OPEN ACCESS

Volume 9 (3), July-September 2025, 852-862

E-ISSN:2580-1643

Jurnal JTIK (Jurnal Teknologi Informasi dan Komunikasi)

DOI: https://doi.org/10.35870/jtik.v9i3.3562

Implementasi Algoritma C4.5 Untuk Memprediksi Penjualan Sepatu Boots Terlaris di Toko Parabellum

Mochamad Kamil [1], Yustika Erliani [2*]

[1,2] Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Mercu Buana, Kota Jakarta Barat, Daerah Khusus Ibukota Jakarta, Indonesia.*

a r t i c l e i n f o

Article history:
Received 10 December 2024
Received in revised form
20 December 2024
Accepted 15 January 2025
Available online Juli 2025.

Keywords:
C4.5 Algorithm; Sales; Boots.

Kata Kunci:
Algoritma C4.5; Penjualan; Sepatu Boots.

a b s t r a c t

In the busines world, especially the shoe shop business, a marketing strategy is needed in order to increase product sales. One of the shoe shop businesses is parabellum. The high number of visitors to Parabellum's online and offlie stores resulted in the stock of boots being sold out, so that when customers looked for a variety og boots, these boots were no longer available. From this problem, the author helps predict by analyzing data processing using a comparison og the C4.5 algorithm, where the C4.5 algorithm is an algorithm, it will have accurate result in predicting shoe sales data at the Prabellum Store.

a b s t r a k

Dalam dunia bisnis, terutama bisnis toko sepatu diperlukan strategi dalam pemasaran agar dapat meningkatkan penjulanan produk. Salah satu bisnis toko sepatu adalah Parabellum. Tingginya jumlah pengunjung di online atau offline store Parabellum mengakibatkan persediaan sepatu boots habis terjual, sehingga ketika customer mencari variasi sepatu boots tesebut sudah tidak tersedia. Dari masalah tersebut, penulis membantu memprediksi dengan melakukan analisa pengolahan data dengan menggunakan perbandingan algoritma C4.5 yang dimana algoritma C4.5 merupakan algoritma untuk membantu mengambil keputusan dengan menggunakan struktur pohon keputusan. Berdasarkan algoritma tersebut akan memiliki hasil yang akurat dalam memprediksi data penjualan sepatu di Toko Parabellum.

Corresponding Author. Email: Yustika.erliani@mercubuana.ac.id [2].

ACM Computing Classification System (CCS)
EBSCOhost
Communication and Mass Media Complete (CMMC)

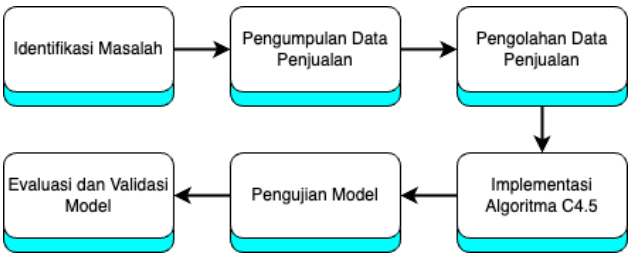
1. Pendahuluan

Dalam persaingan dunia bisnis yang semakin ketat, pengusaha dituntut untuk terus berpikir tentang strategi dan terobosan baru agar usaha yang dijalani dapat terus berkembang, terutama dalam bisnis sepatu yang saat ini sangat pesat. Oleh karena itu, untuk menghadapi persaingan yang semakin intensif dalam industri sepatu, diperlukan peningkatan dalam penjualan dan pemasaran produk dengan memanfaatkan data penjualan yang ada. Salah satu bisnis sepatu tersebut adalah Parabellum, yang berlokasi di Jl. Ciputat Raya No.89A, RT.1/RW.12, Pd. Pinang, Kec. Kby. Lama, Kota Jakarta Selatan, Daerah Khusus Ibukota. Parabellum menjual berbagai jenis sepatu boots untuk pria dan wanita. Meningkatnya jumlah pengunjung di *online* dan *offline* store Parabellum menyebabkan persediaan sepatu boots cepat habis terjual. Akibatnya, saat pelanggan mencari varian sepatu boots, produk tersebut sudah tidak tersedia. Kondisi ini mempengaruhi penilaian pelanggan terhadap toko Parabellum. Oleh karena itu, penelitian ini bertujuan untuk membantu Toko Parabellum dalam memprediksi stok sepatu boots di masa mendatang. Berdasarkan masalah yang ada, diperlukan analisis pengolahan data untuk melihat pola penjualan yang dapat digunakan untuk memprediksi penjualan di masa depan. Penentuan persediaan stok sepatu berdasarkan data penjualan saat ini masih menghadapi kendala.

Oleh karena itu, diperlukan metode yang dapat membantu penulis dalam mengetahui produk mana yang perlu ditingkatkan sesuai dengan data hasil penjualan sepatu boots. Metode yang dipilih oleh penulis adalah penerapan algoritma *C4.5* untuk memprediksi penjualan di masa depan. Berdasarkan penelitian sebelumnya yang berjudul "*Penerapan Algoritma C4.5 untuk Penentuan Ketersediaan Barang E-commerce*" (Pritalia, 2021) dapat disimpulkan bahwa algoritma *C4.5* efektif dalam menentukan ketersediaan barang *e-commerce*. Algoritma ini membantu bagian bisnis dalam mengetahui status barang yang siap dijual, sehingga dapat menjadi acuan dalam pengambilan keputusan untuk menerima pesanan dan menjaga ketersediaan barang tanpa backorder. Hal ini penting untuk mempertahankan kepercayaan pelanggan dalam transaksi barang. Berdasarkan penelitian sebelumnya, algoritma *C4.5* merupakan algoritma pohon keputusan (Ferdiansyah, 2020) yang memiliki keunggulan dalam mengklasifikasikan sekumpulan data, seperti data penjualan, untuk merumuskan prediksi terkait stok sepatu yang diminati pelanggan di toko Parabellum. Dengan demikian, penelitian ini akan membantu dalam mengklasifikasikan arah kecenderungan data penjualan, memungkinkan pemahaman yang lebih baik mengenai prediksi penjualan sepatu terlaris di Toko Parabellum, yang pada gilirannya akan mempengaruhi ketersediaan stok produk berdasarkan data yang sudah dianalisis. Maka dari itu, penulis ingin menerapkan algoritma *C4.5* untuk memprediksi penjualan berdasarkan dataset yang ada, dengan harapan algoritma ini dapat memberikan hasil yang lebih akurat dalam memprediksi data penjualan sepatu di Toko Parabellum.

2. Metodologi Penelitian

Desain penelitian yang disusun oleh peneliti akan digunakan sebagai panduan dalam pelaksanaan penelitian. Tujuan dari desain penelitian ini adalah memberikan arahan yang jelas dan terstruktur bagi peneliti dalam melaksanakan seluruh tahapan penelitian. Di bawah ini disajikan gambar yang menggambarkan alur desain penelitian terkait dengan prediksi penjualan sepatu boots terlaris di Toko Parabellum:



Gambar 1. Alur Penelitian

Identifikasi Masalah

Identifikasi masalah dalam penelitian ini adalah memprediksi stok sepatu boots yang selalu tersedia dan sesuai dengan keinginan atau permintaan pelanggan di Toko Parabellum. Hal ini dilakukan dengan menggunakan algoritma *C4.5* untuk memperoleh prediksi yang akurat.

Pengumpulan Data Penjualan

Data yang dikumpulkan sebanyak 6934 data, berupa dataset penjualan sepatu boots yang diambil dari Toko Parabellum selama periode Januari 2023 hingga Desember 2023. Atribut-atribut yang digunakan dalam penelitian ini meliputi *Product Name*, *Category Name*, *Size*, *Price*, *Quantity Cat*, dan *Label*.

Pengolahan Data Penjualan

Data yang telah terkumpul memerlukan tahap *pre-processing*, yang merupakan rangkaian proses dalam *data mining* untuk mengungkapkan informasi baru yang sebelumnya tidak diketahui. Kinerja algoritma *data mining* juga dipengaruhi oleh atribut-atribut yang digunakan dalam tahap klasifikasi, mengingat tidak semua data atau atribut dalam dataset akan digunakan dalam proses ini. Tujuan dari tahap ini adalah untuk memastikan bahwa data yang digunakan sesuai dengan kebutuhan analisis. Dalam penelitian ini, beberapa jenis *pre-processing* akan diterapkan, yang mencakup tahapan sebagai berikut:

1) Data Cleaning

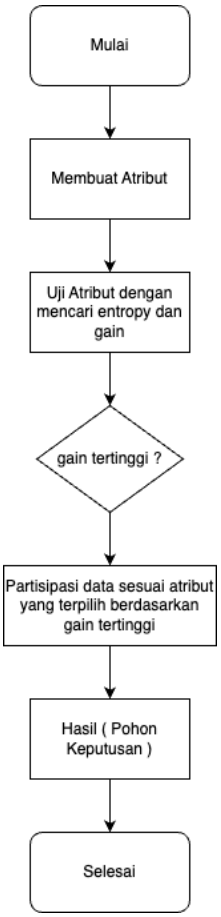
   Pada tahap ini, langkah pertama adalah membersihkan data penjualan sepatu boots di Toko Parabellum dari *missing values* dan duplikasi. *Missing values* merujuk pada kondisi di mana nilai atribut tidak valid atau hilang. Proses *data cleaning* mencakup eliminasi duplikasi data, verifikasi konsistensi data yang tidak sesuai, serta perbaikan data. Selama tahap ini, seluruh dataset akan diperiksa untuk mendeteksi *missing values* dan kemiripan baris (*duplicated rows*).

2) Data Transformation

   Data transformation pada penelitian ini bertujuan untuk mengubah tipe data. Tipe data yang semula dalam bentuk *object/string* akan diubah menjadi tipe data yang dapat dibaca oleh komputer, yaitu *integer*, agar dapat diproses lebih lanjut untuk implementasi algoritma.

3) Implementasi Algoritma C4.5

   Setelah data melalui tahap *pre-processing*, langkah selanjutnya adalah implementasi algoritma *C4.5* yang akan digunakan untuk memprediksi penjualan sepatu boots berdasarkan dataset yang telah dibersihkan dan diubah menjadi format yang sesuai untuk analisis.



Gambar 2. Diagram Implementasi Algoritma C4.5

Pada tahapan pertama, data *training* melibatkan perhitungan *split*, *gain*, dan *entropy value* untuk mendapatkan *gain ratio value*. Dalam algoritma pohon keputusan (*decision tree*), atribut-atribut biasanya dinyatakan dalam bentuk tabel yang mencakup atribut dan rekaman data. Atribut ini berfungsi sebagai parameter yang digunakan sebagai kriteria dalam pembentukan pohon keputusan, di mana simpul akar (*root node*) pada struktur pohon keputusan dipilih berdasarkan atribut dengan nilai *gain ratio* tertinggi. Selanjutnya, perhitungan dilakukan kembali untuk setiap atribut dengan mengecualikan atribut yang telah dipilih sebelumnya. Atribut dengan nilai *gain ratio* tertinggi akan dipilih sebagai simpul pohon berikutnya. Proses ini akan diulang hingga semua atribut memiliki kelas yang sesuai. Setelah semua atribut terklasifikasi, pohon keputusan awal akan ditampilkan dan aturan keputusan awal akan dihasilkan.

Pengujian Model

Pengujian model dilakukan pada data penjualan sepatu boots di Toko Parabellum menggunakan Google Colab dengan bahasa pemrograman Python. Percobaan dilakukan dalam empat tahap pembagian data, yaitu:

1) 60% data digunakan sebagai data *training* dan 40% lainnya digunakan sebagai data *testing*
2) 70% data digunakan sebagai data *training* dan 30% lainnya digunakan sebagai data *testing*
3) 80% data digunakan sebagai data *training* dan 20% lainnya digunakan sebagai data *testing*
4) 90% data digunakan sebagai data *training* dan 10% lainnya digunakan sebagai data *testing*

Evaluasi dan Validasi Model

Pada tahap evaluasi, hasil evaluasi yang telah dilakukan akan dibahas. Evaluasi bertujuan untuk menganalisis dan mengukur sejauh mana tingkat akurasi hasil yang telah dicapai. Hasil evaluasi akan dibandingkan untuk setiap percobaan yang dilakukan. Pada fase validasi, tujuan dari validasi adalah untuk memastikan bahwa model yang dihasilkan secara akurat merepresentasikan sistem sesuai dengan pemodelan yang telah dilakukan. Validasi bertujuan untuk memastikan bahwa semua kesimpulan yang dapat diambil dari hasil simulasi sistem tersebut dapat diterima dan dipertanggungjawabkan dalam batasan-batasan yang telah ditentukan.

3. Hasil dan Pembahasan

Hasil

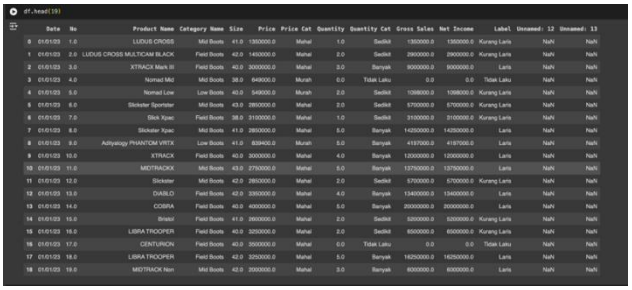
Dataset yang digunakan dalam penelitian ini merupakan data penjualan produk selama satu tahun, mulai dari Januari 2023 hingga Desember 2023. Dataset yang berhasil diimpor ke dalam Google Colab ini terdiri dari 6934 data dan memiliki atribut-atribut sebagai berikut: *Date*, *No*, *Product Name*, *Category Name*, *Size*, *Price*, *Price Cat*, *Quantity*, *Quantity Cat*, *Gross Sales*, *Net Income*, dan *Label*.

```
[ ] df.info()

<class 'pandas.core.frame.DataFrame'>
Index: 6935 entries, 0 to 6934
Data columns (total 12 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   Date           6935 non-null   object
 1   No             6935 non-null   float64
 2   Product Name   6935 non-null   object
 3   Category Name  6935 non-null   object
 4   Size           6897 non-null   float64
 5   Price          6935 non-null   float64
 6   Price Cat      6935 non-null   object
 7   Quantity       6897 non-null   float64
 8   Quantity Cat   6935 non-null   object
 9   Gross Sales    6935 non-null   float64
 10  Net Income     6935 non-null   float64
 11  Label          6935 non-null   object
dtypes: float64(6), object(6)
memory usage: 704.3+ KB
```

Gambar 3. Atribut Dataset

Berikut dijelaskan dibawah ini adalah 19 data teratas dalam dataset dan 19 data terbawah yang keseluruhan memiliki data yang masih mentah dalam dataset yang akan digunakan dalam penelitian ini.



Gambar 5. Dataset 19 Teratas

Gambar 6. Dataset 19 Terbawah

Preprocessing Dataset

Preprocessing bertujuan untuk membersihkan dataset mentah yang akan digunakan agar dapat diolah atau diproses lebih lanjut dalam pengolahan algoritma. Terdapat beberapa langkah dalam *preprocessing*, yaitu *data cleaning* dan *data transformation*.

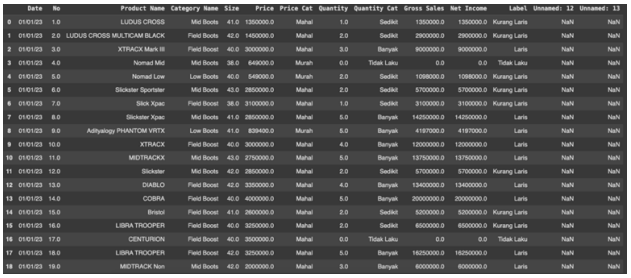
Data Cleaning

Pada penelitian ini, *data cleaning* dilakukan untuk menghapus atau memperbaiki kolom dan baris yang

tidak relevan dalam dataset. Tahapan yang dilakukan meliputi penghapusan kolom atau baris yang tidak digunakan, karena ketika dataset diimpor ke Python atau Google Colab, terdapat banyak kolom dan baris yang tidak terpakai namun tetap terbawa dalam dataset.

Pembersihan Kolom

Terdapat dua kolom yang tidak terpakai dalam dataset penjualan sepatu. Berikut ini adalah tampilan dataset sebelum dan sesudah dilakukan *cleaning*.



	Date	No	Product Name	Category Name	Size	Price	Price Cat	Quantity	Quantity Cat	Gross Sales	Net Income	Label	Unnamed: 12	Unnamed: 13
0	01/01/23	1.0	LUDUS CROSS	Mid Boots	41.0	1350000.0	Mahal	1.0	Sedikit	1350000.0	1350000.0	Kurang Laris	NaN	NaN
1	01/01/23	2.0	LUDUS CROSS MULTICAM BLACK	Field Boots	42.0	1450000.0	Mahal	2.0	Sedikit	2900000.0	2900000.0	Kurang Laris	NaN	NaN
2	01/01/23	3.0	XTRACX Mark III	Field Boost	40.0	3000000.0	Mahal	3.0	Banyak	9000000.0	9000000.0	Laris	NaN	NaN
3	01/01/23	4.0	Nomad Mid	Mid Boots	38.0	649000.0	Murah	0.0	Tidak Laku	0.0	0.0	Tidak Laku	NaN	NaN
4	01/01/23	5.0	Nomad Low	Low Boots	40.0	549000.0	Murah	2.0	Sedikit	1098000.0	1098000.0	Kurang Laris	NaN	NaN
5	01/01/23	6.0	Slickster Sportster	Mid Boots	43.0	2850000.0	Mahal	2.0	Sedikit	5700000.0	5700000.0	Kurang Laris	NaN	NaN
6	01/01/23	7.0	Slick Xpac	Field Boost	38.0	3100000.0	Mahal	1.0	Sedikit	3100000.0	3100000.0	Kurang Laris	NaN	NaN
7	01/01/23	8.0	Slickster Xpac	Mid Boots	41.0	2850000.0	Mahal	5.0	Banyak	14250000.0	14250000.0	Laris	NaN	NaN
8	01/01/23	9.0	Adityalogy PHANTOM VRTX	Low Boots	41.0	839400.0	Murah	5.0	Banyak	4197000.0	4197000.0	Laris	NaN	NaN
9	01/01/23	10.0	XTRACX	Field Boost	40.0	3000000.0	Mahal	4.0	Banyak	12000000.0	12000000.0	Laris	NaN	NaN
10	01/01/23	11.0	MIDTRACKX	Mid Boots	43.0	2750000.0	Mahal	5.0	Banyak	13750000.0	13750000.0	Laris	NaN	NaN
11	01/01/23	12.0	Slickster	Mid Boots	42.0	2850000.0	Mahal	2.0	Sedikit	5700000.0	5700000.0	Kurang Laris	NaN	NaN
12	01/01/23	13.0	DIABLO	Field Boost	42.0	3350000.0	Mahal	4.0	Banyak	13400000.0	13400000.0	Laris	NaN	NaN
13	01/01/23	14.0	COBRA	Field Boost	40.0	4000000.0	Mahal	5.0	Banyak	20000000.0	20000000.0	Laris	NaN	NaN
14	01/01/23	15.0	Bristol	Field Boost	41.0	2600000.0	Mahal	2.0	Sedikit	5200000.0	5200000.0	Kurang Laris	NaN	NaN
15	01/01/23	16.0	LIBRA TROOPER	Field Boost	40.0	3250000.0	Mahal	2.0	Sedikit	6500000.0	6500000.0	Kurang Laris	NaN	NaN
16	01/01/23	17.0	CENTURION	Field Boost	40.0	3500000.0	Mahal	0.0	Tidak Laku	0.0	0.0	Tidak Laku	NaN	NaN
17	01/01/23	18.0	LIBRA TROOPER	Field Boost	42.0	3250000.0	Mahal	5.0	Banyak	16250000.0	16250000.0	Laris	NaN	NaN
18	01/01/23	19.0	MIDTRACK Non	Mid Boots	42.0	2000000.0	Mahal	3.0	Banyak	6000000.0	6000000.0	Laris	NaN	NaN

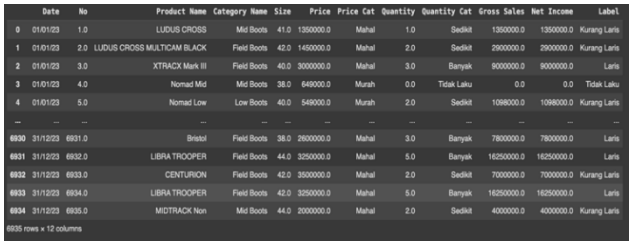
Gambar 7. Dataset sebelum dilakukan penghapusan baris

```
# Remove empty rows
df = df.dropna(how='all')

# Remove duplicate rows
df = df.drop_duplicates()

# Check the number of rows after cleaning
print(len(df))
```

Gambar 8. Code untuk membersihkan baris



	Date	No	Product Name	Category Name	Size	Price	Price Cat	Quantity	Quantity Cat	Gross Sales	Net Income	Label
0	01/01/23	1.0	LUDUS CROSS	Mid Boots	41.0	1350000.0	Mahal	1.0	Sedikit	1350000.0	1350000.0	Kurang Laris
1	01/01/23	2.0	LUDUS CROSS MULTICAM BLACK	Field Boots	42.0	1450000.0	Mahal	2.0	Sedikit	2900000.0	2900000.0	Kurang Laris
2	01/01/23	3.0	XTRACX Mark III	Field Boots	40.0	3000000.0	Mahal	3.0	Banyak	9000000.0	9000000.0	Laris
3	01/01/23	4.0	Nomad Mid	Mid Boots	38.0	649000.0	Murah	0.0	Tidak Laku	0.0	0.0	Tidak Laku
4	01/01/23	5.0	Nomad Low	Low Boots	40.0	549000.0	Murah	2.0	Sedikit	1098000.0	1098000.0	Kurang Laris
...	...	...	...	...	...	...	...	...	...	...	...	...
6930	31/12/23	6931.0	Bristol	Field Boots	38.0	2600000.0	Mahal	3.0	Banyak	7800000.0	7800000.0	Laris
6931	31/12/23	6932.0	LIBRA TROOPER	Field Boots	44.0	3250000.0	Mahal	5.0	Banyak	16250000.0	16250000.0	Laris
6932	31/12/23	6933.0	CENTURION	Field Boots	42.0	3500000.0	Mahal	2.0	Sedikit	7000000.0	7000000.0	Kurang Laris
6933	31/12/23	6934.0	LIBRA TROOPER	Field Boots	42.0	3250000.0	Mahal	5.0	Banyak	16250000.0	16250000.0	Laris
6934	31/12/23	6935.0	MIDTRACK Non	Mid Boots	44.0	2000000.0	Mahal	2.0	Sedikit	4000000.0	4000000.0	Kurang Laris

6935 rows × 12 columns

Gambar 9. Hasil dari pembersihan baris

Pembersihan Atribut

Dalam penelitian ini, penulis memilih beberapa atribut yang digunakan untuk memastikan hasil yang lebih akurat. Atribut-atribut yang ada sebelum dilakukan *cleaning* antara lain: *Date*, *No*, *Product Name*, *Category Name*, *Size*, *Price*, *Price Cat*, *Quantity*, *Quantity Cat*, *Gross Sales*, *Net Income*, dan *Label*. Untuk membersihkan atribut yang tidak terpakai, dilakukan pengkodean sebagai berikut:

```
df.drop(columns=['Date' , 'No' , 'Price Cat' , 'Quantity' , 'Gross Sales' , 'Net Income'], inplace=True)
df
```

Gambar 10. Code untuk menghapus kolom

	Product Name	Category Name	Size	Price	Quantity Cat	Label
0	LUDUS CROSS	Mid Boots	41.0	1350000.0	Sedikit	Kurang Laris
1	LUDUS CROSS MULTICAM BLACK	Field Boots	42.0	1450000.0	Sedikit	Kurang Laris
2	XTRACX Mark III	Field Boots	40.0	3000000.0	Banyak	Laris
3	Nomad Mid	Mid Boots	38.0	649000.0	Tidak Laku	Tidak Laku

Gambar 11. Dataset setelah dilakukan pembersihan

Gambar di atas menunjukkan daftar atribut yang akan digunakan dalam penelitian ini. Atribut yang digunakan meliputi *Product Name*, *Category Name*, *Size*, *Price*, *Quantity Cat*, dan *Label*.

Data Transformation

Setelah data dibersihkan pada tahap *data cleaning*, langkah selanjutnya adalah *data transformation*. *Data transformation* bertujuan untuk mengubah tipe data agar data dapat dibaca dan diproses dalam pengolahan lebih lanjut serta digunakan untuk melakukan prediksi. Berikut adalah kode yang digunakan untuk melakukan transformasi data:

```
df['Product Name'] = enc.fit_transform(df['Product Name'].values)
df['Category Name'] = enc.fit_transform(df['Category Name'].values)
df['Quantity Cat'] = enc.fit_transform(df['Quantity Cat'].values)
df['Label'] = enc.fit_transform(df['Label'].values)
df['Size'] = enc.fit_transform(df['Size'].values)
df['Price '] = enc.fit_transform(df['Price '].values)
```

Gambar 12. Code transformasi data

```
<class 'pandas.core.frame.DataFrame'>
Index: 6935 entries, 0 to 6934
Data columns (total 6 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   Product Name   6935 non-null   int64
 1   Category Name  6935 non-null   int64
 2   Size           6935 non-null   int64
 3   Price          6935 non-null   int64
 4   Quantity Cat   6935 non-null   int64
 5   Label          6935 non-null   int64
dtypes: int64(6)
memory usage: 637.3 KB
```

Gambar 13. Dataset setelah transformasi

Pada gambar 13 ditampilkan hasil dari *data transformation*, yang bertujuan untuk mengubah tipe data menjadi tipe data yang dapat dibaca oleh algoritma *C4.5*, yaitu tipe data *integer* menggunakan *label encoder*.

Algoritma C4.5

Dalam pengujian penelitian ini, data dibagi menjadi data *training* dan data *testing* untuk menghasilkan hasil yang akurat. Pembagian data dilakukan dalam empat metode, yaitu:

1) 10% data untuk *testing* dan 90% data untuk *training*
2) 20% data untuk *testing* dan 80% data untuk *training*
3) 30% data untuk *testing* dan 70% data untuk *training*
4) 40% data untuk *testing* dan 60% data untuk *training*

Berikut adalah hasil dari pengujian *data testing* menggunakan algoritma *C4.5*:

Tingkat akurasi Algoritma C4.5 Decision Tree

	precision	recall	f1-score	support
0	1.00	1.00	1.00	41
1	1.00	1.00	1.00	26
2	1.00	1.00	1.00	42
3	1.00	1.00	1.00	42
4	1.00	1.00	1.00	41
5	1.00	1.00	1.00	80
6	1.00	1.00	1.00	33
7	1.00	1.00	1.00	48
8	1.00	1.00	1.00	33
9	1.00	1.00	1.00	36
10	1.00	1.00	1.00	38
11	1.00	1.00	1.00	37
12	1.00	1.00	1.00	34
13	0.57	0.53	0.55	43
14	0.56	0.62	0.59	32
15	0.43	0.41	0.42	29
16	0.42	0.37	0.39	30
17	0.42	0.48	0.45	29
accuracy			0.88	694
macro avg	0.86	0.86	0.86	694
weighted avg	0.88	0.88	0.88	694

Tingkat akurasi: 88.0403 persen

Gambar 14. Hasil Evaluasi Pengujian Ke-1

Tingkat akurasi Algoritma C4.5 Decision Tree

	precision	recall	f1-score	support
0	1.00	1.00	1.00	67
1	1.00	1.00	1.00	69
2	1.00	1.00	1.00	82
3	1.00	1.00	1.00	71
4	1.00	1.00	1.00	65
5	1.00	1.00	1.00	160
6	1.00	1.00	1.00	65
7	1.00	1.00	1.00	85
8	1.00	1.00	1.00	70
9	1.00	1.00	1.00	71
10	1.00	1.00	1.00	75
11	1.00	1.00	1.00	70
12	1.00	1.00	1.00	76
13	0.52	0.56	0.54	78
14	0.55	0.53	0.54	66
15	0.56	0.53	0.55	75
16	0.45	0.44	0.45	66
17	0.53	0.54	0.53	76
accuracy			0.88	1387
macro avg	0.87	0.87	0.87	1387
weighted avg	0.88	0.88	0.88	1387

Tingkat akurasi: 87.5991 persen

Gambar 15. Hasil Evaluasi Pengujian Ke-2

Tingkat akurasi Algoritma C4.5 Decision Tree

	precision	recall	f1-score	support
0	1.00	1.00	1.00	107
1	1.00	1.00	1.00	105
2	1.00	1.00	1.00	119
3	1.00	1.00	1.00	108
4	1.00	1.00	1.00	106
5	1.00	1.00	1.00	227
6	1.00	1.00	1.00	102
7	1.00	1.00	1.00	117
8	1.00	1.00	1.00	108
9	1.00	1.00	1.00	102
10	1.00	1.00	1.00	110
11	1.00	1.00	1.00	118
12	1.00	1.00	1.00	110
13	0.53	0.59	0.56	110
14	0.56	0.54	0.55	107
15	0.55	0.50	0.52	104
16	0.49	0.75	0.60	105
17	0.57	0.30	0.40	116
accuracy			0.88	2081
macro avg	0.87	0.87	0.87	2081
weighted avg	0.88	0.88	0.88	2081

Tingkat akurasi: 87.8424 persen

Gambar 16. Hasil Evaluasi Pengujian Ke-3

Tingkat akurasi Algoritma C4.5 Decision Tree

	precision	recall	f1-score	support
0	1.00	1.00	1.00	139
1	1.00	1.00	1.00	137
2	1.00	1.00	1.00	145
3	1.00	1.00	1.00	156
4	1.00	1.00	1.00	150
5	1.00	1.00	1.00	304
6	1.00	1.00	1.00	140
7	1.00	1.00	1.00	150
8	1.00	1.00	1.00	149
9	1.00	1.00	1.00	136
10	1.00	1.00	1.00	148
11	1.00	1.00	1.00	152
12	1.00	1.00	1.00	145
13	0.51	0.66	0.57	140
14	0.61	0.38	0.47	144
15	0.48	0.53	0.51	136
16	0.50	0.60	0.55	151
17	0.51	0.41	0.45	152
accuracy			0.87	2774
macro avg	0.87	0.87	0.86	2774
weighted avg	0.88	0.87	0.87	2774

Tingkat akurasi: 87.3468 persen

Gambar 17. Hasil Evaluasi Pengujian Ke-4

Algoritma C4.5

Berdasarkan dari 4 tahap pengujian yang dilakukan dengan nilai data testing dan data training yang berbeda bila dimasukan kedalam table adalah sebagai berikut:

Tabel 1. Evaluasi Hasil Pengujian ke-1

Data Testing		Precision	Recall	F1-score	Accuracy
LUDUS CROSS	10%	01.00	01.00	01.00	88,0403%
LUDUS CROSS MULTICAM BLACK		01.00	01.00	01.00	
XTRACX Mark III		01.00	01.00	01.00	
Nomad Mid		01.00	01.00	01.00	
Nomad Low		01.00	01.00	01.00	
Slickster Sportster		01.00	01.00	01.00	
Slick Xpac		01.00	01.00	01.00	
Slickster Xpac		01.00	01.00	01.00	
Adityalogy PHANTOM VRTX		01.00	01.00	01.00	
XTRACX		01.00	01.00	01.00	
MIDTRACKX		01.00	01.00	01.00	
Slickster		01.00	01.00	01.00	
DIABLO		01.00	01.00	01.00	
COBRA		00.57	00.53	00.55	
Bristol		00.56	0,04305556	00.59	
LIBRA TROOPER		00.43	00.41	00.42	
CENTURION		00.42	00.37	00.39	
MIDTRACK Non		00.42	00.48	00.45	

Tabel 2. Evaluasi Hasil Pengujian ke-2

Data Testing		Precision	Recall	F1-score	Accuracy
LUDUS CROSS	20%	01.00	01.00	01.00	87,5991%
LUDUS CROSS MULTICAM BLACK		01.00	01.00	01.00	
XTRACX Mark III		01.00	01.00	01.00	
Nomad Mid		01.00	01.00	01.00	
Nomad Low		01.00	01.00	01.00	
Slickster Sportster		01.00	01.00	01.00	
Slick Xpac		01.00	01.00	01.00	
Slickster Xpac		01.00	01.00	01.00	
Adityalogy PHANTOM VRTX		01.00	01.00	01.00	
XTRACX		01.00	01.00	01.00	
MIDTRACKX		01.00	01.00	01.00	
Slickster		01.00	01.00	01.00	
DIABLO		01.00	01.00	01.00	
COBRA		00.52	00.56	00.54	
Bristol		00.55	00.53	00.54	
LIBRA TROOPER		00.56	00.53	00.55	
CENTURION		00.45	00.44	00.45	
MIDTRACK Non		00.53	00.54	00.53	

Tabel 3. Evaluasi Hasil Pengujian ke-3

Data Testing		Precision	Recall	F1-score	Accuracy
LUDUS CROSS	30%	01.00	01.00	01.00	87,8424%
LUDUS CROSS MULTICAM BLACK		01.00	01.00	01.00	
XTRACX Mark III		01.00	01.00	01.00	
Nomad Mid		01.00	01.00	01.00	
Nomad Low		01.00	01.00	01.00	
Slickster Sportster		01.00	01.00	01.00	
Slick Xpac		01.00	01.00	01.00	
Slickster Xpac		01.00	01.00	01.00	
Adityalogy PHANTOM VRTX		01.00	01.00	01.00	
XTRACX		01.00	01.00	01.00	
MIDTRACKX		01.00	01.00	01.00	
Slickster		01.00	01.00	01.00	
DIABLO		01.00	01.00	01.00	
COBRA		00.53	00.59	00.56	
Bristol		00.56	00.54	00.55	
LIBRA TROOPER		00.55	00.50	00.52	
CENTURION		00.49	0,05208333	0,04166667	
MIDTRACK Non		00.57	00.30	00.40	

Tabel 4. Evaluasi Hasil Pengujian ke-4

Data Testing		Precision	Recall	F1-score	Accuracy
LUDUS CROSS	40%	01.00	01.00	01.00	87,3468%
LUDUS CROSS MULTICAM BLACK		01.00	01.00	01.00	
XTRACX Mark III		01.00	01.00	01.00	
Nomad Mid		01.00	01.00	01.00	
Nomad Low		01.00	01.00	01.00	
Slickster Sportster		01.00	01.00	01.00	
Slick Xpac		01.00	01.00	01.00	
Slickster Xpac		01.00	01.00	01.00	
Adityalogy PHANTOM VRTX		01.00	01.00	01.00	
XTRACX		01.00	01.00	01.00	
MIDTRACKX		01.00	01.00	01.00	
Slickster		01.00	01.00	01.00	
DIABLO		01.00	01.00	01.00	
COBRA		00.51	0,04583333	00.57	
Bristol		0,04236111	00.38	00.47	
LIBRA TROOPER		00.48	00.53	00.51	
CENTURION		00.50	0,04166667	00.55	
MIDTRACK Non		00.51	00.41	00.45	

Berdasarkan hasil dari table 1 dapat disimpulkan bahwa pengujian dengan data *testing* sebesar 10% menghasilkan nilai yang lebih baik dibandingkan dengan tiga pengujian algoritma *C4.5* lainnya yang terdapat dalam tabel tersebut. Dengan menggunakan data *training* 90% dan data *testing* 10%, diperoleh akurasi sebesar 88.0403%. Oleh karena itu, dapat disimpulkan bahwa pengujian ke-1 memiliki akurasi tertinggi dan dapat dijadikan acuan sebagai data uji.

Tabel 5. Confusion Matrix Pengujian ke-1

Confusion Matrix																	
41	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	26	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	42	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	42	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	41	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	80	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	33	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	48	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	33	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	36	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	38	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	37	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	34	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	23	9	11	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	7	20	5	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	10	7	12	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	11	19
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	15	14

Matriks yang diberikan kemungkinan besar merupakan *confusion matrix*, yang sering digunakan untuk mengevaluasi kinerja model klasifikasi. Matriks ini menunjukkan bahwa ada 18 kelas yang dievaluasi (kelas 0 hingga 17). Baris pada matriks merepresentasikan kelas sebenarnya (*actual class*), sementara kolom merepresentasikan prediksi model (*predicted class*). Elemen pada diagonal utama (misalnya, 41 di [0, 0], 26 di [1, 1], dll.) menunjukkan jumlah sampel yang diklasifikasikan dengan benar untuk masing-masing kelas. Contohnya:

1) Pada kelas 0, terdapat 41 sampel yang diklasifikasikan dengan benar.
2) Pada kelas 1, terdapat 26 sampel yang diklasifikasikan dengan benar, dan seterusnya.

Elemen-elemen di luar diagonal utama menunjukkan jumlah sampel yang salah diklasifikasikan. Contohnya:

1) Pada kelas 13, terdapat 23 sampel yang diklasifikasikan dengan benar (diagonal utama), namun 9 sampel diklasifikasikan sebagai kelas 14, dan 11 sampel diklasifikasikan sebagai kelas 15.

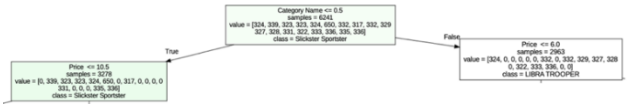
Confusion matrix ini menunjukkan bahwa model bekerja sangat baik untuk sebagian besar kelas karena sebagian besar nilai terkonsentrasi pada diagonal utama. Namun, terdapat beberapa kesalahan klasifikasi pada kelas-kelas tertentu, terutama di kelas 13 hingga 15, di mana distribusi prediksi tersebar ke beberapa kelas lain. Dalam *confusion matrix*, berikut adalah penjelasan tentang *True Positive* (TP), *True*

Negative (TN), *False Positive* (FP), dan *False Negative* (FN) berdasarkan matriks pengujian ke-1:

Tabel 6. Hasil Perhitungan Confusion Matrix

Kelas	TP	FP	FN	TN
0	41	0	0	653
1	26	0	0	668
2	42	0	0	652
3	42	0	0	652
4	41	0	0	653
5	80	0	0	614
6	33	0	0	661
7	48	0	0	646
8	33	0	0	661
9	36	0	0	658
10	38	0	0	656
11	37	0	0	657
12	34	0	0	660
13	23	17	20	634
14	20	16	12	646
15	12	16	17	649
16	11	15	19	649
17	14	19	15	646

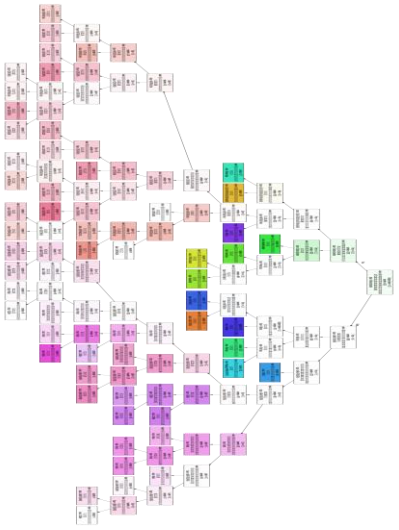
Matriks yang diberikan kemungkinan besar merupakan *confusion matrix*, yang digunakan untuk mengevaluasi kinerja model klasifikasi. Matriks ini menunjukkan adanya 18 kelas yang dievaluasi, dari kelas 0 hingga 17. Baris dalam matriks merepresentasikan kelas sebenarnya (*actual class*), sementara kolom merepresentasikan prediksi model (*predicted class*). Elemen-elemen pada diagonal utama, seperti 41 di [0, 0] dan 26 di [1, 1], menunjukkan jumlah sampel yang diklasifikasikan dengan benar untuk masing-masing kelas. Sebagai contoh, pada kelas 0, terdapat 41 sampel yang diklasifikasikan dengan benar, sementara pada kelas 1 terdapat 26 sampel yang diprediksi dengan benar. Di sisi lain, elemen-elemen di luar diagonal utama menunjukkan jumlah sampel yang salah diklasifikasikan. Sebagai contoh, pada kelas 13, terdapat 23 sampel yang diklasifikasikan dengan benar, tetapi 9 sampel diprediksi sebagai kelas 14 dan 11 sampel diprediksi sebagai kelas 15. *Confusion matrix* ini menunjukkan bahwa model bekerja sangat baik untuk sebagian besar kelas, karena sebagian besar nilai terkonsentrasi pada diagonal utama. Namun, ada beberapa kesalahan klasifikasi pada kelas tertentu, terutama di kelas 13 hingga 15, di mana distribusi prediksi tersebar ke beberapa kelas lain. Penjelasan untuk setiap kelas menunjukkan bahwa untuk kelas 0 hingga 12, model berhasil memprediksi semua sampel dengan sangat baik, dengan TP yang signifikan dan FP serta FN bernilai 0. Ini menunjukkan bahwa model sangat efektif dalam memprediksi kelas-kelas ini, dengan kemampuan tinggi untuk mengenali sampel dari kelas lain sebagai bukan bagian dari kelas tersebut. Namun, untuk kelas 13 hingga 17, terdapat beberapa kesalahan prediksi. Sebagai contoh, pada kelas 13, model memprediksi dengan benar 23 sampel (TP), namun 17 sampel dari kelas lain salah diprediksi sebagai kelas 13 (FP), dan 20 sampel dari kelas 13 salah diprediksi sebagai kelas lain (FN). Meskipun demikian, model masih berhasil memprediksi 634 sampel sebagai bukan bagian dari kelas ini (TN). Hal serupa terjadi pada kelas 14, 15, 16, dan 17, di mana model berhasil memprediksi sebagian besar sampel dengan benar (TP), tetapi juga terdapat sejumlah kesalahan prediksi yang tercermin pada FP dan FN. Misalnya, pada kelas 15, hanya 12 sampel yang diprediksi dengan benar (TP), sementara 16 sampel dari kelas lain salah diprediksi sebagai kelas 15 (FP), dan 17 sampel kelas 15 salah diprediksi sebagai kelas lain (FN). Kemudian, untuk hasil pohon keputusan singkat dari pengujian ke-1, dapat disimpulkan bahwa meskipun ada beberapa kesalahan prediksi pada kelas-kelas tertentu, model ini menunjukkan kinerja yang cukup baik dalam memprediksi sebagian besar kelas, dengan akurasi yang tinggi pada kelas-kelas yang memiliki distribusi prediksi yang lebih terkonsentrasi.



Gambar 6. Gambaran Singkat Decision Tree Pengujian ke-1

Pada gambar di atas, dapat diketahui bahwa pohon keputusan singkat yang diperoleh adalah sebagai berikut: Simpul awal dimulai dengan membagi data berdasarkan atribut *Category Name* dengan kondisi *Category Name <= 0.5*. Data yang memenuhi kondisi ini akan masuk ke cabang True, sementara yang tidak memenuhi akan masuk ke cabang False. Pada simpul awal, terdapat 6.241 sampel yang mencakup semua data pada kondisi awal. Distribusi kelas pada simpul ini menunjukkan jumlah sampel dalam masing-masing kelas, dengan kelas 5 (*Slickster Sportster*) memiliki

jumlah sampel terbanyak, yaitu 650 sampel. Oleh karena itu, prediksi pada simpul ini adalah kelas *Slickster Sportster*. Pada cabang True, data yang memenuhi kondisi *Category Name <= 0.5* dibagi lagi berdasarkan atribut *Price* dengan kondisi *Price <= 10.5*. Cabang ini berisi 3.278 sampel, dengan distribusi kelas menunjukkan kelas 5 (*Slickster Sportster*) sebagai kelas mayoritas. Karena kelas 5 memiliki jumlah sampel terbesar (650), prediksi pada cabang ini tetap *Slickster Sportster*. Sementara itu, pada cabang False, data yang tidak memenuhi kondisi *Category Name <= 0.5* dibagi lagi berdasarkan atribut *Price* dengan kondisi *Price > 10.5*. Cabang ini berisi 2.963 sampel, dan distribusi kelas menunjukkan bahwa kelas 6 (*LIBRA TROOPER*) memiliki jumlah sampel terbanyak (332). Oleh karena itu, prediksi pada cabang ini adalah *LIBRA TROOPER*. Secara keseluruhan, hasil dari pohon keputusan menunjukkan bahwa *Category Name* adalah atribut paling penting untuk membagi data pada simpul akar. Mayoritas data pada simpul ini diklasifikasikan ke dalam kelas *Slickster Sportster*. Cabang True, dengan kondisi *Category Name <= 0.5* dan *Price <= 10.5*, memprediksi kelas mayoritas sebagai *Slickster Sportster*, sementara cabang False, dengan kondisi *Category Name <= 0.5* dan *Price > 10.5*, menghasilkan prediksi *LIBRA TROOPER* berdasarkan kelas mayoritas di cabang tersebut. Berikut adalah gambar pohon keputusan secara keseluruhan yang diperoleh dari penelitian ini:



Gambar 7. Hasil Pohon Keputusan / Decision Tress

Pembahasan

Hasil pohon keputusan yang dihasilkan dari algoritma *C4.5* menunjukkan bahwa atribut *Category Name* dan *Price* memegang peranan penting dalam memprediksi permintaan sepatu di Toko Parabellum. Sejalan dengan temuan Pritalia (2021), yang menyatakan bahwa algoritma *C4.5* efektif untuk menentukan ketersediaan barang di e-commerce, penelitian ini juga menunjukkan bagaimana algoritma ini dapat digunakan untuk memprediksi stok sepatu yang sesuai dengan permintaan pelanggan berdasarkan kategori dan harga produk. Pembagian data berdasarkan atribut *Category Name* pada simpul awal berhasil memisahkan produk yang lebih terjangkau dan lebih mahal, yang kemudian diprediksi sebagai *Slickster Sportster* dan *LIBRA TROOPER* pada cabang-cabang selanjutnya. Selanjutnya, pada cabang True, data yang memenuhi kondisi *Category Name <= 0.5* dan *Price <= 10.5* menghasilkan prediksi yang lebih kuat untuk kelas *Slickster Sportster*. Hal ini konsisten dengan penelitian sebelumnya yang dilakukan oleh Ferdiansyah dan Goeirmanto (2020), yang menggunakan algoritma *C4.5* untuk memprediksi loyalitas karyawan terhadap perusahaan berdasarkan berbagai faktor, termasuk harga dan kategori, yang relevansinya sama dalam prediksi permintaan barang di e-commerce. Di sisi lain, pada cabang False, di mana harga lebih tinggi, model memprediksi kelas *LIBRA TROOPER* sebagai kelas mayoritas, yang menunjukkan bahwa harga juga menjadi faktor utama dalam menentukan preferensi pelanggan terhadap produk.

Hasil distribusi sampel yang menunjukkan kelas mayoritas di setiap cabang juga sejalan dengan temuan yang dilaporkan oleh Sukri dan Handrianto (2024), yang menjelaskan bagaimana pohon keputusan dapat secara efektif mengidentifikasi dan mengklasifikasikan data berdasarkan atribut-atribut penting yang mempengaruhi keputusan pembelian atau keputusan lainnya. Sebagai contoh, pada cabang False, kelas dengan harga lebih tinggi, yaitu *LIBRA TROOPER*, memiliki sampel terbanyak di antara kelas lainnya, menunjukkan pengaruh kuat dari harga terhadap prediksi permintaan produk. Selain itu, dalam penelitian ini, atribut *Price* digunakan untuk lebih memperjelas prediksi berdasarkan kondisi pasar, yang juga ditekankan dalam penelitian oleh Azwanti dan Yulia (2018) yang menganalisis penggunaan algoritma

C4.5 untuk memprediksi penjualan motor dan penggunaan listrik rumah tangga, keduanya melibatkan analisis harga sebagai faktor penting dalam prediksi. Penelitian ini juga membuktikan bahwa model yang dihasilkan memiliki kemampuan untuk memprediksi kelas yang lebih banyak dengan menggunakan atribut yang relevan, yang secara langsung berhubungan dengan preferensi konsumen. Secara keseluruhan, hasil dari penelitian ini mendukung efektivitas penggunaan algoritma *C4.5* dalam memprediksi stok barang yang sesuai dengan permintaan pelanggan, sebagaimana dijelaskan oleh Pritalia (2021) dalam penelitian *e-commerce*. Algoritma ini dapat mengklasifikasikan data berdasarkan kategori dan harga, yang sangat berguna untuk pengambilan keputusan yang lebih baik dalam manajemen stok barang, terutama untuk bisnis yang bergerak di bidang retail seperti Toko Parabellum. Dengan pemodelan yang lebih akurat, pengelola bisnis dapat mengoptimalkan strategi pemasaran dan pengelolaan persediaan untuk meningkatkan kepuasan pelanggan.

4. Kesimpulan

Berdasarkan hasil penelitian dan pembahasan yang telah dilakukan, implementasi Algoritma C4.5 berhasil diterapkan untuk memprediksi penjualan sepatu boots terlaris di Toko Parabellum. Algoritma ini menunjukkan performa yang baik dalam klasifikasi data penjualan, dengan akurasi pengujian terbaik mencapai 88,04% pada skenario data training 90% dan data testing 10%. Atribut yang paling berpengaruh terhadap prediksi penjualan adalah *Category Name*, *Quantity Cat* dan *Price* berdasarkan hasil pohon keputusan yang dihasilkan. Atribut ini memberikan pembagian data yang signifikan dalam membentuk keputusan. Algoritma C4.5 memberikan hasil analisis yang dapat diandalkan untuk mendukung pengambilan keputusan bisnis, terutama dalam menentukan kebutuhan stok sepatu yang populer. Penelitian ini juga membuktikan bahwa pendekatan data mining berbasis algoritma pohon keputusan dapat diadaptasi pada konteks bisnis lain untuk meningkatkan efisiensi operasional dan kepuasan pelanggan.

5. Daftar Pustaka

Abdillah, M. A., Setyanto, A., & Sudarmawan, S. (2020). Implementasi Decision Tree Algoritma C4. 5 Untuk Memprediksi Kesuksesan Pendidikan Karakter. *Respati*, *15*(2), 59-69.

AMIK, F. (2017). Algoritma C4. 5 Dalam Data Mining Untuk Menentukan Klasifikasi Kelulusan Calon Mahasiswa Baru (Sudi Kasus: AMIK-DP). *JURNAL ILMU PENGETAHUAN & SISTEM INFORMASI (JIPSI)*, *4*(November), 12-33.

Azwanti, N. (2018). Analisa Algoritma C4. 5 Untuk Memprediksi Penjualan Motor Pada Pt. Capella Dinamik Nusantara Cabang Muka Kuning. *Inform. Mulawarman J. Ilm. Ilmu Komput*, *13*(1), 33.

Ferdiansyah, B., & Goeirmanto, L. (2020). Prediksi Loyalitas dalam Keterikatan Karyawan terhadap Perusahaan Menggunakan Algoritma C4. 5*(Studi Kasus PT. XYZ). *JUSTIN (Jurnal Sistem dan Teknologi Informasi)*, *8*(1), 87-97. https://doi.org/10.26418/justin.v8i1.33606.

Pambudi, R. H. (2017). *Penerapan Algoritma C4. 5 Untuk Memprediksi Nilai Kelulusan Siswa Sekolah Menengah Berdasarkan Faktor Eksternal* (Doctoral dissertation, Universitas Brawijaya).

Prayogi, A., & Kurniawan, M. A. (2024). Pendekatan Kualitatif dan Kuantitatif: Suatu Telaah. *Complex: Jurnal Multidisiplin Ilmu Nasional*, *1*(2), 30-37.

Pritalia, G. L. (2018). Penerapan Algoritma C4. 5 untuk Penentuan Ketersediaan Barang E-commerce. *Indonesian Journal of Information Systems*, *1*(1), 47-56. https://doi.org/10.24002/ijis.v1i1.1727.

Putri, R. P. S., & Waspada, I. (2018). *Penerapan Algoritma C4. 5 Pada Aplikasi Prediksi Kelulusan Mahasiswa Prodi Teknik Informatika* (Doctoral dissertation, Universitas Diponegoro).

Rambe, D. I., Nasution, M., & Ah, R. M. (2024). Penerapan Metode Algoritma C4. 5 Untuk Memprediksi Loyalitas Karyawan Pada PT. Tolan Tiga Indonesia Perlabian Estate. *INFORMATIKA*, *12*(2), 132-138. https://doi.org/10.36987/informatika.v12i2.5646.

Sagala, N., & Tampubolon, H. (2018). Komparasi Kinerja Algoritma Data Mining pada Dataset Konsumsi Alkohol Siswa. *Khazanah Informatika: Jurnal Ilmu Komputer Dan Informatika*, *4*(2), 98-103. https://doi.org/10.23917/khif.v4i2.7061.

Salsabila, C. P., & Wijayanto, A. (2023). Implementasi Algoritma C. 45 Dalam Memprediksi Kualitas Aset Kendaraan Kantor. *Journal of Information System Research (JOSH)*, *4*(3), 840-846.

Sukri, M. H., & Handrianto, Y. (2024). Penerapan Algoritma C4. 5 Dalam Menentukan Prediksi Prestasi Siswa Pada SMPN 51 Jakarta. *Informatics and Computer Engineering Journal*, *4*(1), 11-24. https://doi.org/10.31294/icej.v4i1.2582.

Wirasena, M. R., & Warmansyah, J. (2024). Penerapan Algoritma C4. 5 Untuk Prediksi Kelayakan Pengajuan Kartu Kredit Visa Bagi Nasabah. *TeknoIS: Jurnal Ilmiah Teknologi Informasi dan Sains*, *14*(2), 296-302.

Yulia, Y., & Azwanti, N. (2018). Penerapan Algoritma C4. 5 Untuk Memprediksi Besarnya Penggunaan Listrik Rumah Tangga di Kota Batam. *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, *2*(2), 584-590. https://doi.org/10.29207/resti.v2i2.503.