iJRASET

International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

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

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

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*

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

Keerti Ghodke[1], Rashmi Patil[2], Indira Umarji[3], Dr Vidyagouri Hemadri[4], Dr U P Kulkarni[5]
*[1]Student, [2, 3, 4]Assit Professor, [5]Professor, 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 streams: Take immediate action on knowledge and insight from actual data streaming platforms such as Kafka. Make your 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.

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

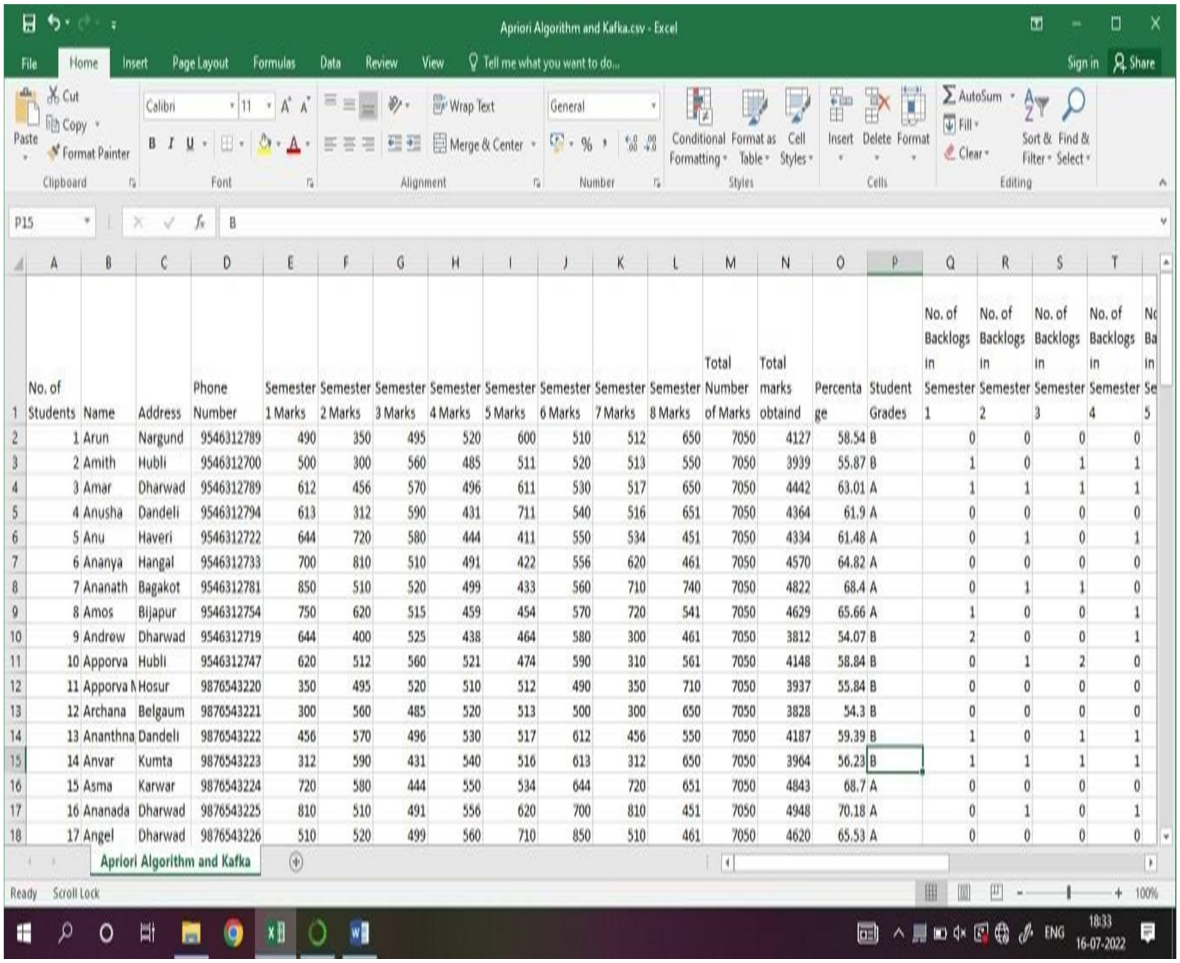
Create centralized dataset to improve the capability of level-wise frequent generation student dataset, an vital Apriori property is being used, which aids in reducing the search time. space. All subsets of a frequent student dataset should be common (Apriori property).

## V. DATSET DESCRIPTION

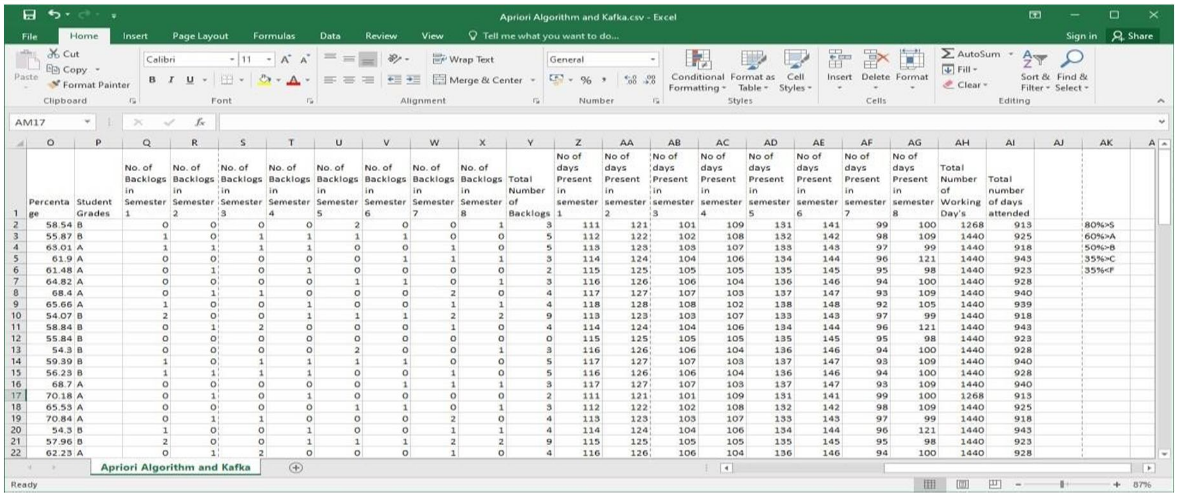
The followings are the steps involves design and dataset.

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



| No. of Students | Name | Address | Phone Number | Semester 1 Marks | Semester 2 Marks | Semester 3 Marks | Semester 4 Marks | Semester 5 Marks | Semester 6 Marks | Semester 7 Marks | Semester 8 Marks | Total Number of Marks | Total marks obtaind | Percentage | Student Grades | No. of Backlogs in Semester 1 | No. of Backlogs in Semester 2 | No. of Backlogs in Semester 3 | No. of Backlogs in Semester 4 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | Arun | Nargund | 9546312789 | 490 | 350 | 495 | 520 | 600 | 510 | 512 | 650 | 7050 | 4127 | 58.54 | B | 0 | 0 | 0 | 0 |
| 2 | Amith | Hubli | 9546312700 | 500 | 300 | 560 | 485 | 511 | 520 | 513 | 550 | 7050 | 3939 | 55.87 | B | 1 | 0 | 1 | 1 |
| 3 | Amar | Dharwad | 9546312789 | 612 | 456 | 570 | 496 | 611 | 530 | 517 | 650 | 7050 | 4442 | 63.01 | A | 1 | 1 | 1 | 1 |
| 4 | Anusha | Dandeli | 9546312794 | 613 | 312 | 590 | 431 | 711 | 540 | 516 | 651 | 7050 | 4364 | 61.9 | A | 0 | 0 | 0 | 0 |
| 5 | Anu | Haveri | 9546312722 | 644 | 720 | 580 | 444 | 411 | 550 | 534 | 451 | 7050 | 4334 | 61.48 | A | 0 | 1 | 0 | 1 |
| 6 | Ananya | Hangal | 9546312733 | 700 | 810 | 510 | 491 | 422 | 556 | 620 | 461 | 7050 | 4570 | 64.82 | A | 0 | 0 | 0 | 0 |
| 7 | Ananath | Bagakot | 9546312781 | 850 | 510 | 520 | 499 | 433 | 560 | 710 | 740 | 7050 | 4822 | 68.4 | A | 0 | 1 | 1 | 0 |
| 8 | Amos | Bijapur | 9546312754 | 750 | 620 | 515 | 459 | 454 | 570 | 720 | 541 | 7050 | 4629 | 65.66 | A | 1 | 0 | 0 | 1 |
| 9 | Andrew | Dharwad | 9546312719 | 644 | 400 | 525 | 438 | 464 | 580 | 300 | 461 | 7050 | 3812 | 54.07 | B | 2 | 0 | 0 | 1 |
| 10 | Apporva | Hubli | 9546312747 | 620 | 512 | 560 | 521 | 474 | 590 | 310 | 561 | 7050 | 4148 | 58.84 | B | 0 | 1 | 2 | 0 |
| 11 | Apporva N | Hosur | 9876543220 | 350 | 495 | 520 | 510 | 512 | 490 | 350 | 710 | 7050 | 3937 | 55.84 | B | 0 | 0 | 0 | 0 |
| 12 | Archana | Belgaum | 9876543221 | 300 | 560 | 485 | 520 | 513 | 500 | 300 | 650 | 7050 | 3828 | 54.3 | B | 0 | 0 | 0 | 0 |
| 13 | Ananthna | Dandeli | 9876543222 | 456 | 570 | 496 | 530 | 517 | 612 | 456 | 550 | 7050 | 4187 | 59.39 | B | 1 | 0 | 1 | 1 |
| 14 | Anvar | Kumta | 9876543223 | 312 | 590 | 431 | 540 | 516 | 613 | 312 | 650 | 7050 | 3964 | 56.23 | B | 1 | 1 | 1 | 1 |
| 15 | Asma | Karwar | 9876543224 | 720 | 580 | 444 | 550 | 534 | 644 | 720 | 651 | 7050 | 4843 | 68.7 | A | 0 | 0 | 0 | 0 |
| 16 | Ananada | Dharwad | 9876543225 | 810 | 510 | 491 | 556 | 620 | 700 | 810 | 451 | 7050 | 4948 | 70.18 | A | 0 | 1 | 0 | 1 |
| 17 | Angel | Dharwad | 9876543226 | 510 | 520 | 499 | 560 | 710 | 850 | 510 | 461 | 7050 | 4620 | 65.53 | A | 0 | 0 | 0 | 0 |



| | O | P | Q | R | S | T | U | V | W | X | Y | Z | AA | AB | AC | AD | AE | AF | AG | AH | AI | AJ | AK |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | Percentage | Student Grades | No. of Backlogs in Semester 1 | No. of Backlogs in Semester 2 | No. of Backlogs in Semester 3 | No. of Backlogs in Semester 4 | No. of Backlogs in Semester 5 | No. of Backlogs in Semester 6 | No. of Backlogs in Semester 7 | No. of Backlogs in Semester 8 | Total Number of Backlogs | No of days Present in semester 1 | No of days Present in semester 2 | No of days Present in semester 3 | No of days Present in semester 4 | No of days Present in semester 5 | No of days Present in semester 6 | No of days Present in semester 7 | No of days Present in semester 8 | Total Number of Working Day's | Total number of days attended | | |
| 2 | 58.54 | B | 0 | 0 | 0 | 0 | 2 | 0 | 0 | 1 | 3 | 111 | 121 | 101 | 109 | 131 | 141 | 99 | 100 | 1268 | 913 | | 80%>S |
| 3 | 55.87 | B | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 5 | 112 | 122 | 102 | 108 | 132 | 142 | 98 | 109 | 1440 | 925 | | 60%>A |
| 4 | 63.01 | A | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 5 | 113 | 123 | 103 | 107 | 133 | 143 | 97 | 99 | 1440 | 918 | | 50%>B |
| 5 | 61.9 | A | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 3 | 114 | 124 | 104 | 106 | 134 | 144 | 96 | 121 | 1440 | 943 | | 35%>C |
| 6 | 61.48 | A | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 2 | 115 | 125 | 105 | 105 | 135 | 145 | 95 | 98 | 1440 | 923 | | 35%<F |
| 7 | 64.82 | A | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 3 | 116 | 126 | 106 | 104 | 136 | 146 | 94 | 100 | 1440 | 928 | | |
| 8 | 68.4 | A | 0 | 1 | 1 | 0 | 0 | 0 | 2 | 0 | 4 | 117 | 127 | 107 | 103 | 137 | 147 | 93 | 109 | 1440 | 940 | | |
| 9 | 65.66 | A | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 4 | 118 | 128 | 108 | 102 | 138 | 148 | 92 | 105 | 1440 | 939 | | |
| 10 | 54.07 | B | 2 | 0 | 0 | 1 | 1 | 1 | 2 | 2 | 9 | 113 | 123 | 103 | 107 | 133 | 143 | 97 | 99 | 1440 | 918 | | |
| 11 | 58.84 | B | 0 | 1 | 2 | 0 | 0 | 0 | 1 | 0 | 4 | 114 | 124 | 104 | 106 | 134 | 144 | 96 | 121 | 1440 | 943 | | |
| 12 | 55.84 | B | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 115 | 125 | 105 | 105 | 135 | 145 | 95 | 98 | 1440 | 923 | | |
| 13 | 54.3 | B | 0 | 0 | 0 | 0 | 2 | 0 | 0 | 1 | 3 | 116 | 126 | 106 | 104 | 136 | 146 | 94 | 100 | 1440 | 928 | | |
| 14 | 59.39 | B | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 5 | 117 | 127 | 107 | 103 | 137 | 147 | 93 | 109 | 1440 | 940 | | |
| 15 | 56.23 | B | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 5 | 116 | 126 | 106 | 104 | 136 | 146 | 94 | 100 | 1440 | 928 | | |
| 16 | 68.7 | A | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 3 | 117 | 127 | 107 | 103 | 137 | 147 | 93 | 109 | 1440 | 940 | | |
| 17 | 70.18 | A | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 2 | 111 | 121 | 101 | 109 | 131 | 141 | 99 | 100 | 1268 | 913 | | |
| 18 | 65.53 | A | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 3 | 112 | 122 | 102 | 108 | 132 | 142 | 98 | 109 | 1440 | 925 | | |
| 19 | 70.84 | A | 0 | 1 | 1 | 0 | 0 | 0 | 2 | 0 | 4 | 113 | 123 | 103 | 107 | 133 | 143 | 97 | 99 | 1440 | 918 | | |
| 20 | 54.3 | B | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 4 | 114 | 124 | 104 | 106 | 134 | 144 | 96 | 121 | 1440 | 943 | | |
| 21 | 57.96 | B | 2 | 0 | 0 | 1 | 1 | 1 | 2 | 2 | 9 | 115 | 125 | 105 | 105 | 135 | 145 | 95 | 98 | 1440 | 923 | | |
| 22 | 62.23 | A | 0 | 1 | 2 | 0 | 0 | 0 | 1 | 0 | 4 | 116 | 126 | 106 | 104 | 136 | 146 | 94 | 100 | 1440 | 928 | | |

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 => 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 {the variety|the amount|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 => 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)

## 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.

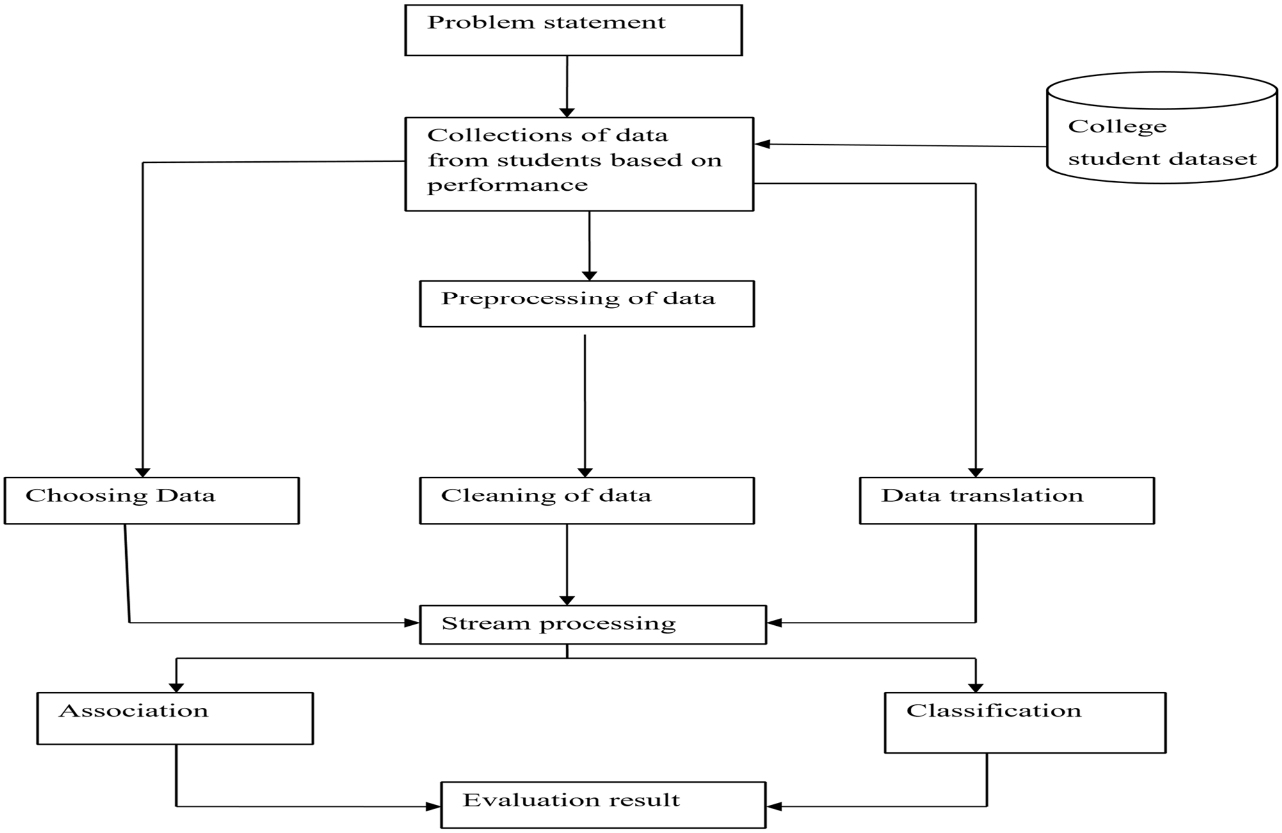
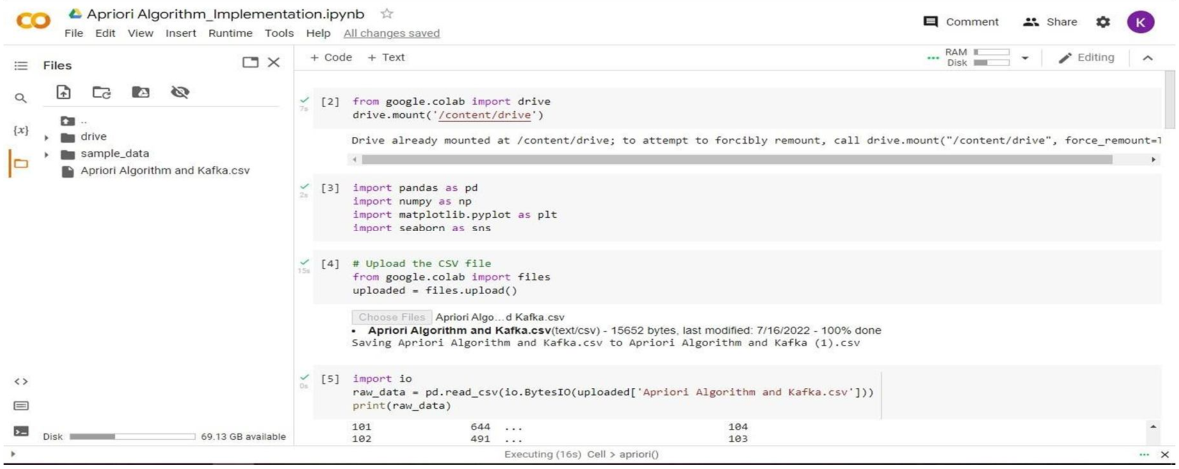


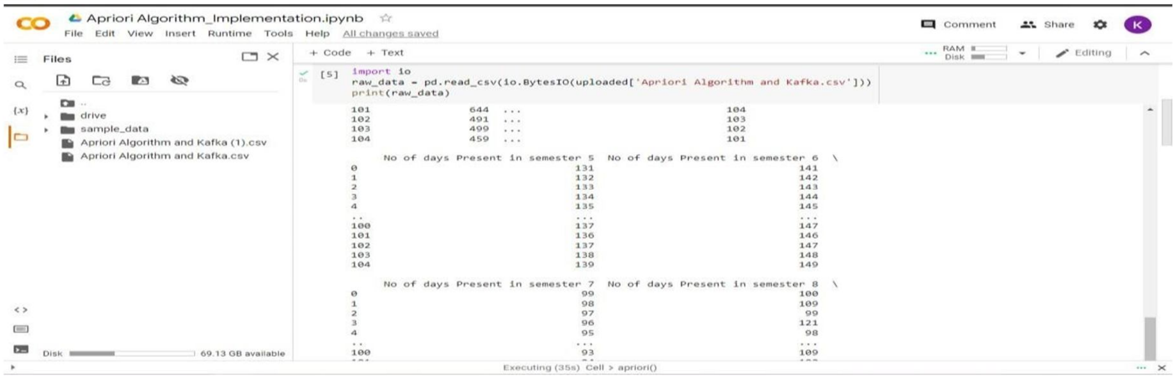
Figure 2: System Design

## VIII. PERFORMANCE ANALYSIS

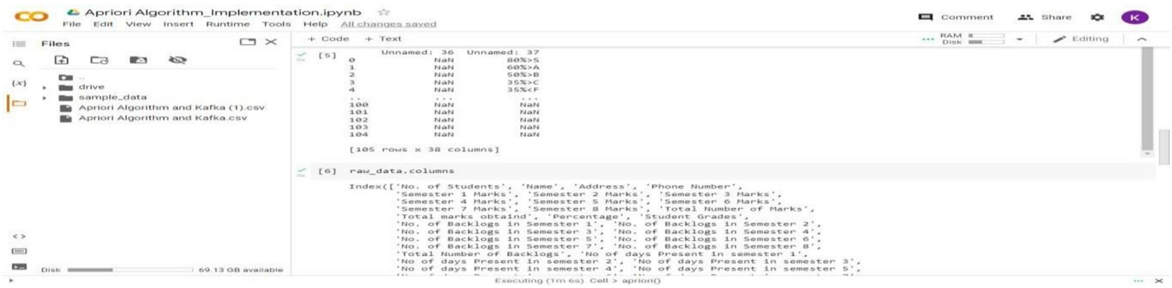
Fig 3 Shows the apriori algorithm to predict and analysis of student dataset and Kafka Implementation...



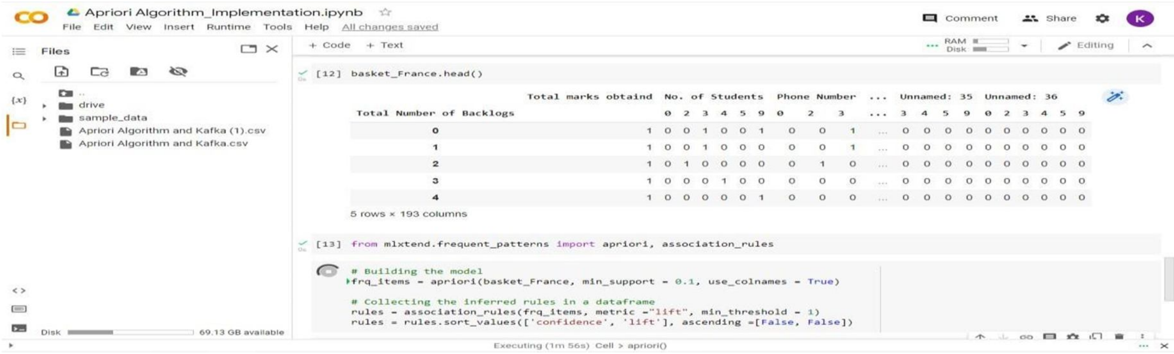
```
[2] from google.colab import drive
    drive.mount('/content/drive')

    Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force_remount=1

[3] import pandas as pd
    import numpy as np
    import matplotlib.pyplot as plt
    import seaborn as sns

[4] # Upload the CSV file
    from google.colab import files
    uploaded = files.upload()

    • Apriori Algorithm and Kafka.csv(text/csv) - 15652 bytes, last modified: 7/16/2022 - 100% done
    Saving Apriori Algorithm and Kafka.csv to Apriori Algorithm and Kafka (1).csv

[5] import io
    raw_data = pd.read_csv(io.BytesIO(uploaded['Apriori Algorithm and Kafka.csv']))
    print(raw_data)

    101            644  ...                      104
    102            491  ...                      103
```


Apriori Algorithm_Implementation.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment Share
Files
+ Code + Text
RAM Disk
Editing
drive
sample_data
Apriori Algorithm and Kafka (1).csv
Apriori Algorithm and Kafka.csv
Disk 69.13 GB available
Executing (35s) Cell > apriori()

```
[5] import io
    raw_data = pd.read_csv(io.BytesIO(uploaded['Apriori Algorithm and Kafka.csv']))
    print(raw_data)

    101           644  ...                                        104
    102           491  ...                                        103
    103           499  ...                                        102
    104           459  ...                                        101

         No of days Present in semester 5  No of days Present in semester 6  \
    0                                 131                               141
    1                                 132                               142
    2                                 133                               143
    3                                 134                               144
    4                                 135                               145
    ..                                ...                               ...
    100                               137                               147
    101                               136                               146
    102                               137                               147
    103                               138                               148
    104                               139                               149

         No of days Present in semester 7  No of days Present in semester 8  \
    0                                  99                               100
    1                                  98                               109
    2                                  97                                99
    3                                  96                               121
    4                                  95                                98
    ..                                ...                               ...
    100                                93                               109
```

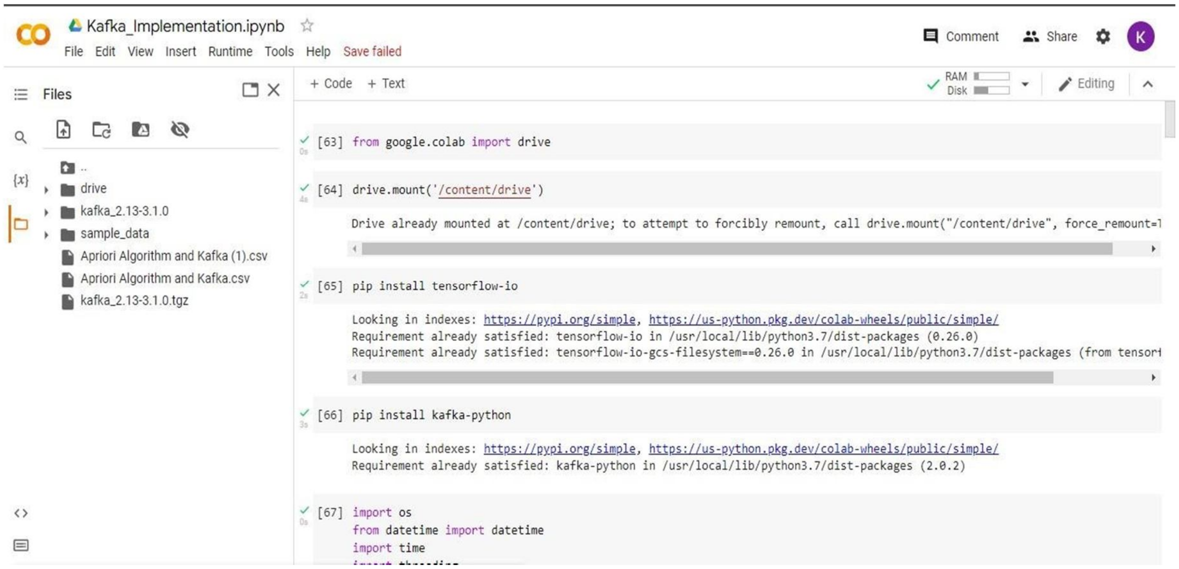

Apriori Algorithm_Implementation.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment Share
Files
+ Code + Text
Editing
drive
sample_data
Apriori Algorithm and Kafka (1).csv
Apriori Algorithm and Kafka.csv
Disk 69.13 GB available
Executing (1m 6s) Cell > apriori()

```
[5]      Unnamed: 36  Unnamed: 37
    0            NaN       80%>S
    1            NaN       60%>A
    2            NaN       50%>B
    3            NaN       35%>C
    4            NaN       35%<F
    ..           ...         ...
    100          NaN         NaN
    101          NaN         NaN
    102          NaN         NaN
    103          NaN         NaN
    104          NaN         NaN

    [105 rows x 38 columns]

[6] raw_data.columns

    Index(['No. of Students', 'Name', 'Address', 'Phone Number',
           'Semester 1 Marks', 'Semester 2 Marks', 'Semester 3 Marks',
           'Semester 4 Marks', 'Semester 5 Marks', 'Semester 6 Marks',
           'Semester 7 Marks', 'Semester 8 Marks', 'Total Number of Marks',
           'Total marks obtaind', 'Percentage', 'Student Grades',
           'No. of Backlogs in Semester 1', 'No. of Backlogs in Semester 2',
           'No. of Backlogs in Semester 3', 'No. of Backlogs in Semester 4',
           'No. of Backlogs in Semester 5', 'No. of Backlogs in Semester 6',
           'No. of Backlogs in Semester 7', 'No. of Backlogs in Semester 8',
           'Total Number of Backlogs', 'No of days Present in semester 1',
           'No of days Present in semester 2', 'No of days Present in semester 3',
           'No of days Present in semester 4', 'No of days Present in semester 5',
```

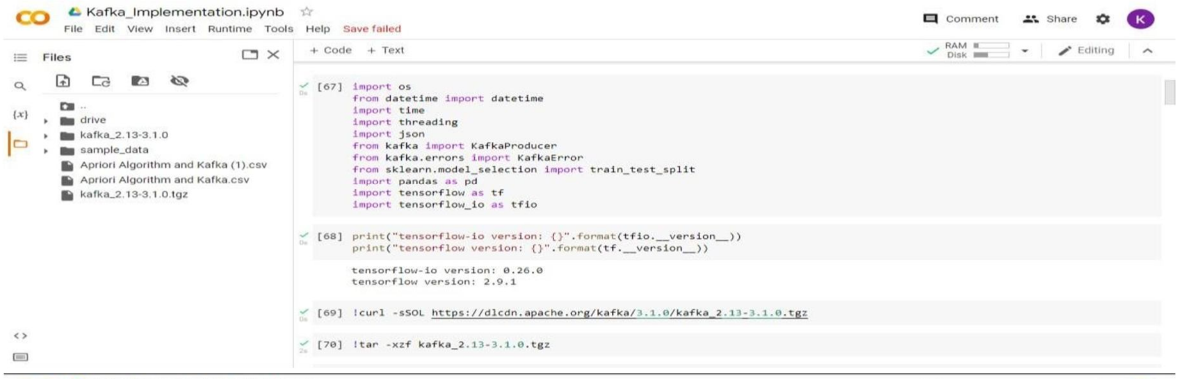

Apriori Algorithm_Implementation.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Comment Share
Files
+ Code + Text
Editing
drive
sample_data
Apriori Algorithm and Kafka (1).csv
Apriori Algorithm and Kafka.csv
Disk 69.13 GB available
Executing (1m 56s) Cell > apriori()

```
[12] basket_France.head()

                          Total marks obtaind  No. of Students      Phone Number  ...  Unnamed: 35  Unnamed: 36
     Total Number of Backlogs                  0 2 3 4 5 9     0  2  3  ...  3 4 5 9  0 2 3 4 5 9
                0                           1  0 0 1 0 0 1     0  0  1  ...  0 0 0 0  0 0 0 0 0 0
                1                           1  0 0 1 0 0 0     0  0  1  ...  0 0 0 0  0 0 0 0 0 0
                2                           1  0 1 0 0 0 0     0  1  0  ...  0 0 0 0  0 0 0 0 0 0
                3                           1  0 0 0 1 0 0     0  0  0  ...  0 0 0 0  0 0 0 0 0 0
                4                           1  0 0 0 0 0 1     0  0  0  ...  0 0 0 0  0 0 0 0 0 0
     5 rows × 193 columns

[13] from mlxtend.frequent_patterns import apriori, association_rules

     # Building the model
     frq_items = apriori(basket_France, min_support = 0.1, use_colnames = True)

     # Collecting the inferred rules in a dataframe
     rules = association_rules(frq_items, metric ="lift", min_threshold = 1)
     rules = rules.sort_values(['confidence', 'lift'], ascending =[False, False])
```

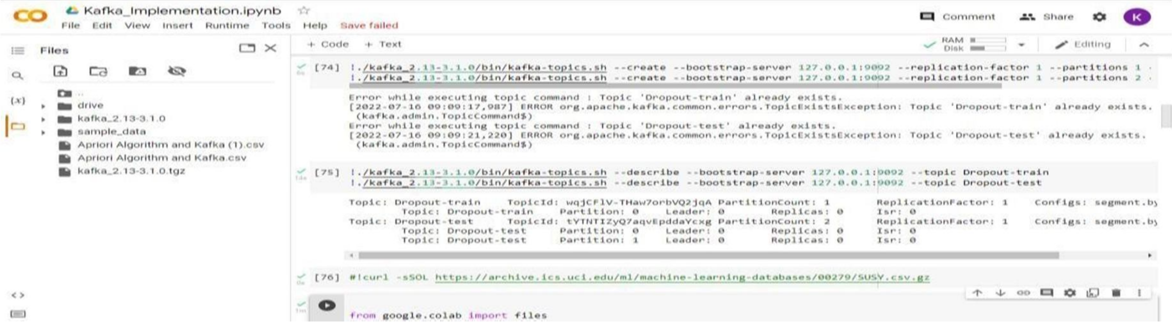
Perform stream processing: Use any open source project such as TensorFlow, NumPy, SciPy, or Matplotlib, making it easy to run machine learning models on streaming data. The following figure shows Streaming data with kafka.


Kafka_Implementation.ipynb
File Edit View Insert Runtime Tools Help Save failed
Comment Share
Files
+ Code + Text
RAM Disk
Editing
drive
kafka_2.13-3.1.0
sample_data
Apriori Algorithm and Kafka (1).csv
Apriori Algorithm and Kafka.csv
kafka_2.13-3.1.0.tgz

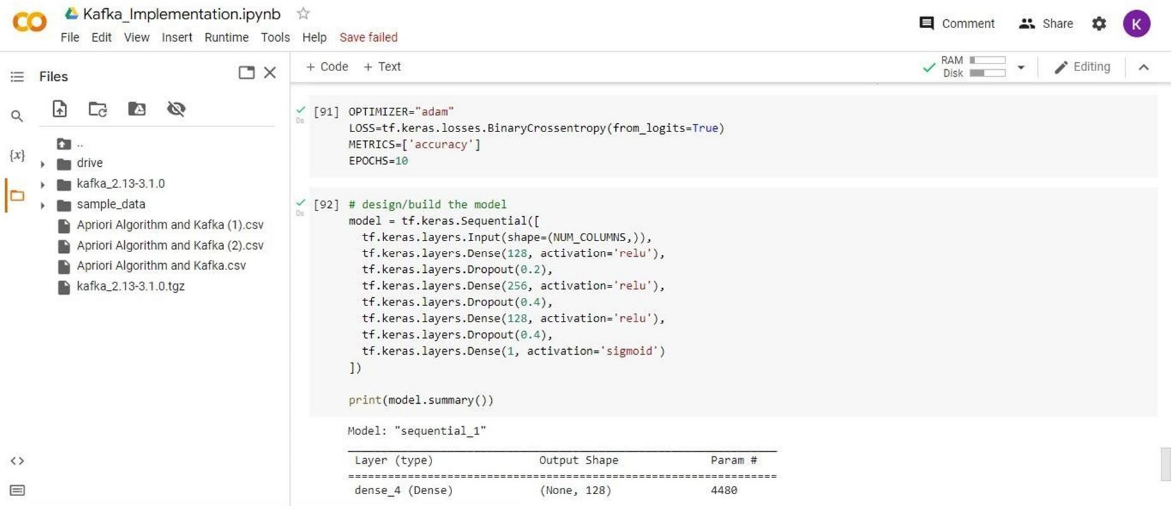
```
[63] from google.colab import drive

[64] drive.mount('/content/drive')

     Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force_remount=T

[65] pip install tensorflow-io

     Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
     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 tensorf

[66] pip install kafka-python

     Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
     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
```


Kafka_Implementation.ipynb
File Edit View Insert Runtime Tools Help Save failed
Comment Share
Files
+ Code + Text
RAM Disk
Editing
drive
kafka_2.13-3.1.0
sample_data
Apriori Algorithm and Kafka (1).csv
Apriori Algorithm and Kafka.csv
kafka_2.13-3.1.0.tgz

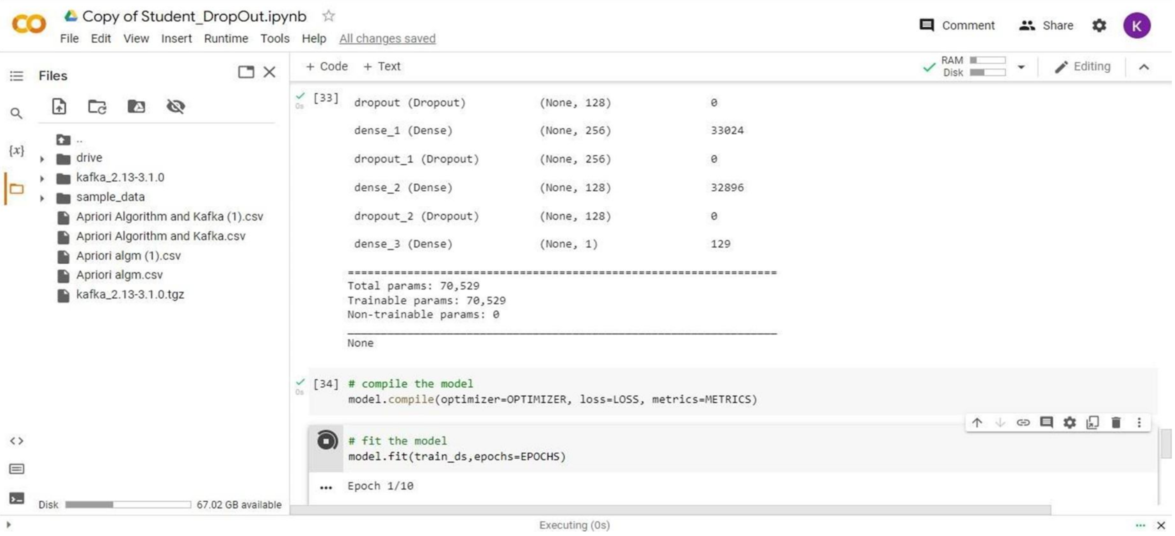
```
[67] import os
     from datetime import datetime
     import time
     import threading
     import json
     from kafka import KafkaProducer
     from kafka.errors import KafkaError
     from sklearn.model_selection import train_test_split
     import pandas as pd
     import tensorflow as tf
     import tensorflow_io as tfio

[68] print("tensorflow-io version: {}".format(tfio.__version__))
     print("tensorflow version: {}".format(tf.__version__))

     tensorflow-io version: 0.26.0
     tensorflow version: 2.9.1

[69] !curl -sSOL https://dlcdn.apache.org/kafka/3.1.0/kafka_2.13-3.1.0.tgz

[70] !tar -xzf kafka_2.13-3.1.0.tgz
```


Kafka_Implementation.ipynb
File Edit View Insert Runtime Tools Help Save failed
Comment Share
Files
+ Code + Text
RAM Disk
Editing
drive
kafka_2.13-3.1.0
sample_data
Apriori Algorithm and Kafka (1).csv
Apriori Algorithm and Kafka.csv
kafka_2.13-3.1.0.tgz

```
[74] !./kafka_2.13-3.1.0/bin/kafka-topics.sh --create --bootstrap-server 127.0.0.1:9092 --replication-factor 1 --partitions 1
     !./kafka_2.13-3.1.0/bin/kafka-topics.sh --create --bootstrap-server 127.0.0.1:9092 --replication-factor 1 --partitions 2

     Error while executing topic command : Topic 'Dropout-train' already exists.
     [2022-07-16 09:09:17,987] ERROR org.apache.kafka.common.errors.TopicExistsException: Topic 'Dropout-train' already exists.
      (kafka.admin.TopicCommand$)
     Error while executing topic command : Topic 'Dropout-test' already exists.
     [2022-07-16 09:09:21,220] ERROR org.apache.kafka.common.errors.TopicExistsException: Topic 'Dropout-test' already exists.
      (kafka.admin.TopicCommand$)

[75] !./kafka_2.13-3.1.0/bin/kafka-topics.sh --describe --bootstrap-server 127.0.0.1:9092 --topic Dropout-train
     !./kafka_2.13-3.1.0/bin/kafka-topics.sh --describe --bootstrap-server 127.0.0.1:9092 --topic Dropout-test

     Topic: Dropout-train    TopicId: wqjCFlV-THaw7orbVQ2jqA PartitionCount: 1       ReplicationFactor: 1    Configs: segment.by
             Topic: Dropout-train    Partition: 0    Leader: 0       Replicas: 0     Isr: 0
     Topic: Dropout-test     TopicId: tYTNTIZyQ7aqvEpddaYcxg PartitionCount: 2       ReplicationFactor: 1    Configs: segment.by
             Topic: Dropout-test     Partition: 0    Leader: 0       Replicas: 0     Isr: 0
             Topic: Dropout-test     Partition: 1    Leader: 0       Replicas: 0     Isr: 0

[76] #!curl -sSOL https://archive.ics.uci.edu/ml/machine-learning-databases/00279/SUSY.csv.gz

     from google.colab import files
```


Kafka_Implementation.ipynb
File Edit View Insert Runtime Tools Help Save failed
Comment Share
Files
+ Code + Text
RAM Disk
Editing
drive
kafka_2.13-3.1.0
sample_data
Apriori Algorithm and Kafka (1).csv
Apriori Algorithm and Kafka (2).csv
Apriori Algorithm and Kafka.csv
kafka_2.13-3.1.0.tgz

```
[91] OPTIMIZER="adam"
     LOSS=tf.keras.losses.BinaryCrossentropy(from_logits=True)
     METRICS=['accuracy']
     EPOCHS=10

[92] # design/build the model
     model = tf.keras.Sequential([
       tf.keras.layers.Input(shape=(NUM_COLUMNS,)),
       tf.keras.layers.Dense(128, activation='relu'),
       tf.keras.layers.Dropout(0.2),
       tf.keras.layers.Dense(256, activation='relu'),
       tf.keras.layers.Dropout(0.4),
       tf.keras.layers.Dense(128, activation='relu'),
       tf.keras.layers.Dropout(0.4),
       tf.keras.layers.Dense(1, activation='sigmoid')
     ])

     print(model.summary())

     Model: "sequential_1"
     _________________________________________________________________
      Layer (type)                Output Shape              Param #
     =================================================================
      dense_4 (Dense)             (None, 128)               4480
```



```
[33]  dropout (Dropout)           (None, 128)               0

      dense_1 (Dense)             (None, 256)               33024

      dropout_1 (Dropout)         (None, 256)               0

      dense_2 (Dense)             (None, 128)               32896

      dropout_2 (Dropout)         (None, 128)               0

      dense_3 (Dense)             (None, 1)                 129

     =================================================================
     Total params: 70,529
     Trainable params: 70,529
     Non-trainable params: 0
     _________________________________________________________________
     None

[34] # compile the model
     model.compile(optimizer=OPTIMIZER, loss=LOSS, metrics=METRICS)

     # fit the model
     model.fit(train_ds,epochs=EPOCHS)

 ... Epoch 1/10
```

## 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.

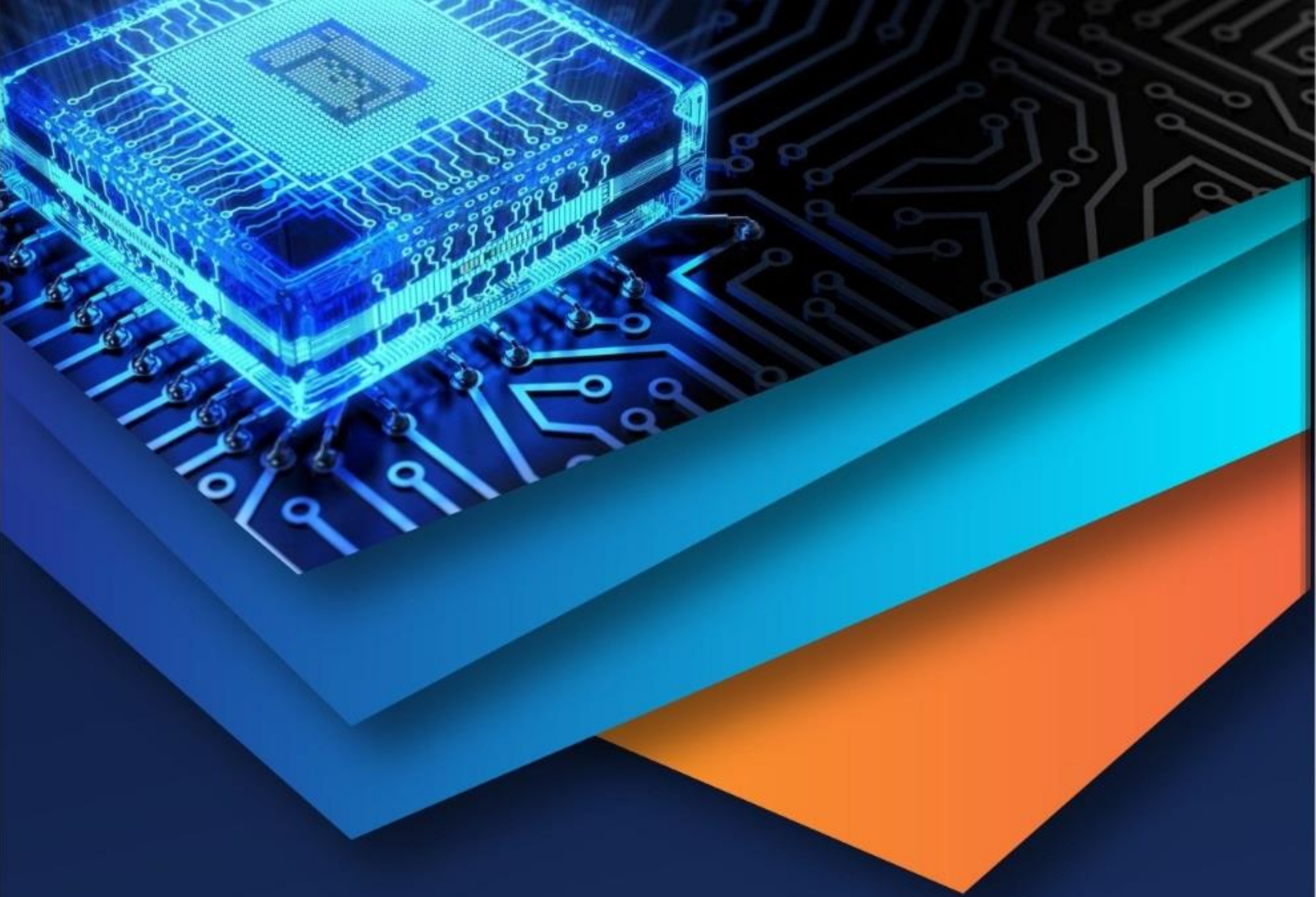
[2] M. Goyal and R. Vohra, “Applications of information Mining in Higher Education”,IJCSI International Journal of engineering problems, Vol. 9, Issue2, No 1, March 2012.

[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. http://dx.doi.org/10.47760/ijcsmc.2021.v10i07.003.

[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.



10.22214/IJRASET

INDEX COPERNICUS

45.98

IMPACT FACTOR:
7.129

TOGETHER WE REACH THE GOAL

IMPACT FACTOR:
7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY