MODUL

Fundamental Data Analyst

Minggu ke-7

Universitas Bina Sarana Informatika

Sekolah Tinggi Informatika & Komputer Indonesia

2024

Kata Pengantar

Puji syukur Alhamdulillah, penulis panjatkan kehadirat Allah ta'ala, yang telah melimpahkan Rahmat dan Karunia-Nya sehingga pada akhirnya penulis dapat menyelesaikan modul ini dengan baik. Dimana modul ini penulis sajikan dalam bentuk modul yang sederhana. Adapun modul ini penulis buat untuk menambah wawasan para pembaca pada umumnya dan untuk menambah bahan materi untuk mata kuliah Fundamental Data Analyst bagi mahasiswa prodi Sistem Informasi Universitas Bina Sarana Informatika.

Sebagai bahan penulisan diambil berdasarkan pencarian di beberapa sumber, seperti buku, internet dan masih banyak lagi yang lainnya. Dalam modul ini menjelaskan materi Fundamental Data Analyst pertemuan 7 yang membahas tentang Unsupervised Learning (Association Rule). Penulis menyadari bahwa tanpa bimbingan dan dorongan dari semua pihak, maka penulisan dan pembuatan modul ini tidak akan berjalan dengan lancar.

Penulis mengucapkan terima kasih kepada tim sehingga bisa menyelesaikan penyusunan modul ini. Semoga modul ini berguna bagi para pembaca baik mahasiswa ataupun siapapun yang bisa dijadikan bahan referensi untuk pembelajaran.

Agustus 2024

Tim Penyusun
Unit Pengembangan Akademik
Program Studi Sistem Informasi

Daftar Isi

Kata Pengantar 1

Daftar Isi 3

PEMBAHASAN 4

1. Unsupervised Learning (Association Rule) 4
2. Peran Association Rule dalam Data Mining 4
3. Tahapan Implementasi Association Rule pada Data Mining 5
4. CRISP-DM (Cross-Industry Standard Process for Data Mining) 5
5. Library Python untuk Association Rule 5
6. Implementasi Model CRISP-DM pada Algoritma Association Rule 6

Referensi 12

PEMBAHASAN

1. Unsupervised Learning (Association Rule)

Association rule dalam data mining merupakan teknik untuk menemukan hubungan atau keterkaitan antar item dalam data transaksi. Hal Ini sering digunakan dalam analisis keranjang belanja (data retail) untuk menemukan pola pembelian pelanggan, misalnya, hubungan antara pembelian roti dan mentega dan lain sebagainya.

Terdapat beberapa komponen pada Association rule diantaranya:

a. **Itemset**: Kumpulan item yang muncul bersama-sama dalam satu transaksi.
b. **Support**: Ukuran seberapa sering itemset muncul dalam basis data.
c. **Confidence**: Ukuran seberapa sering item B muncul dalam transaksi yang mengandung item A
d. **Lift**: Ukuran seberapa besar peningkatan kemungkinan B muncul dalam transaksi yang mengandung A dibandingkan dengan jika B muncul secara acak.

Association rule sangat berguna dalam berbagai bidang yang melibatkan analisis data besar untuk menemukan pola tersembunyi yang dapat mendukung pengambilan keputusan bisnis.

2. Peran Association Rule dalam Data Mining

a. **Peningkatan Penjualan**: Menemukan pola pembelian membantu toko atau bisnis menata produk sehingga pelanggan cenderung membeli lebih banyak barang.
b. **Manajemen Persediaan**: Memungkinkan pengelola toko untuk mengetahui produk mana yang sering dibeli bersama sehingga dapat mengatur persediaan dengan lebih efisien.
c. **Strategi Pemasaran**: Informasi tentang kebiasaan belanja dapat digunakan untuk membuat promosi yang lebih efektif, seperti memberikan diskon pada barang yang sering dibeli bersama.
d. **Rekomendasi Produk**: Membantu dalam sistem rekomendasi, seperti yang digunakan oleh Amazon atau Netflix, untuk menyarankan produk atau konten berdasarkan perilaku belanja atau menonton sebelumnya.
e. **Peningkatan Layanan Pelanggan**: Dengan memahami kebutuhan dan preferensi pelanggan, perusahaan dapat meningkatkan layanan dan kepuasan pelanggan

3. Tahapan Implementasi Association Rule pada Data Mining

Implementasi association rule dalam Python bisa dilakukan dengan menggunakan berbagai pustaka data mining seperti mlxtend. Berikut adalah tahapan implementasi association rule dalam Python:

a. Persiapan Data:

1) Memuat dan membersihkan data.
2) Mengubah data transaksi menjadi format yang sesuai untuk analisis association rule.

b. Membuat Frequent Itemsets:

Menggunakan algoritma seperti Apriori untuk menemukan itemset yang sering muncul.

c. **Membuat Association Rules**:

Menghasilkan aturan asosiasi dari itemset yang sering muncul.

d. **Evaluasi dan Interpretasi**:

1) Mengevaluasi aturan berdasarkan metrik seperti support, confidence, lift dan lain-lain.
2) Menginterpretasikan hasil untuk tindakan bisnis.

4. CRISP-DM (Cross-Industry Standard Process for Data Mining)

CRISP-DM (Cross-Industry Standard Process for Data Mining) adalah metodologi yang sering digunakan untuk proyek data mining dan data science. Metodologi ini terdiri dari enam langkah utama yang membantu dalam merencanakan, melaksanakan, dan mengelola proyek data mining, yaitu:

a. Business Understanding
b. Data Understanding
c. Data Preparation
d. Modeling
e. Evaluation
f. Deployment

5. Library Python untuk Association Rule

Beberapa library python yang dibutuhkan dalam analisis data menggunakan algoritma Association Rule adalah:

```
import pandas as pd
import numpy as np
import datetime
from mlxtend.frequent_patterns import apriori,
association_rules
from mlxtend.preprocessing import TransactionEncoder
```

penjelasan dari library diatas adalah:

1. Manipulasi dan analisis data tabular dimjuat dalam bentuk dataframe
2. Manipulasi data dalam bentuk array
3. Untuk konversi tanggal dan waktu
4. mengimpor dua fungsi utama dari pustaka mlxtend, yaitu apriori dan association_rules
5. mengimpor kelas TransactionEncoder dari modul preprocessing dalam pustaka mlxtend

6. Implementasi Model CRISP-DM pada Algoritman Association Rule

a. Business Understanding

dataset atau studi kasus yang akan digunakan pada analisis kali ini adalah House Prices. Langkah pertama ini berfokus pada pemahaman tujuan bisnis dan persyaratan proyek. Ini melibatkan:

1) Menentukan tujuan bisnis dan kebutuhan yang spesifik.
   a) Mengelompokkan data harga rumah (SalePrices) ke dalam beberapa kelas (Low, Medium, High, Very High)
   b) Mengetahui atau Memprediksi kelas harga rumah berdasarkan fitur yang dipilih
2) Mengidentifikasi masalah bisnis yang ingin diselesaikan.
   a) Apa saja faktor-faktor utama yang mempengaruhi variasi harga rumah di Boston?
   b) Seberapa besar dampak faktor-faktor ini terhadap harga rumah?
   c) Bagaimana model prediktif dapat digunakan untuk memperkirakan harga rumah berdasarkan karakteristik spesifik?
3) Menyusun rencana proyek, termasuk sasaran, anggaran, dan waktu.

b. Data Understanding

Langkah kedua melibatkan pengumpulan data dan familiarisasi dengan data yang tersedia. Tahapan Data Understanding yang digunakan pada analisis kali ini diantaranya:

1) Pengumpulan Data

Dataset : Data Penjualan Retail (https://s.id/DatasetFDA)

Deskripsi **:** dataset Penjualan Retail memiliki 23 fitur, diantaranya seperti InvoiceNo, InvoiceDate, Branch_SPLR dan seterusnya (check informasi dasar dari dataset pada pengolahan data)

Fitur : 'OverallQual', 'GrLivArea', 'GarageCars', 'GarageArea', 'TotalBsmtSF', 'FullBath', 'YearBuilt'

Target : SalePrice

2) Deskripsi Data

Identifikasi Tipe Data: Menentukan jenis data dari setiap fitur (numerik, kategorikal, dll.). Dalam dataset ini, sebagian besar fitur adalah numerik, untuk menampilkan deskripsi data menggunakan bahasa pemrograma python dapat menggunakan Script berikut ini:

```
data = pd.read_excel('data_retail2.xlsx')
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 541909 entries, 0 to 541908
Data columns (total 23 columns):
 #   Column              Non-Null Count   Dtype
---  ------              --------------   -----
 0   InvoiceNo           541909 non-null  object
 1   InvoiceDate         541909 non-null  datetime64[ns]
 2   BRANCH_SPLR         541909 non-null  int64
 3   BRANCHNAME_SPLR     541909 non-null  object
 4   warehouseProductsID 541909 non-null  object
 5   BARCODEID           541909 non-null  int64
 6   StockCode           541909 non-null  object
 7   PRODUCT             541909 non-null  object
 8   PRODUCT_CATEGORY    541909 non-null  object
 9   Quantity            541909 non-null  int64
 10  UnitPrice           541909 non-null  float64
 11  UnitPriceRupiah     541909 non-null  float64
 12  oldCUSTID           541909 non-null  object
 13  CustomerID          406829 non-null  float64
 14  CUSTNAME            541909 non-null  object
 15  ADDRESS             541737 non-null  object
 16  KOTA                525237 non-null  object
 17  PROVINSI            527069 non-null  object
 18  NEGARA              541909 non-null  object
 19  CHANNELID_SPLR      541909 non-null  int64
 20  CHANNELNAME_SPLR    541909 non-null  object
 21  SUBDISTID           541909 non-null  int64
 22  SUBDIST_NAME        541909 non-null  object
dtypes: datetime64[ns](1), float64(3), int64(5), object(14)
```

Penjelasan:

a) `data = pd.read_csv` memuat dataset dengan format .csv

b) `data.info()` digunakan untuk menampilkan Deskripsi Data

c. Data Preparation

Langkah ketiga ini melibatkan pembersihan dan transformasi data agar siap untuk pemodelan. Beberapa Tahapan Data Preparation yang dilakukan pada analisis kali ini diantaranya:

1) Memeriksa apakah ada nilai dalam dataset yang “Kosong” atang “NaN”

Memeriksa apakah ada nilai dalam dataset yang "kosong" atau "NaN" (Not a Number) adalah langkah penting dalam analisis data. Nilai-nilai kosong atau NaN dapat muncul karena berbagai alasan, seperti kesalahan saat memasukkan data, data yang hilang, atau ketidaksempurnaan dalam proses pengumpulan data. untuk melakukan pengecheckan data kosong pada python dapat menggunakan script berikut:

```
print(data.isnull().values.any())
```

```
print(data.isnull().values.any())
✓ 0.0s
True
```

Scrip diatas hanya melakukan pengecheckan apakah pada dataset masih memiliki kolom dengan baris yang kosong, Apabila ingin melakukan pemeriksaan data isnull perkolom gunakan script berikut:

```
print(data.isnull().sum())
```

2) Mengisi Nilai yang Hilang Missing Values

Dalam kasus ini akan dilakukan pengisian data yang kosong untuk kolom dengan type numeric saja dan akan diisi dengan nilai rata-rata yang ada pada variable tersebut menggunakan script berikut:

```
data = data.fillna(data.mean(numeric_only=True))
print(data.isnull().sum())
```

3) Memilih Fitur (Feature) dan Target

Dari total 81 variable yang ada, dalam kasus ini akan menggunakan beberapa variable saja yang ditentukan untuk fitur dan target:

Fitur : ‘OverallQual', 'GrLivArea', 'GarageCars', 'GarageArea', 'TotalBsmtSF', 'FullBath', 'YearBuilt'

Target: SalePrice

Pemilihan fitur dan target menggunakan python dapat menggunakan script berikut:

```
features = ['OverallQual', 'GrLivArea', 'GarageCars',
           'GarageArea', 'TotalBsmtSF', 'FullBath', 'YearBuilt’]
X = data[features]
y = data['SalePrice']
```

4) Standarisasi fitur

Dalam kasus ini akan menggunakan StandardScaler() dalam konteks clustering adalah untuk menstandarisasi fitur-fitur dalam dataset sebelum melakukan clustering. Standardisasi sangat penting dalam clustering karena banyak algoritma clustering (seperti K-Means) sensitif terhadap skala fitur.

Dengan menggunakan StandardScaler, Anda memastikan bahwa semua fitur memiliki skala yang sama sebelum melakukan clustering, yang dapat meningkatkan akurasi dan kinerja dari algoritma clustering.

```
# Standarisasi fitur
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)
```

5) Membagi dataset menjadi data pelatihan dan pengujian

Untuk membagi dataset menjadi data pelatihan dan pengujian dengan python dapat menggunakan script berikut:

```
X_train, X_test, y_train, y_test = train_test_split(X, y_binned,
test_size=0.2, random_state=42)
print("Shape of X_train:", X_train.shape)
print("Shape of X_test:", X_test.shape)
print("Shape of Y_train:", y_train.shape)
print("Shape of Y_test:", y_test.shape)
```

```
Shape of X_train: (1168, 7)
Shape of X_test: (292, 7)
Shape of Y_train: (1168,)
Shape of Y_test: (292,)
```

pada script pembagian data latih dan data uji diatas dibagi menjadi 20% Test dan 70% Train, Nilai random state memungkinkan untuk dirubah tergantu metode acak yang akan digunakan, contoh Random State=0

d. Modelling

Langkah keempat adalah membangun model menggunakan teknik data mining. Dalam tahap pembelajaran ini menggunakan algoritma Association Rule. Dalam bahasa python teknik implementasi model Association Rules dapat menggunakan script berikut:

```
rules1 = association_rules(frequent_itemsets, metric='lift', min_threshold=1 )
rules1.head()
```

Keterangan:

gunakan algoritma model asosiasi berdasarkan subset data frequent itemsets
dengan menggunakan metrik “lift”
dan ambang batas minimum sebesar 1.
Hasilnya akan disimpan dalam variabel rules1

e. Evaluation

Langkah kelima melibatkan evaluasi model untuk memastikan model memenuhi tujuan bisnis dan persyaratan proyek, Adapun Evaluasi model Association Rule mencakup :

1. Antecedents (A)
2. Consequents (B)
3. Support
4. Confidence
5. Support dan Confidence
6. Lift
7. Leverage
8. Conviction

9. Leverage dan Conviction

Untuk menampilkan hasil evaluasi model dalam bentuk visualisasi yang lebih menarik dapat menggunakan script berikut:

```
result1=rules1[(rules1['lift'] >= 1) & (rules1['confidence']) >= 0.8]
apriori_result=result1.sort_values(by='confidence', ascending=False)
apriori_result.head(20)

# Tabel hasil model
```

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction	zhangs_metric
1238	(SABUN & SAMPHOO, OBATAN, PARFUM)	(KOSMETIK)	0.115503	0.412757	0.102081	0.883795	2.141200	0.054406	5.053522	0.602571
1252	(SABUN & SAMPHOO, SUSU, PARFUM)	(KOSMETIK)	0.121290	0.412757	0.106760	0.880203	2.132497	0.056697	4.901986	0.604370
1086	(SABUN & SAMPHOO, OBATAN, BISKUIT)	(MINUMAN)	0.117596	0.383327	0.102697	0.873298	2.278206	0.057619	4.867128	0.635829
1182	(SABUN & SAMPHOO, MINUMAN, PARFUM)	(KOSMETIK)	0.140500	0.412757	0.122522	0.872042	2.112725	0.064530	4.589344	0.612772
	(BISKUIT, SABUN & SAMPHOO,									

f. Deployment.

Langkah terakhir adalah mengimplementasikan model ke dalam lingkungan operasional. Hal Ini mencakup:

1. Merencanakan dan menjalankan implementasi model dalam sistem produksi.
2. Memantau dan memelihara model untuk memastikan kinerjanya tetap optimal.
3. Mengkomunikasikan hasil dan manfaat model kepada pemangku kepentingan.

Namun untuk pembelajaran kali ini tidak membahas tahap Deployment.

Referensi

Steele, B., Chandler, J., Reddy, S. (2016). Algorithms for Data Science. Jerman: Springer International Publishing.

Abdussomad, dkk. (2021). Dasar Pemrograman Python. Yogyakarta: Teknosain.

Saeful Bahri, dkk (2019), Data mining : algoritma klasifikasi & penerapannya dalam aplikasi, Grha Ilmu

Segmentasi Pelanggan Menggunakan Python. (n.d.). (n.p.): Kreatif

Data Mining Menggunakan Android, Weka, dan SPSS. (2020). (n.p.): Airlangga University Press.

Abdussomad, A., Kurniawan, I., & Wibowo, A. (2023). Implementation of the Decission Tree Algorithm to Determine Credit Worthiness. *Compiler, 12*(2), 103-108