JURNAL ILMIAH FIFO **P-ISSN 2085-4315 / E-ISSN 2502-8332**
DOI: http://dx.doi.org/10.22441/fifo.2023.v15i2.002

Penerapan Algoritma Naïve Bayes Dengan Feature Selection Pada Data Penjualan Konstruksi

Fajar Muji Anto[1], Lintang Setiaji Abimanyu[2], Tazkiyah Herdi*[3]

[1,2,3]*Fakultas Ilmu Komputer, Universitas Mercu Buana*
Jl. Raya Meruya Selatan No.1, Jakarta Barat, Indonesia
[1]41818210014@student.mercubuana.ac.id, [2]41819210015@student.mercubuana.ac.id,
[3*]tazkiyah.herdi@mercubuana.ac.id

*) Corresponding author

Abstract

In construction services, the application of machine learning is often used in processing large amounts of data. One example of the application of machine learning is using machine learning algorithms to classify sales data from a company, the final result of which is information to be used as a basis for decision making. In data classification, the application of the Naïve Bayes algorithm method is widely used because it only requires a small amount of training data to determine the parameters in the classification process. PT. Maju Jaya Makmur Sejahtera is a company engaged in digital transformation in the construction services sector. Based on the results of the interview, the problems found at PT. Maju Jaya Makmur Sejahtera is a large number of client data for consultations amounting to approximately 700 lines, making it difficult to obtain relevant information so that data analysis is needed in determining decisions at PT. Maju Jaya Makmur Sejahtera. The results of the processing and classification of the Recursive Feature Elimination algorithm selected 10 dataset features into a total of 6 features and overall, the accuracy obtained from the Naïve Bayes algorithm model was 88%, precision 87%, recall 85%, and F1-score 86%. The classification results can be said to be quite good, but have shortcomings in terms of dataset attributes resulting in an average score below 90%.

***Keyword**: Classification, Machine Learning, Feature Selection, Naïve Bayes*

Abstrak

Dalam jasa konstruksi, penerapan *machine learning* kerap digunakan pada proses pengolahan data yang berjumlah besar, salah satu contoh dari penerapan machine learning adalah menggunakan algoritma *machine learning* untuk mengklasifikasi data - data penjualan dari sebuah perusahaan yang hasil akhirnya berupa informasi untuk digunakan sebagai landasan pengambilan keputusan. Dalam klasifikasi data penerapan metode algoritma *naïve bayes* banyak digunakan karena hanya membutuhkan jumlah data pelatihan yang sedikit untuk menentukan parameter dalam proses klasifikasi. PT. Maju Jaya Makmur Sejahtera adalah perusahaan yang bergerak di bidang digital transformation di sektor jasa konstruksi. Berdasarkan hasil wawancara, masalah yang terdapat pada PT. Maju Jaya Makmur Sejahtera adalah banyaknya data client untuk konsultasi yang sebanyak kurang lebih 700 baris, menyebabkan sulitnya untuk mendapatkan informasi yang relevan sehingga diperlukannya analisis data dalam menentukan keputusan di PT. Maju Jaya Makmur Sejahtera. Hasil dari *processing* dan klasifikasi Algoritma *Recursive Feature Elimination* menyeleksi 10 fitur *dataset* menjadi total 6 fitur dan secara keseluruhan, akurasi yang didapatkan dari model algoritma *naïve bayes* sebesar 88%, *precision* 87%, *recall* 85%, dan *F1-score* 86%. Hasil klasifikasi dapat dikatakan cukup bagus, tapi memiliki kekurangan dari segi atribut dataset sehingga menghasil skor rata rata dibawah 90%.

Kata Kunci: Klasifikasi, Machine Learning, Feature Selection, Naïve Bayes

I. Pendahuluan

Proses pengolahan data yang besar biasanya menggunakan *machine learning*. Salah satu contohnya adalah penggunaan algoritma pengajaran mesin untuk mengklasifikasi data penjualan perusahaan, hasilnya adalah informasi untuk digunakan sebagai landasan pengambilan keputusan [1]. Dalam klasifikasi data, *naïve bayes*

adalah algoritma *machine learning* yang banyak digunakan. Ini karena hanya membutuhkan jumlah data pelatihan yang sedikit untuk menentukan parameter proses klasifikasi, menghasilkan hasil pengujian dan penelitian yang sangat akurat, dan memiliki tingkat nilai error yang lebih rendah ketika dataset yang besar digunakan. Selain itu, algoritma *naïve bayes* memiliki tingkat akurasi yang lebih tinggi daripada algoritma lain dalam klasifikasi data [2].

Menurut perbandingan pemilihan fitur pada *naïve bayes*, pemilihan fitur adalah komponen penting yang harus diperhatikan saat melakukan preprocessing data karena memungkinkan untuk menghilangkan fitur fitur yang tidak relevan dari *dataset*, yang berdampak pada peningkatan hasil akurasi dari algoritma pengajaran mesin serta masalah yang sering muncul dalam klasifikasi khusus. Pengajaran mesin umumnya adalah menemukan cara untuk mengurangi dimensi n dari dataset. Jadi, metode *feature selection* dibutuhkan untuk menghindari *overfitting*. Untuk menghindari risiko *overfitting*, digunakan *feature selection*. Metode seleksi fitur *Recursive Feature Elimination (RFE)* bekerja dengan algoritma pembelajaran mesin berdasarkan seberapa baik algoritma klasifikasi bekerja. Metode ini melakukannya dengan menghapus atribut secara rekursif dan kemudian membuat model untuk atribut yang tersisa. Metode ini pertama akan membuat model dari set fitur dan menilai setiap fitur berdasarkan pengaruh dan kepentingannya pada variabel target. Pada saat yang sama, metode ini menghilangkan fitur yang tidak perlu dan lemah atau yang paling tidak mempengaruhi keberhasilan model klasifikasi, dan pada saat yang sama mempertahankan fitur yang efektif dan kuat yang meningkatkan keberhasilan model. Setelah itu, fitur yang tidak penting akan dihapus pada setiap langkah, dan model akan dibangun kembali dengan menghitung ulang pentingnya setiap fitur hingga tingkat keberhasilan model tertinggi dicapai[3].

Jasa konstruksi adalah konsultasi tentang perencanaan, pelaksanaan, dan pengawasan proyek konstruksi dapat ditemukan di perusahaan jasa konstruksi. Data tentang stok barang dan penjualan sebuah perusahaan jasa konstruksi berisi lebih dari 700 variabel dan diperbarui setiap bulan. Metode pembelajaran mesin diperlukan untuk mengelola data yang sangat besar. PT. Maju Jaya Makmur Sejahtera beroperasi di bidang transformasi digital dan menyediakan layanan seperti pembangunan, pengoperasian, pemeliharaan, pembongkaran, dan pembangunan kembali bangunan. Berdasarkan hasil wawancara, PT. Maju Jaya Makmur Sejahtera menghadapi masalah dalam mendapatkan informasi yang relevan karena banyaknya data konsultasi, yang mencakup lebih dari 700 baris. Akibatnya, proses pengambilan keputusan di PT. Maju Jaya Makmur Sejahtera membutuhkan analisis data. *Feature selection* digunakan untuk mencari dan menghilangkan variabel pada dataset yang tidak penting untuk meningkatkan kinerja hasil prediksi algoritma dan meningkatkan efisiensi waktu dan informasi dalam mendapatkan *insights*. Solusi yang tepat untuk masalah PT. Maju Jaya Makmur Sejahtera harus ditemukan berdasarkan masalah tersebut. Solusi yang dimaksud adalah algoritma *naïve bayes*, yang tidak memerlukan banyak data baris untuk mendapatkan tingkat akurasi yang tinggi. Oleh karena itu, algoritma ini sangat cocok untuk digunakan dalam pemilihan fitur pada data penjualan. Dalam penelitian ini, data dikumpulkan melalui wawancara, observasi, dokumentasi, dan studi pustaka.

Data yang digunakan pada penelitian ini merupakan data sekunder berupa dataset yang berisikan data penjualan PT. Maju Jaya Makmur selama 1 Tahun. Dataset yang digunakan berisikan lebih dari 700 baris data dari tahun 2022 sampai 2023. Berdasarkan uraian diatas, maka peneliti membuat laporan tugas akhir atau karya tulis ilmiah dengan judul "Penerapan Algoritma naïve bayes Dengan Feature Selection Pada Data Penjualan Konstruksi (Studi Kasus Pt. Maju Jaya Makmur Sejahtera)".

Berikut adalah hasil *literature review* dari penelitian terdahulu berdasarkan permasalahan yang diangkat oleh peneliti. Penelitian terdahulu berjudul *Implementation Of Data Mining With Classification And Forecasting Method Use Model Gaussian Naïve Bayes For Building Store* (Studi Case: Tb Sinar Jaya) [2]. Penelitian selanjutnya berjudul Klasifikasi Kualitas Biji Kopi Menggunakan *Multilayer Perceptron* Berbasis Fitur Warna LCH [12]. Penelitian selanjutnya berjudul Klasifikasi Citra Daging Menggunakan *Deep Learning* dengan Optimisasi *Hard Voting*[13].

Berdasarkan penelitian sebelumnya, algoritma *naïve bayes* digunakan untuk menentukan seleksi fitur yang digunakan untuk memberikan informasi kepada sebuah dataset, yang kemudian dapat digunakan sebagai referensi saat membuat keputusan. Penelitian terdahulu menggunakan algoritma *naïve bayes* untuk menentukan klasifikasi berita yang dataset-nya diproses melalui *text mining*. Hasil klasifikasi menghasilkan model dengan nilai evaluasi akurasi, *recall*, dan presisi sebesar 73,2% yang kemudian ditambahkan dengan peningkatan *bayesian* menghasilkan nilai evaluasi yang sama besar, 73.2%. Informasi yang diperoleh dari klasifikasi ini menunjukkan bahwa seleksi fitur *gain information* tidak berdampak signifikan pada peningkatan hasil performa terhadap kondisi label *polynomial*.

Penelitian kami berbeda dari penelitian sebelumnya karena kami menggunakan algoritma *naïve bayes* sebagai dasar untuk membuat model klasifikasi pada dataset yang berisi data penjualan penjualan jasa yang diperoleh dari PT. Maju Jaya Makmur Sejahtera. Tujuan dari penelitian kami adalah untuk meningkatkan efisiensi kolom data dengan menggunakan feature selection dan naïve bayes untuk mengidentifikasi variable variable yang relevan pada masing-masing kolom dan menenangkan dataset dengan lebih baik.

II. Metode Penelitian

Naïve bayes adalah metode pengklasifikasian sederhana yang digunakan untuk menghitung kemungkinan dengan menjumlahkan frekuensi dan nilai dari data yang digunakan. Itu didasarkan pada teorema Bayes, yang digunakan untuk menghitung kemungkinan untuk setiap kelas dengan mengasumsikan kelas satu dengan kelas lain secara independen (tidak ada keterkaitan)[7]. Ada juga definisi lain dari naïve bayes, yang digunakan untuk memprediksi kemungkinan yang akan datang dengan menggunakan data yang digunakan.

$$P(H|X) = \frac{P(X|H).P(H)}{P(X)}$$

Penjelasan:

X: Data yang kelasnya tidak diketahui.

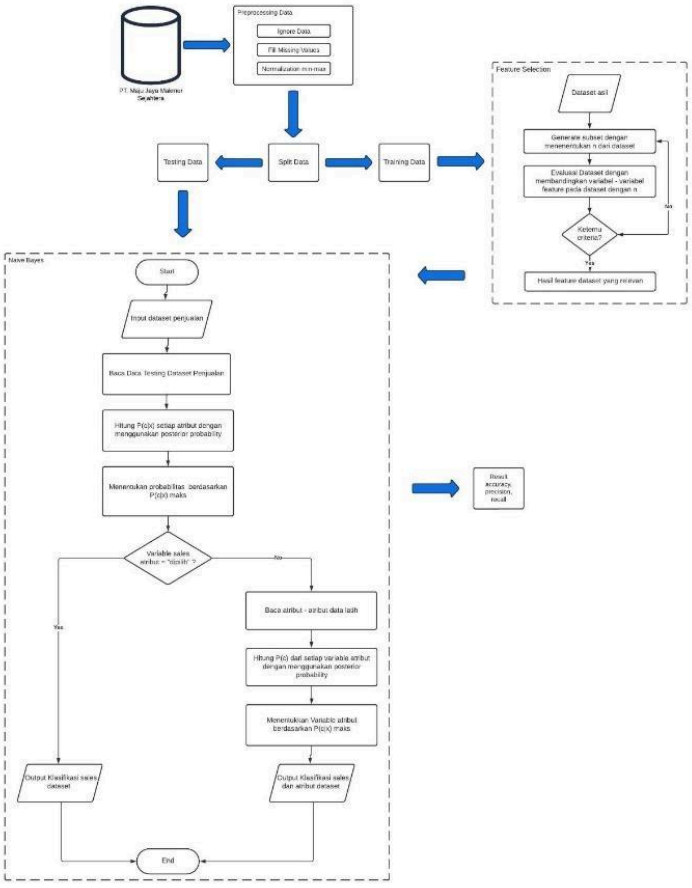
H: Data hipotesis X pada sebuah kelas.

P(H|X): Probabilitas dari hipotesis berdasarkan dari kondisi (*posteriori* probabilitas).

P(H): Probabilitas dari hipotesis H (sebelum probabilitas).

P(X|H): Probabilitas dari X berdasarkan dari kondisi dalam hipotesis.

H P(X): Probabilitas X [8].



Gambar 1. Diagram Alir Peneltian

Berikut ini merupakan penjelesan Gambar 1

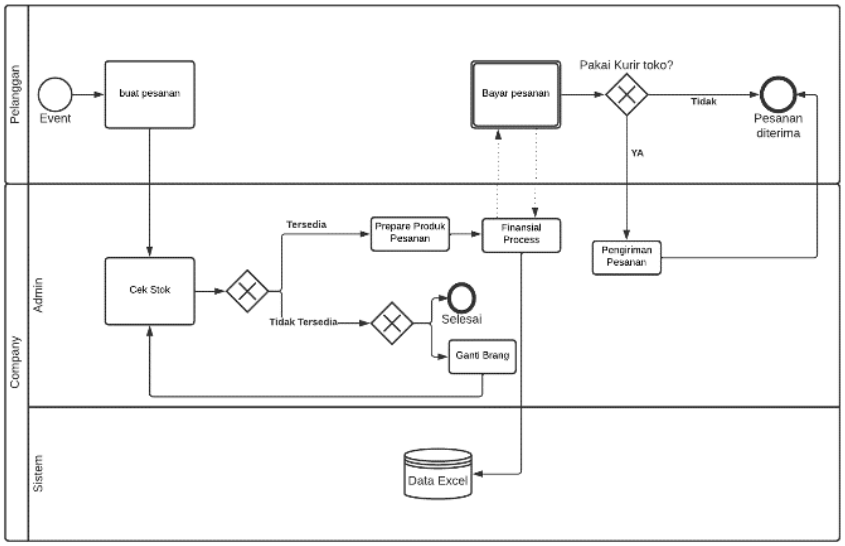
1. *Dataset*
   Pada tahap ini, data mentah yang diperoleh dari PT. Maju Jaya Makmur Sejahtera diformat menjadi format yang lebih rapi dengan berbagai jenis data, sehingga lebih mudah untuk memprosesnya.
2. *Preprocessing Data*
   a. *Ignore Data*
      Pada tahap ini, mengabaikan baris data yang memiliki tingkat *null* yang mencapai kapasitas maksimal untuk meningkatkan hasil akurasi dari klasifikasi.
   b. *Fill Missing Values*
      Pada tahap ini yaitu mengisi data yang memiliki nilai kosong atau *null* dengan menggunakan IQR (*Interquartile Range*).
   c. *Normalization min-max*
      Pada tahap ini dilakukan pengecekan jenis distribusi normalisasi data dengan melihat nilai terendah dan tertinggi dari hasil visualisasi dengan menggunakan *histogram*.
3. *Split Data*
   Split data adalah proses pembagian dataset menjadi 2 data baru. Data pertama disebut data latih (*training*), dan yang kedua adalah data test.
4. *Training Data*
   Training data adalah proses melakukan pengkodean supaya *machine learning* dapat mencari korelasinya sendiri atau belajar pola dari data yang diberikan.
5. *Feature Selection*
   Dataset asli PT. Maju Jaya Makmur Sejahtera dilakukan *generate subset* dengan menentukan variabel dari dataset yang kemudian dilakukan *feature selection* dengan menggunakan *library python scikit-learn* untuk mengurangi variabel yang masuk dengan cara menggunakan data - data yang relevan dan mengurangi *noise* pada dataset.
   a. Dataset asli
      Dataset didapatkan dari PT. Maju Jaya Makmur Sejahtera yang sebelumnya sudah dilakukan tahap *preprocessing* data dan *training data.*
   b. *Generate subset* dengan menentukan n dari *dataset*
      Subset dilakukan generasi dengan melihat jumlah *row* yang ada pada dataset dibandingkan dengan semua *feature* pada *dataset*.
   c. Evaluasi *dataset* dengan membandingkan variabel variabel *feature* pada dataset dengan n
      Hasil generasi kemudian membandingkan variabel variabel *feature* pada dataset dengan n
   d. Ketemu kriteria?
      Apakah kriteria ditemukan atau tidak?
   e. Jika ya, maka akan menghasilkan variabel *dataset* yang relevan.
   f. Jika tidak, maka kembali ke *generate subset* dengan menentukan n dari dataset PT. Maju Jaya Makmur Sejahtera.
6. *Testing Data*
   Testing data adalah data yang digunakan untuk dijadikan sebagai perbandingan untuk mengukur akurasi dari hasil data yang telah dilatih pada tahap *training data.*
7. *Naïve Bayes*
   Implementasi algoritma *naïve bayes* dengan menggunakan *library python scikit-learn* pada dataset PT. Maju Jaya Makmur Sejahtera yang akan dilakukan dengan method fit (*X_train, y_train*) untuk menentukan atribut pada dataset.
   a. Input Data Penjualan
      Input data penjualan berupa dataset penjualan PT. Maju Jaya Makmur dari tahun 2022 - 2023.
   b. Baca Data Testing Dataset Penjualan
      Dataset penjualan yang telah dibagi menjadi data latih dan testing, data testingnya dibaca pada tahapan ini.
   c. Hitung P(c|x) setiap atribut dengan menggunakan *posterior probability*
      Posterior probability dihitung dengan *likelihood* dikali *class prior probability*, dibagi dengan *predictor prior probability*.
   d. Menentukan probabilitas berdasarkan P(c|x) maks
      Dari hasil perhitungannya kemudian ditentukan untuk *maks probabilitas* dari terpilihnya atribut pada *feature*.

e. Variable sales atribut = ‘dipilih?’
Apakah atribut dipilih atau tidak.
f. Jika ya, *output* klasifikasi *sales dataset*
Apabila ya, *output*-nya berupa *sales dataset* penjualan PT. Maju Jaya Makmur.
g. Jika tidak, baca atribut data latih
Jika atribut yang terpilih adalah tidak maka data latih akan dibaca untuk penentuan *posterior probability*.
h. Hitung P(c) dari setiap variabel atribut dengan menggunakan *prior probability*
Setiap variabel dataset dihitung dengan *prior probability* untuk mendapatkan *prediktor* dari setiap atribut *dataset*.
i. Menentukan variabel atribut berdasarkan P(c|x) maks
Setiap variabel pada dataset dihitung berdasarkan *posterior probability* dari variabel yang diberikan P(c).
j. Output klasifikasi *sales* dan atribut *dataset*
Menghasilkan klasifikasi *sales* dan atribut dari dataset yang sudah ditentukan setiap variabel dengan rumus $P(H|X) = \frac{P(X|H).P(H)}{P(X)}$

8. *Result accuracy, precision, recall*
Pada tahap akhir yaitu hasil dari penerapan algoritma *Naïve bayes* yang menghasilkan *result accuracy, precision* dan *recall*.

III. Hasil dan Pembahasan

3.1 Analisis Proses Bisnis Berjalan



Gambar 2. Analisis Proses Berjalan

Pada gambar 2 diagram *BPMN* menjabarkan bagaimana proses bisnis penjualan, setiap proses di bagian sistem, admin, dan pelanggan memiliki alur berbeda dan mendukung satu sama lain.

3.2 Analisis SWOT

Penentuan untuk dapat menentukan SWOT peneliti telah melakukan pengumpulan data dengan datang langsung ke PT. Maju Jaya Makmur Sejahtera untuk menganalisis proses bisnis yang sedang berjalan melalui observasi. Selain itu peneliti juga melakukan wawancara dengan pihak PT. Maju Jaya Makmur Sejahtera, untuk mendapatkan informasi - informasi yang diperlukan untuk memahami permasalah - permasalahan yang ada pada PT. Maju Jaya Makmur Sejahtera.

INTERNAL / EXTERNAL		
	Strength (S) 1. Ketersediaan produk beragam. 2. Lokasi Strategis. 3. Melakukan pemanfaatan teknologi dalam hal data penjualan.	**Weakness (W)** 1. Tingginya persaingan. 2. Tidak melakukan riset penjualan.
Opportunity (O) 1. Kepercayaan konsumen terhadap produk yang ada. 2. Antusiasme masyarakat untuk membeli produk. 3. Perkembangan teknologi yang sudah mencapai industri 4.0	**Strategi (SO)** 1. Mempertahankan hubungan dengan konsumen 2. Meningkatkan keragaman produk dengan mempertahankan kualitas produk 3. Melakukan penerapan klasifikasi pada data penjualan	**Strategi (WO)** 1. Memanfaatkan kepercayaan konsumen sebagai sumber daya saing 2. Melakukan riset terhadap produk yang memiliki nilai antusiasme dengan dibagi menjadi 4 kategori 3. Menerapkan teknologi machine learning dalam memingkatkan tingkat daya saing.
Threats (T) 1. Persaingan dengan toko online yang kiat berkembang yang cenderung memiliki cost yang relatif rendah. 2. Biaya operasional yang cenderung semakin tinggi dikarenakan menurunnya ekonomi secara global.	**Strategi (ST)** 1. Memanfaatkan teknologi pasar online untuk melakukan transaksi secara online.	**Strategi (WT)** 1. Melakukan riset pasar agar mampu bersaing secara offline dan online. 2. Beradaptasi dengan pesaing dengan membuka toko online.

Gambar 3. Analisis SWOT

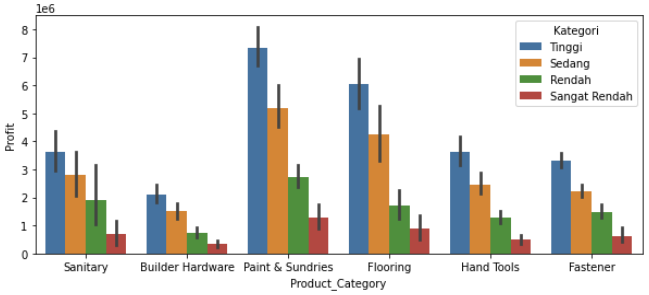
Pada gambar 3 bisa kita lihat bahwa peran dari faktor *internal* dan *external* sangat mendukung terjadinya proses bisnis, dari kedua faktor tersebut sebuah bisnis bisa mengembangkan data penjualan dengan cara mengidentifikasi prospek terbaik. Dengan cara ini maka dapat membantu perusahaan untuk mengambil keputusan yang lebih baik dan lebih cepat.

3.3 Pengumpulan Data

Dataset pada penelitian ini adalah data primer yang didapatkan langsung dari PT. Maju Jaya Makmur Sejahtera. *Dataset* berisikan data penjualan yang berlangsung selama 4 tahun, yitu 2019 - 2022. Pada dataset berisikan kurang lebih 3000 row data, dan berisikan 10 *feature*. *Feature - feature* nya adalah *order id, order date, order quantity, price_per_unit, cost_of_sales, sales, profit, profit_margin, channel*, dan *product_category*. *Dataset* terdiri dari 10 kolom, penjelasan setiap kolomnya secara berurutan sebagai berikut.

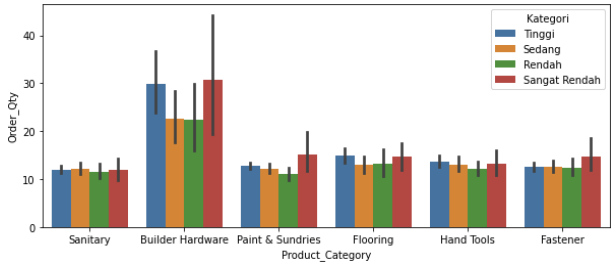
1. *Order_ID*: Id order.
2. *Order_Date*: tanggal transaksi yang dilakukan oleh konsumen
3. *Order_Quantity*: jumlah yang dibeli oleh konsumen berdasarkan tipe produk kategori.
4. *Price_per_unit*: harga produk kategori.
5. *Cost_Of_Sales*: harga bahan baku.
6. *Sales*: Penjualan yang dilakukan oleh konsumen.
7. *Profit*: total keuntungan yang dihasilkan.
8. *Profit_Margin*: presentase laba yang dihasilkan perusahaan.
9. *Channel*: pembagian segmen berdasarkan sales (Reseller, Store, dan Telpon).
10. *Product_Category*: tipe produk berdasarkan kategori.

Tujuan pengolahan data adalah untuk menentukan 4 kategori berdasarkan profit margin yaitu kategori tinggi (profit besar kuantitas kecil), kategori sedang (profit sedang dengan kuantitas kecil), kategori rendah (profit rendah dengan kuantitas besar), dan kategori sangat rendah (profit sangat rendah dengan kuantitas besar).



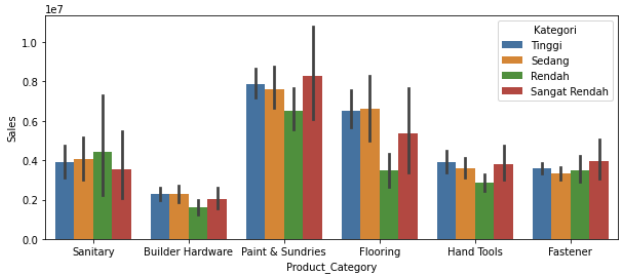
Gambar 4. Visualisasi Product Category Berdasarkan Profit

Pada gambar 4 memperlihatkan jika *Paint & Sundries* menghasilkan profitabilitas tertinggi disusul dengan *Flooring*, sementara itu profit terendah dimiliki oleh *builder hardware*.



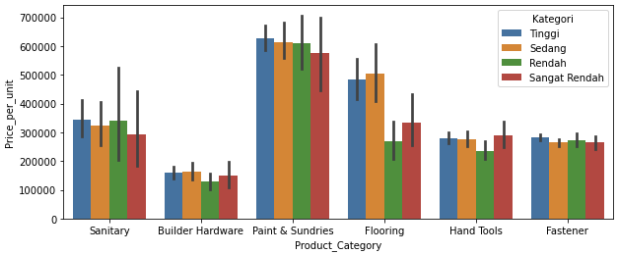
Gambar 5. Visualisasi Product Category Berdasarkan Order Quantity

Pada gambar 5 menghasilkan produk kategori *builder hardware* mempunyai *order quantity* tertinggi, dan terendah oleh *sanitary*.



Gambar 6. Visualisasi Product Category Berdasarkan Sales

Pada gambar 6 memperlihatkan jika *paint & sundries* dan *flooring* mempunyai *sales* tertinggi, sementara *builder hardware* menghasilkan *sales* terendah.



Gambar 7. Visualisasi Product Category Berdasarkan Price Per Unit

Pada gambar 7 memperlihatkan jika *paint & sundries* dan *flooring* menghasilkan *price per unit* tertinggi, sementara *builder hardware* menghasilkan *price per unit* terendah. Kesimpulannya, visualisasi data memperlihatkan bahwa kategori *paint & sundries* dan *flooring* menghasilkan kategori tertinggi dari setiap *frequency* dibandingkan dengan kategori yang lain. Sementara itu *builder hardware* merupakan kategori terendah berdasarkan *average frequency*.

3.5 Ground Truth

1. Feature selection

   Pemilihan variabel pada dataset PT. Maju Jaya Makmur Sejahtera yang bertujuan untuk memilih variabel yang paling optimal dengan menggunakan metode *Feature Selection Recursive Feature Elimination.*

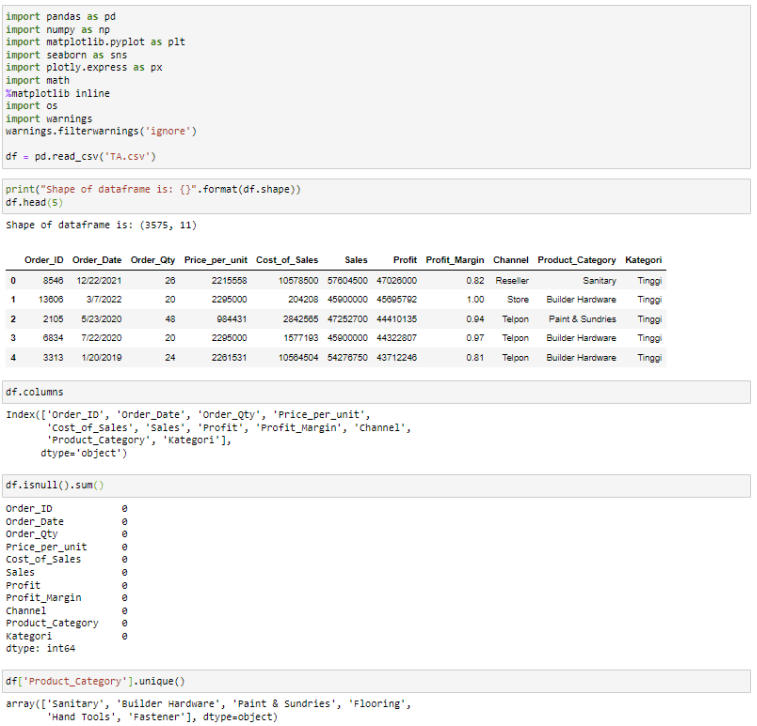
2. Naïve Bayes

   Naïve Bayes pada penelitian ini digunakan untuk menentukan klasifikasi dari data penjualan PT. Maju Jaya Makmur Sejahtera, berdasarkan variabel yang telah diseleksi pada tahap *feature selection* kemudian di klasifikasi dengan 4 kategori yaitu tinggi dengan order quantity >700, profit >10000000, sedang dengan order quantity >50 dan <700, profit <10000000 dan >5000000, rendah dengan order quantity >30 dan <50, profit <5000000 dan >3000000, dan sangat rendah dengan order quantity <30 dan profit <3000000. Selanjutnya *dataset* dibagi menjadi 2, yaitu data latih dan data tes. Data latih adalah data yang dilatih dengan memasukan data kedalam model *Naïve bayes* yang kemudian dijalankan perhitungan - perhitungan *Naïve bayes* yang kemudian hasilnya dibandingkan dengan data tes untuk dihitung tingkat akurasi dari model yang telah dibuat.

3.6 Preprocessing Data

1. PT. Maju Jaya Makmur Sejahtera
   a. Data Extraction

   Mengambil dan memilih data dari dataset PT. Maju Jaya Makmur Sejahtera.



```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import plotly.express as px
import math
%matplotlib inline
import os
import warnings
warnings.filterwarnings('ignore')

df = pd.read_csv('TA.csv')
```

```
print("Shape of dataframe is: {}".format(df.shape))
df.head(5)
```

```
Shape of dataframe is: (3575, 11)
```

	Order_ID	Order_Date	Order_Qty	Price_per_unit	Cost_of_Sales	Sales	Profit	Profit_Margin	Channel	Product_Category	Kategori
0	8546	12/22/2021	26	2215558	10578500	57604500	47026000	0.82	Reseller	Sanitary	Tinggi
1	13606	3/7/2022	20	2295000	204208	45900000	45695792	1.00	Store	Builder Hardware	Tinggi
2	2105	5/23/2020	48	984431	2842565	47252700	44410135	0.94	Telpon	Paint & Sundries	Tinggi
3	6834	7/22/2020	20	2295000	1577193	45900000	44322807	0.97	Telpon	Builder Hardware	Tinggi
4	3313	1/20/2019	24	2261531	10564504	54276750	43712246	0.81	Telpon	Builder Hardware	Tinggi

```
df.columns
```

```
Index(['Order_ID', 'Order_Date', 'Order_Qty', 'Price_per_unit',
       'Cost_of_Sales', 'Sales', 'Profit', 'Profit_Margin', 'Channel',
       'Product_Category', 'Kategori'],
      dtype='object')
```

```
df.isnull().sum()
```

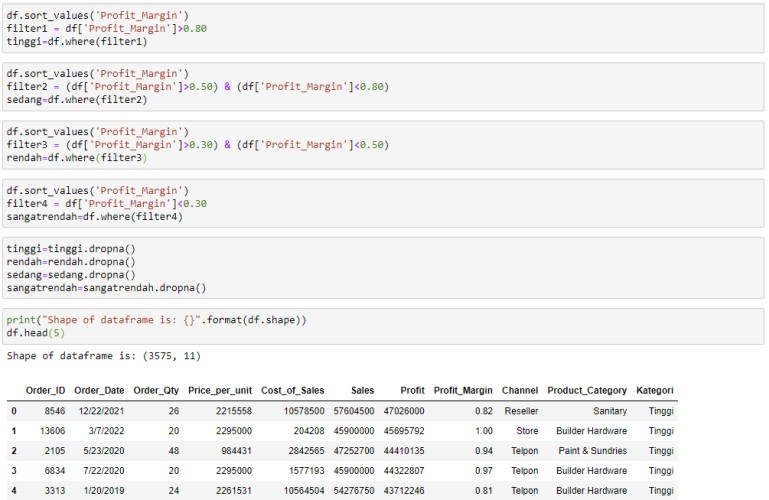
```
Order_ID          0
Order_Date        0
Order_Qty         0
Price_per_unit    0
Cost_of_Sales     0
Sales             0
Profit            0
Profit_Margin     0
Channel           0
Product_Category  0
Kategori          0
dtype: int64
```

```
df['Product_Category'].unique()
```

```
array(['Sanitary', 'Builder Hardware', 'Paint & Sundries', 'Flooring',
       'Hand Tools', 'Fastener'], dtype=object)
```

Gambar 8. Data Extraction

Pada gambar 8 dapat dilihat proses ekstraksi data yang dilakukan pada dataset penjualan PT. Maju Jaya Makmur Sejahtera dengan menggunakan *library python* pada *jupyter notebook*. Pada gambar diatas dijalankan beberapa sintaks untuk melihat bentuk data, format data, *feature - feature* yang ada pada data, melihat jumlah *row* yang memiliki nilai *null*, dan melihat *value* unik dari *row* pada *feature product_name*.



```
df.sort_values('Profit_Margin')
filter1 = df['Profit_Margin']>0.80
tinggi=df.where(filter1)
```

```
df.sort_values('Profit_Margin')
filter2 = (df['Profit_Margin']>0.50) & (df['Profit_Margin']<0.80)
sedang=df.where(filter2)
```

```
df.sort_values('Profit_Margin')
filter3 = (df['Profit_Margin']>0.30) & (df['Profit_Margin']<0.50)
rendah=df.where(filter3)
```

```
df.sort_values('Profit_Margin')
filter4 = df['Profit_Margin']<0.30
sangatrendah=df.where(filter4)
```

```
tinggi=tinggi.dropna()
rendah=rendah.dropna()
sedang=sedang.dropna()
sangatrendah=sangatrendah.dropna()
```

```
print("Shape of dataframe is: {}".format(df.shape))
df.head(5)
```

```
Shape of dataframe is: (3575, 11)
```

	Order_ID	Order_Date	Order_Qty	Price_per_unit	Cost_of_Sales	Sales	Profit	Profit_Margin	Channel	Product_Category	Kategori
0	8546	12/22/2021	26	2215558	10578500	57604500	47026000	0.82	Reseller	Sanitary	Tinggi
1	13606	3/7/2022	20	2295000	204208	45900000	45695792	1.00	Store	Builder Hardware	Tinggi
2	2105	5/23/2020	48	984431	2842565	47252700	44410135	0.94	Telpon	Paint & Sundries	Tinggi
3	6834	7/22/2020	20	2295000	1577193	45900000	44322807	0.97	Telpon	Builder Hardware	Tinggi
4	3313	1/20/2019	24	2261531	10564504	54276750	43712246	0.81	Telpon	Builder Hardware	Tinggi

Gambar 9. Pemilihan Kategori Tinggi, Sedang, Rendah dan Sangat Rendah

Pada gambar 9 adalah proses pemilihan kategori berdasarkan kategori tinggi (profit besar kuantitas kecil), kategori sedang (profit sedang dengan kuantitas kecil), kategori rendah (profit rendah dengan kuantitas besar), dan kategori sangat rendah (profit sangat rendah dengan kuantitas besar) sehingga memiliki rekap profit yang dihasilkan dari penjualan produk yang diambil berdasarkan kuantitas dan jumlah terjualnya dan tujuan akhir klasifikasi data penjualan berdasarkan profit.

3.7 Mengembangkan Model

Mengembangkan *feature selection* dan memberikan rekomendasi berdasarkan hasil *train/test dataset.*

a. Feature selection

Feature Selection Dengan RFE

```
print("Shape of dataframe is: {}".format(df.shape))
df.head()
```

```
Shape of dataframe is: (3575, 11)
```

	Order_ID	Order_Date	Order_Qty	Price_per_unit	Cost_of_Sales	Sales	Profit	Profit_Margin	Channel	Product_Category	Kategori
0	8546	12/22/2021	26	2215558	10578500	57604500	47026000	0.82	1	1	1
1	13606	3/7/2022	20	2295000	204208	45900000	45695792	1.00	2	2	1
2	2105	5/23/2020	48	984431	2842565	47252700	44410135	0.94	3	3	1
3	6834	7/22/2020	20	2295000	1577193	45900000	44322807	0.97	3	4	1
4	3313	1/20/2019	24	2261531	10564504	54276750	43712246	0.81	3	4	1

```
df=df.drop(["Order_Date"], axis = 1)
```

```
df = df.astype('int')
```

```
# Feature Extraction with RFE
from pandas import read_csv
from sklearn.feature_selection import RFE
from sklearn.linear_model import LogisticRegression
# load data
dataframe = df
array = dataframe.values
X = array[:,0:10]
Y = array[:,6]
# feature extraction
rfe = RFE(estimator=LogisticRegression(solver='lbfgs'), n_features_to_select=6)
fit = rfe.fit(X, Y)
print("Num Features: %d" % fit.n_features_)
print("Selected Features: %s" % fit.support_)
print("Feature Ranking: %s" % fit.ranking_)
```

```
Num Features: 6
Selected Features: [ True  True  True  True  True False  True False]
Feature Ranking: [1 1 1 1 1 3 1 2]
```

```
df_selected_features = df.iloc[:,fit.support_]
df_selected_features
```

	Order_Qty	Price_per_unit	Cost_of_Sales	Sales	Profit	Product_Category
0	26	2215558	10578500	57604500	47026000	1
1	20	2295000	204208	45900000	45695792	2
2	48	984431	2842565	47252700	44410135	3
3	20	2295000	1577193	45900000	44322807	4
4	24	2261531	10564504	54276750	43712246	4
...	...	...	...	...	...	...
3570	10	15900	154439	159000	4561	4
3571	4	17005	64364	68020	3656	4
3572	12	268000	3212485	3216000	3515	6
3573	6	9990	55444	59940	3496	4
3574	10	28990	288593	289900	1307	4

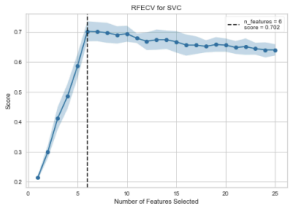
3575 rows × 6 columns

```
from sklearn.svm import SVC
from yellowbrick.model_selection import RFECV

X, y = df_selected_features(
    n_samples=1000, n_features=25, n_informative=6, n_redundant=2,
    n_repeated=0, n_classes=8, n_clusters_per_class=1, random_state=0
)

visualizer = RFECV(SVC(kernel='linear', C=1))

visualizer.fit(X, y)
visualizer.show()
```

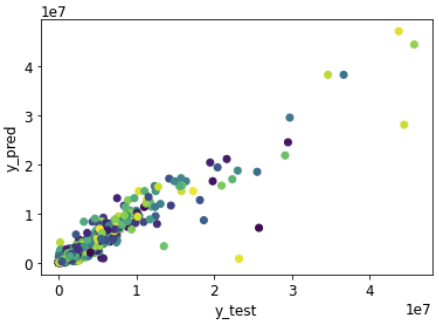


Gambar 10. Feature Selection

Setelah proses encode, kemudian data masuk pada tahap *feature selection*. algoritma yang digunakan pada proses ini adalah *RFE* dan *logistic regression* yang diambil dari *library scikit learn*. Value dari data yang bentuk awalnya objek dimasukan kedalam *array*, kemudian dibuat variabel penampung baru x dan y. pada data x array discipline value *index* awalnya adalah 0 sampai indeks 11, dan untuk data penampung y berisikan *array* yang di *splice* dari indeks angka 0 hingga indeks ke 6. Kemudian kedua variabel tersebut dimasukan kedalam model algoritma *RFE*, dengan konfigurasi untuk memilih 6 *feature* dari total 11 *feature*. Kemudian hasil dari kalkulasi di *print* menjadi bentuk *array* yang berisikan nilai relevansi *feature* dengan *dataset* dan mendapatkan hasil score akurasi 0.702.

Implementasi algoritma *Naïve Bayes* dengan *sklearn:*

```
N = 1073
colors = np.random.rand(N)
plt.scatter(
    x=y_test,
    y=y_pred,
    c=colors
)
plt.xlabel("y_test")
plt.ylabel("y_pred")
plt.show()
```



Gambar 10. Visualisasi Scatter Plot

Pseudocode:
1: **Algoritma:** *Feature Selection* RFE untuk seleksi *best feature*.
2: ***Prosedure***:
3: *Repeat for l in {1-a}*
4: *Rank set fe using RF(Tr;Fe)*
5: *F*<- last ranked feature fe*
6: *R(a-i+1)<-F**
7: *Fe<- remove last ranked feature from set of features*
8: *End*
9: *Return ranked liat R*
10: *End procedure.*

Train/Test Split Dataset

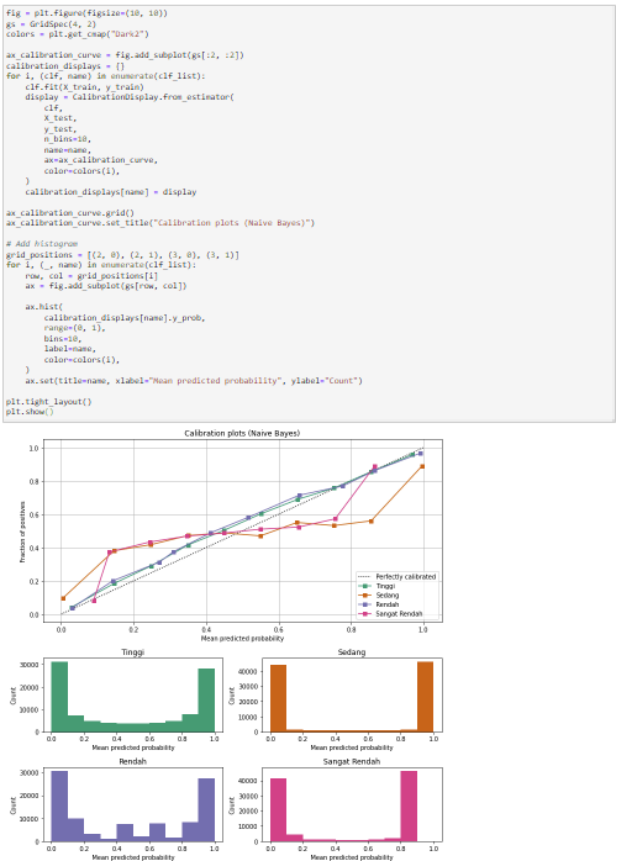
```
Sanitary = df_selected_features[df_selected_features.Product_Category == "1"]
BuilderHardware = df_selected_features[df_selected_features.Product_Category == "2"]
PaintdanSundries = df_selected_features[df_selected_features.Product_Category == "3"]
Flooring= df_selected_features[df_selected_features.Product_Category == "4"]
HandTools = df_selected_features[df_selected_features.Product_Category == "5"]
Fastener = df_selected_features[df_selected_features.Product_Category == "6"]
```

```
x = df_selected_features.drop(["Product_Category"], axis = 1)
y = df_selected_features.Product_Category.values
```

```
from sklearn.model_selection import train_test_split
X_train, x_test, y_train, y_test = train_test_split(x, y, test_size = 0.25, random_state = 5)
```

Gambar 11. *Train/Test Split Dataset*

Pada gambar 11 adalah proses *train/test split dataset* yang bertujuan untuk menilai hasil algoritma dari *Naïve bayes* dan *feature selection* dengan menggunakan *X_train* dan *y_train*. Data latih dan data *training* diambil dari kedua variabel yang telah dibuat sebelumnya. yang kemudian dimasukan kedalam sintaks train_test_split, dengan variabel x dan y sebagai parameter dan test sizenya 0.25, dan random statenya 5. Kemudian data yang telah dibagi dimasukan kedalam model *Naïve bayes* dan di print hasil dari kalkulasi algoritmanya.



Gambar 12. Probabilitas Pada Setiap Kategori

Pada gambar 12 memperlihatkan jika kategori tinggi menghasilkan probabilitas yang lebih baik daripada kategori sedang, rendah dan sangat rendah. Hasil penelitian menunjukan bahwa skor probabilitas pada kategori tinggi dan rendah yang lebih akurat dibandingkan dengan kategori sedang dan sangat rendah. Kategori sedang dan sangat rendah melampaui kurva line chart sehingga mendapatkan skor yang kurang baik.

Tabel 1. Hasil Probabilitas Pada Product Category

Product Category	Score Probabilitas
Builder Hardware Tinggi	0.9478
Builder Hardware Sedang	0.7576
Builder Hardware Rendah	0.7576
Builder Hardware Sangat Rendah	1
Paint & Sundries Tinggi	0.6000
Paint & Sundries Sedang	1
Paint & Sundries Rendah	1
Paint & Sundries Sangat Rendah	1
Sanitary Tinggi	0.9375
Sanitary Sedang	1
Sanitary Rendah	1
Sanitary Sangat Rendah	1
Hand Tools Tinggi	0.9359
Hand Tools Sedang	1
Hand Tools Rendah	1
Hand Tools Sangat Rendah	1
Fastener Tinggi	0.9550
Fastener Sedang	1
Fastener Rendah	1
Fastener Sangat Rendah	1

Flooring Tinggi	0.9379
Flooring Sedang	1
Flooring Rendah	1
Flooring Sangat Rendah	1

Pada tabel 1 memperlihatkan jika *product category builder hardware* menghasilkan *score* akurasi *probability* terendah. Hasil penelitian menunjukan bahwa skor probabilitas pada *product category fastener, sanitary, paint & sundries, flooring* dan *hand tools* lebih akurat dibandingkan dengan *product category builder hardware.*

3.8 Evaluasi dan *Tuning*

Classifier	Brier loss	Log loss	Roc auc	Precision	Recall	F1
Tinggi	0.098921	0.323178	0.937457	0.872009	0.851408	0.861586
Sedang	0.117608	0.782246	0.940374	0.857400	0.875941	0.866571
Rendah	0.098332	0.368412	0.938613	0.883065	0.836224	0.859007
Sangat Rendah	0.108880	0.368896	0.940201	0.861106	0.871277	0.866161

Gambar 13. Hasil Confusion Matrix

Pada gambar 13, *brier loss* dan *log loss* didefinisikan jika semakin kecil skor maka semakin baik, *ROC-AUC* didefinisikan jika score mendekati 1 maka menunjukan bahwa uji diagnostik semakin baik, precision didefinisikan semakin kecil *false positive (FP)* membuat *precision* semakin besar, *recall* didefinisikan semakin kecil *false negative (FN)* membuat *recall* semakin besar, *f1-score* adalah *harmonic mean* dari *precision* dan *recall* yang mengindikasikan bahwa jika score *f1-score* mendekati 1 maka *precision* dan *recall* juga mempunyai *score* yang baik yaitu mendekati 1. Hasil dari *confusion matrix* pada kategori tinggi dan rendah mendapati hasil klasifikasi yang terbaik dibandingkan dengan kategori sedang dan sangat rendah dengan melihat *score brier loss, log loss* dan *f1-score* nya.

IV. Kesimpulan

Dari penelitian yang telah dilakukan, peneliti menemukan jika analisis untuk setiap peran yang ada pada sistem yang sedang berjalan pada PT. Maju Jaya Makmur Sejahtera walaupun memiliki alir yang berbeda - beda (antara system, admin, dan pelanggan), peran - peran tersebut saling berkaitan dan mendukung satu sama lain, contohnya ada pada bagian bagian admin dan pelanggan yang telah menyelesaikan transaksi atau *financial process* kemudian data tersebut direkap oleh sistem melalui excel. Dengan penerapan algoritma *naïve bayes* dengan *feature selection* pada *dataset* PT. Maju Jaya Makmur Sejahtera memberikan probabilitas tinggi, sedang, rendah, dan sangat rendah serta peran *feature selection* dengan algoritma *recursive feature elimination* menyeleksi 10 fitur *dataset* menjadi total 6 fitur. Hasil akurasi probabilitas pada *builder hardware* 1.59, *fastener* 0.11, *flooring* 0.11, *hand tools* 0.38, *paint & sundries* 9.5, *sanitary* 0.31. Sedangkan pada implementasi *naïve bayes* dengan *python* menghasilkan *score* probabilitas pada *builder hardware* 1, *fastener* 0.95, *flooring* 0.93, *hand tools* 1, *paint & sundries* 0.60, *sanitary* 0.93. Secara keseluruhan, akurasi yang didapatkan dari model algoritma *naïve bayes* sebesar 88%, *precision* 87%, *recall* 85%, dan *F1-score* 86%. Saran untuk penelitian ini dapat menggunakan algoritma untuk optimasi seperti *Binary Particle Swarm Optimization* (BPSO) untuk meningkatkan akurasi model *naïve bayes.* Selain itu hasil dari perhitungan *naïve bayes* pada penelitian ini dapat dikembangkan untuk dijadikan sebagai dasar *requirement* untuk pengembangan aplikasi.

Daftar Pustaka

[1] N. Deepa, J. Sathya Priya, and T. Devi, “Towards applying internet of things and machine learning for the risk prediction of COVID-19 in pandemic situation using Naïve Bayes classifier for improving accuracy,” *Mater Today Proc*, vol. 62, pp. 4795–4799, 2022, doi: 10.1016/j.matpr.2022.03.345.

[2] Valentinus Fernando, Sujono Fabian, Ariansyah Ilham, and Capah Dwi Ade Handayani, “IMPLEMENTATION OF DATA MINING WITH CLASSIFICATION AND FORECASTING METHOD USE MODEL GAUSSIAN NAÏVE BAYES FOR BUILDING STORE (STUDI CASE: TB SINAR JAYA),” *Jurnal Teknik Informatika*, vol. 4, no. 2, 2023.

[3] Yoga Religia and A. Amali, "Perbandingan Optimasi Feature Selection pada Naïve Bayes untuk Klasifikasi Kepuasan Airline Passenger," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 5, no. 3, pp. 527–533, Jun. 2021, doi: 10.29207/resti.v5i3.3086.
[4] I. Romli, Fairuz Kharida, and Chandra Naya, "Determination of Customer Satisfaction of Tax Service Office Services Using C4.5 and PSO," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 4, no. 2, pp. 296–302, Apr. 2020, doi: 10.29207/resti.v4i2.1718.
[5] Ş. Özlem and O. F. Tan, "Predicting cash holdings using supervised machine learning algorithms," *Financial Innovation*, vol. 8, no. 1, p. 44, Dec. 2022, doi: 10.1186/s40854-022-00351-8.
[6] A. Ambarwari, Q. Jafar Adrian, and Y. Herdiyeni, "Analysis of the Effect of Data Scaling on the Performance of the Machine Learning Algorithm for Plant Identification," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 4, no. 1, pp. 117–122, Feb. 2020, doi: 10.29207/resti.v4i1.1517.
[7] A. Nugroho and Y. Religia, "Analisis Optimasi Algoritma Klasifikasi Naïve Bayes menggunakan Genetic Algorithm dan Bagging," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 5, no. 3, pp. 504–510, Jun. 2021, doi: 10.29207/resti.v5i3.3067.
[8] Y. Astuti, Yova Ruldeviyani, Faris Salbari, and Aldiansah Prayogi, "Sentiment Analysis of Electricity Company Service Quality Using Naïve Bayes," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 7, no. 2, pp. 389–396, Mar. 2023, doi: 10.29207/resti.v7i2.4627.
[9] B. S. Prakoso, D. Rosiyadi, H. S. Utama, and D. Aridarma, "Klasifikasi Berita Menggunakan Algoritma Naïve Bayes Classifer Dengan Seleksi Fitur Dan Boosting," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 3, no. 2, pp. 227–232, Aug. 2019, doi: 10.29207/resti.v3i2.1042.
[10] B. S. Prakoso, D. Rosiyadi, H. S. Utama, and D. Aridarma, "Klasifikasi Berita Menggunakan Algoritma Naïve Bayes Classifer Dengan Seleksi Fitur Dan Boosting," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 3, no. 2, pp. 227–232, Aug. 2019, doi: 10.29207/resti.v3i2.1042.
[11] Dores Ardiansyah and Herdi Tazkiyah, "Arsitektur Enterprise untuk Lembaga Swadaya Masyarakat berdasarkan The Open Group Architecture Framework (TOGAF) (Enterprise Architecture for Non-Governmantal Organization based on The Open Group Architecture Framework (TOGAF))," *IPTEK-KOM*, vol. 23, no. 2, pp. 155–168, 2021.
[12] I. Ilhamsyah, A. Y. Rahman, and I. Istiadi, "Klasifikasi Kualitas Biji Kopi Menggunakan MultilayerPerceptron Berbasis Fitur Warna LCH," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 5, no. 6, pp. 1008–1017, Dec. 2021, doi: 10.29207/resti.v5i6.3438.
[13] Kade Bramasta Vikana Putra, I Putu Agung Bayupati, and Dewa Made Sri Arsa, "Klasifikasi Citra Daging Menggunakan Deep Learning dengan Optimisasi Hard Voting," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 5, no. 4, pp. 656–662, Aug. 2021, doi: 10.29207/resti.v5i4.3247.
[14] Evi Purnamasari, D. P. Rini, and Sukemi, "Feature Selection using Particle Swarm Optimization Algorithm in Student Graduation Classification with Naïve Bayes Method," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 4, no. 3, pp. 469–475, Jun. 2020, doi: 10.29207/resti.v4i3.1833.
[15] H. Xie, L. Zhang, C. P. Lim, Y. Yu, and H. Liu, "Feature Selection Using Enhanced Particle Swarm Optimisation for Classification Models," *Sensors*, vol. 21, no. 5, p. 1816, Mar. 2021, doi: 10.3390/s21051816.
[16] P. Arsi, R. Wahyudi, and R. Waluyo, "Optimasi SVM Berbasis PSO pada Analisis Sentimen Wacana Pindah Ibu Kota Indonesia," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 5, no. 2, pp. 231–237, Apr. 2021, doi: 10.29207/resti.v5i2.2698.
[17] A. Septiarini, Rizqi Saputra, Andi Tejawati, and Masna Wati, "Deteksi Sarung Samarinda Menggunakan Metode Naïve Bayes Berbasis Pengolahan Citra," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 5, no. 5, pp. 927–935, Oct. 2021, doi: 10.29207/resti.v5i5.3435.
[18] A. I. S. Azis, Budy Santoso, and Serwin, "LL-KNN ACW-NB: Local Learning K-Nearest Neighbor in Absolute Correlation Weighted Naïve Bayes for Numerical Data Classification," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 4, no. 1, pp. 28–36, Feb. 2020, doi: 10.29207/resti.v4i1.1348.
[19] N. Umar and M. Adnan Nur, "Application of Naïve Bayes Algorithm Variations On Indonesian General Analysis Dataset for Sentiment Analysis," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 6, no. 4, pp. 585–590, Aug. 2022, doi: 10.29207/resti.v6i4.4179.
[20] M. Guntur, J. Santony, and Y. Yuhandri, "Prediksi Harga Emas dengan Menggunakan Metode Naïve Bayes dalam Investasi untuk Meminimalisasi Resiko," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 2, no. 1, pp. 354–360, Apr. 2018, doi: 10.29207/resti.v2i1.276.
[21] Budiyanto Arif and Dwiasnati Saruni, "The Prediction of Best-Selling Product Using Naïve Bayes Algorithm," *International Journal of Computer Techniques*, vol. 5, no. 6, pp. 68–74, 2018.
[22] D. M. Tarigan, Dian Palupi Rini, and Samsuryadi, "Feature Selection in Classification of Blood Sugar Disease Using Particle Swarm Optimization (PSO) on C4.5 Algorithm," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 4, no. 3, pp. 569–575, Jun. 2020, doi: 10.29207/resti.v4i3.1881.