-1);

After apply this function the above will become it will remove the last comma

Complete\_String =I1,I2,I5,I6,I2,I4,I8,I2,I3,I1,I2,I4,I1,I3,I5,I2,I3,I1,I3,I1,I2,I3,I5

After this we will call class String\_Duplicacy

B-class String\_Duplicacy

In this class there is one method remove\_duplicacy() In this method we remove duplicate string from given transaction .In this method we use String method split(",");

The split() method in Java is used to split the given string followed by regular expression in Java. For example you can check the string for "," "/" etc.. and split the string into multiple Java String Objects.

I1

I2

I5

I6

I2

I4

I8

I2

I3

I1

I2

I4

I1

Like that……………

I1,I2,I5,I6,I2,I4,I8,I2,I3,I1,I2,I4,I1,I3,I5,I2,I3,I1,I3,I1,I2,I3,I5

remove\_duplicacy() this method will give result like this

I1,I2,I5,I6,\*,I4,I8,\*,I3,\*,\*,\*,\*,\*,\*,\*,\*,\*,\*,\*,\*,\*,\*

final\_String=final\_String.concat(str\_arr[i]+ ",");

so final\_String=I1,I2,I5,I6,I4,I8,I3

returning to the AprioriAlgo class

Complete\_final\_String=str\_duplicate.remove\_duplicacy();

Complete\_final\_String= I1,I2,I5,I6,I4,I8,I3

Complete\_String =I1,I2,I5,I6,I2,I4,I8,I2,I3,I1,I2,I4,I1,I3,I5,I2,I3,I1,I3,I1,I2,I3,I5

String temp1[];

temp1=Complete\_final\_String.split(",");

void addElement(Object obj)

Adds the specified component to the end of this vector, increasing its size by one.

1-for(int i=0; i<temp1.length;i++)

{

2-for(int before=0, next=0; before!=-1; before=Complete\_String.indexOf(",",next))

{

int pos=Complete\_String.indexOf(temp1[i], before);

if(pos>-1)

{

count++;

next=pos;

}

else

break;

}

System.out.println(count + " " + Min\_Sup);

if(count>=Min\_Sup)

{

Sup\_Count[0].addElement(count);

temp[0].addElement(temp1[i]);

}

}

System.out.println(Complete\_final\_String);

temp1=Complete\_final\_String.split(",");//String temp1[].It is declare in get\_apriori() function of AprioriAlgo class.

//temp1 will store the string which does not contain any duplicate string inside.

Complete\_String

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| I1 | , | I2 | , | I5 | , | I6 | , | I2 | , | I4 | , | I8 | , | I2 | , | I3 | , | I1 | , | I2 |

|  |  |
| --- | --- |
| temp1[0] | I1 |
| temp1[1] | I2 |
| temp1[2] | I5 |
| temp1[3] | I6 |
| temp1[4] | I4 |
| temp1[5] | I8 |
| temp1[6] | I3 |

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20

In first for loop i=0

And second loop before =0,next =0

pos=Complete\_String.indexOf(temp1[i], before);

pos=Complete\_String.indexOf(I1, 0);

pos=0;

if(pos>-1)

count++=1

next=pos=0

before=Complete\_String.indexOf(",",next)

before=Complete\_String.indexOf(",",0)

before=1;

pos=Complete\_String.indexOf(I1, 1);

pos=18;

count++=2

next=pos=18;

before=Complete\_String.indexOf(",",18)

before=19;

pos=Complete\_String.indexOf(I1, 19);

pos=-1

so break

so total no of I1 equal to count.

The below result is given by

|  |  |
| --- | --- |
| temp[0] | Sup\_Count[0] |
| I1 | 2 |
| I2 | 4 |
| I5 | 1 |
| I6 | 1 |
| I4 | 1 |
| I8 | 1 |
| I3 | 1 |

After calling method int get\_apriori() will return itemcount =0. Which was initial -1.

In PolicyGeneration class it is used as

int itemcount=Apriori.get\_apriori();

itemcount is passing to get\_Candidate(itemcount) .

while(true)

{

if(Apriori.get\_Candidate(itemcount)==0)

{

System.out.println(itemcount);

if(itemcount==0)

{

System.out.println("No matches can be found");

result=new String[1];

result[0]="No matches can be found";

System.out.println("Not matched");

}

else

result=get\_Association(itemcount);

return result;

}

Itemcount++;

System.out.println ("Item count= " +itemcount);

}

Lets see what is working inside int get\_Candidate (int current\_candidate)

If(temp[current\_candidate].size()==1) then it will temp[0].size()==1 it’s means vector contains only one object inside.

temp[current\_candidate+1]=new Vector();

temp[1]=new Vector();

temp[1] // see code from 180 to 200.

|  |
| --- |
| I1 , I2 |
| I1 , I5 |
| I1 , I6 |
| I1 , I4 |
| I1 , I8 |
| I1 , I3 |
| I2 , I5 |
| I2 , I6 |
| I2 , I4 |
| I2 , I8 |
| ………………….. |

temp[current\_candidate+1]=vector\_redundant(temp[current\_candidate+1]);

Vector vector\_redundant(Vector initial) //we are passing temp[1] in vector\_redundant .

B-Class MyCustomFilter

Operations:

1-public Boolean accept(File file)// // Allow only directories, or files with ".txt" extension

2-public String getDescription () // This description will be displayed in the dialog,

// hard-coded = ugly, should be done via I18N

This function will display (\*.csv) Comma Delimited .

C-FutureStore

Operations:

1-BtnOKActionPerformed(java.awt.event.ActionEvent)

/\* Under this operation the following action performed

1-check the file is selected or not,

2-check whether the file is in correct format or not,

3-to check whether minimum support and confidence values are set properly or not if not it will give message” Specify correct support and confidence value... i.e. greater than 1”

4-to check whether minimum support is properly set or not if not it will give message "Specify correct support value... i.e. greater than 1 ".

5-to check whether confidence is properly set or not if it will give message "Specify correct confidence value... i.e. greater than 1 ".

If all above condition are correct then this will be executed to find the association rules.

Function BtnOKActionPerformed(java.awt.event.ActionEvent) will call constructor of PoliceGeneration from it’s inside .

Like

PolicyGeneration Calculate= new PolicyGeneration(TxtDataFile.getText(), Integer.parseInt(SpnMinSupport.getValue().toString()), Integer.parseInt(SpnMinConfidence.getValue().toString()));

And one the method of class PoliceGeneration Set\_Result().

Set\_Result() is used to find the final association rules for each frequent n-item set

In Set\_Result method we apply the method get\_apriori () of AprioriAlgo class on it’s(AprioriAlgo) object Apriori to no of item

Class String\_Duplicacy has method String remove\_duplicacy() which is used by AprioriAlgo class to remove the duplicate item from the Complete\_String .

Class Transaction File This class will initialize the transaction file given by user and this class is used in

AprioriAlgo class and FuTureStore class