

# Aprioriiiiii.pdf

*by* Raisa Turnitin

---

**Submission date:** 08-Jun-2024 07:09AM (UTC+0530)

**Submission ID:** 2395545606

**File name:** Aprioriiiiii.pdf (500.12K)

**Word count:** 1062

**Character count:** 6945

# MENGIMPLEMENTASIKAN ALGORITMA APRIORI DALAM SUPERMARKET SALES DATA

1<sup>st</sup> Jovinka Aphellia Salva  
Program Studi Ilmu Komputer  
Universitas Sumatera Utara  
Medan, Indonesia  
jovinkav@gmail.com

2<sup>nd</sup> Najwa Afifi Situmorang  
Program Studi Ilmu Komputer  
Universitas Sumatera Utara  
Medan, Indonesia  
najwaafifi121@gmail.com

3<sup>rd</sup> Dewi Sartika Br Ginting S.Kom.,  
M.Kom  
Program Studi Ilmu Komputer  
Universitas Sumatera Utara  
Medan, Indonesia  
dewidintingdg90@gmail.com

**Abstrak** – Dengan menggunakan teknik penambangan aturan asosiasi, kami bertujuan untuk mendistribusikan kelompok item yang umum muncul dan menghasilkan aturan asosiasi yang bermakna yang dapat memberikan wawasan berharga tentang perilaku pembelian pelanggan.

Kumpulan data terdiri dari detail selanjutnya seperti ID faktur, lini produk, dan model pembayaran dan telah diproses sebelumnya untuk menyederhanakan proses utama. Analisis kami mengungkapkan bahwasanya nilai support ke Fashion accessories (0,178) yang merupakan nilai support tertinggi di antara lainnya.

**Kata Kunci**—asosiasi, apriori, penjualan, data

## I. PENDAHULUAN

Algoritma apriori adalah proses penggalian informasi dari database dan seringkali menghasilkan elemen atau kumpulan elemen dan kandidat untuk membentuk asosiasi rule mining untuk memperoleh nilai dukungan minimum dan keyakinan minimum.

Untuk database yang cukup besar, algoritma apriori akan menghasilkan pola item/itemset yang sering digunakan dalam jumlah besar, karena algoritma ini perlu membuat kandidat dan mencatat database yang berulang.

Pada penelitian ini, kami menerapkan algoritma Aprior pada dataset toko supermarket yang berisi data penjualan transaksi supermarket. Tujuan utama penelitian ini adalah untuk menemukan kelompok produk umum yang sering dibeli konsumen secara bersamaan. Dengan mengidentifikasi perilaku pembelian ini, pemilik supermarket dapat memperoleh wawasan tentang taktik pemasaran dan teknik penjualan mereka untuk meningkatkan kinerja secara keseluruhan.

Sebelum menerapkan algoritma Aprior, kami melakukan penambangan data awal untuk memahami karakteristik kumpulan data dan mengidentifikasi informasi penting seperti produk terlaris, metode pembayaran terpopuler, dan penjualan berdasarkan jenis kelamin dan bulan. Tujuan dari langkah ini adalah untuk mendapatkan wawasan tentang pola pembelian pelanggan dan menyiapkan data untuk analisis lebih lanjut.

## II. TEORI

Apriori adalah algoritme untuk menambang kumpulan target dan mempelajari aturan asosiasi secara berulang melalui database relasional. Prosesnya dengan mengidentifikasi objek tertentu yang ditemukan berulang kali dalam database dan kemudian secara bertahap meningkatkan jumlah objek tersebut hingga tersedia untuk dianalisis.

Menambang produk yang sering digunakan dan aturan asosiasinya dicapai melalui penggunaan algoritma Apriori. Secara umum, algoritma apriori bekerja pada

database yang berisi transaksi dalam jumlah besar. Misalnya barang konsumsi, tapi di Grand Bazaar.

Support, confidence, dan lift menjadi tiga komponen utama dalam proses data mining asosiasi menggunakan algoritma apriori.

Bentuk persentase jumlah kemunculan untuk kombinasi item tertentu disebut dengan nilai Support(s).

$$\text{Support}, s(X \rightarrow Y) = \frac{(X \cup Y)}{N}$$

Pentingnya nilai-nilai dukungan dalam aturan asosiasi ditekankan, karena tingkat dukungan yang rendah menunjukkan bahwa asosiasi jarang terjadi dalam kumpulan data (semua data peristiwa).

Menghitung persentase dari keakuratan aturan asosiasi yang akan dihasilkan disebut dengan nilai Confident(c).

$$\text{Confident}, c(X \rightarrow Y) = \frac{(X \cup Y)}{X}$$

Besarnya Y didefinisikan sebagai keyakinan yang tinggi untuk peristiwa yang mengandung X.

**Lift Ratio** adalah parameter yang mengukur kekuatan aturan asosiasi yang diciptakan oleh nilai-nilai dukungan dan keyakinan.

## III. METODOLOGI PENELITIAN

Penelitian ini menerapkan algoritma apriori dari Data Mining untuk mengekstrak data terkait penjualan supermarket.

Data yang digunakan adalah catatan penjualan supermarket yang berisi informasi tentang seluruh transaksi penjualan, seperti produk yang dibeli, metode pembayaran dan demografi pelanggan.

### 1. Data dan Preprocessing

#### 1.1. Sumber Data :

Dataset yang digunakan berasal dari file CSV Kaggle berjudul "supermarket\_sales.csv". Data yang ada pada dataset ini mencakup berbagai atribut seperti ID transaksi, kota, anggota, jenis kelamin, produk yang dibeli, harga, jumlah, tanggal, metode pembayaran, dan lainnya.

#### 1.2. Pembacaan Data :

Dataset dibaca menggunakan pustaka `pandas`, `numpy`, `matplotlib.pyplot`, `seaborn`, `itertools` dan `warnings` dengan perintah berikut :

```
# Import pustaka yang diperlukan
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import itertools
import warnings
warnings.filterwarnings("ignore")
from mxtend.frequent_patterns import apriori, association_rules
from tabulate import tabulate
```

### 1.3. Preprocessing Data :

Data diproses untuk mempersiapkannya bagi analisis lebih lanjut.

- Mengonversi kolom tanggal ke tipe datetime.
- Menambah kolom yang memisahkan informasi hari, bulan, dan tahun dari kolom tanggal.
- Menghapus kolom yang tidak diperlukan untuk analisis.

```
data['Date'] = pd.to_datetime(data['Date'])

months = ["january", "february", "march", "april", "may", "june", "july",
          "august", "septembre", "octobre", "novembre", "decembre"]

data["day"] = data["Date"].apply(lambda x : x.day)
data["Month Name"] = data["Date"].dt.month.apply(lambda x: months[x-1])
data["year"] = data["Date"].apply(lambda x : x.year)
```

## 2. Analisis Data Eksploratif

### 2.1. Visualisasi Jumlah Produk Yang Dibeli

```
val_counts = dict(data["Product line"].value_counts()[:10])

plt.figure(figsize=(12,6))
sns.barplot(x=list(val_counts.keys()), y=list(val_counts.values()),
            palette="Blues_r")
```

### 2.2. Visualisasi Fitur Metode Pembayaran

```
payment = dict(data.groupby("Payment")["Product line"]
               .count().sort_values(ascending=False))

explode = [0] * len(payment)
explode[1] = 0.01 if len(payment) > 1 else 0
explode[2] = 0.2 if len(payment) > 2 else 0

plt.figure(figsize=(10, 6))
plt.pie(payment.values(), labels=payment.keys(), explode=explode,
        colors=sns.color_palette("Set2")[:len(payment)], autopct='%2f%%')
plt.legend(loc='best')
plt.tight_layout()
plt.show()
```

### 2.3. Visualisasi Fitur Jenis Kelamin

```
gender = dict(data.groupby("Gender")["Product line"]
              .count().sort_values(ascending=False))

plt.figure(figsize=(10,6))
plt.pie(gender.values(), labels=gender.keys(), explode = [0, 0.01],
        colors = sns.color_palette("Set2")[5:7], autopct='%2f%%')
plt.tight_layout()
plt.legend()
plt.show()
```

### 2.4. Visualisasi Fitur Bulan

```
months = ["January", "February", "March", "April", "May", "June",
          "July", "August", "September", "October", "November", "December"]
month = dict(data.groupby("Month Name")["Product line"].count())

ordered_months = ["January", "February", "March", "April", "May", "June",
                  "July", "August", "September", "October", "November", "December"]
sorted_month = {k: month[k] for k in ordered_months if k in month}

plt.figure(figsize=(14, 6))
sns.barplot(x=list(sorted_month.keys()), y=list(sorted_month.values()),
            palette="Purples_r")
plt.title('Jumlah Product Line per Bulan')
plt.xlabel('Month')
plt.ylabel('Jumlah Product line')
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()
```

## 3. Preprocess Data (untuk Penambangan Data Pola)

### 3.1. Penjumlahan Tiap Produk Yang Dibeli

```
val_counts = data["Product line"].value_counts()
val_counts
```

### 3.2. Penampilan ID Transaksi

```
invoices = []
for action in data["Invoice ID"].unique():
    if action not in excluded:
        invoice = data[data["Invoice ID"] == action]["Product line"].tolist()
        if len(invoice) > 0:
            invoices.append(invoice)
```

## 4. Implementasi Algoritma Apriori

```
from itertools import combinations
```

```
def item_counter(data):
    counts = {}
    for action in data:
        for item in action:
            counts[item]=0
    for action in data:
        for item in action:
            counts[item] += 1

    return counts
```

```
def remove_non_sup(dic, min_sup):
    non_freq = []
    for k,v in dic.items():
        if v < min_sup:
            non_freq.append(k)
    [dic.pop(key) for key in non_freq]
    return dic
```

```
def check_valid_pairs(data, pairs):
    valid_pairs=[]
    for action in data:
        for pair in pairs:
            if all(x in action for x in pair):
                valid_pairs.append(pair)
    return list(set(valid_pairs))
```

## 5. Mengambil Hasil Akhir dan Membandingkannya Dengan Algoritma Yang Asli/Fakta

### 5.1. Menampilkan TransaksiEncoder

```
from mlxtend.preprocessing import TransactionEncoder

te = TransactionEncoder()
te_ary = te.fit(invoices).transform(invoices)
df = pd.DataFrame(te_ary, columns=te.columns_)
df
```

### 5.2. Pembentukan Frequent Itemsets

Frequent itemsets dibentuk dengan menggunakan fungsi `apriori` dari pustaka `mlxtend.frequent\_patterns` dengan minimal support yang telah ditentukan.

### 5.3. Pembentukan Nilai Terbesar pada Frequent Itemsets

```
frequent_itemsets.nlargest(n = 15, columns = 'support')
```

### 5.4. Visualisasi Gambar Frequent Itemsets

```
plt.figure(figsize=(12,6))
plt.xticks(rotation=90)
sns.barplot(x='itemsets', y='support', data=frequent_itemsets.nlargest(n = 15, columns = 'support'), palette="Purples_r")
```

### 6. Interpretasi Aturan Asosiasi

```
confidence_association = association_rules(frequent_itemsets, metric='lift', min_threshold=0.2)
confidence_association.head(10)
```

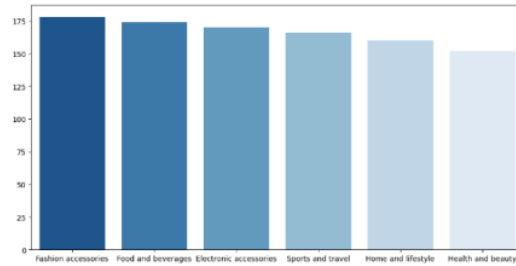
## IV. HASIL DAN PEMBAHASAN

Index	Invoice ID	Branch	City	Customer type	Gender	Product line	Unit price	Quantity
0	750-67-8428	A	Yangon	Member	Female	Health and beauty	74.69	7
1	226-31-3081	C	Naypyitaw	Normal	Female	Electronic accessories	15.28	5
2	631-41-3108	A	Yangon	Normal	Male	Home and lifestyle	46.33	7
3	123-19-1176	A	Yangon	Member	Male	Health and beauty	58.22	8
4	373-73-7910	A	Yangon	Normal	Male	Sports and travel	86.31	7

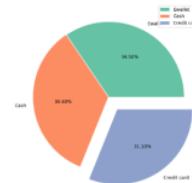
Gambar 1 : Data Belajar sudah Diambil dan Dijalankan

	Invoice ID	Branch	City	Customer type	Gender	Product line	Unit price	Quantity
0	750-67-8428	A	Yangon	Member	Female	Health and beauty	74.69	7
1	226-31-3081	C	Naypyitaw	Normal	Female	Electronic accessories	15.28	5
2	631-41-3108	A	Yangon	Normal	Male	Home and lifestyle	46.33	7
3	123-19-1176	A	Yangon	Member	Male	Health and beauty	58.22	8
4	373-73-7910	A	Yangon	Normal	Male	Sports and travel	86.31	7

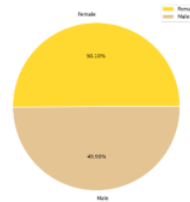
Gambar 2 : Preprocessing Data



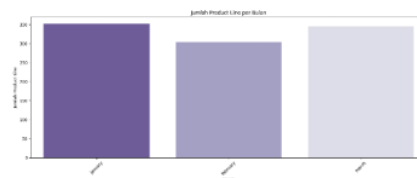
Gambar 3 : Visualisasi Jumlah Produk Yang Dibeli



Gambar 4 : Visualisasi Fitur Metode Pembayaran



Gambar 5 : Visualisasi Fitur Jenis Kelamin



Gambar 6 : Visualisasi Fitur Bulan

```
Product line
Fashion accessories    178
Food and beverages     174
Electronic accessories 170
Sports and travel      166
Home and lifestyle     160
Health and beauty      152
Name: count, dtype: int64
```

Gambar 7 : Jumlah Tiap Produk yang Dibeli

	Electronic accessories	Fashion accessories	Food and beverages	Health and beauty	Home and lifestyle	Sports and travel
0	False	False	False	True	False	False
1	True	False	False	False	False	False
2	False	False	False	False	True	False
3	False	False	False	True	False	False
4	False	False	False	False	False	True
...	...	...	...	...	...	...
995	False	False	False	True	False	False
996	False	False	False	False	True	False
997	False	False	True	False	False	False
998	False	False	False	False	True	False
999	False	True	False	False	False	False

1000 rows x 6 columns

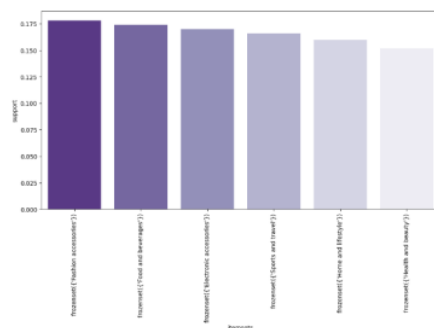
Gambar 8 : Tabel Transaksi Ercorder

	support	itemsets	length
0	17.8	(Fashion accessories)	1
1	17.4	(Food and beverages)	1
2	17.0	(Electronic accessories)	1
3	16.6	(Sports and travel)	1
4	16.0	(Home and lifestyle)	1

Gambar 9 : Tabel Support 1 Itemset

	support	itemsets
1	0.178	(Fashion accessories)
2	0.174	(Food and beverages)
0	0.170	(Electronic accessories)
5	0.166	(Sports and travel)
4	0.160	(Home and lifestyle)
3	0.152	(Health and beauty)

Gambar 10 : Tabel Support



Gambar 11 : Gambar Figur dari Support

## V. KESIMPULAN

Kami menggunakan algoritma Aprior untuk menerapkan analisis penambangan pola berulang pada data supermarket penjualan acara dalam penelitian ini. Fungsi seperti memasak "penghitung\_item", "remove\_non\_sup", "check\_valid\_pairs", "pair\_counter", "u

nique\_elements", dan "apriori" dibuat untuk mengimplementasikan algoritme Apriori secara manual.

Analisis ini menunjukkan kelompok umum atau kelompok produk yang sering dibeli oleh pelanggan supermarket dalam jumlah besar dalam periode waktu yang berbeda (misalnya 2, 3, 4, 5 item kehabisan stok). Kumpulan item dengan panjang item 2 dan 3 mewakili pola pembelian yang paling sering diamati dalam data transaksi.

Penelitian ini tidak hanya mencari jumlah item yang sering dilihat, tetapi juga menghitung nilai kepercayaan antar item atau kelompok produk tertentu. Kepercayaan dihitung menggunakan fungsi equity\_on\_items dan equity\_on\_sets. Ini menunjukkan seberapa sering suatu produk atau lini produk dibeli ketika produk atau lini produk lain dibeli dalam transaksi yang sama.

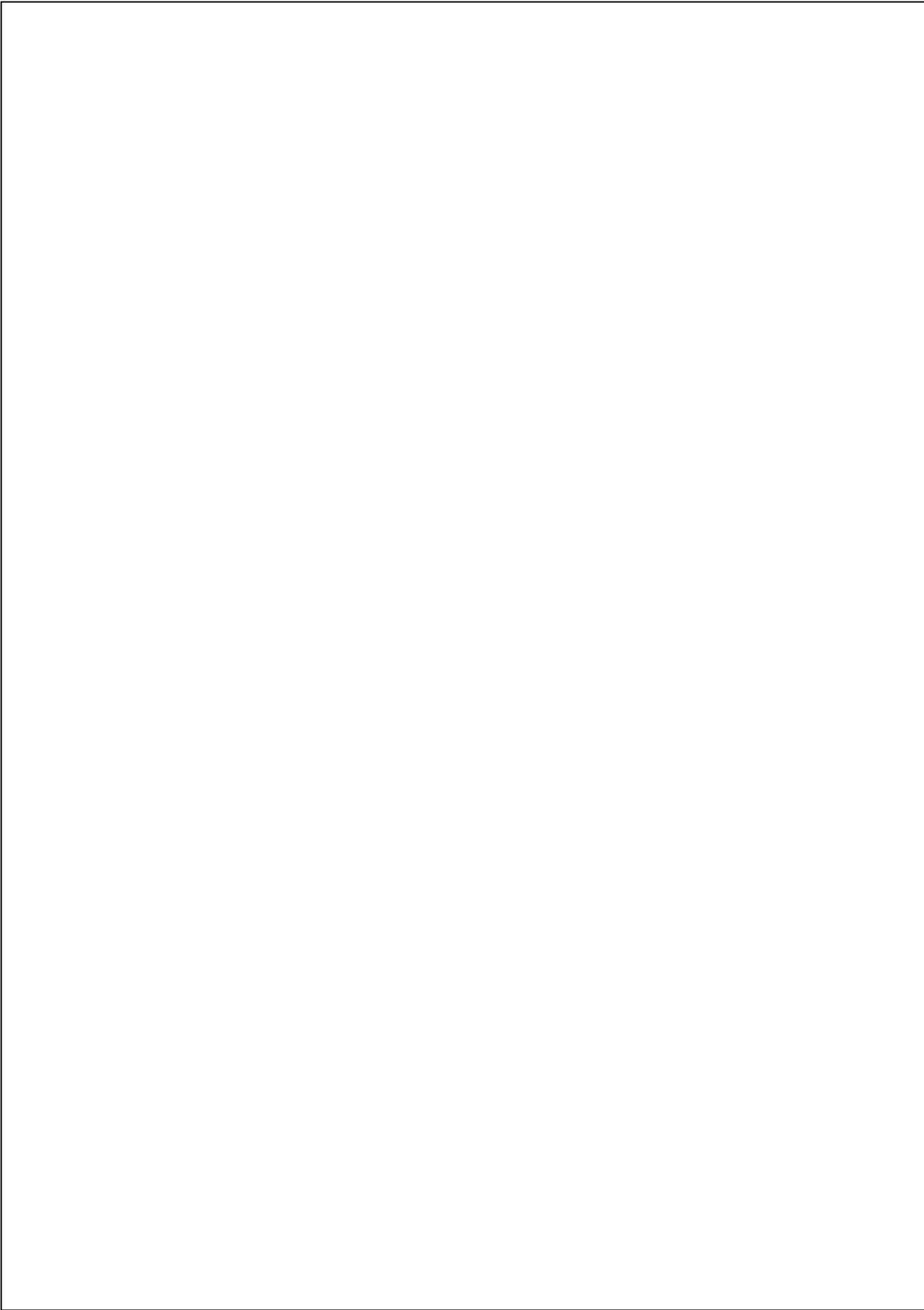
Hasil penambangan model dan analisis kepercayaan berkelanjutan membantu supermarket memahami perilaku pembelian konsumen. Informasi ini dapat digunakan dalam strategi pemasaran, kampanye, penempatan produk atau pengembangan produk baru untuk memenuhi preferensi pelanggan. Secara keseluruhan, penelitian ini menunjukkan bahwa algoritma Aprior dapat menemukan pola belanja yang berguna untuk pengambilan keputusan bisnis melalui analisis pattern mining pada data transaksi supermarket.

## VI. DAFTAR PUSTAKA

<sup>2</sup>  
[https://en-m-wikipedia-org.translate.goog/wiki/Apriori\\_algorithm?\\_x\\_tr\\_sl=en&\\_x\\_tr\\_tl=id&\\_x\\_tr\\_hl=id&\\_x\\_tr\\_pto=tc](https://en-m-wikipedia-org.translate.goog/wiki/Apriori_algorithm?_x_tr_sl=en&_x_tr_tl=id&_x_tr_hl=id&_x_tr_pto=tc)

<sup>1</sup>  
[https://www.javatpoint-com.translate.goog/apriori-algorithm?\\_x\\_tr\\_sl=en&\\_x\\_tr\\_tl=id&\\_x\\_tr\\_hl=id&\\_x\\_tr\\_pto=tc](https://www.javatpoint-com.translate.goog/apriori-algorithm?_x_tr_sl=en&_x_tr_tl=id&_x_tr_hl=id&_x_tr_pto=tc)

[https://colab.research.google.com/drive/13bAvG56tLkY113duD9pCNRIs1qXFFUB?authuser=0#scrollTo=tnSj\\_c8nI311](https://colab.research.google.com/drive/13bAvG56tLkY113duD9pCNRIs1qXFFUB?authuser=0#scrollTo=tnSj_c8nI311)



# Aprioriiii.pdf

## ORIGINALITY REPORT

8%

SIMILARITY INDEX

8%

INTERNET SOURCES

1%

PUBLICATIONS

3%

STUDENT PAPERS

## PRIMARY SOURCES

1

[e-jurnal.lppmunsera.org](http://e-jurnal.lppmunsera.org)

Internet Source

2%

2

[neuroquantology.com](http://neuroquantology.com)

Internet Source

2%

3

[publikasi.dinus.ac.id](http://publikasi.dinus.ac.id)

Internet Source

2%

4

[fasilkom-ti.usu.ac.id](http://fasilkom-ti.usu.ac.id)

Internet Source

1%

5

[export.arxiv.org](http://export.arxiv.org)

Internet Source

1%

Exclude quotes Off

Exclude matches Off

Exclude bibliography On