

Using the top-k frequent item set for mining non-overlapping patterns Configuration Manual

> Shivam Gulve Student ID: x20181400

School of Computing National College of Ireland

Supervisor: Dr. Hicham Rafai

National College of Ireland Project Submission Sheet School of Computing



Student Name:	Shivam Gulve
Student ID:	x20181400
Programme:	Data Analytics
Year:	2022
Module:	MSc Research Project
Supervisor:	Dr. Hicham Rafai
Submission Due Date:	15/12/2022
Project Title:	Using the top-k frequent item set for mining non-overlapping
	patterns
	Configuration Manual
Word Count:	730
Page Count:	11

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

<u>ALL</u> internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	Shivam Gulve
Date:	14th December 2022

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).		
Attach a Moodle submission receipt of the online project submission, to		
each project (including multiple copies).		
You must ensure that you retain a HARD COPY of the project, both for		
your own reference and in case a project is lost or mislaid. It is not sufficient to keep		
a copy on computer.		

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Using the top-k frequent item set for mining non-overlapping patterns Configuration Manual

Shivam Gulve x20181400

Developer Configuration Environment

1 Overview

The objective of the study is to use top k counts to find maximal sequential patterns that are non-overlapping. The project is created in Apache NetBeans 12.4, and Java 15.0.1 JDK 11 is used to implement the improved NMSP algorithm. A step-by-step process for running the program and creating the necessary environment will be provided in the setup manual Babich (1986).

2 Hardware Requirement

Processor Intel(R) Core(TM) i5-7700HQ CPU @ 2.80GHz 2.80 GHz RAM 8 to 16 GB Internal Storage 60 GB System Type 64-bit-based processor Operating System Windows 10 or 11

3 Prerequisite

The prerequisites that must be met in order to start this project are described in this section. Prerequisites could include downloading an IDE in order to run software, and this section will explain the type of environment that must be set up in order to execute an algorithm.

3.1 Setting up IDE

An integrated development environment (IDE) is a software program that assists inside the successful creation of software code by developers. By merging functions like software editing, building, testing, and packaging in a user-friendly program, it increases software quality.

3.1.1 Apache NetBeans Software

Java has such an integrated development environment named NetBeans. That used a collection of application software functional units called are modules, NetBeans allows the development of projects. Linux, Mac OS, Windows, and Solaris can accept NetBeans. For this project minimum requirement for IDE is Apache NetBeans version 12.4 and that can be download from this link https://netbeans.apache.org/download/nb124/ nb124.html. The installation process is shown in fig 1 and fig 2

0	Apache NetBeans IDE Installe	er 🗕 🗖 🗙
Summary Click Install to start the installation.		Apache NetBeans IDE
Base IDE Installation Folder: C:\Program Files\WetBeans-15		
[V] Check for Updates The NetBeans installer can automatical using your Internet connection.	y check for updates of installed plugins	
Total Installation Size: 791.4 MB		
		< Back Install Cancel

Figure 1: NetBeans Installation

0	Apache NetBeans IDE Installer	_ 🗆 X
	Welcome to the Apache NetBeans IDE 15 Installer	
	The installer will install the NetBeans IDE with the following packs and run Click Customize to select the packs and runtimes to install.	times.
	Java SE Java EE HTML5/JavaScript PHP	
~	Castania.	Tankallaki an Gana 701 4 MP
NetBeans IDE	CORPORT P AC est	Ensumeron Size: 791-110
		Next > Cancel

Figure 2: NetBeans Installation 2

3.1.2 Java JDK

Oracle provides Java Technology through the Java Development Kit. This offers the Standard Edition of a Java Interface for Application Programming and includes the Java Language Specifications and Java Virtual Machine Specifications. The minimum requirement for JDK is 32 bit processor. For this project minimum requirement for JDK is JDK 11 and that can be download from here https://www.oracle.com/ie/ java/technologies/javase/jdk11-archive-downloads.html The installation of JDK is shown in fig 3 and to check if jdk is installed or not we can use a command i.e., javac in command prompt show in fig 4



Figure 3: JDK Installation



Figure 4: JDK Installation Validation

3.1.3 Data Set

The data set we will be using for this project is about DNA, protein, and viral sequences, baby products and much more. All type of data set can be download from this link https://github.com/shivamgulve4/research-project-data-set-/raw/main/DataSet.zip

4 Project Configuration

After completing all the above section now is the time to configure the project. In this section we will be configuring the project i.e., how to setup the project for execution we will see here.

C A	pache NetBeans IDE 1 Edit View Naviga New Project New File	2.4 te Source Refacto Ctrl+Shift+N Ctrl+N	r Run Debug Profile Tez	m Tools Window Help				Q. Search	- • ×
	Open Project	Ctrl+Shift+O	acer rege						
	Close Project Close Other Projects Close All Projects	,		후 Apache NetBeans IDE	Learn & Discover	My NetBeans	What's New	Show On Startup 🚽	
	Open File Open Recent File	,		My NetBeans					
	Project Groups Project Properties			Recent Projects	Install Plugins		Activate F	eatures	
	Import Project Export Project	3			Add support for other technologies by installir	anguages and g plugins from the	NetBeans to Start creati	ums on functionality as you use it. Ig and opening projects and the IDE	
	Save As Save All	Ctrl+Shift+S		<no project="" recent=""></no>	NetBeans Update Cent	er.	will just acti your exper Alternativel	vate the features you need, making lence quicker and cleaner. r, you can activate features manually.	
	Page Setup Print Print to HTML	Ctrl+Alt+Shift+P		/					
	Exit								
	<no avi<="" td="" view=""><td>ilab le ></td><td>Output ×</td><td></td><td></td><td></td><td></td><td></td><td>-</td></no>	ilab le >	Output ×						-
									INS 10/44 DM
÷	C Type her	e to search	💆 😻 o	<u> </u>				~ @ = 0	** 13/12/2022

Figure 5: Opening Project

In above fig 5 shows how to locate and in next fig 6 it shows how can we open a project.

Martin Config>	_ ⊌ - 17 18 P + 10 + 13	t - i sastatvaerove. C? C?					
Vices * Ster Dabbee Servers Merce Reportere National Software Subders Dabbee Subders Dabbee Stere Software Selection Server	Page ×	Apache NetBeans IDE st tet tet tet tet tet tet tet	Learn & Discove	My HetBoans	What's New X	Shew On Statury 2	< > ▼
igaar ×	Documents The PC Network	File name: add/National Colege o Files of type: project Folder	f Indend (Moduke)/SEM 3 (Resourch Projec	t(Project(Shiv am-Application)	Open Project Cancel		
- 1 3 3 4 2 3 3 4 2 4 3 4 2 4 3 4 2 4 3 4 2 4 2	ut ×						

Figure 6: Locating Project

The next fig 7 shows which type of GUI are build and we can see that there are total five GUI build.

Shivam-Application - Apache NetBeans IDE 12.4 File Edit View Navigate Source Refactor Run Deb	ug Profile Team Tools Window Help			Search (Ctri+I)	σ×
Caerous contg>					() * 5
Projects × Services - Start Page	*				() • U
E Source Packages					
gui DispbyPatterns.java	Apache NetBeans IDE	Learn & Discover My NetBeans	What's New	Show On Startup 🖌	
D Logn.jwa D PatternD Bcovery.jøva D controPaneljøva mages	My NetBeans				
Int AlgorithmExisting.java AlgorithmProposed.java Montr_cand.java	Recent Projects	Install Plugins	Activate Features		
cournes.java cournes.java cournes.java cournes.java cournes.java cournes.java cournes.java cournes.java cournes.interventes.	<no project="" recent=""></no>	Add support for other languages and technologies by notating plugns from the NetBeans Update Center.	NetBeans turns on functional Start creating and opening pr will gust activate the features your experience quicker and Alternatively, you can activate	y as you use t. oyects and the ID E fou med, making cleaner, e features manually.	
Tot Lines Tot Lines Tot Jines Coupt 2					
				inci	
				1052	PM
Type nere to search	N N N N N N N N N N N N N N N N N N N			^ 🥂 🗖 पण 13/12/	2022 -

Figure 7: Implemented GUI's

So, how to run this page is shown in fig 8.

Shivam Application - Apache NetBeans IDE	124				- 0 X
Pile Edit View Navigate Source Refact	Cor Kun Debug Profile Team Tools Window	N612.0MB			Swaren (centra)
Projects V Services	Start Pana X B Look two X				 () v D
E Shivam-Application	Source Design History		ି କାହା 💧 🗆	8	
Source Packages	<pre>import gui.controlamel/ import gui.controlamel/ import java.util.regox.Mu /</pre>	Lober J. Navigate Show Javadoc Find Usages Call Hierarchy Insert Code Fix Imports Refactor Format	Alt+F1 Alt+F7 Alt+Insert Ctrl+Shift+I 2 Alt+Shift+F	Project Properties.	
eccurrence.java seqdb.java sub_ptn_struct.java	15 * 4 16 * @author admin 17 */	Run File Debug Hie	Shift+F6 Ctri+Snitt+F5		
😸 🏣 Libraries	18 public class Login exten 19	Test File Debug Test File	Ctrl+F6 Ctrl+Shift+F6		
🐮 🧝 Test Libraries	<u>></u>	Run Focused Test Method	Currainterro		×
Login java - Navigator × - Members × <empty> × 10 Solution - Solution - Sol</empty>	Output - Shivam-Application (run) × run: BUILD SUCCESSFUL (total time: 24 -	Debug Focused Test Metho Run Into Method New Watch Toggle Line Breakpoint Profile	Ctrl+Shift+F7 Ctrl+F8 >		-
 jToggk8uttonSActionPerformed(Ac main(String[] args) jLabell : JLabel iLabell 3: Label 	66	Cut Copy Paste	Ctrl+X Ctrl+C Ctrl+V		
 j.abe0 : J.label j.abe4 : J.label j.abe5 : J.label j.abe5 : J.label 		Code Folds Select in Projects text	>		
PesswordField 2: JPasswordField					,
					5:25 DIS
Type here to search	À 涂 o 🖽 🧧				∧ ((😉 ⊄)) 1058 PM

Figure 8: Execution

Execution Configuration Environment

The first one is Login page that consist of user authentication and for temporary the credentials are Username: "admin" Password : "123456". shown in fig 9. For now there is no back-end to this so, the login data is static.



Figure 9: Login Page

After user have successfully login we will get on next page i.e., Control Panel. The control panel have three option shown in fig 10 and those are Pattern discovery from Data Set, Pattern Discovery from Input and Sign out.



Figure 10: Control Panel

After select Pattern discovery from Data Set, it goes to next page 11. User have to browse the data set from our Local Drive 12 and then just provide all the details given below.

Find Circular Pattern	MININ	E PATTERNS		- a >
1. 54				
-09.26	Select File:		Browse	
-90	minimum Gap:	Maximum Gap:		
	• Pattern Count(k):	Pattern Size:		
	Minimum Support:			
		Submit		
P Type here to search	🚴 💥 o 🗄 🖬 🍖 🌘 !			へ 🥢 🔚 (1) 11:38 PM

Figure 11: Pattern discovery from Data Set

at Find Circular Pattern		
	MINING PATTERNS	
	🗟 Open X	
Select File:	Lookju 🗑 DataSet 🔻 🍘 🗇 💭 🗐 月 Browse	
minimum Gap:	Bally1 tot DM2 bit MHR SOUVLA SDB3 bit Bally2 bit DM3 bit SMR SOUVLA SDB4 bit Bally2 bit DM4 bit SMR SOUVLA SDB5 bit Bally2 bit DM4 bit SMR SOUVLA SDB5 bit Bally2 bit DM4 bit SMR SOUVLA SDB5 bit Bally4 bit DM4 bit SMR SOUVLA SDB5 bit DM4 bit DM4 bit SDB2 bit SDB5 bit	
• Pattern Count(k)	۹	
Minimum Suppor	File Name: Files of Type: All Files	
	Open Cancel	
	Submit	
・ P Type here to search 👌 🕸 O 日 📠	S S	ヽ 🦟 📹 (Þi) 11:38 PM 📮

Figure 12: Browse Data Set

🔊 Find Circular Pattern			– a ×
		MINING PATTERNS	
	Select File: minimum Gap:	C:\Users\shiva\OneDrive\Documents\DataSet\Baby1.txt	Browse
	Pattern Count(k):Minimum Support:	100 Pattern Size: 3	
		Submit	
Type here to search	👌 💸 o 🖽 🖿 🔇	A • M B	^ ≪ ₩ 40 1283 AM UM122022 □

Figure 13: Filling Parameters

Now, just adding the parameters we will get NMSP patterns shown in fig 13 and when we submit we will get a processing window fig 14 and then after getting out patterns shown in fig 15

		MINING PATTERNS	
6770	Select File:	C/\Users\shiva\OneDrive\Documents\DataSet\Baby1.txt	Browse
	Pattern CoMinimum S	Please wait *	
		Submit	

Figure 14: Waiting Window

we can also save our output and we can see how to do that in fig 16. It means we can save the output as a notepad

🛓 Find Circular Pattern		- o ×
	MINING PATTERN	
	Pattern:	
	No of patterns found :36 [{ "pattern":"ccc","support" : 44352] , { "pattern":"caa","support" : 24599} , { "pattern":"cdd","support" : 17513} , { "pattern":"cdd","support" : 17509} , { "pattern":"aaa","support" : 17509 , { "pattern":"aad","support" : 10652} , { "pattern":"dad","support" : 10599} , { "pattern":"bcc","support" : 1886}	*
	Save	Close
📕 🔎 Type here to search 🛛 👌	🔊 o 🖽 🚍 🍖 🏮 🌂 🛃	へ <i>派</i> (筆 句) ¹²⁰² AM 1402/2022 □

Figure 15: Pattern Output

So, if we want to compare our output we would be able to do that and just to get the preview of the last output we don't have to re-run the whole program again.

Pattern: Void patterns found :36 [{*pattern*:*ccc*,*support*: 44352}; • (*pattern*:*cca*,*support*: 24927) If Select Destination Tooler If Selec	💩 Find Circular Pattern		٥	×
Pattern: No of patterns found :36 [{"pattern":"ccc", "support" : 44352}; , {"pattern":"cca", "support" : 24927) if Select Destination Noiser if winn-Agediation		MINING PATTERN		
No of patterns found :36 [{"pattern":"ccc","support": :44352] , {"pattern":"caa","support": :24927) If Select Defination Tooler Image: Select Defination Tooler <th></th> <th>Pattern:</th> <th></th> <th></th>		Pattern:		
Folder banne: oge of reland Modules SEM 3 Research Project/Project/Shvam-Application Files of Type:		No of patterns found :36 [{ "pattern":"ccc","support" : 44352} , { "pattern":"caa","support" : 24927} @ Select Demination folder		
		Folder Jame: Telder		

Figure 16: Save Output

we can also check different experiments cast study i.e., processing time, memory used by program shown in fig $17\,$



Figure 17: Experiments

Now, if we select another option i.e., Pattern Discovery from Input 18. In this we have to provide the pattern as an input and in remaining parameters we have to give same details as we gave in fig 13 So, after giving the input we get the output as shown in fig 19

≰j Find Circular Pattern	MINING PATTERN	×
	Add Data Attoatoacatoa	
	minimum Gap: ¹ Maximum Gap: ⁴ ^o Pattern Count(k): <u>18</u> Pattern Size: <u>3</u>	
	Minimum Support: Submit	

Figure 18: Pattern Discovery from Input



Figure 19: Pattern Discovery Output

To check the experiments like what was the processing time and memory used can be seen in fig $20\,$

Shivam-Application - Apache NetBeans IDE	124	- 0	×
ble Edit View Navigate Source Refacto	tor Kun Debug Bronie Team Tools Window Help	Search (Ctri+1)	
- Y Y II II II - V C - C - C - C - C - C - C - C - C -			
Projects × Services -	Start Page 🛪 📸 Login.java 🗙 🔂 PatternDiscovery.java 🛪 🔂 FindPatternsFromFile.java 🛪		() ¥ 🗆
🖃 🆢 Shivam-Application	Source Design Hatory 🔯 🐘 - 📓 - 🖏 - 🖏 - 🗣 🖶 🛄 🛷 🐁 🗞 🗐 🗐 🗰 💷		88
🖶 🛅 Source Packages			A
🕀 🌐 gui	2		
Display Patterns. java	3 import java, awt, BorderLavout;		-
FindPatternsFromFile.java	4 import java.io.File;		-
Login.java	S import java.io.IOException;		=
PatternDiscovery.java	6 import java.util.ArrayList;		=
Control-anecjava	S import java.util.Enumeration:		=
a anges	import java.util.Hashtable;		-
AborthmExisting lava	import java.util.logging.Level;		
AkorthmProposed.tava	import java.util.logging.Logger;		-
freArr_cand.java	import java.util.regex.Matcher;		-
ode.java	import java.util.regex.Pattern;		_
- 📓 occurrence.java	13 import javax.swing.ubjalog/		_
seqdb.jev a	in the second se		
	is import javax.swing.IntionDane:		-
🕸 🧱 Test Packages	17 import javas, sving. Danel:		
E Draries	18 import javax.swing.SwingWorker;		
H S Test Libraries	and the state of the second se		~
Navigator × -			^
Members V <empty> V</empty>	Output - Shivam-Application (run) ×		-
E & FindPatternsFromFile :: JFrame	- W - >>3		^
FindPatternsFromFile()	>>3		
() intComponents()			
- iRadioButton1ActionPerformed(Act	(1) pattern * Add		
jRadioButton2ActionPerformed(Act	""" , ("pattern":"ICC","support" : 6)		
JToggleButton3ActionPerformed(Ac	, ("pattern":"AIT","support" : 5)		
JToggleButton4ActionPerformed(Ac	, ("pattern":"TAA", "support": 5)		
JToggleButton5ActionPerformed(Ac	<pre>("pattern": "AAA", "support" : 4) ("pattern": CTTE " #UNTERN : 1 2)</pre>		
main(String[] args)	, (pattern', 'CCC', "support : 3)		
buttonGroup1 : ButtonGroup	("pattern":"ITT","support" : 3)		
crcPat : Hashtable <string, integer=""></string,>			
jLabell : JLabel	Processing time: 0.04600000009840697		
Jraberto : Jraber	Used memory is bytes: 22201040		
	SUILD SUCCESSIU (total time: 1 minute 42 meconds)		
🏶 🗆] 🏛 🚫 🚳 85			~
		16:32 INS Un	« (LF)
		. 2:13 AM	_
P Type here to search		へ 信 告 切) 14(12(20)	

Figure 20: Pattern Discovery Experiments

References

Babich, W. A. (1986). Software configuration management: coordination for team productivity, Addison-Wesley Longman Publishing Co., Inc.