



IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 10 Issue: VII Month of publication: July 2022

DOI: https://doi.org/10.22214/ijraset.2022.45884

www.ijraset.com

Call: 🕥 08813907089 🔰 E-mail ID: ijraset@gmail.com



Stream Processing for Association Rule to Generate Student Dataset using Apriori Algorithm

Keerti Ghodke¹, Rashmi Patil², Indira Umarji³, Dr Vidyagouri Hemadri⁴, Dr U P Kulkarni⁵

¹Student, ^{2, 3, 4}Assit Professor, ⁵Professor, Dept of Computer Science and Engineering, SDM College of Engineering and Technology, Dharwad, Karnataka, India

Abstract: Analytical techniques have been used for many years to analyse and predict academic achievement from various perspectives. One of the most challenging problems for higher education is predicting students' paths through the education system. Many factors influence successful student outcome prediction in the early course stage. Apriori algorithm techniques use a variety of methods to find out and collect based on stored data patterns student information. Colab and Python applications are used in this project to predict each student based on characteristics in the given dataset. Each student's information is included in the dataset. Because it arrives as it is being created, received real-world data is referred to that as streaming data. Keywords: Stream Processing, Kafka, Apriori Algorithm, Data mining, Student Dataset

I. INTRODUCTION

Data mining is an important part of educational today's organizations, as well as one of the most important research areas, with the goal of extraction of useful information from huge datasets of data. Educational data mining (EDM) is an important research field that can predict future useful information from educational datasets in order to improve academic outcomes, better understand, and assess students' active learning. Data mining is the process of extracting information from massive amounts of data. To put it another way, data mining is the process of trying to extract knowledge from data. Knowledge discovery refers to the technique of learning from data collection (KDD). We can make educated guesses based on the data provided. we have used Apriori algorithm and LinkedIn generated Apache Kafka, which is an accessible stream platform. It was later given to the Apache Foundation and accessible in 2011. Real-time processing of data scientists available: By attaching to the broker to discover data, train designs, deploy them to producers, monitor them, and They become more self-sufficient throughout the development lifecycle if they are quickly re-trained on new data. Apriori Algorithm :The Apriori is a popular method for mining frequently occurring item sets for Logic association rules. Apriori employs a "bottom up" approach in which frequent subsets are expanded one at a period. Kafka:LinkedIn created Apache Kafka, an open-source stream platform. It was later transferred to the Apache Foundation and accessible in 2011.

II. LITERATURE SURVEY

Dr. Vikesh Kumar & Samrat Singh [1] Data mining is a powerful tool for improving academic performance. Educational Data Mining is concerned with developing new methods for extracting information from educational data sets that can be used for decision making in the educational system.

M. Goyal and R. Vohra [2] Data analysis is vital for decision support in any industry, including manufacturing and education. While data mining techniques such as clustering, decision trees, and association are applied to higher education processes, this can help to improve student performance, life cycle management, course selection, retention rate, and grant fund management.

Seema Purohit and Neelam Naik [4]. Quality higher education is required for the country's growth and development. One of the pillars of higher education is professional education. Data mining techniques seek to uncover hidden knowledge in existing educational data, forecast the future, and apply it to the benefit of higher education institutions and students.

K. Rajeswari, Suchita Borkar [7]. Education Data mining is an interesting area that has a major impact on predicting students' academic performance. The performance of students is evaluated in this paper using the association rule mining algorithm. There has been research done on evaluating student performance based on various attributes. Important rules are generated in our study to measure the correlation between various attributes, which will help in enhancing the student's academic performance.



International Journal for Research in Applied Science & Engineering Technology (IJRASET) ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538 Volume 10 Issue VII July 2022- Available at www.ijraset.com

M. Tiwari, Randhir Singh, and Neeraj Vimal [8]. Educational institutions are important parts of our society, and they play an important role in the nation's growth and development. Predicting student performance in educational settings is also important. Personal, social, as well as psychological factors all effect a student's academic performance.

III. PROPOSED METHODOLOGY

The Apriori had a significant issue with various scan results through entire data set. It took a lot of spacetime. The change in our paper implies that we really do not scan the database structure to add up the support for each attribute. This is accomplished by keeping track of the minimum support count and comparing it to the support of each attribute. An attribute's support is only counted until it reaches its minimum support value. It is not necessary to know the support for just an attribute up to that point. This feature is achieved by using a value called flag in the technique. When the value of flag changes, the loop is divided and the benefit for support is recorded.

IV. PROBLEM DEFINITION

Create centralized dataset to improve the capability of level-wise frequent generation student dataset, an vital Apriori property is being used, which aids in reducing the search time. space. All subsets of a frequent student dataset should be common (Apriori property).

V. DATSET DESCRIPTION

The followings are the steps involves design and dataset.

We have chosen a dataset and attributes: created dataset contains the analysis of each students from 1 to 8 semester. The dataset contains 105 instances and 35 attributes. The data file has to be in either in 'CSV' format.

Here is the sample of our dataset which is in 'CSV' format

۵	5 C - +							Apriori	Algorithm ar	nd Kafka.csv	- Excel							m -		
File	Home	1sert Pa	ge Løyout 🛛 Fo	ormulas	Data R	eview 1	View 🛛	Tell me what	it you want t	o do								Sig	nin As	Share
Paste	★ Cut Im Copy + ✓ Format Painte Clipboard	Calibri B I		* A* A* * A *	= = <u> </u>	No constant	🔐 Wrap 🗐 Merge	Text & Center =	General		0.0	itional Form titing * Tab Styles		nsert (Delete Forr	T Fill	I* 5	ort & Find ilter = Select) Bi	
P15	• 1	XV	∫x B																	
4	A 8	c	D	E	F	G	н	1)	К	ι	м	N	0	р	Q	R	s	T	
1.00	o. of		Phone					r Semester					Total marks	Percenta		No. of Backlogs In Semeste	in r Semeste			ir er Se
St	udents Name 1 Arun	Address	Number 9546312789		2 Marks 350	3 Marks 495	4 Marks 520			7 Marks 512	8 Marks 650	of Marks 7050	obtaind 4127	ge 58.54	Grades	1	2	3	4	0
	2 Amith	Hubli	9546312789															()()*		1
	3 Amar	Dharwad																		1
	4 Anusha	Dandeli	9546312794														7			0
	5 Anu	Haveri	9546312722						550								0			1
1	6 Ananya	Hangal	9546312733		810	510	49		556	620	46	7050	4570	64.82	A		0 0) (0
	7 Ananath	and the second second second	9546312781		510	520	499			710	740	7050	4822				0 1			0
	8 Amos	Bijapur	9546312754	750	620	515	455	454	570	720	54	7050	4629	65.66	A		1 () ()	1
6	9 Andrew	Dharwad	9546312719	644	400	525	438	464	580	300	46	7050	3812	54.07	B		2 () ()	1
1	10 Apporva	Hubli	9546312747	620	512	560	521	474	590	310	56	7050	4148	58.84	В		0 1		2	0
8	11 Apporva	NHosur	9876543220	350	495	520	510	512	490	350	710	7050	3937	55.84	В		0 () ()	0
	12 Archana	Belgaum	9876543221	300	560	485	520	513	500	300	650	7050	3828	54.3	В		0 () ()	0
	13 Ananthr	a Dandeli	9876543222	456	570	496	530	517	612	450	550	7050	4187	59.39	В		1 () 1	i l	1
5	14 Anvar	Kumta	9876543223	312	590	431	. 540	516	613	312	650	7050	3964	56.23	В		1 1	. 1	1	1
	15 Asma	Karwar	9876543224	720	580	444	550	534	644	720	65:	1 7050	4843	68.7	A		0 () ()	0
	16 Ananada	Dharwad	9876543225	810	510	491	550	620	700	810	45:	7050	4948	70.18	A		0 1	. ()	1
1	17 Angel	Dharwad	9876543226	510	520	499	560) 710	850	510	46:	1 7050	4620	65.53	A		0 () ()	0
1	Aprie	ri Algorith	m and Kafka	۲								1	0.000							
ady	Scroll Lock															H [[]	E -	1	+	1(
-	and the second second second	100		and the second	100		-	-	-		_	_			_		-			÷.



International Journal for Research in Applied Science & Engineering Technology (IJRASET)

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538 Volume 10 Issue VII July 2022- Available at www.ijraset.com

E	3 5-										Apriori Alge	orithm and	Kafka.csv											
	le H	ome		Page Layo			Data	Review	View		ne what yo											Sig	n in A	Shar
	to to	py *	Calib			- ^ ^				/rap Text lerge & Ce		General	, 58 4		itional For	mat as			Format	∑ Autos	Z	* 5		
	Clipboa	mat Pain rd	ier . G		ont				lignment	ierge & Ce	river -	Num				able - Sty	les -	ells	-	Clear 🧶	Filte	er * Selec	t-	
	117	¥ 1:	× •	- fx																				
	0	P	0	R	s	т	U	v	w	×	Y	z	AA	AB	AC	AD	AE	AF	AG	AH	AI	AJ	AK	
	Percenta	Student	No. of Backlogs in	No. of	No. of	in	No. of Backlogs in Semester	in	No. of Backlogs in	No. of	Total Number	No of days Present in semester	Total Number of Working	Total number of days										
		Grades		2	3	4	5	6	7	Semester 8	Backlogs		2	3	4	5	6	7	8	Day's	attended			
	58.54		0	0	C	0	2	0	0	1	3	111	121	101	109	131	141	99	100	1268	913		80%>5	
	55.87		1	0	1	1	1	1	0	0		112							109		925		60%>A	
	68.01		1	1	1 1	1	0		1	0	5	113											50%>8	
	61.9		0	0			0		1	1	3	114											35%>C	
	64.82		0	1				1	0			115											337641	
	68.4		0	1	1	0		0		0		117											1	
	65.66		1	0			0			1	4	118												
	54.07		2	0	0	1	1	1	2	2	9	113				133			99	1440				
	58.84		0	1	2	0	C	0	1	0	4										943		1	
	55.84	в	0	0	C	0	C	0	0	0	C	115	125	105	105	135	145	95	98	1440	923			
	54.3		0	0	0	0	2	0	0	1	3	116		106	104		146						1	
	59.39		1	0	1 1	1	1	1	0			117			103		147		109					
	56.23		1	1	1	1	0		1	0	5	116												
	68.7		0	0	C		C		1	1	3	117					147							
	70.18		0	1	C		C	0			2	111									913		-	
	65.53		0	0	C			1	0		3	112											-	
	70.84		0	1	1	0				0	4	113							99					
	54.3		1	0	-		0	0	1	1	4	114								1440			1	
	62.23		2	0	0	1	1	1	2	2	9	115				135					923 928		1	
	02.20		iori Algor	ithm and	Kafka	(+)						110	120	100	1 4	150	140	. 94	100	1440	928		1	-
		Apr	ion Algor	i and		Ð									- hard				1110000	and a second				
	dy																			[[[]]]	m		+	87

Fig 1. Student dataset

VI. EVALUATION METRICS

The Apriori algorithmic rule is that it assumes all elements of a frequently occurring item set to be frequent.

Similarly, for any sporadic item set, all its supersets should even be sporadic.

Support

Confidence

List

Conviction

Support- The amount of support for a law $X \Rightarrow Y$ is calculated by dividing the number of transaction data that fulfil the law, N (X=>Y), by the total number of transactions, N.

(X=>Y) Support =N (X=>Y) / N

The frequency of activities that each of the rule's LHS and RHS hold true is thus the support. The bigger and more powerful the information that each type of event occurs along, the higher the support.

Support of item x is nothing however the quantitative relation of $\{\text{the variety}|\text{the amount}|\text{the quantity}\}\$ of transactions within which item x seems to the full number of transactions.

Support = Support = = 0.66667

Confidence is calculated by dividing the number of transaction data that fulfils the guideline N (X=>Y) by the transactions that consist the rule's body, X.

(X => Y) Confidence = N (X => Y) / N (X)

The belief is that the RHS will hold true if the LHS proves true. A high likelihood that the LHS event will end up there in RHS event assumes feat or apply statistical dependence.

Lift- The lift of the rule $X \Rightarrow Y$ is the deviation of the full rule's support from the Support assumed below self rule given the support systems of the LHS (X) and also the Right hand side (Y).

Lift = self-assurance (X=>Y) / help (Y)= help (X=>Y) / help (X). support (Y)

Lift is a measure of the impact that information from the LHS has on chances of The RHS being true. Then raise is a value that provides data on the increase in likelihood of the "then" (subsequent RHS) half handed the "if" (antecedent LHS) half.

Lift is exactly one: There was no outcome (LHS and RHS independent). There is no connection between Events.

Greater than one lift: Positive outcome (if the LHS holds true, the RHS of operational risk management is more likely to hold true). Positive relationship between events

Lift is less than one: Negative outcome (whenever the LHS holds good, the RHS is less likely to hold true). Dependence between events that is negative.

Leverage is the amount of extra examples covered by both the element and also the outcome that is greater than what would be required if the cause and outcome were independent of each other, and finally. lev(X,Y) = supp(X,Y) sup(X) (Y)

Conviction is a live, related to Leverage, that mechanisms the departure from freedom. conv(X

Y) = supp(X)(1-supp(Y)) / supp(X) - supp(X) (X,Y)



ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538 Volume 10 Issue VII July 2022- Available at www.ijraset.com

VII. SYSTEM DESIGN

System design at the first stage we have problem statement once the problem statement defines the the what we are carring out for the project is defined we collect the student dataset which predicate and analysis the performance of each student once it is done by data is preprocessed Data preprocessing, that defines Any type of processing performed on original data to prepare it for further data processing is referred to as data preparation. Filters that convert the data in ways can be defined in the preprocess section. At the third stage we have data Data cleanup and data translation options Software is an information management technique involving ingesting an ongoing data stream and rapidly analysing, filtering, transforming, or improving the data in real time. Classification and relationship describe how components and object types will be further defined by linking to sources of information.

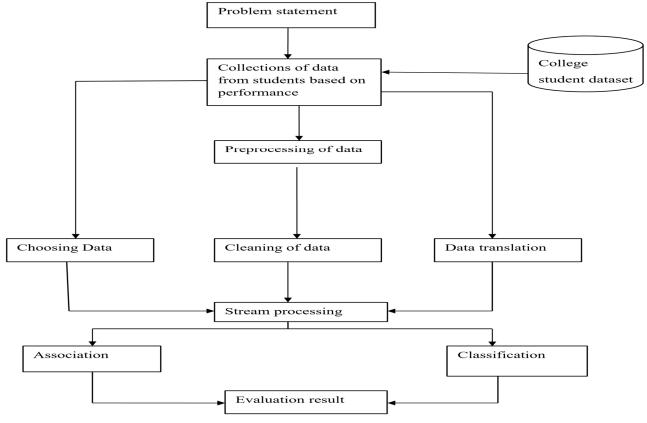


Figure 2: System Design

VIII. PERFORMANCE ANALYSIS

Fig 3 Shows the apriori algorithm to predict and analysis of student dataset and Kafka Implementation...

Files Code + Text: Editing Code + Text: <th>File Edit View Insert Runtime Tools</th> <th>Help All changes saved</th> <th>🗖 Comment 👫 Share 🌣 K</th>	File Edit View Insert Runtime Tools	Help All changes saved	🗖 Comment 👫 Share 🌣 K
 (x) a simple_data Apriori Algorithm and Kafka.csv (z) from google.colab import drive drive.mount('/content/drive') Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force_rements (a) import pandas as pd import numpy as np import multiple as pt import matplotible.pyplot as pt import reahorn at snt (a) Upload the CSV file from google.colab import files uploaded = files.upload() Choose Files Apriori Algorithm and Kafka.csv (b) Apriori Algorithm and Kafka.csv (c) Saving Apriori Algorithm and Kafka.csv to Apriori Algorithm and Kafka (1).csv (c) import io 	≣ Files ⊡ ×	+ Code + Text	Disk Editing
from google.colab import files uploaded = files.upload() Choose Files Apriori Algorithm and Kafka.csv Apriori Algorithm and Kafka.csv to Apriori Algorithm and Kafka.csv (>) ()	(x)	<pre>drive.mount('<u>/content/drive</u>') Drive already mounted at /content/drive; to attempt to forcibly r () [3] import pandas as pd import numpy as np import mumpy as np import matplotlib.pyplot as plt</pre>	remount, call drive.mount("/content/drive", force_remount-
Saving Apriori Algorithm and Kafka.csv to Apriori Algorithm and Kafka (1).csv		from google.colab import files uploaded = files.upload() Choose Files Apriori Algod Kafka.csv	
E print(raw_data)		<pre>raw_data = pd.read_csv(io.BytesIO(uploaded['Apriori Algorithm and</pre>	i Kafka.csv']))
Disk 69.13 GB available 101 644 104 102 491 103 103 103	Disk 69.13 GB available		-

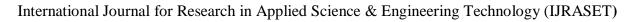


International Journal for Research in Applied Science & Engineering Technology (IJRASET) ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538 Volume 10 Issue VII July 2022- Available at www.ijraset.com

C					n_Impleme Runtime Te				玲 anges s	aved																	nment	4	sh	are 🕸	K
100	Files					<	+ C	ode -	Text																	RAM Disk	-	-	- L - A	 Editing 	^
Q	₽	C 3		2			(s	naw	_data . nt(raw_	- pd.re	ad_cs	v(10.8	BytesIO	(uploa	ded['A	prior	·i Alg	gorit	hm a	ind K	afka	.csv	. 1))								
{x}			Algorith		Kafka (1).csv			101 102 103 104			644 491 499 459	····						10 10 10	3 2 1												-
	iii	Apriori	Algorith	im and	Kafka.csv			0 1 2 3 4 100 101 102 103 104						131 132 133 134 135 137 136 137 138 139	No of c						141 142 143 144 145 147 146 147 148 149	`									
								0	NO OI	days	presei	nt in	semeste	99 80	No of c	lays	prese	int 1	n se	mest	100 100	`									
< >								2						97							99										
								4						95							98										
2	Disk I	_			69.13 GB availa	ble		100						93							109										
,												Exec	uting (35	is) Cell	> apriori)															×
C	•	e Edit	View	Insert	n_Impleme Runtime To	ools	Help	All.ch		aved																	ment	-	A sh	1965-197 <u>7</u> 00	R
100	Files						+ 0		- Text	med: 36	Unna	amed :	37													RAM Disk	-	- · ·	<u> </u>	Editing	^
a,	E	C3		100				1 2		ned: 36 NaN NaN NaN		80% 60% 50%	i>5 i>A i>B																		
{ <i>x</i> }	- BB	drive	data					34		NaN		353	INC.																		
		Apriori	Algorith	m and	Kafka (1).csv Kafka.csv			100 101 102		NaN		N	IAN IAN																		
	_							103		NaN		PA	IaN																		
								[10	i rows	× 38 c	olumns	-1																			
							- [G] raw	data.	columns																					
•	Disk =				69.13 OB availat			Ind	**(['No 'Se 'Se 'No 'No 'No 'No 'No 'No 'No 'No 'No 'No), of 5 imester imester stal ma), of 8), of 8	tudent 1 Mar 4 Mar 7 Mar acklog acklog acklog acklog acklog mber c ys Pre	toks',',', noks'', notaind gs in gs in gs in gs in sent	Name', 'Semest 'Semest Semeste Semeste Semeste klogs', in seme	'Addr ter 2 ter 5 ter 8 rcenta er 1', er 3', er 7', No ester	ess', Marks', Marks', ge', 's 'No. c 'No. c 'No. c of days 2', 'No 4', 'No	Phon 'Se 'To tude f Ba f Ba f Ba	meste meste tal N nt Gr cklog cklog cklog sent days	iber 3 1 in 6 1 in 6 1 in 6 1 in 6 1 in 6 1 in 6 1 in 5 in 1 in 5 Press	Mark Mark r of Sem Sem Sem emes ent	s', Marl este este ter in s	K5', r 4', r 6', r 8', 1', omest	er 3									
+	Disk =				69.13 GB availat	Die			No	of da	ys Pre	Execu	ting (1m	65) Cel	1 > aprior	10	days	prese	ent	in s	emest	er s									×
~	6	Aprior	i Algo	rithm	_Implemen	itatic	on.ip	nb	ŝ																-		ment		sha		0
-	File	Edit	view I	Insert	Runtime Too	ols I	Help	All cha	iges say	ved																		-	• 5Hd	ne 💊	
=	Files						+ Co	le +	Text																	RAM Disk		•	-	Editing	~
Q	æ	C 3	4	3		~	[12]	bask	t_Fran	nce.hea	d()																				
	-											т	otal ma	arks of	btaind	No.	of 5	tuder	nts	Phor	ie Nu	mber		Un	named	1: 35	Unn	amed	36	0	
<i>{x}</i>	· • • •	lrive ample_c	lata					Tota	1 Numb	per of	Backlo					0 :	2 3	4 5		0	2	3			4 5			2 3	4 5		
		priori Al	gorithm		afka (1).csv					0					1	0 0	0 1	0 0	1	0	0	1		. 0	0	0 0	0	0 0	0 0	0	
		priori Al	gorithn	n and Ka	afka.csv					1					1	0 0	0 1	0 0	0	0	0	1				0 0		0 0	0 0		
										2					1	0		0 0		0	1	0				0 0		0 0	0 0		
										3								1 0		0	0	0			0	0 0		0 0		0	
								5 row	s × 193	4 columns	s				1	0 0		0 0	1	0	0	0		. 0	0 1	0 0	0	0 0	0 0	0	
						04	-					patter	ons impo	ort ap	riori,	asso	ociati	Lon_r	ules												
							0	# Bu	lding	the mo	del ri(bas	sket_F	rance,	min_s	upport	- 0.	1, us	se_co	Inam	ies -	True	•)									
<>								# Co	llectin	ng the	inferr	red ru	les in	a dat	aframe																
								rule	s - ass	sociati	on_rul	les(fr	q_item:	s, met	'lift'	1ft'], a	, min	thr.	esho =[Fa	ld -	1) Fal:	se])									
,	Disk -			6	9.13 GB available	0									> apriori												A	- 00	•	sta iCi i	: ×

Perform stream processing: Use any open source project such as TensorFlow, NumPy, SciPy, or Matplotlib, making it easy to run machine learning models on streaming data. The following figure shows Streaming data with kafka.

File Edit View Insert Runtime Tools	Help Save failed	-
Files 🖸 🗶	+ Code + Text V RAM Disk C Editing	
Ø 🛛 51 6	✓ [63] from google.colab import drive	
 drive 	<pre>/ [64] drive.mount('/content/drive')</pre>	
Mafka_2.13-3.1.0 manual sample_data Apriori Algorithm and Kafka (1).csv	Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force_re	emour
 Apriori Algorithm and Kafka.csv kafka_2.13-3.1.0.tgz 	<pre>> [65] pip install tensorflow-io 25</pre>	
	Looking in indexes: <u>https://pygi.org/simple, https://us-python.okg.dev/colab-wheels/public/simple/</u> Requirement already satisfied: tensorflow-io in /usr/local/lib/python3.7/dist-packages (0.26.0) Requirement already satisfied: tensorflow-io-gcs-filesystem==0.26.0 in /usr/local/lib/python3.7/dist-packages (from	ten
	4	
	<pre></pre>	
	Looking in indexes: <u>https://pygi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/</u> Requirement already satisfied: kafka-python in /usr/local/lib/python3.7/dist-packages (2.0.2)	
	[67] import os from datetime import datetime	
	import time	





ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538 Volume 10 Issue VII July 2022- Available at www.ijraset.com

Kafka_Implementation.ipynb File Edit View Insert Runtime Tool:	s Help				RAM
Files 🖸 🗙	+ Cod	e + Text			V RAM Disk K Cliting V
Ø 🖾 5J 🖬	∑ [67]	<pre>import os from datetime import dat</pre>	etime		
drive		import time import threading	ecane		
 kafka_2.13-3.1.0 sample_data 		import json from kafka import KafkaP	roducer		
Apriori Algorithm and Kafka (1).csv		<pre>from kafka.errors import from sklearn.model_selec</pre>	KafkaError	split	
 Apriori Algorithm and Kafka.csv kafka_2.13-3.1.0.tgz 		<pre>import pandas as pd import tensorflow as tf import tensorflow_io as</pre>	tfio		
	[68]	print("tensorflow-io ver		version))	
	Os	print("tensorflow versio tensorflow-io version: 0	n: {}".format(tfvers	ion_))	
		tensorflow version: 2.9.	1		
	[69] [69]	<pre>!curl -sSOL https://dlcd</pre>	n.apache.org/kafka/3.1.	0/kafka_2.13-3.1.0.tgz	
	2 [70]	ltar -xzf kafka_2.13-3.1	.0.tgz		
👝 💪 Kafka_Implementation.ipynb	14	a colore a			🗖 Comment 🔐 Share 🏟 🗰
File Edit View Insert Runtime Tools		Save failed e + Text			Disk Editing
		1./kafka_2.13-3.1.0/bin/	kafka-topics.shcreat	ebootstrap-server 127.0.0	1:9892replication-factor 1partitions 1
Ca ···		1./kafka_2.13-3.1.0/bin/	kafka-topics.shcreat	ebootstrap-server 127.0.0.	1:9092replication-factor 1partitions 2
anve kafka_2.13-3.1.0 sample_data Apriori Algorithm and Kafka (1).csv		[2022-07-16 09:09:17,987 (kafka.admin.TopicComma Error while executing to [2022-07-16 09:09:21,220 (kafka.admin.TopicComma] ERROR org.apache.kafk	a.common.errors.TopicExistsE> opout-test' already exists. a.common.errors.TopicExistsE>	cception: Topic 'Dropout-train' already exist cception: Topic 'Dropout-test' already exists
Apriori Algorithm and Kafka.csv kafka_2.13-3.1.0.tgz	[75]	1./kafka_2.13-3.1.0/bin/	kafka-topics.shdescr	ibebootstrap-server 127.0.	0.1:0092topic Dropout-train 0.1:9092topic Dropout-test
		Topic: Propout-train	TopicId: woiCELV-THaw20	rbV0210A PartitionCount: 1	ReplicationFactor: 1 Configs: segment
		Topic: Dropout-test Topic: Dropout-t	TopicId: tYTNTIZyQ7aqvE est Partition: 0	pddaYcxg PartitionCount: 2 Leader: 0 Replicas: 0	Isr: 0 ReplicationFactor: 1 Configs: segment Isr: 0 Isr: 0
		Topic: Dropout-t	est Partition: 1	Leader: 0 Replicas: 0	Isr: 0
	2 [76]	#!curl -sSOL https://arc	hive.ics.uci.edu/ml/mac	hine-learning-databases/00279	>/SUSV.csv.gz ↑ ↓ 00 日 ☆ [] 首 1
	<u> </u>	from google.colab import	files		
🚫 🍐 Kafka_Implementation.ipynb	☆				🗖 Comment 🛛 😫 Share 🔹 🔀
File Edit View Insert Runtime Tools					RAM
Files 🖸 🗙	+ Cod	e + Text			Disk Editing
🐼 🖾 5J 🖬	<pre>[91]</pre>	OPTIMIZER="adam"			
CII		LOSS=tf.keras.losses.Bin METRICS=['accuracy']	aryCrossentropy(from_lo	gits=True)	
drive		EPOCHS=10			
 kafka_2.13-3.1.0 sample_data 	× [02]	<pre># design/build the model</pre>			
Apriori Algorithm and Kafka (1).csv	On [Jac]	<pre>model = tf.keras.Sequent</pre>	ial([
Apriori Algorithm and Kafka (2).csv		<pre>tf.keras.layers.Input(tf.keras.layers.Dense(</pre>	<pre>shape=(NUM_COLUMNS,)), 128, activation='relu')</pre>		
Apriori Algorithm and Kafka.csv		tf.keras.layers.Dropou	t(0.2),		
kafka_2.13-3.1.0.tgz		tf.keras.layers.Dropou	<pre>256, activation='relu') t(0.4),</pre>	,	
		<pre>tf.keras.layers.Dense(tf.keras.layers.Dropou</pre>	<pre>128, activation='relu') t(0,4)</pre>	,	
			 activation='sigmoid')	
])			
		<pre>print(model.summary())</pre>			
		Model: "sequential_1"			
		Layer (type)	Output Shape	Param #	
		dense_4 (Dense)	(None, 128)	4480	
Copy of Student_DropOut.ipy					🗖 Comment 🔉 Share 🏟 候
File Edit View Insert Runtime Tools		All changes saved e + Text			
Files CX					Disk Editing
🧭 🖾 51 🖬	°s [33]	dropout (Dropout)	(None, 128)	0	
Cu		dense_1 (Dense)	(None, 256)	33024	
 drive kafka 2 13-3 1 0 		dropout_1 (Dropout)	(None, 256)	0	
 kafka_2.13-3.1.0 sample_data 		dense_2 (Dense)	(None, 128)	32896	
Apriori Algorithm and Kafka (1).csv		dropout_2 (Dropout)	(None, 128)	0	
 Apriori Algorithm and Kafka.csv Apriori algm (1).csv 		dense_3 (Dense)	(None, 1)	129	
Apriori algm.csv		Total params: 70,529			
kafka_2.13-3.1.0.tgz		Trainable params: 70,529 Non-trainable params: 0			
		None			
	2 [34]	# compile the model			
	0.0	<pre>model.compile(optimizer=</pre>	OPTIMIZER, loss=LOSS, m	etrics=METRICS)	
					^ ↓ ⊕ 🗖 ‡ 🖟 📋 🗄
	0	# fit the model			
		<pre># fit the model model.fit(train_ds,epoch Epoch 1/10</pre>	s=EPOCHS)		



International Journal for Research in Applied Science & Engineering Technology (IJRASET)

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538 Volume 10 Issue VII July 2022- Available at www.ijraset.com

IX. CONCLUSION

We use the apriori algorithm in this paper to predict and analyse student database which calculates the confidence and support with L1 and L2 to perform apriori algorithm. We also introduce the term called kafka which does the stream processing, In the future we are combining the apriori algorithm with Hash-based technique, Transaction Reduction, Portioning, Sampling, and Dynamic item counting.

The authors are also willing to collaborate on data from tests and examinations for each course in the future in order to determine what types of students succeed in what types of courses. It may specify the types of courses that are tailored to each student's model who shares similar characteristics. It can also generate a variety of multi - dimensional reports and reshape pedagogical practises. learning paths.

REFERENCES

- [1] Samrat Singh, Dr. Vikesh Kumar, "Performance Analysis of Engineering Students for achievement mistreatment Classification data processing Techniques ",IJCSET Feb 2013.
- M. Goyal and R. Vohra, "Applications of information Mining in Higher Education", IJCSI International Journal of engineering problems, Vol. 9, Issue2, No 1, March 2012.
- [3] Jason Brownlee, "How to avoid wasting Your Machine Learning Model and create Predictions in Weka", August 3, 2016.
- [4] Neelam Naik & Seema Purohit, "Prediction of ultimate Result and Placement of scholars mistreatment Classification Algorithm"International Journal of pc Applications (0975 – 8887) Volume 56– No.12, Gregorian calendar month 2012.
- [5] Tirumalasetty, Sudhir, A. Aruna, A. Padmini, D. Vijaya Sagaru, and A. Tejeswini. "An increased Apriori with interest of Patterns mistreatment cSupport and rSupport." International Journal of engineering and Mobile Computing ten, no. seven (July 2021): 20–27. <u>http://dx.doi.org/10.47760/ijcsmc.2021.v10i07.003</u>.
- [6] Cortez P. and timberland A. (2008). mistreatment data processing to Predict Secondary Student Performance. In EUROSIS, A. Brito and J. Teixeira (Eds.), pp.5 -12.











45.98



IMPACT FACTOR: 7.129







INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 🕓 (24*7 Support on Whatsapp)