

VOYAGE: SISTEM RAMALAN PERMINTAAN PELANGGAN

DANISH DEAN BIN NOOR SALLAHUDDIN

TS. DR. NOR SAMSAH SANI

*Fakulti Teknologi & Sains Maklumat, Universiti Kebangsaan Malaysia, 43600 UKM Bangi,
Selangor Darul Ehsan, Malaysia*

ABSTRAK

Dalam era digital dan teknologi yang semakin maju, industri perniagaan menghadapi cabaran besar dalam memastikan pengurusan inventori dan menguruskan permintaan pengguna yang semakin dinamik. Berdasarkan laporan, ketidakseimbangan dalam pengurusan inventori dapat mengakibatkan lebih atau kekurangan stok. Lebih stok meningkatkan kos penyimpanan dan menurunkan nilai produk terutamanya bagi barangan yang mempunyai jangka hayat yang pendek manakala kekurangan stok pula boleh mengakibatkan kekecewaan pelanggan dan kehilangan jualan. Dalam menghadapi cabaran ini, projek ini bertujuan membangunkan Sistem Ramalan Permintaan Pelanggan yang berasaskan pembelajaran mesin untuk menghasilkan ramalan permintaan yang lebih tepat sekaligus membantu syarikat dalam mengoptimumkan strategi pengurusan inventori. Kunci utama bagi ramalan permintaan yang tepat adalah demi mengoptimumkan keuntungan dan mengelakkan kos yang tidak seharusnya. Sistem ini memanfaatkan teknik pembelajaran mesin untuk menganalisis data jualan lampau dengan mengesan pola-pola permintaan bermusim dan mengenal pasti faktor luaran yang mempengaruhi jualan seperti perubahan ekonomi, musim perayaan dan promosi. Dengan menilai faktor sedemikian, sistem ramalan yang dibangunkan ini dapat menyediakan anggaran permintaan yang lebih tepat bagi setiap produk. Data jualan lampau berfungsi sebagai input utama yang membolehkan sistem mempelajari pola jualan sepanjang tahun. Sistem ini turut mempertimbangkan faktor seperti promosi, diskaun dan pelancaran produk terbaru yang mempengaruhi trend pembelian.

Kata kunci: Ramalan Permintaan Pelanggan, pengurusan inventori, data jualan lampau

PENGENALAN

Dalam era digital dan dengan pertambahan jumlah data yang pesat, sektor perniagaan dan perdagangan menghadapi pelbagai cabaran dalam mengurus permintaan pengguna yang dinamik serta memastikan pengurusan inventori yang cekap. Pengurusan inventori yang tidak tepat boleh membawa kepada masalah ketidakseimbangan stok, di mana lebih stok akan meningkatkan kos penyimpanan, manakala kekurangan stok pula akan menyebabkan peluang jualan terlepas dan kekecewaan pelanggan. Kebimbangan lain timbul kerana ketiadaan laporan industri yang boleh memberikan panduan dalam meramalkan permintaan pelanggan.

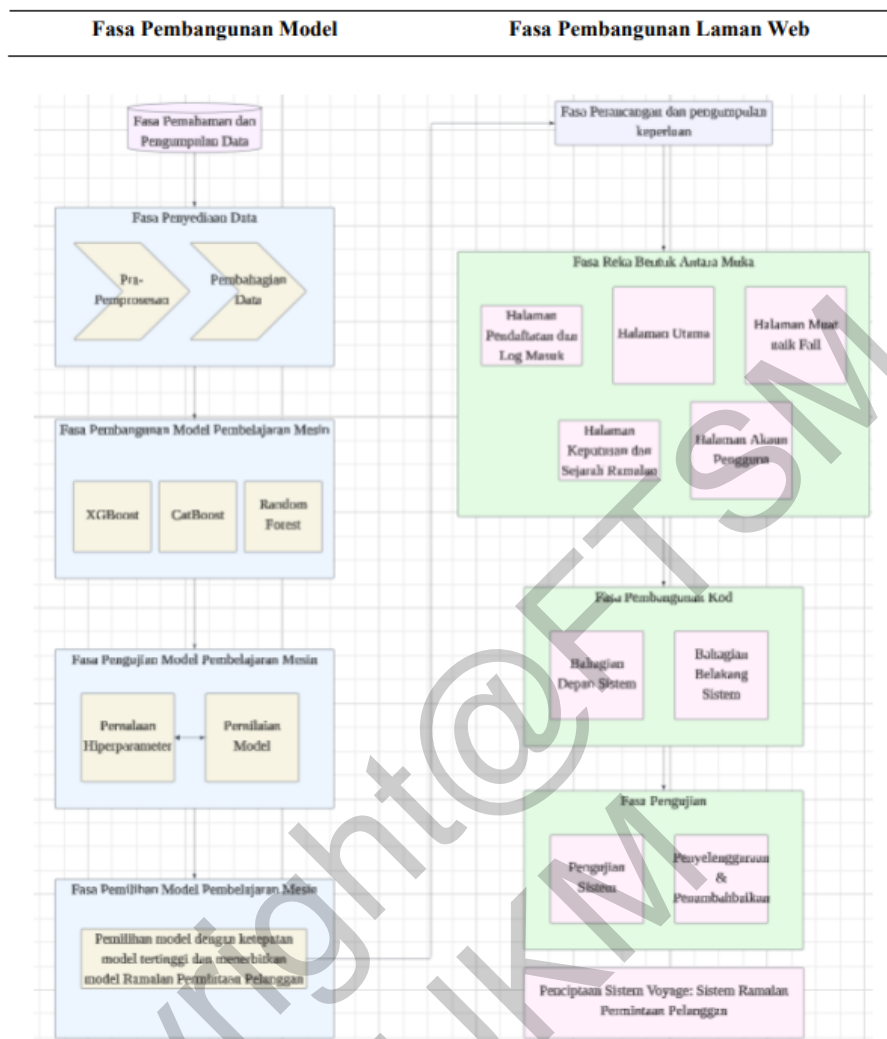
Ini disebabkan oleh ketidakmampuan kebanyakan syarikat untuk meramalkan permintaan dengan tepat, yang sering membawa kepada penurunan keuntungan kerana stok tidak dapat disesuaikan dengan perubahan permintaan pasaran. Model regresi logistik yang digunakan menunjukkan bahawa pembelajaran kecerdasan buatan sangat sesuai dan berkesan untuk menganalisis data POS sesebuah syarikat (Kaneko & Yada 2016).

Sektor perniagaan dan perdagangan menghadapi pelbagai cabaran dalam mengurus permintaan pengguna yang dinamik serta memastikan pengurusan inventori yang cekap. Pandemik Covid-19 telah menjadikan pengurusan inventori dalam sesebuah syarikat semakin membimbangkan. (Idris et al. 2023). Pengurusan inventori yang tidak tepat boleh membawa kepada masalah ketidakseimbangan stok di mana lebih stok akan meningkatkan kos penyimpanan dan kekurangan stok pula akan mengakibatkan peluang jualan yang terlepas serta menyebabkan kekecewaan pelanggan. Menjadi satu lagi kebimbangan adalah kerana tiada laporan industri yang dapat dikesan tentang ramalan permintaan pelanggan. Ini mungkin juga kerana privasi 2 syarikat dan mungkin juga kerana kebanyakan syarikat tidak mampu meramalkan permintaan dengan baik dan sering berhadapan dengan penurunan keuntungan. Hal ini adalah kerana mereka tidak dapat menyesuaikan stok dengan permintaan pasaran.

Bagi menangani cabaran ini, projek ini bertujuan untuk membangunkan sebuah Sistem Ramalan Permintaan Pelanggan yang berasaskan pembelajaran mesin yang akan memberikan anggaran permintaan yang lebih tepat berdasarkan pola jualan masa lampau. Pembelajaran mesin merujuk kepada kaedah analitik dalam bidang kecerdasan buatan yang membolehkan sistem untuk belajar daripada data tanpa perlu di program secara eksplisit untuk setiap tugas. Dengan hanya menganalisis corak data tersebut, pembelajaran mesin dapat membolehkan sistem membuat ramalan atau keputusan berdasarkan data yang telah dipelajari (Agil et al. 2024).

Dalam projek ini, teknik pembelajaran mesin akan digunakan untuk menganalisis sejarah data jualan, mengenal pasti pola permintaan bermusim dan mengesan faktor luaran seperti promosi, perayaan serta keadaan ekonomi yang mempengaruhi jualan (Othman et al. 2020). Dengan inisiatif ini, sistem ramalan permintaan pelanggan yang dibangunkan dapat membantu syarikat dalam membuat keputusan yang lebih tepat dalam pengurusan inventori dan merancang strategi pemasaran yang efektif. Melalui sistem ini, syarikat dapat mengoptimumkan pengurusan inventori dan mengurangkan kos penyimpanan tetapi juga dapat meningkatkan kepuasan pelanggan dengan memastikan produk sentiasa tersedia sesuai dengan permintaan pasaran.

METODOLOGI KAJIAN



Rajah 1 Carta Alir Pembangunan Model dan Sistem

Rajah 1 menunjukkan carta alir bagi fasa pembangunan model dan sistem bagi laman web ramalan permintaan pelanggan, Voyage. Fasa pembangunan model akan mengaplikasi bahasa pengaturcaraan Python dan Jupyter Notebook sebagai platform untuk melakukan pengaturcaraan membina model. Model yang digunakan bagi tujuan perbandingan adalah model XGBoost, CatBoost dan RandomForest. Di fasa penilaian model, ketiga-tiga model akan menjalani ujian t-test untuk menentukan model yang terbaik. Model tersebut kemudian akan diintegrasikan ke dalam sistem web yang dibina 67 menggunakan Visual Studio Code menggunakan bahasa pengaturcaraan HTML, Javascript dan CSS. Selepas model diintegrasikan ke dalam sistem web, keseluruhan sistem akan diuji untuk memastikan objektif kajian dan keperluan pengguna tercapai dan sistem dapat digunakan dengan baik.

Fasa Pembangunan Model

a. Fasa Pemahaman dan Pengumpulan data

Fasa pertama di dalam fasa pembangunan model ini, fasa pemahaman dan

pengumpulan data memainkan peranan penting dalam memahami keperluan model dan situasi sebenar syarikat perniagaan untuk membuat ramalan permintaan pelanggan. Data dikumpulkan dari pelbagai sumber seperti Kaggle, Google dan syarikat yang berada Malaysia mahupun luar negara dengan keizinan menggunakan data syarikat tersebut. Ciri seperti tarikh, unit yang terjual, harga, diskaun, jualan unit tersebut dan lain-lain sangat dititikberatkan untuk ke fasa seterusnya. Contoh data yang dikumpulkan adalah seperti berikut.

Date	Store ID	Product ID	Category	Region	Inventory Level	Units Sold	Units Ordered	Demand Forecast Price	Discount	Weather Condition	Holiday/Promotion	Competitor Price	Seasonality
1/1/2022	S001	P0001	Groceries	North	231	127	55	135.47	33.5	20 Rainy	0	29.69	Autumn
1/1/2022	S001	P0002	Toys	South	204	150	66	144.04	63.01	20 Sunny	0	66.16	Autumn
1/1/2022	S001	P0003	Toys	West	102	65	51	74.02	27.99	10 Sunny	1	31.32	Summer
1/1/2022	S001	P0004	Toys	North	469	61	164	62.18	32.72	10 Cloudy	1	34.74	Autumn
1/1/2022	S001	P0005	Electronics	East	166	14	135	9.26	73.64	0 Sunny	0	68.95	Summer
1/1/2022	S001	P0006	Groceries	South	138	128	102	139.82	76.83	10 Sunny	1	79.35	Winter
1/1/2022	S001	P0007	Furniture	East	359	97	167	108.92	34.16	10 Rainy	1	36.55	Winter
1/1/2022	S001	P0008	Clothing	North	380	312	54	329.73	97.99	5 Cloudy	0	100.09	Spring
1/1/2022	S001	P0009	Electronics	West	183	175	135	174.15	20.74	10 Cloudy	0	17.66	Autumn
1/1/2022	S001	P0010	Toys	South	108	28	196	24.47	59.99	0 Rainy	1	61.21	Winter
1/1/2022	S001	P0011	Furniture	South	258	150	153	152.74	58.53	10 Sunny	1	61.42	Spring
1/1/2022	S001	P0012	Clothing	West	66	24	70	26.75	58.25	20 Snowy	0	62.21	Spring
1/1/2022	S001	P0013	Toys	South	96	42	85	41.46	43.6	0 Cloudy	0	46.31	Spring
1/1/2022	S001	P0014	Clothing	West	193	12	187	6.8	78.11	0 Sunny	0	80.06	Spring
1/1/2022	S001	P0015	Clothing	North	379	369	154	363.46	92.99	15 Snowy	0	95.8	Winter
1/1/2022	S001	P0016	Electronics	North	363	255	69	255.74	21.9	5 Cloudy	1	20.27	Autumn
1/1/2022	S001	P0017	Toys	West	318	246	177	255.37	21.07	20 Sunny	0	16.49	Winter
1/1/2022	S001	P0018	Clothing	South	241	151	47	147.27	19.57	5 Cloudy	0	23.13	Autumn
1/1/2022	S001	P0019	Clothing	East	352	257	186	267.38	73.28	10 Cloudy	0	77.26	Winter
1/1/2022	S001	P0020	Toys	East	274	99	166	115.23	30.24	5 Cloudy	0	27.1	Spring
1/1/2022	S002	P0001	Groceries	South	343	104	144	112.55	32.8	20 Sunny	1	30.78	Spring
1/1/2022	S002	P0002	Clothing	West	373	141	151	135.62	46.16	10 Rainy	1	45.21	Winter
1/1/2022	S002	P0003	Furniture	East	445	182	119	176.31	20.46	20 Snowy	1	22	Spring
1/1/2022	S002	P0004	Toys	West	191	63	115	54.57	26.19	0 Cloudy	0	28.06	Autumn
1/1/2022	S002	P0005	Toys	West	281	156	25	150.9	83.29	20 Rainy	0	80.64	Autumn
1/1/2022	S002	P0006	Furniture	North	492	250	168	242.57	85.59	20 Cloudy	0	86.82	Winter

Rajah 2 Dataset yang dikumpulkan dari Kaggle

b. Fasa Penyediaan Data

Fasa penyediaan data adalah langkah yang kritikal dalam proses pembangunan model pembelajaran mesin yang melibatkan pra pemprosesan dan pembahagian data. Di peringkat ini data mentah yang biasanya tidak mempunyai nilai, data yang tidak 68 konsisten dan duplikasi data akan dibersihkan agar data lebih berkualiti. Kemudian data dibahagikan untuk tujuan latihan dan ujian.

i. Pra-pemprosesan

Proses Pra-pemprosesan data ini dimulakan dengan membuang baris yang mengandungi nilai hilang dalam nilai hilang dalam kolom rangka data. Proses ini bertujuan untuk memastikan data yang digunakan ada memastikan data yang digunakan adalah lengkap dan tidak mengandungi kekosongan yang boleh menjejaskan analisis atau model pembelajaran mesin. Selepas itu, nilai hilang bagi kolom kategori tertentu seperti 'Store ID', 'Product ID', 'Category', 'Region', 'Weather Condition', 'Holiday/Promotion', dan 'Seasonality', diisi dengan nilai mod iaitu nilai yang paling kerap muncul dalam kolom tersebut. Langkah ini membantu mengekalkan baris data tanpa kehilangan maklumat penting dan memastikan nilai kategori yang hilang digantikan dengan nilai yang logik dan biasa. Seterusnya, semua nilai dalam kolom jenis objek ditukar menjadi huruf kecil kecuali kolom 'Store ID' dan 'Product ID'. Ini dilakukan untuk menyamakan format teks supaya tiada perbezaan

disebabkan oleh huruf besar atau kecil, contohnya "Holiday" dan "holiday" dianggap sama. Akhir sekali, mesej pengesahan dipaparkan untuk menunjukkan bahawa proses pra pemprosesan telah selesai, bersama dengan beberapa baris pertama DataFrame yang telah diproses. Proses ini memastikan data yang digunakan adalah bersih, lengkap dan konsisten untuk mendapatkan hasil analisis atau model yang tepat dan berkesan.

```
# Drop rows with missing values
df = df.dropna()

# --- Handle missing categorical values (fill with mode) ---
categorical_cols = ['Store ID', 'Product ID', 'Category', 'Region',
                    'Weather Condition', 'Holiday/Promotion', 'Seasonality']
for col in categorical_cols:
    if col in df.columns:
        df[col] = df[col].fillna(df[col].mode()[0])

# Lowercase all object-type columns except 'Store ID' and 'Product ID'
for col in df.select_dtypes(include='object').columns:
    if col not in ['Store ID', 'Product ID']:
        df[col] = df[col].str.lower()

print("✅ Preprocessing complete. Here's the final DataFrame:")
df.head()
```

Rajah 3 Kod Proses Pra-Pemprosesan

Seterusnya menukarkan kolom 'Date' kepada format tarikh supaya mudah untuk melakukan operasi berasaskan masa seperti mengenal pasti hari hujung minggu. Beberapa ciri baharu dibina melalui proses *feature engineering*, antaranya adalah harga selepas diskaun (*price_after_discount*) yang dikira dengan menolak nilai diskaun daripada harga asal, serta penambahan ciri Jualan (*Sales*) dikira dengan mendarabkan harga selepas diskaun dan unit yang terjual (*Units Sold*). Selain itu, satu ciri binari ditambah untuk menandakan sama ada transaksi berlaku pada hujung minggu. Kemudian, hanya kolom numerik yang relevan dipilih untuk mengkaji hubungan antara satu sama lain melalui matriks korelasi. Matriks ini menunjukkan kekuatan dan arah hubungan linear antara ciri-ciri tersebut. Matriks korelasi ini divisualisasikan dalam bentuk heatmap menggunakan Seaborn, di mana warna dan nilai korelasi dipaparkan supaya dapat mengenal pasti ciri-ciri yang mempunyai hubungan kuat atau lemah antara satu sama lain. Akhir sekali, matriks korelasi tersebut juga dicetak dalam bentuk jadual supaya dapat dirujuk dengan mudah.

```

import seaborn as sns
import matplotlib.pyplot as plt
# Convert date
df['Date'] = pd.to_datetime(df['Date'])

# Feature engineering
df['price_after_discount'] = df['Price'] - df['Discount']
df['price_vs_competitor'] = df['Price'] - df['Competitor Pricing'] / df['Competitor Pricing'].replace(0, 1e-10)
df['is_weekend'] = df['Date'].dt.dayofweek >= 5
df['Sales'] = df['price_after_discount'] * df['Units Sold']

# Select numerical columns for correlation matrix
numerical_cols = [
    'Inventory Level', 'Units Sold', 'Units Ordered', 'Demand Forecast',
    'Price', 'Discount', 'Competitor Pricing', 'price_after_discount',
    'price_vs_competitor', 'Sales'
]

# Ensure all numerical columns exist in the dataset
numerical_cols = [col for col in numerical_cols if col in df.columns]

# Compute correlation matrix
correlation_matrix = df[numerical_cols].corr()

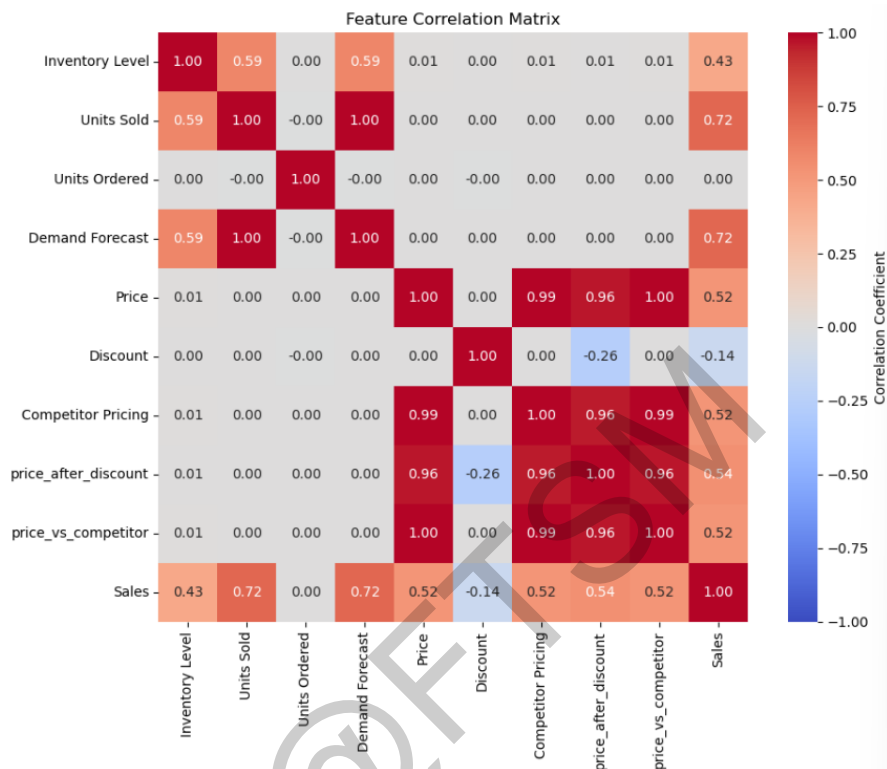
# Visualize correlation matrix as a heatmap
plt.figure(figsize=(10, 8))
sns.heatmap(
    correlation_matrix,
    annot=True, # Show correlation values
    cmap='coolwarm', # Color scheme
    vmin=-1, vmax=1, # Scale from -1 to 1
    center=0, # Center the colormap at 0
    fmt='.2f', # Format numbers to 2 decimal places
    square=True, # Make cells square
    cbar_kws={'label': 'Correlation Coefficient'}
)
plt.title('Feature Correlation Matrix')
plt.tight_layout()
plt.show()

# Print correlation matrix for reference
print("\n Feature Correlation Matrix:")
print(correlation_matrix.round(2))

```

Rajah 4 Kod Kejuruteraan Ciri dan Kolerasi Ciri

Penghasilan Kolerasi Ciri penting kerana ia membantu untuk mengenal pasti hubungan linear antara pembolehubah dalam dataset. Bagi rajah 4.5, ia memaparkan kolerasi antara semua ciri dalam dataset. Warna merah pekat mewakili korelasi positif yang sangat tinggi (hampir +1), manakala warna biru menunjukkan korelasi negatif. *Units Sold* dan *Demand Forecast* menunjukkan korelasi positif yang sangat kuat dengan Sales (0.72), bermaksud semakin tinggi ramalan permintaan dan unit dijual, semakin tinggi jualan. Ciri *Price*, *Competitor Pricing*, dan *price_after_discount* mempunyai korelasi sederhana positif dengan Sales (sekitar 0.52–0.54), menunjukkan harga dan harga pesaing turut berkait dengan prestasi jualan. Sebaliknya, *Discount* menunjukkan sedikit korelasi negatif (-0.14) terhadap Sales, mencadangkan diskaun lebih besar tidak semestinya meningkatkan jualan.

Rajah 5 Rajah *Heatmap* Kolerasi Ciri

ii. Pembahagian data

Sebelum proses pembahagian data berlaku, data asal perlu diproses dengan membuang kolum yang menyebabkan kebocoran seperti 'Demand Forecast', 'Product ID' dan 'Store ID', seterusnya ciri kategori ditukar kepada bentuk numerik menggunakan kaedah one-hot encoding dan sebarang nilai hilang digantikan dengan sifar. Kolum tarikh juga dibuang jika masih wujud. Selepas itu, data dibahagikan kepada dua bahagian utama iaitu set latihan (training set) dan set ujian (holdout/test set) menggunakan fungsi 'train_test_split', di mana 80% data digunakan untuk latihan dan 20% lagi disimpan untuk penilaian akhir model.

```
# Drop Leakage-prone columns if any
leakage_cols = ['Demand Forecast', 'Product ID', 'Store ID']
X = df.drop(columns=leakage_cols)
y = df['Demand Forecast']

# One-hot encode categorical features
X = pd.get_dummies(X, drop_first=True)

# Ensure no missing values after encoding
X = X.fillna(0)

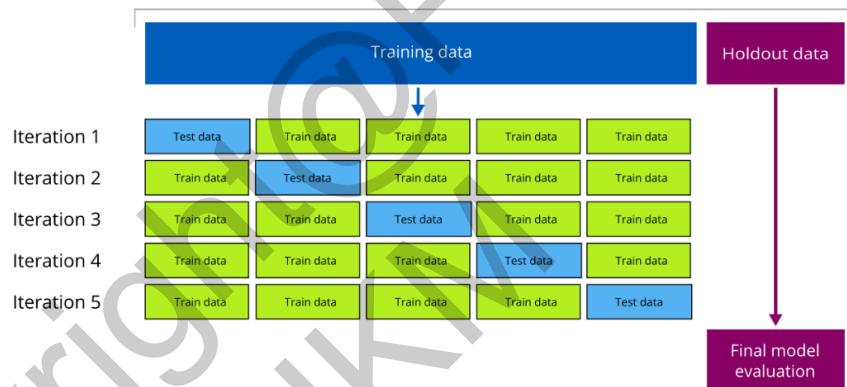
# Drop datetime column
if 'Date' in X.columns:
    X = X.drop(columns=['Date'])

# Train-test split
X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.2, random_state=42
)
```

Rajah 6 Kod Proses *train_test_split*

Rajah 6 untuk proses `train_test_split`, data dibahagikan kepada dua bahagian utama iaitu data latihan (Training data) dan data pengujian (Test Data). Training data digunakan untuk membina dan menala model manakala Test Data disimpan untuk penilaian akhir setelah semua proses selesai. training data ini melalui proses *k-fold cross-validation*. Training data dibahagikan kepada lima bahagian yang sama besar untuk 5-fold cross-validation, $cv=5$ atau $k=5$. Pada setiap iterasi, satu bahagian dipilih sebagai test data manakala empat bahagian selebihnya digunakan sebagai train data. Proses ini diulang sebanyak lima kali sehingga setiap bahagian pernah menjadi test data sekali. Dengan cara ini, model diuji pada pelbagai subset data latihan untuk memastikan prestasinya konsisten dan tidak terlalu bergantung kepada satu subset sahaja.

Default holdout and cross-validation



Rajah 7 Proses *k-Fold Cross-Validation*

Seterusnya, proses penalaan hiperparameter untuk ketiga-tiga model iaitu model XGBoost, CatBoost dan RandomForest menggunakan kaedah *RandomizedSearchCV* yang merupakan satu kaedah untuk mencari kombinasi hiperparameter terbaik bagi model. `param_grid` mendefinisikan ruang carian untuk hiperparameter seperti `n_estimators` (bilangan pohon), `learning_rate` (kadar pembelajaran), dan `max_depth` (kedalaman maksimum pohon), di mana *RandomizedSearchCV* akan secara rawak memilih 10 kombinasi (ditentukan oleh `n_iter=10`) daripada ruang carian ini. Proses pencarian ini akan dinilai menggunakan metrik `neg_root_mean_squared_error` (negatif *RMSE*) dan paling penting, ia melibatkan *5-fold cross-validation* (ditentukan oleh `cv=5`) yang beroperasi pada training data sahaja seperti yang digambarkan dalam Rajah 4.7. Bagi setiap 10 kombinasi hiperparameter yang diuji, data latihan akan dibahagikan kepada lima bahagian (*folds*) dan model dilatih serta dinilai lima kali, dengan setiap bahagian bertindak sebagai data ujian secara bergilir untuk memastikan pemilihan hiperparameter yang stabil dan

mengelakkan overfitting sebelum model akhir diuji pada *holdout data* yang berasingan.

```
catboost = CatBoostRegressor(
    random_state=42,
    verbose=0
)

random_search = RandomizedSearchCV(
    estimator=catboost,
    param_distributions=param_grid,
    n_iter=10,
    scoring='neg_root_mean_squared_error',
    cv=5,
    verbose=1,
    random_state=42,
    n_jobs=-1
)
```

Rajah 8 Penalaan Hiperparameter dan Penepatan *K-Fold*

c. Fasa Pembangunan Model Pembelajaran Mesin

Fasa pembangunan model bagi sistem ramalan permintaan pelanggan dalam sistem Voyage melibatkan pendekatan berstruktur untuk membina, melatih, menala, dan menilai model berdasarkan data. Setelah penalaan hiperparameter dilakukan untuk ketiga-tiga model iaitu CatBoost, XGBoost, dan Random Forest, model terbaik dikenal pasti melalui pencarian rawak (*random_search.best_estimator_*). Model terbaik ini digunakan untuk meramalkan nilai sasaran bagi set data latihan (*X_train*) dan set data ujian (*X_test*), menghasilkan ramalan *y_train_pred* dan *y_test_pred*. Prestasi model dinilai menggunakan metrik seperti *Root Mean Squared Error* (RMSE) yang mengukur saiz purata ralat, *Mean Absolute Error* (MAE) yang mengukur purata magnitud ralat tanpa mengira arah, dan *R-squared* (R^2) yang menunjukkan proporsi varians dalam pemboleh ubah bersandar yang dapat dijelaskan oleh pemboleh ubah bebas. Selain itu, *Mean Absolute Percentage Error* (MAPE) dikira khusus untuk set ujian bagi memberikan peratusan ralat purata yang sangat berguna dalam konteks perniagaan. Analisis kepentingan ciri juga dijalankan untuk mengenal pasti ciri-ciri input paling berpengaruh dalam ramalan, diikuti dengan visualisasi SHAP (*SHapley Additive exPlanations*) yang memberikan pemahaman lebih mendalam tentang bagaimana setiap ciri mempengaruhi keputusan model.

```

xgb = XGBRegressor(random_state=42)

random_search = RandomizedSearchCV(
    estimator=xgb,
    param_distributions=param_grid,
    n_iter=10,
    scoring='neg_root_mean_squared_error',
    cv=5,
    verbose=1,
    random_state=42,
    n_jobs=-1
)

# Fit model
random_search.fit(X_train, y_train)
best_model = random_search.best_estimator_

# Predictions
y_train_pred = best_model.predict(X_train)
y_test_pred = best_model.predict(X_test)

# Evaluation
xg_train_rmse = np.sqrt(mean_squared_error(y_train, y_train_pred))
xg_test_rmse = np.sqrt(mean_squared_error(y_test, y_test_pred))
xg_train_mae = mean_absolute_error(y_train, y_train_pred)
xg_test_mae = mean_absolute_error(y_test, y_test_pred)
xg_train_r2 = r2_score(y_train, y_train_pred)
xg_test_r2 = r2_score(y_test, y_test_pred)

# Creates a mask to identify test set values where y_test is not zero. This prevents division by zero in the MAPE calculation.
nonzero_mask = y_test != 0
mape = np.mean(np.abs((y_test[nonzero_mask] - y_test_pred[nonzero_mask]) / y_test[nonzero_mask])) * 100

```

Rajah 9 Pembangunan Model XGBoost

```

catboost = CatBoostRegressor(
    random_state=42,
    verbose=0
)

random_search = RandomizedSearchCV(
    estimator=catboost,
    param_distributions=param_grid,
    n_iter=10,
    scoring='neg_root_mean_squared_error',
    cv=5,
    verbose=1,
    random_state=42,
    n_jobs=-1
)

# Fit model (pass categorical column indices)
cat_features = [X.columns.get_loc(col) for col in categorical_cols if col in X.columns]
random_search.fit(X_train, y_train, cat_features=cat_features)
best_model = random_search.best_estimator_

# Predictions
y_train_pred = best_model.predict(X_train)
y_test_pred = best_model.predict(X_test)

# Evaluation
cat_train_rmse = np.sqrt(mean_squared_error(y_train, y_train_pred))
cat_test_rmse = np.sqrt(mean_squared_error(y_test, y_test_pred))
cat_train_mae = mean_absolute_error(y_train, y_train_pred)
cat_test_mae = mean_absolute_error(y_test, y_test_pred)
cat_train_r2 = r2_score(y_train, y_train_pred)
cat_test_r2 = r2_score(y_test, y_test_pred)

# MAPE
nonzero_mask = y_test != 0
mape = np.mean(np.abs((y_test[nonzero_mask] - y_test_pred[nonzero_mask]) / y_test[nonzero_mask])) * 100

```

Rajah 10 Pembangunan Model CatBoost

```

rf = RandomForestRegressor(random_state=42)
param_grid = {
    'n_estimators': [50, 100],
    'max_depth': [None, 10, 20],
    'min_samples_split': [2, 5]
}

# Perform randomized search for hyperparameter tuning
random_search = RandomizedSearchCV(
    estimator=rf,
    param_distributions=param_grid,
    n_iter=10,
    scoring='neg_root_mean_squared_error',
    cv=5,
    verbose=1,
    random_state=42,
    n_jobs=-1
)

# Fit the model
random_search.fit(X_train, y_train)
best_rf = random_search.best_estimator_

# Make predictions
y_train_pred = best_rf.predict(X_train)
y_test_pred = best_rf.predict(X_test)

# Calculate evaluation metrics
rf_train_rmse = np.sqrt(mean_squared_error(y_train, y_train_pred))
rf_test_rmse = np.sqrt(mean_squared_error(y_test, y_test_pred))
rf_train_mae = mean_absolute_error(y_train, y_train_pred)
rf_test_mae = mean_absolute_error(y_test, y_test_pred)
rf_train_r2 = r2_score(y_train, y_train_pred)
rf_test_r2 = r2_score(y_test, y_test_pred)

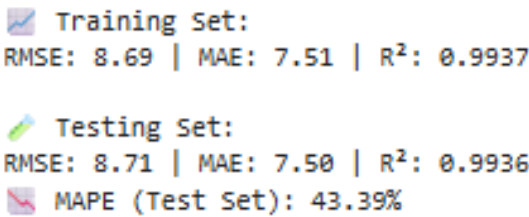
# Calculate MAPE for non-zero values
nonzero_mask = y_test != 0
mape = np.mean(np.abs((y_test[nonzero_mask] - y_test_pred[nonzero_mask]) / y_test[nonzero_mask])) * 100

```

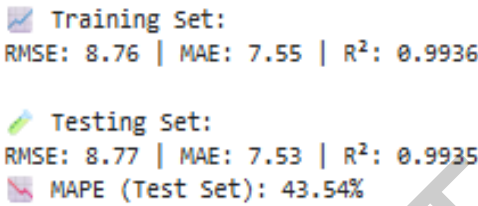
Rajah 11 Pembangunan Model Random Forest

d. Fasa Pengujian Model Pembelajaran Mesin

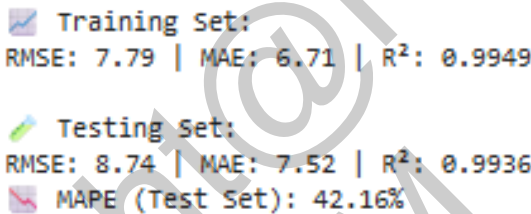
Fasa pengujian model dalam pembelajaran mesin merupakan tahap penting untuk menilai sejauh mana model yang dibina dapat membuat ramalan yang tepat terhadap data baru yang belum pernah dilihat sebelumnya. Pada fasa ini, model yang telah dilatih akan diuji menggunakan set data ujian untuk mengukur prestasinya secara objektif, dengan menggunakan pelbagai metrik penilaian seperti *Root Mean Squared Error* (RMSE), *Mean Absolute Error* (MAE), dan *R-squared* (R^2) yang memberikan gambaran tentang kesalahan purata dan kebolehan model menjelaskan varians data sebenar. Model yang digunakan termasuk XGBoost, CatBoost dan Random Forest yang merupakan algoritma pembelajaran berasaskan pokok keputusan dengan pendekatan ensemble yang berbeza. XGBoost dan CatBoost menggunakan teknik boosting untuk meningkatkan ketepatan ramalan manakala Random Forest menggunakan teknik bagging untuk mengurangkan overfitting. Evaluasi menggunakan RMSE dan MAE membantu mengukur magnitud ralat ramalan, sementara R^2 menunjukkan sejauh mana varians data sasaran dapat diterangkan oleh model.



Rajah 12 Keputusan Penilaian Model XGBoost



Rajah 13 Keputusan Penilaian Model CatBoost



Rajah 14 Keputusan Penilaian Model Random Forest

- i. **Root Mean Square Error (RMSE)**
Root Mean Square Error (RMSE) merupakan metrik yang digunakan untuk mengukur purata ralat antara nilai yang diramal dan nilai sebenar dalam satu set data. Ia dikira dengan mengambil punca kuasa dua daripada min ralat kuasa dua, yang memberi penekanan lebih kepada ralat yang lebih besar. Nilai RMSE yang lebih rendah menunjukkan ketepatan ramalan yang lebih baik, dan ia biasanya digunakan untuk menilai prestasi model regresi.

Jadual 1 Perbandingan RMSE bagi set latihan dan set ujian

Set Data	XGBoost	CatBoost	Random Forest
Set Latihan	8.69	8.76	7.79
Set Ujian	8.71	8.77	8.74

Jadual 1 menunjukkan perbandingan nilai RMSE bagi set latihan dan set ujian bagi tiga model pembelajaran mesin iaitu XGBoost, CatBoost, dan Random Forest. Pada set latihan, Random Forest mencatatkan nilai

RMSE terendah iaitu 7.79, menunjukkan prestasi terbaik dalam menyesuaikan diri dengan data latihan berbanding XGBoost (8.69) dan CatBoost (8.76). Namun pada set ujian, nilai RMSE Random Forest meningkat kepada 8.74, hampir sama dengan XGBoost (8.71) dan sedikit lebih baik daripada CatBoost (8.77) yang menunjukkan bahawa Random Forest mungkin mengalami sedikit overfitting berbanding XGBoost yang lebih konsisten antara set latihan dan ujian. Secara umum, ketiga-tiga model ini adalah algoritma ensemble yang popular dalam pembelajaran mesin; XGBoost dan CatBoost menggunakan teknik boosting untuk meningkatkan ketepatan ramalan secara berperingkat, manakala Random Forest menggunakan teknik bagging untuk mengurangkan varians dan overfitting. RMSE yang lebih rendah menandakan ralat ramalan yang lebih kecil, dan perbezaan nilai antara set latihan dan ujian memberikan gambaran tentang kestabilan model dalam mengendalikan data baru.

ii. **Mean Absolute Error (MAE)**

Mean Absolute Error (MAE) adalah metrik yang mengukur purata magnitud ralat antara nilai yang diramal dan nilai sebenar tanpa mempertimbangkan arah ralat. Ia dikira dengan mengambil nilai mutlak perbezaan antara ramalan dan nilai sebenar, kemudian dipuratakan nilai tersebut. Nilai MAE yang lebih rendah menunjukkan ketepatan ramalan yang lebih baik dan ia digunakan untuk menilai prestasi model dengan memberi penekanan yang sama kepada semua ralat.

Jadual 2 Perbandingan MAE bagi set latihan dan set ujian

Set Data	XGBoost	CatBoost	Random Forest
Set Latihan	7.51	7.55	6.71
Set Ujian	7.50	7.53	7.52

Jadual 2 merupakan perbandingan nilai MAE bagi set latihan dan set ujian untuk model XGBoost, CatBoost, dan Random Forest. Pada set latihan, Random Forest mencatatkan MAE terendah iaitu 6.71 menunjukkan ralat purata mutlak yang paling kecil berbanding XGBoost (7.51) dan CatBoost (7.55). Namun, pada set ujian, nilai MAE Random Forest meningkat kepada 7.52, hampir sama dengan XGBoost (7.50) dan lebih baik daripada CatBoost (7.53), menandakan Random Forest mungkin mengalami sedikit penurunan prestasi apabila diuji pada data baru. Secara umum, MAE mengukur purata nilai mutlak perbezaan antara ramalan model dan nilai sebenar, memberikan gambaran langsung tentang ketepatan ramalan tanpa mengambil kira arah ralat. Perbezaan nilai MAE antara set latihan dan ujian juga membantu menilai kestabilan model dalam mengendalikan data yang tidak pernah dilihat sebelum ini.

iii. **R-Squared (R^2)**

R-squared (R^2) adalah metrik yang menunjukkan peratusan varians dalam pemboleh ubah bergantung yang dapat dijelaskan oleh pemboleh ubah bebas dalam model. Nilai R^2 adalah antara 0 hingga 1 di mana nilai 1 menunjukkan bahawa model mampu menangkap 100% varians data manakala nilai 0 menunjukkan tiada varians yang digambarkan oleh model. Nilai yang lebih tinggi menunjukkan kesesuaian model yang lebih baik dengan data, walaupun ia juga perlu diambil kira bersama metrik lain untuk menilai ketepatan keseluruhan model terutamanya dalam set data yang kompleks atau apabila terdapat lebih banyak pemboleh ubah.

Jadual 3 Perbandingan R^2 bagi set latihan dan set ujian

Set Data	XGBoost	CatBoost	Random Forest
Set Latihan	99.37%	99.36%	99.49%
Set Ujian	99.36%	99.35%	99.36%

Jadual 3 menunjukkan perbandingan nilai *R-squared* (R^2) bagi set latihan dan set ujian untuk model XGBoost, CatBoost, dan Random Forest. Ketiga-tiga model menunjukkan prestasi yang sangat tinggi dengan nilai R^2 melebihi 99%, menandakan bahawa model-model ini mampu menangkap lebih daripada 99% varians dalam data sasaran. Pada set latihan, Random Forest mencatatkan nilai R^2 tertinggi iaitu 99.49%, sedikit lebih baik daripada XGBoost (99.37%) dan CatBoost (99.36%). Namun, pada set ujian, nilai R^2 untuk ketiga-tiga model hampir sama, iaitu sekitar 99.35% hingga 99.36%, menunjukkan ketiga-tiga model mempunyai kebolehan generalisasi yang sangat baik terhadap data baru. R^2 yang tinggi ini mencerminkan ketepatan model dalam meramalkan nilai sebenar dan menandakan bahawa model-model ini sangat sesuai untuk tugas regresi yang dijalankan. Secara keseluruhan, perbezaan kecil antara set latihan dan ujian menunjukkan bahawa model-model ini tidak mengalami masalah overfitting yang kritikal dan mampu memberikan prestasi yang stabil.

iv. **Pengujian t-test dan p-value**

```
Paired t-test results: RMSE
Random Forest vs XGBoost:  t = 3.6059, p = 0.0226
Random Forest vs CatBoost: t = 9.0658, p = 0.0008
XGBoost vs CatBoost:      t = 4.7664, p = 0.0089

Paired t-test results: MAE
Random Forest vs XGBoost:  t = 2.4177, p = 0.0729
Random Forest vs CatBoost: t = 4.3056, p = 0.0126
XGBoost vs CatBoost:      t = 2.9798, p = 0.0407

Paired t-test results: R2
Random Forest vs XGBoost:  t = -3.5531, p = 0.0237
Random Forest vs CatBoost: t = -9.1508, p = 0.0008
XGBoost vs CatBoost:      t = -4.8333, p = 0.0084
```

Rajah 15 Keputusan t-test dan p-value

T-test digunakan untuk menguji sama ada perbezaan RMSE, MAE atau R^2 antara set latihan dan set ujian adalah signifikan secara statistik. Sebagai contoh, terdapat peningkatan pada RMSE set latihan kepada RMSE set ujian, maka t-test boleh menentukan sama ada peningkatan ini bermakna atau tidak.

P-value pula akan membantu menilai kebarangkalian peningkatan ini. Jika p-value kurang daripada 0.05, ia menunjukkan perbezaan tersebut tidak berlaku secara rawak dan mungkin mencerminkan ketidakstabilan model. Untuk ketiga-tiga model, dengan R^2 yang konsisten pada 0.99, t-test boleh mengesahkan sama ada kesesuaian ini kekal signifikan di antara set latihan dan ujian, manakala p-value yang rendah akan menguatkan kesimpulan bahawa model menjelaskan varians data dengan baik.

Jadual 4 Keputusan perbandingan t-test dan p-value model bagi RMSE

Model vs Model	t-Statistic	p-Value
Random Forest vs XGBoost	3.6059	0.0226
Random Forest vs CatBoost	9.0658	0.0008
XGBoost vs CatBoost	4.7664	0.0089

Jadual 4 menunjukkan keputusan paired t-test bagi perbandingan RMSE antara model Random Forest, XGBoost, dan CatBoost di mana semua perbandingan mempunyai nilai $p < 0.05$, menandakan perbezaan prestasi antara model adalah signifikan secara statistik. Nilai t yang positif menunjukkan bahawa CatBoost mempunyai RMSE yang lebih rendah dan dengan itu prestasi yang lebih baik berbanding Random Forest dan XGBoost, manakala XGBoost juga lebih baik daripada Random Forest. Kesimpulannya, CatBoost adalah model yang paling baik dari segi ketepatan ramalan berdasarkan analisis ini.

Jadual 5 Keputusan perbandingan t-test dan p-value model bagi MAE

Model vs Model	t-Statistic	p-Value
Random Forest vs XGBoost	2.4177	0.0729
Random Forest vs CatBoost	4.3056	0.0126
XGBoost vs CatBoost	2.9798	0.0407

Jadual 5 menunjukkan keputusan paired t-test bagi perbandingan MAE antara model Random Forest, XGBoost, dan CatBoost di mana perbandingan Random Forest vs XGBoost mempunyai nilai $p > 0.05$, menandakan tiada perbezaan signifikan secara statistik antara kedua model

ini. Namun, perbandingan Random Forest vs CatBoost dan XGBoost vs CatBoost menunjukkan nilai $p < 0.05$, menandakan perbezaan prestasi yang signifikan secara statistik. Nilai t yang positif menunjukkan bahawa CatBoost mempunyai MAE yang lebih rendah dan dengan itu prestasi yang lebih baik berbanding Random Forest dan XGBoost. Kesimpulannya, CatBoost adalah model yang paling baik dari segi ketepatan ramalan berdasarkan metrik MAE.

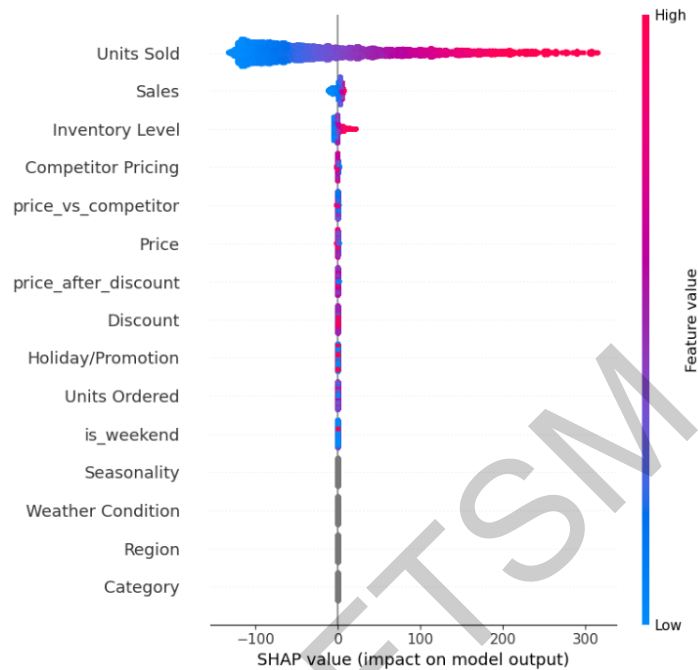
Jadual 6 Keputusan perbandingan t-test dan p-value model bagi R^2

Model vs Model	t-Statistic	p-Value
Random Forest vs XGBoost	-3.5531	0.0237
Random Forest vs CatBoost	-9.1508	0.0008
XGBoost vs CatBoost	-4.8333	0.0084

Jadual 6 menunjukkan keputusan paired t-test bagi perbandingan R^2 antara model Random Forest, XGBoost, dan CatBoost di mana semua perbandingan mempunyai nilai $p < 0.05$, menandakan perbezaan prestasi antara model adalah signifikan secara statistik. Nilai t yang negatif menunjukkan bahawa CatBoost mempunyai nilai R^2 yang lebih tinggi dan dengan itu prestasi yang lebih baik berbanding Random Forest dan XGBoost, manakala XGBoost juga lebih baik daripada Random Forest. Kesimpulannya, CatBoost adalah model yang paling baik dari segi ketepatan ramalan berdasarkan metrik R^2 dalam analisis ini.

v. **Visualisasi Fitur Penting Menggunakan SHAP Analisis**

SHAP (SHapley Additive exPlanations) analysis adalah kaedah dalam pembelajaran mesin yang digunakan untuk menjelaskan sumbangan setiap ciri terhadap output model ramalan, dengan mengagihkan sumbangan secara adil berdasarkan teori permainan Shapley value melalui kesan pertukaran dalam semua kombinasi mungkin. SHAP boleh membantu mengenal pasti ciri-ciri yang paling mempengaruhi ketepatan ramalan atau ralat sekaligus membantu model untuk membuat keputusan dengan lebih tepat.



Rajah 16 Keputusan Penilaian Model Random Forest

Plot SHAP ini menunjukkan kepentingan dan kesan setiap ciri terhadap ramalan model di mana ciri *Units Sold* mempunyai impak terbesar (kedudukannya paling atas dengan nilai SHAP yang diplot ke kiri dan kanan), bermakna jumlah unit terjual sangat mempengaruhi output model secara positif atau negatif bergantung pada nilainya. Ciri *Sales* dan *Inventory Level* juga penting, tetapi dengan pengaruh yang lebih sederhana. Titik berwarna mewakili nilai ciri (merah = tinggi, biru = rendah), contohnya, titik merah di sebelah kanan menunjukkan nilai tinggi *Units Sold* meningkatkan ramalan. Sebaliknya, ciri seperti *Weather Condition*, *Region*, dan *Category* mempunyai impak hampir sifar kerana titik-titik mereka berkumpul di sekitar paksi tengah, menunjukkan ciri ini kurang berperanan dalam keputusan model. Oleh itu, ciri yang terpilih adalah *Units Sold*, *Sales* dan *Inventory Level*.

vi. **Perbandingan Semua Ciri dan Ciri Terpilih berdasarkan nilai RMSE dan R-Squared**

Perbandingan antara semua ciri dan ciri terpilih berdasarkan nilai RMSE dan R-Squared dapat memberikan gambaran tentang keberkesanan pemilihan ciri dalam meningkatkan prestasi model. Nilai RMSE yang lebih rendah untuk ciri terpilih berbanding semua ciri menunjukkan pengurangan ralat ramalan, seperti yang mungkin dilihat jika RMSE untuk set ujian menurun dari nilai asal mencerminkan ketepatan yang lebih baik. Sementara itu, nilai R-Squared yang kekal tinggi atau meningkat menunjukkan bahawa ciri terpilih masih dapat menjelaskan varians data dengan baik.

Jadual 7 Perbandingan semua ciri dan ciri terpilih RMSE, MAE & R² bagi set latihan model CatBoost

Ciri	CatBoost		
	RMSE	MAE	R ²
Semua Ciri (Date, Store ID, Product ID, Category, Region, Inventory Level, Units Sold, Units Ordered, Price, Discount, Weather Condition, Holiday/Promotion, Competitor Pricing, Seasonality, price_after_discount, price_vs_competitor, is_weekend, is_promo, sales)	8.76	7.55	99.36%
Ciri Terpilih (Units Sold, Sales, Inventory Level, Date)	8.75	7.54	99.36%

Jadual 7 menunjukkan perbandingan prestasi model CatBoost menggunakan dua set ciri berbeza pada set latihan berdasarkan metrik RMSE, MAE, dan R². Model dengan semua ciri (termasuk *Date*, *Store ID*, *Product ID*, *Category*, *Region*, *Inventory Level*, *Units Sold*, *Units Ordered*, *Price*, *Discount*, *Weather Condition*, *Holiday/Promotion*, *Competitor Pricing*, *Seasonality*, *price_after_discount*, *price_vs_competitor*, *is_weekend*, *is_promo*, *sales*) mencatatkan RMSE sebanyak 8.76, MAE pula sebanyak 7.55 dan R² yang sangat tinggi iaitu 99.36%. Manakala model dengan ciri terpilih (*Units Sold*, *Sales*, *Inventory Level*, *Date*) menunjukkan prestasi yang hampir sama dengan RMSE sebanyak 8.75, MAE pula sebanyak 7.54 dan R² juga tinggi iaitu sebanyak 99.36%.

Jadual 8 Perbandingan semua ciri dan ciri terpilih RMSE, MAE & R² bagi set ujian model CatBoost

Ciri	CatBoost		
	RMSE	MAE	R ²
Semua Ciri (Date, Store ID, Product ID, Category, Region, Inventory Level, Units Sold, Units Ordered, Price, Discount, Weather Condition, Holiday/Promotion, Competitor Pricing, Seasonality, price_after_discount, price_vs_competitor,	8.77	7.53	99.35%

is_weekend, is_promo,
sales)

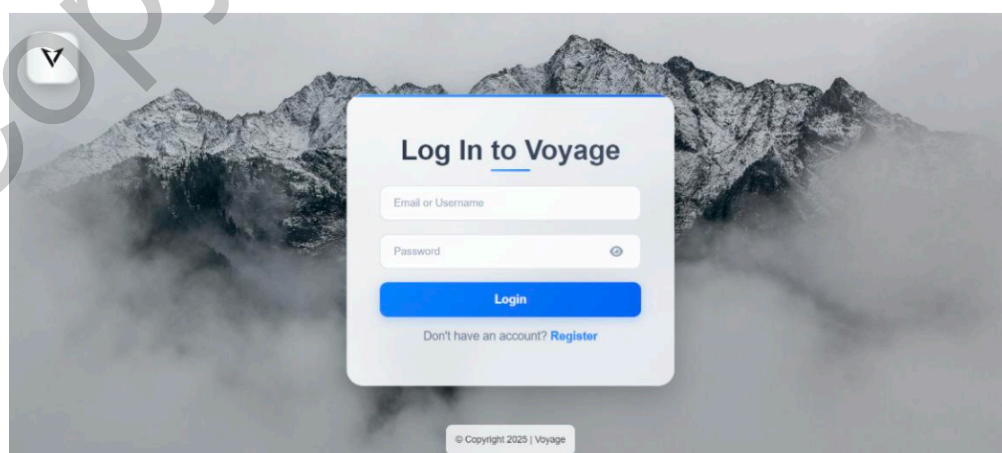
Ciri Terpilih (Units Sold, Sales, Inventory Level, Date)	8.78	7.54	99.35%
--	------	------	--------

Jadual 8 memperlihatkan perbandingan prestasi model CatBoost pada set ujian menggunakan dua set ciri yang berbeza berdasarkan metrik RMSE, MAE, dan R2. Model dengan semua ciri mencatatkan nilai RMSE sebanyak 8.77, MAE pula sebanyak 7.53 dan R2 yang sangat tinggi iaitu 99.35%. Model dengan ciri terpilih (Units Sold, Sales, Inventory Level, Date) menunjukkan prestasi yang hampir sama dengan RMSE sebanyak 8.78, MAE pula sebanyak 7.54, dan R2 juga 99.35%. Keputusan ini menunjukkan bahawa model yang menggunakan ciri terpilih dapat mengekalkan ketepatan ramalan yang setanding dengan model yang menggunakan semua ciri, walaupun dengan jumlah ciri yang lebih sedikit. Ini membuktikan bahawa pemilihan ciri yang tepat dapat mengurangkan kerumitan model tanpa mengorbankan prestasi sekaligus meningkatkan kecekapan dan kebolegunaan model CatBoost dalam set ujian.

Fasa Pembangunan Sistem

a. Reka Bentuk Antara Muka Laman Web

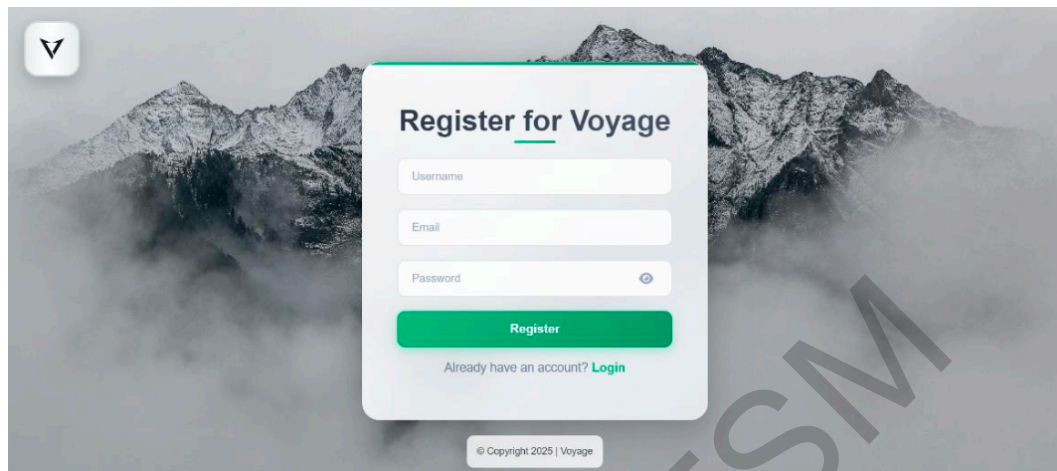
Laman web Voyage ini mempunyai 6 laman antara muka iaitu halaman pendaftaran akaun, halaman log masuk pengguna, halaman utama, halaman muat naik fail, halaman keputusan dan sejarah ramalan, dan halaman akaun. Voyage ini dibangunkan menggunakan rangka kerja flask dengan bahasa pengaturcaraan Python. Perisian yang digunakan adalah Microsoft Visual Studio Code. Rajah 17 sehingga 22 berikut menunjukkan laman antara muka laman web Voyage.



Rajah 17 Antara muka Log Masuk Pengguna

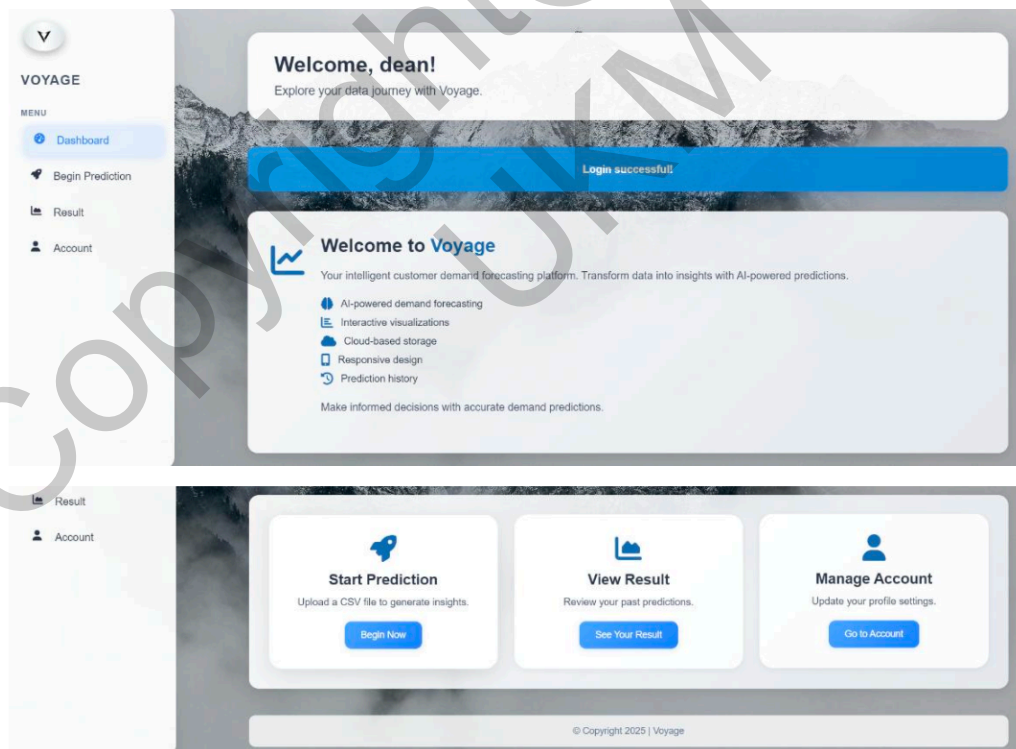
Rajah 17 menunjukkan halaman pendaftaran akaun laman web Voyage. Pengguna perlu mengisi setiap maklumat yang diperlukan. Maklumat yang perlu diisi adalah nama pengguna atau emel dan kata laluan. Selain itu pengguna perlu

juga memilih untuk membuat pendaftaran akaun jika belum lagi mendaftar.



Rajah 18 Antara muka Pendaftaran akaun

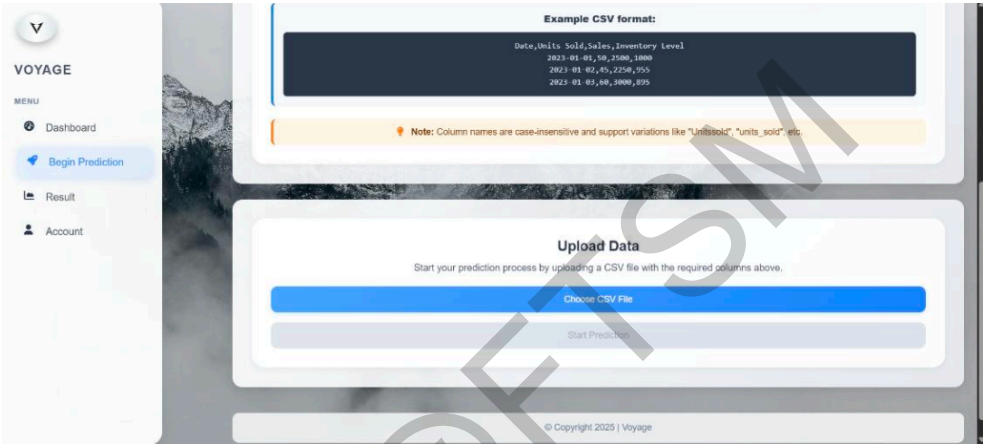
Berdasarkan rajah 18 menunjukkan halaman pendaftaran akaun laman web Voayge. Halaman ini akan dipaparkan selepas pengguna menekan butang ‘register’ untuk membuat pendaftaran akaun dengan mengisi nama pengguna, emel dan password. Setelah selesai membuat pendaftaran akaun, pengguna akan perlu log masuk akaun di halaman log masuk.



Rajah 19 Antara muka Halaman utama

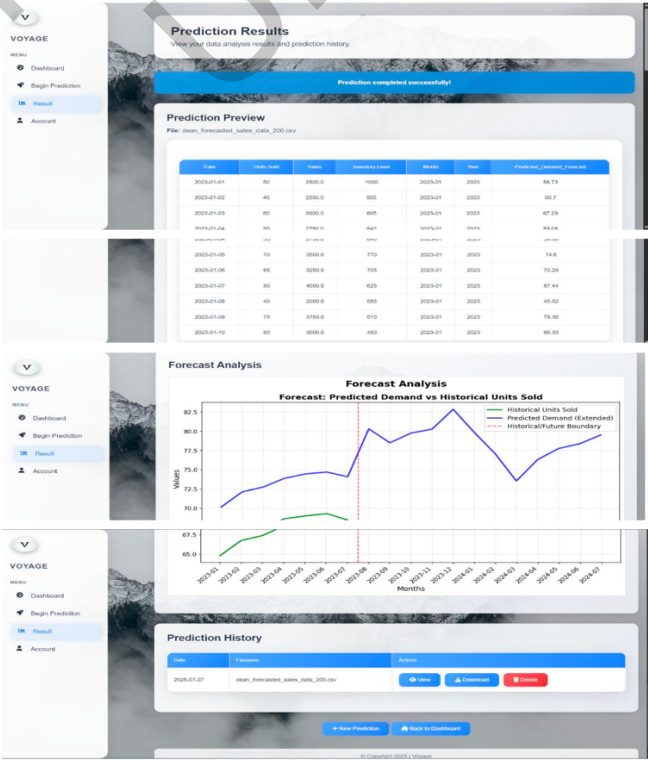
Berdasarkan rajah 19 menunjukkan halaman utama laman web Voayge. Halaman utama ini akan dipaparkan selepas pengguna log masuk ke dalam akaun. Halaman utama ini memberikan pilihan kepada pengguna untuk melayar kepada 3

halaman yang berbeza. ‘Begin Prediction’ adalah halaman untuk pengguna muat naik fail dataset berdasarkan format tertentu dan memulakan ramalan. ‘Result’ merupakan halaman keputusan ramalan untuk fail yang dimuat naik oleh pengguna, Selain itu, pengguna juga dapat melihat Sejarah ramalan di halaman ini. Akhir sekali yang merupakan halaman untuk pengguna membuat pengubahsuaian nama pengguna, emel atau kata laluan iaitu halaman akaun.



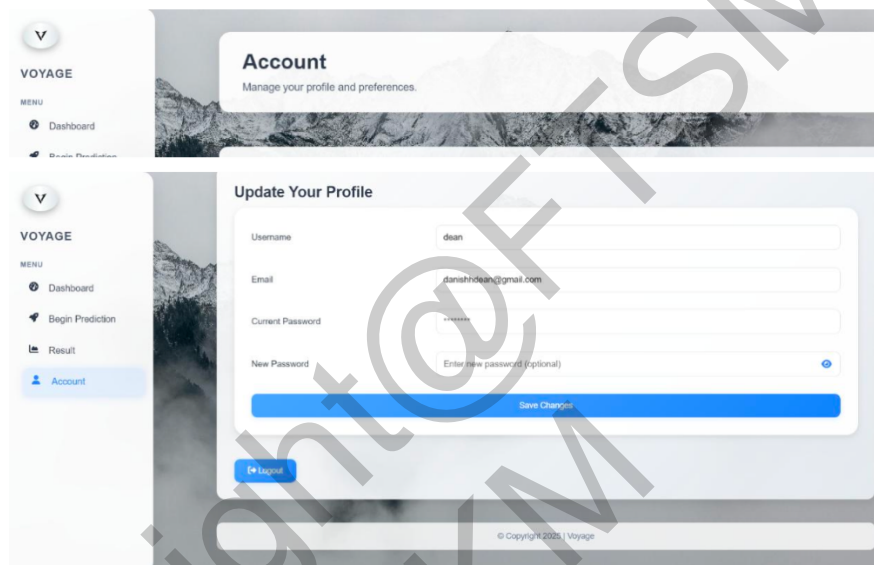
Rajah 20 Antara muka Muat Naik Fail

Rajah 20 menunjukkan halaman muat naik fail. Halaman ini akan dipaparkan sekiranya pengguna menekan butang ‘Begin Prediction’ di halaman utama. Di halaman ini, pengguna boleh memilih fail dataset untuk tujuan ramalan kemudian menekan butang ‘Start Prediction’ untuk memberikan ramalan permintaan pelanggan. Fail yang dimuat naik oleh pengguna perlulah mempunyai kolum ciri yang diminta.



Rajah 21 Antara muka Keputusan Ramalan dan Sejarah Ramalan

Rajah 21 menunjukkan halaman Keputusan ramalan dan Sejarah ramalan bagi laman web Voyage. Halaman ini akan memaparkan pratonton bagi 10 baris Keputusan bagi ramalan terakhir yang dilakukan oleh pengguna. Selain itu, pengguna dapat melihat visualisasi berdasarkan graf garis yang menunjukkan ramalan permintaan pelanggan dan unit yang terjual serta ramalan permintaan pelanggan untuk satu tahun akan datang selepas bulan terakhir dalam dataset. Pengguna juga boleh melihat semula fail yang telah diramal di bawah visualisasi graf tersebut untuk tujuan perbandingan. Fail ramalan ini turut boleh dimuat turun oleh pengguna dalam format csv.



Rajah 22 Antara muka Akaun Pengguna

Rajah 22 menunjukkan halaman akaun pengguna. Halaman ini bertujuan untuk membolehkan pengguna untuk mengubah nama pengguna, emel dan kata laluan. Selepas itu, pengguna perlu menekan butang 'Save Changes' untuk pengesahan. Pengguna juga dapat log keluar akaun dengan menekan butang 'Logout'.

b. Fasa Pembangunan Laman Web

Voyage ialah laman web yang direka untuk meramalkan permintaan pelanggan menggunakan algoritma model CatBoost. Sistem ini mempunyai dua bahagian utama iaitu pembangunan bahagian front-end dan back-end untuk memastikan sistem berfungsi dengan baik dan memudahkan pengguna untuk menggunakan sistem ini.

i. Front-end

Front-end merujuk kepada bahagian antaramuka pengguna dalam sesebuah aplikasi atau laman web, iaitu apa yang dilihat dan digunakan secara langsung oleh pengguna. Pembangunan front-end biasanya menggunakan teknologi seperti HTML untuk struktur kandungan, CSS untuk gaya visual, dan JavaScript untuk fungsi interaktif.


```

1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>Voyage - Register</title>
7   <link rel="stylesheet" href="{{ url_for('static', filename='style.css') }}">
8   <link rel="stylesheet" href="{{ url_for('static', filename='register.css') }}">
9   <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.4.0/css/all.min.css">
10 </head>
11 <body>
12   <!-- Header Image -->
13   
14
15   <!-- Main Content -->
16   <div class="register-container">
17     <h1>Register for Voyage</h1>
18     <!-- Flash Messages -->
19     {% with messages = get_flashed_messages(with_categories=true) %}
20       {% if messages %}
21         <div class="flash-messages">
22           {% for category, message in messages %}
23             <p class="{{ category }}">{{ message }}</p>
24           {% endfor %}
25         </div>
26       {% endif %}
27     {% endwith %}
28
29     <!-- Register Form -->
30     <form action="{{ url_for('register') }}" method="POST">
31       <input type="text" name="username" placeholder="Username" required>
32       <input type="email" name="email" placeholder="Email" required>
33       <div class="password-wrapper">
34         <input type="password" name="password" placeholder="Password" required>
35         <span class="fas fa-eye toggle-password" id="togglePassword" aria-label="Toggle password visibility"></i>
36       </div>
37       <button type="submit">Register</button>
38     </form>
39     <p>Already have an account? <a href="{{ url_for('login') }}">Login</a></p>
40   </div>
41
42   <!-- Footer -->
43   <footer>
44     <p>© Copyright 2025 | Voyage</p>
45   </footer>
46
47   <!-- JavaScript -->
48   <script>
49     document.getElementById('togglePassword').addEventListener('click', function () {
50       const passwordInput = document.getElementById('password');
51       const isVisible = passwordInput.type === 'text';
52       passwordInput.type = isVisible ? 'password' : 'text';
53       this.classList.toggle('fa-eye');
54       this.classList.toggle('fa-eye-slash');
55     });
56   </script>
57 </body>
58 </html>

```

Rajah 23 Kod Halaman Pendaftaran Akaun

```

1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>Voyage - Login</title>
7   <link rel="stylesheet" href="{{ url_for('static', filename='style.css') }}">
8   <link rel="stylesheet" href="{{ url_for('static', filename='login.css') }}">
9   <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.4.0/css/all.min.css">
10 </head>
11 <body>
12   <!-- Header Image -->
13   
14
15   <!-- Main Content -->
16   <div class="login-container">
17     <h1>Log In to Voyage</h1>
18     <!-- Flash Messages -->
19     {% with messages = get_flashed_messages(with_categories=true) %}
20       {% if messages %}
21         <div class="flash-messages">
22           {% for category, message in messages %}
23             <p class="{{ category }}">{{ message }}</p>
24           {% endfor %}
25         </div>
26       {% endif %}
27     {% endwith %}
28
29     <!-- Login Form -->
30     <form action="{{ url_for('login') }}" method="POST">
31       <input type="text" name="login_id" placeholder="Email or Username" required>
32       <div class="password-wrapper">
33         <input type="password" name="password" placeholder="Password" required>
34         <span class="fas fa-eye toggle-password" id="togglePassword" aria-label="Toggle password visibility"></i>
35       </div>
36       <button type="submit">Login</button>
37     </form>
38     <p>Don't have an account? <a href="{{ url_for('register') }}">Register</a></p>
39   </div>
40
41   <!-- Footer -->
42   <footer>
43     <p>© Copyright 2025 | Voyage</p>
44   </footer>
45
46   <!-- JavaScript -->
47   <script>
48     document.getElementById('togglePassword').addEventListener('click', function () {
49       const passwordInput = document.getElementById('password');
50       const isVisible = passwordInput.type === 'text';
51       passwordInput.type = isVisible ? 'password' : 'text';
52       this.classList.toggle('fa-eye');
53       this.classList.toggle('fa-eye-slash');
54     });
55   </script>
56 </body>
57 </html>

```

Rajah 24 Kod Halaman Log Masuk Pengguna

Dalam pembangunan sistem Voyage, teknologi front-end seperti HTML, CSS, dan JavaScript telah digunakan untuk membina antaramuka yang moden, responsif, dan mesra pengguna, bagi memastikan pengalaman penggunaan yang lancar merentasi pelbagai jenis peranti. Rajah 23 dan 24 menunjukkan rangka kod halaman pendaftaran akaun dan log masuk pengguna. Struktur kedua halaman ini hampir sama tetapi berbeza di ruang untuk mengisi maklumat. Halaman pendaftaran mempunyai ruang untuk mengisi 'username', 'email' dan 'password' manakala halaman log masuk hanya memerlukan 'login_id' iaitu sama ada nama pengguna atau emel dan kata laluan. Kedua halaman ini mempunyai pautan yang membolehkan pengguna untuk ke halaman pendaftaran akaun atau log masuk dengan mudah. Jika maklumat yang diisi adalah salah, mesej maklum balas 'flash messages' akan meminta pengguna untuk cuba semula.

ii. Back-end

Back-end pula merujuk kepada bahagian system yang berfungsi di belakang tabir dan tidak dapat dilihat oleh pengguna. Walau bagaimanapun, ia sangat penting kerana ia mengendalikan logik aplikasi, pemprosesan data, penyimpanan maklumat ke dalam pangkalan data dan komunikasi server atau sistem lain.

```

4 from flask import Flask, render_template, request, redirect, url_for, session, flash, send_file, jsonify
5 import numpy as np
6 import joblib
7 import io
8 import pandas as pd
9 from pyairtable import Api
10 from werkzeug.security import generate_password_hash, check_password_hash
11 from werkzeug.utils import secure_filename
12 import os
13 from datetime import datetime, timedelta
14 from sklearn.metrics import mean_squared_error, mean_absolute_error, r2_score
15 import logging
16 import matplotlib.pyplot as plt
17
18 # Set up logging
19 logging.basicConfig(level=logging.DEBUG)
20 logger = logging.getLogger(__name__)
21
22 MODEL_PATH = r'C:\Users\danis\OneDrive\Desktop\Danish Dean\sem 6\Projek\Training.pkl'
23 try:
24     model = joblib.load(MODEL_PATH)
25     logger.info("Model loaded successfully.")
26 except Exception as e:
27     logger.error(f"Error loading model: {str(e)}")
28     model = None
29
30 app = Flask(__name__)
31 app.secret_key = os.getenv('SECRET_KEY', 'your_secret_key')
32 app.permanent_session_lifetime = timedelta(minutes=30)
33
34 # Airtable configuration
35 AIRTABLE_API_KEY = os.getenv('AIRTABLE_API_KEY', 'patGIYW30PTUI97L.e07d0a2a06eeaf59f879dbdb31448f7379cd2bf9c1916c43fef9d085fa20b09')
36 AIRTABLE_BASE_ID = 'app55TzuINT0DxjL'
37 USERS_TABLE_NAME = 'Users'
38 PREDICTIONS_TABLE_NAME = 'Predictions'
39
40 # Initialize Airtable clients
41 api = Api(AIRTABLE_API_KEY)
42 users_table = api.table(AIRTABLE_BASE_ID, USERS_TABLE_NAME)
43 predictions_table = api.table(AIRTABLE_BASE_ID, PREDICTIONS_TABLE_NAME)
44 ..

```

Rajah 25 Rangka Kod Flask dengan Integrasi Airtable dan Fungsi Log Masuk

Rajah 25 menunjukkan rangka kod Flask dengan integrasi Airtable dan fungsi log masuk. Ia merangkumi sistem pengesahan kata laluan menggunakan Werkzeug untuk penyulitan kata laluan serta integrasi Airtable untuk pengurusan pengguna dan penyimpanan data.


```

141 @app.route('/upload_csv', methods=['POST'])
142 def upload_csv():
143     if 'username' not in session:
144         return redirect(url_for('login'))
145
146     file = request.files.get('csv-file')
147     if not file or file.filename == '':
148         flash('No file selected.')
149         return redirect(url_for('begin'))
150
151     if allowed_file(file.filename):
152         # Add artificial delay to show loading states (remove in production)
153         import time
154         time.sleep(2) # Simulate processing time
155         try:
156             data = pd.read_csv(file.stream)
157
158             # Normalize column names (case-insensitive)
159             column_mapping = {}
160             for col in data.columns:
161                 col_lower = col.lower().strip()
162                 if col_lower == 'units sold' or col_lower == 'unitsold' or col_lower == 'units_sold':
163                     column_mapping[col] = 'Units Sold'
164                 elif col_lower == 'sales':
165                     column_mapping[col] = 'Sales'
166                 elif col_lower == 'inventory level' or col_lower == 'inventorylevel' or col_lower == 'inventory_level':
167                     column_mapping[col] = 'Inventory Level'
168                 elif col_lower == 'date':
169                     column_mapping[col] = 'Date'
170
171             # Rename columns to standard format
172             data = data.rename(columns=column_mapping)
173
174             # Check for required columns (case-insensitive)
175             base_required_columns = ['Units Sold', 'Sales', 'Inventory Level']
176             missing_base_columns = [col for col in base_required_columns if col not in data.columns]
177             if missing_base_columns:
178                 flash(f'Missing required columns: {', '.join(missing_base_columns)}. Please ensure your CSV has: Units Sold, Sales, Inventory Level, and Date columns.')
179                 return redirect(url_for('begin'))
180
181             # Check if Date column is present
182             if 'Date' not in data.columns:
183                 flash('Missing required column: Date')
184                 return redirect(url_for('begin'))
185
186             # Drop all columns except the required target columns
187             required_columns = ['Units Sold', 'Sales', 'Inventory Level', 'Date']
188             data = pd.DataFrame(data[required_columns])
189
190             # Process Date column for the model and visualization
191             try:
192                 data['Date'] = pd.to_datetime(data['Date'])
193                 # Format the date back to a readable string for display
194                 data['Date_Display'] = data['Date'].dt.strftime('%Y-%m-%d')
195                 # Extract month and year for visualization
196                 data['Month'] = data['Date'].dt.strftime('%Y-%m')
197                 data['Year'] = data['Date'].dt.year
198             except Exception as e:
199                 flash(f'Error converting Date column: {str(e)}. Please ensure dates are in a valid format (YYYY-MM-DD, MM/DD/YYYY, DD/MM/YYYY, etc.).')
200                 return redirect(url_for('begin'))
201
202             X = data[required_columns].fillna(0)
203             if model is None:
204                 flash('Prediction model is not available.')
205                 return redirect(url_for('begin'))
206
207             predictions = model.predict(X)
208             data['Predicted_Demand_Forecast'] = np.round(predictions, 2)
209
210             # Save output CSV with all columns including Month
211             original_filename = secure_filename(file.filename or 'uploaded_file')
212             name_without_ext = os.path.splitext(original_filename)[0]
213             base_output_filename = f'{session["username"]}_forecasted_{name_without_ext}'
214             output_dir = os.path.join('static', 'predictions')
215             os.makedirs(output_dir, exist_ok=True)
216
217             # Find the next available number for this base filename
218             counter = 1
219             output_filename = f'{base_output_filename}.csv'
220             while os.path.exists(os.path.join(output_dir, output_filename)):
221                 output_filename = f'{base_output_filename}_{counter}.csv'
222                 counter += 1
223
224             output_path = os.path.join(output_dir, output_filename)
225
226             # Create a copy for saving with readable dates
227             data_to_save = data.copy()
228             if 'Date_Display' in data_to_save.columns:
229                 # Replace the Unix timestamp Date column with the readable Date_Display
230                 data_to_save['Date'] = data_to_save['Date_Display']
231                 data_to_save = data_to_save.drop('Date_Display', axis=1)
232
233             data_to_save.to_csv(output_path, index=False)
234
235             # Save prediction to Airtable
236             current_time = datetime.now()
237             prediction_data = {
238                 'username': session['username'],
239                 'date': current_time.strftime('%Y-%m-%d'),
240                 'timestamp': current_time.isoformat(),
241                 'filename': output_filename,
242                 'url': url_for('download_prediction', filename=output_filename, _external=True),
243             }
244             try:
245                 predictions_table.create(prediction_data)
246             except Exception as e:
247                 logger.error(f'Failed to save prediction to Airtable: {str(e)}')
248                 flash(f'Error saving prediction data: {str(e)}')
249
250             flash('Prediction completed successfully!')
251             return redirect(url_for('result'))
252
253             except Exception as e:
254                 logger.error(f'Error processing file: {str(e)}')
255                 flash(f'Error processing file: {str(e)}')
256                 return redirect(url_for('begin'))
257             else:
258                 flash('Invalid file type. Only CSV files are allowed.')
259                 return redirect(url_for('begin'))
260
261 @app.route('/download_prediction/<filename>')
262 def download_prediction(filename):
263     file_path = os.path.join('static', 'predictions', filename)
264     return send_file(
265         file_path,
266         mimetype='text/csv',
267         as_attachment=True,
268         download_name=filename
269     )

```

Rajah 26 menunjukkan rangka kod muat naik fail. Pengguna perlu memuat naik fail yang berformat csv yang mengandungi data sejarah seperti 'Units Sold', 'Sales', 'Inventory Level' dan 'Date'. Sistem akan menormalisasikan nama kolum, menyemak kehadiran kolum yang diminta dan memproses data termasuk penukaran tarikh ke format yang boleh dibaca. Data ini kemudian digunakan oleh model pembelajaran mesin yang telah dilatih untuk menjana nilai ramalan permintaan pelanggan untuk data sejarah dan satu tahun selepas data terakhir. Hasil ramalan akan disimpan dalam fail CSV baharu yang boleh dimuat turun serta direkodkan ke dalam pangkalan data Airtable.

```

361 @app.route('/prediction_plot.png')
362 def prediction_plot():
363     if "username" not in session:
364         return redirect(url_for("login"))
365
366     try:
367         predictions = predictions_table.all(
368             formula=f"({username})={session['username']}",
369             sort=['-date']
370         )
371
372         if predictions:
373             csv_url = predictions[0]["fields"].get("url")
374             if csv_url:
375                 df = pd.read_csv(csv_url)
376                 if 'Month' in df.columns and 'Predicted_Demand_Forecast' in df.columns:
377                     # Group by Month and average the demand
378                     df_grouped = (
379                         df.groupby('Month', as_index=False)['Predicted_Demand_Forecast']
380                         .mean()
381                         .sort_values('Month') # type: ignore
382                     )
383                     labels = df_grouped["Month"].tolist()
384                     values = df_grouped["Predicted_Demand_Forecast"].tolist()
385
386                     if values:
387                         fig, ax = plt.subplots(figsize=(8, 4))
388                         ax.plot(labels, values, marker='o', linestyle='-', color='b', label='Predicted Demand')
389                         ax.set_title('Retail Demand Forecast', fontsize=14, pad=10)
390                         ax.set_xlabel('Month', fontsize=12)
391                         ax.set_ylabel('Predicted Demand', fontsize=12)
392                         ax.grid(True, linestyle='--', alpha=0.7)
393                         ax.legend()
394                         plt.xticks(rotation=45, ha='right', fontsize=10)
395                         plt.yticks(fontsize=10)
396                         fig.tight_layout()
397
398                         img = io.BytesIO()
399                         plt.savefig(img, format='png', dpi=100)
400                         plt.close(fig)
401                         img.seek(0)
402                         return send_file(img, mimetype='image/png')
403                     else:
404                         labels = []
405                         values = []
406
407             except Exception as e:
408                 logger.error(f"Error generating prediction plot: {str(e)}")
409
410             # Fallback: no data image
411             fig, ax = plt.subplots(figsize=(8, 4))
412             ax.text(0.5, 0.5, 'No data available', fontsize=12, ha='center', va='center')
413             img = io.BytesIO()
414             plt.savefig(img, format='png', dpi=100)
415             plt.close(fig)
416             img.seek(0)
417             return send_file(img, mimetype='image/png')

```

Rajah 27 Rangka Kod Penjanaan Graf

Rajah 27 menunjukkan rangka kod penjanaan graf. Selepas pengguna perlu memuat naik fail, fungsi prediction_plot akan menjana dan memaparkan graf ramalan permintaan pelanggan dalam bentuk imej PNG

berdasarkan data ramalan yang disimpan di Airtable. Ia hanya boleh diakses jika pengguna telah log masuk. Fungsi ini mengambil rekod ramalan terkini milik pengguna, memuat turun fail CSV yang mengandungi hasil ramalan dan memproses data untuk mendapatkan purata permintaan bulanan. Jika data tersedia, graf garisan akan dilukis menggunakan Matplotlib untuk menunjukkan trend permintaan yang diramalkan mengikut bulan. Jika tiada data, sistem akan paparkan imej dengan mesej "No data available" sebagai alternatif.

KEPUTUSAN DAN PERBINCANGAN

Voyage, Sistem Ramalan Permintaan Pelanggan telah berjaya dibangunkan dan semua dokumentasinya telah dilengkapkan. Terdapat dua jenis pengujian yang akan dilakukan dalam projek ini. Pengujian pertama adalah pengujian model pembelajaran mesin untuk menilai model yang terbaik antara ketiga-tiga model iaitu XGBoost, CatBoost dan Random Forest. Pengujian kedua pula adalah pengujian kepuasan pengguna untuk menilai prestasi laman web Voyage.

Pengujian Model Pembelajaran Mesin Voyage

Pengujian model merupakan fasa yang penting untuk menentukan keberkesanan model dengan mengira RMSE, MAE dan R^2 . Data yang tidak konsisten, tidak relevan dan mempunyai outlier akan menyebabkan nilai RMSE dan MAE yang tinggi. Nilai R^2 yang tinggi pula bermakna model ramalan mampu untuk menerangkan sebahagian besar variasi dalam data output berdasarkan input atau pemboleh ubah bebas yang digunakan.

Jadual 1 Perbandingan RMSE, MAE dan R^2 bagi kedua-dua set

Model	XGBoost		CatBoost		Random Forest	
Set	Latihan	Ujian	Latihan	Ujian	Latihan	Ujian
RMSE	8.69	8.71	8.76	8.77	7.79	8.74
MAE	7.51	7.50	7.55	7.53	6.71	7.52
R^2	99.37%	99.36%	99.36%	99.35%	99.49%	99.36%

Berdasarkan Jadual 1, ketiga-tiga model menunjukkan ketepatan yang sangat tinggi dengan nilai R^2 yang melebihi 99% bagi set latihan dan ujian. Random Forest mencatatkan prestasi terbaik secara keseluruhan dengan RMSE dan MAE yang paling rendah dalam set latihan menunjukkan ia lebih tepat dalam meramalkan nilai sebenar semasa latihan. Namun, dalam set ujian, ketiga-tiga model menunjukkan prestasi yang hampir sama dengan perbezaan kecil menjadikan ketiga-tiganya sesuai digunakan, walaupun Random Forest sedikit lebih baik dalam latihan manakala XGBoost dan CatBoost lebih seimbang antara latihan dan ujian.

Pengujian t-test juga dijalankan untuk menguji sama ada perbezaan RMSE, MAE atau R^2 antara set latihan dan set ujian adalah signifikan secara statistik. Sebagai contoh, terdapat peningkatan pada RMSE set latihan kepada RMSE set ujian, maka t-test boleh menentukan

sama ada peningkatan ini bermakna atau tidak. P-value pula akan membantu menilai kebarangkalian peningkatan ini. Jika p-value kurang daripada 0.05, ia menunjukkan perbezaan tersebut tidak berlaku secara rawak dan mungkin mencerminkan ketidakstabilan model.

Jadual 2 Perbandingan t-test ketiga-tiga model

Model	RMSE		MAE		R ²	
	t-statistic	p-value	t-statistic	p-value	t-statistic	p-value
Random Forest vs XGBoost	3.6059	0.0226	2.4177	0.0729	-3.5531	0.0237
Random Forest vs CatBoost	9.0658	0.0008	4.3056	0.0126	-9.1508	0.0008
XGBoost vs CatBoost	4.7664	0.0089	2.9798	0.0407	-4.8333	0.0084

Jadual 2 menunjukkan hasil ujian statistik bagi perbandingan prestasi antara model menggunakan t-statistic dan p-value untuk metrik RMSE, MAE, dan R². Jika nilai t-statistik adalah positif maka nilai model di sebelah kiri adalah lebih tinggi. Contohnya, XGBoost vs CatBoost memperoleh RMSE sebanyak 4.7664 bermaksud XGBoost mendapat nilai RMSE yang lebih tinggi berbanding CatBoost. Kesimpulannya, CatBoost adalah model yang terbaik dari segi ketepatan ramalan berdasarkan metrik R² di dalam analisis ini.

Pengujian Kepuasan Pengguna

Pengujian kepuasan pengguna ialah satu proses untuk menilai tahap kepuasan pengguna terhadap sesuatu produk, sistem atau perkhidmatan berdasarkan pengalaman mereka semasa menggunakannya. Antara pengujian yang dapat dijalankan adalah melalui soal selidik, temubual atau pemerhatian untuk mengumpul maklum balas mengenai kemudahan penggunaan, fungsi, reka bentuk dan tahap kepuasan keseluruhan. Tujuan utama pengujian ini adalah untuk mengenal pasti aspek yang memenuhi atau tidak memenuhi jangkaan pengguna supaya penambahbaikan dapat dilakukan. Keputusan daripada ujian ini penting bagi memastikan produk yang dibangunkan mesra pengguna, relevan dan mampu memberikan pengalaman yang positif.

Jadual 3 Tafsiran Skala Skor min

Skor Min	Tafsiran
1.00 – 2.32	Rendah
2.33 – 3.65	Sederhana
3.66 – 5.00	Tinggi

Jadual 3 menunjukkan tafsiran skala skor min bagi pengujian yang dijalankan. Skala 1.00-2.32 menunjukkan tafsiran rendah, 2.33-3.65 menunjukkan tafsiran sederhana dan 3.66-5.00 menunjukkan tafsiran tinggi.

Jadual 4 Skor Min pengujian kepuasan pengguna

No	Item	Min
1	Secara keseluruhan, saya berpuas hati betapa mudahnya menggunakan laman web ini.	4.67
2	Saya rasa laman web ini mudah diguna.	4.67
3	Saya dapat mengakses maklumat dengan cepat menggunakan laman web ini.	4.5
4	Saya percaya saya boleh menjadikan produktif dengan cepat menggunakan laman web ini.	4.17
5	Laman web ini memberikan mesej ralat yang jelas untuk memberitahu saya bagaimana menyelesaikan masalah.	4.67
6	Setiap kali saya melakukan kesilapan menggunakan laman web, saya dapat menyelesaikannya dengan cepat dan mudah.	4.5
7	Maklumat yang diberikan bersama laman web (bantuan dalam talian, paparan mesej di skrin, dokumentasi) yang diberikan adalah jelas.	4.33
8	Laman web ini adalah mudah untuk mencari maklumat yang saya perlukan.	4.67
9	Maklumat adalah berkesan dalam membantu saya menyelesaikan tugas dan hal berkaitan.	4.67
10	Maklumat pada skrin laman web adalah jelas.	4.67
11	Antara muka laman web ini adalah memudahkan.	4.83
12	Saya suka menggunakan antara muka laman web ini.	4.5
13	Laman web ini mempunyai semua fungsi dan keupayaan yang saya jangkakan.	4.67
14	Secara keseluruhan, saya berpuas hati dengan laman web ini.	4.83
	Min Keseluruhan	4.6

Jadual 4 menunjukkan skor min bagi pengujian kepuasan pengguna laman web Voyage. Setiap soalan menunjukkan skor min tafsiran yang tinggi. Soalan 11 dan 14 menerima skor min tertinggi iaitu sebanyak 4.83. Sebaliknya, soalan 4 pula menerima skor min terendah iaitu sebanyak 4.17.

Jadual 5 Rekod Jawapan Soalan Bahagian C

No	Item	Jawapan
1	Pada pendapat anda, apakah kelebihan sistem ini? Sila jelaskan.	interface sangat user-friendly
		Memudahkan pemprosesan data yang rumit
		Mudah digunakan
		Boleh meramal permintaan pada tahun akan datang
		Mesra pengguna
		Sangat memudahkan urusan dan sistem yang mudah difahami

2	Pada pendapat anda, apakah kekurangan sistem ini? Sila jelaskan.	Tiada
		Hanya ada satu features sahaja yang penting
		Masih Baru
		Prestasi sistem agak memuaskan, cuma ada sedikit isu kelancaran pada waktu tertentu
3	Pada pendapat anda, apakah penambahbaikan yang perlu pada sistem ini?	all perfect
		tiada
		tambah button download png untuk graf
		Belum dapat dinyatakan buat masa ini
		Mungkin boleh perbanyakkan lagi function

Jadual 5 merupakan jadual bagi soalan maklum balas bahagian c tentang kelebihan, kekurangan dan cadangan penambahbaikan bagi laman web Voyage. Responden kepada borang maklum balas ini adalah seramai 6 orang daripada 8 orang pengguna. Hasil analisis menunjukkan bahawa 4 orang pengguna sudah berpuas hati dengan laman web Voyage manakala 2 orang pengguna menyatakan bahawa perlu menambah butang muat turun graf yang dihasilkan oleh sistem Voyage dan menambah lebih banyak fungsi kepada sistem ini.

Cadangan Penambahbaikan

Selepas menjalankan kajian, cadangan untuk menambahbaik yang dapat dilakukan pada masa hadapan adalah dengan menjalinkan kerjasama dengan pelbagai syarikat bagi tujuan pengumpulan data yang lebih meluas dan berkualiti untuk meningkatkan ketepatan serta kesesuaian model ramalan yang dibangunkan. Dengan adanya data daripada pelbagai sumber dan jenis industri, model dapat dilatih menggunakan maklumat yang lebih bervariasi sekali gus mengurangkan risiko *bias* terhadap jenis produk atau corak jualan tertentu. Selain itu, pelaksanaan kajian pengesahan luaran dengan menguji model terhadap set data sebenar daripada syarikat yang menjual pelbagai jenis produk juga sangat penting untuk menilai kebolehan model berfungsi dalam persekitaran dunia sebenar.

KESIMPULAN

Secara keseluruhannya, Voyage iaitu sistem ramalan permintaan pelanggan ini telah berjaya dalam mencapai objektifnya dalam memberikan ramalan permintaan pelanggan dengan mengenal pasti corak dan trend data permintaan lampau. Walaupun terdapat beberapa kekangan semasa membangunkan sistem ini, ia berjaya diatasi menggunakan cadangan penyelesaian yang diusulkan oleh pengguna. Dengan adanya sistem Voyage, laman web ini diharapkan supaya memberikan impak yang mapan kepada syarikat di dalam sektor perniagaan dan kepada setiap pengguna laman web ini.

Kekuatan Sistem

Kekuatan sistem Voyage adalah menjana permintaan pelanggan berdasarkan data yang diberikan. Sistem ini direka bentuk untuk membantu perniagaan membuat keputusan yang lebih berinformasi terutamanya dalam aspek pengurusan stok dan inventori. Melalui laman

web Voyage, pengguna hanya perlu memuat naik data yang mengandungi kolum penting seperti 'Units Sold', 'Inventory Level', dan 'Sales' dan sistem akan menghasilkan ramalan permintaan selama satu tahun selepas tarikh terakhir dalam data tersebut. Selain itu, sistem ini turut memastikan data yang dimasukkan adalah konsisten dan sesuai digunakan oleh model ramalan, seterusnya meningkatkan kebolehpercayaan hasil yang dikeluarkan. Dengan ramalan yang diberikan secara terus kepada pengguna, sistem ini bukan sahaja menjimatkan masa dan usaha, malah turut menyokong pengambilan keputusan yang lebih strategik dalam operasi perniagaan harian.

Kelemahan Sistem

Kompleksiti data yang pelbagai telah membawa kepada kekangan sistem ini iaitu masa pemprosesan yang agak panjang terutamanya apabila memproses dataset yang kompleks. Ini juga boleh mengakibatkan berlakunya 'Timeout error' yang akan mengganggu pengalaman pengguna dan menjejaskan kelancaran proses ramalan. Di samping itu, sistem ini sangat bergantung kepada data yang konsisten dari segi struktur dan format. Sekiranya pengguna memuat naik data yang tidak mengikut spesifikasi yang ditetapkan, sistem berkemungkinan gagal berfungsi dengan baik atau menghasilkan ramalan yang kurang tepat. Selain itu, ketiadaan ciri penyimpanan automatik atau butang "save" juga menyukarkan pengguna untuk menyimpan hasil ramalan graf yang dihasilkan secara mudah.

PENGHARGAAN

Alhamdulillah, kesyukuran ke hadrat Ilahi di atas kurniaan-Nya, saya dapat menyiapkan projek ini bagi memenuhi syarat Ijazah Sarjana Sains Komputer dengan Kepujian. Saya amat bersyukur kerana segala cabaran dan dugaan yang dihadapi sepanjang persiapan projek ini dapat diatasi dengan penuh kesabaran dan tekad serta segala pencapaian ini merupakan hasil dari rahmat-Nya.

Selain itu, saya juga menyampaikan ucapan jutaan terima kasih yang tidak terhingga kepada para pensyarah terutamanya pada penyelia saya, Dr. Nor Samsiah Sani, atas bimbingan dan sanggup mengorbankan masa untuk memberikan tunjuk ajar dalam menyiapkan projek akhir tahun ini. Keluasan ilmu dan pengalaman beliau telah banyak membantu saya dalam menyempurnakan projek ini.

Akhir kata, saya ingin mengucapkan terima kasih kepada Fakulti Teknologi dan Sains Maklumat atas sumber dan fasiliti yang telah disediakan. Saya juga dengan rendah dirinya ingin memanjangkan ucapan terima kasih kepada keluarga dan rakan-rakan saya yang telah menjadi tulang belakang sepanjang perjalanan untuk menjayakan projek ini.

RUJUKAN

- Abdelrahman, M.M., Zhan, S. & Chong, A. 2021. A three-tier architecture visual-programming platform for building-lifecycle data management. Department of Building, School of Design and Environment, The National University of Singapore.
- Agil, H., Ahmad, A.L. & Azlan, A.A. 2024. Peranan dan pengaruh kecerdasan buatan (AI) terhadap industri periklanan di Malaysia. *e-Bangi: Journal of Social Sciences & Humanities*, 21(2), Article 25.
- Ahmad, S. & Zulkifli, M.N. 2020. Reka bentuk antara muka pengguna dalam meningkatkan pengalaman pengguna. *E-Journal of Social Sciences (EJOSS)*, 15(2): 45–60.
- Breiman, L. (2001). Random forests. *Machine Learning*, 45(1), 5–32.
- Chen, T. & Guestrin, C. 2016. XGBoost: A scalable tree boosting system. *International Conference on Knowledge Discovery and Data Mining*: 785–794.
- Chin, C.K. & Omar, N. 2020. Bitcoin price prediction based on sentiment of news article and market data with LSTM model. *Asia-Pacific Journal of Information Technology and Multimedia*, 9(1): 1–16.
- Crisostomo, A.S., Chaguile, C.C. & Gustilo, R. 2023. Stock market prediction using linear regression. *Proceedings of the 2023 IEEE 15th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM)*: 1–4.
- Dorogush, A. V., Ershov, V., & Gulin, A. (2018). CatBoost: Gradient boosting with categorical features support. arXiv preprint arXiv:1810.11363.
- Dwitami, N. 2021. Pentingnya analisis data penjualan dalam meningkatkan revenue perusahaan. Sunartha Blog.
- Emioma, C.C. & Edeki, S.O. 2021. Stock price prediction using machine learning on least-squares linear regression basis. *Journal of Physics: Conference Series*, 1734, 012058.
- Graves, A., Mohamed, A.-r. & Hinton, G. 2013. Speech recognition with deep recurrent neural networks. *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*: 6645–6649.
- Hill, R.K. 2016. What an algorithm is *Philosophical Technology*, 29(1): 35–59.
- Ho, W.R. & Abdullah, A.B. 2023. Forecasting natural rubber price in Malaysia by using branching Bi-LSTM. *Fakulti Teknologi & Sains Maklumat, Universiti Kebangsaan Malaysia*.
- Hyndman R.J. & Athanasopoulos G. 2018. *Forecasting: Principles and practice* (2nd ed.). OTexts.
- Idris, A., Bukhari, N.J., Md Yunus, M. & Abdul, S.A. 2023. Penggunaan teknologi digital dalam kalangan usahawan luar bandar dalam era pandemik Covid-19. *Jurnal Pengurusan*, 67: 12.

- Idris, F. N. S., Selamat, N., & Abd Rahman, Y. (2022). A development of the IMKEDA driving institute information management system. *Applied Information Technology and Computer Science*, 3(2), 607–619.
- Istianto, A. F., Hadiana, A. I., & Umbara, F. R. (2023). Prediksi curah hujan menggunakan metode categorical boosting (CatBoost). *JATI (Jurnal Mahasiswa Teknik Informatika)*, 7(4), 2930–2937.
- Jais, N.A.N. & Ahmad, K. 2021. Aplikasi myKesatria. Fakulti Teknologi dan Sains Maklumat, Universiti Kebangsaan Malaysia, PTA-FTSM-2021-033.
- Junaidi, J., Cholisoh, N. & Nurjaman, D. 2024. Pembelajaran mesin untuk ramalan harga rumah menggunakan kaedah regresi linear berganda. *Journal Sensi*.
- Kaneko, Y. & Yada, K. 2016. A deep learning approach for the prediction of retail store sales. *Proceedings of the 2016 IEEE International Conference on Data Mining Workshops (ICDMW)*.
- Loo, Y.L. & Sani, N.S. 2021. Sistem ramalan harga sewa pangsapuri dan kondominium menggunakan model pembelajaran mesin (RENSTIMATE). Fakulti Teknologi & Sains Maklumat, Universiti Kebangsaan Malaysia.
- Mohd Nasir, D.N. & Zainudin, S. 2020. Ramalan kejayaan filem menggunakan kaedah perlombongan data. Fakulti Teknologi & Sains Maklumat, Universiti Kebangsaan Malaysia.
- Morazuki S. R. & Abd Rahman U. N. 2013 Pembangunan perisian sistem pengurusan pangkalan data ujian standard kecergasan fizikal kebangsaan untuk murid sekolah malaysia (SEGAKE). Fakulti Pendidikan, Universiti Teknologi Malaysia.
- Othman, N.H., Baharuddin, S.A. & Abu Hassan, B.R. 2020. Papan iklan luar dan pemilihan sosial: Kajian kes papan iklan di Lebuhraya Elite Expressway E608. *Jurnal Komunikasi: Malaysian Journal of Communication*, 36(4), 258-281.
- Pavlyshenko, B.M. 2019. Machine-learning models for sales time series forecasting. SoftServe, Inc.
- Prokhorenkova, L., Gusev, G., Vorobev, A., Dorogush, A. V., & Gulin, A. (2017). CatBoost: Unbiased boosting with categorical features. *Advances in Neural Information Processing Systems*, 2018-December, 6638–6648.
- Razmi, N.R., Amin, H.M. & Hanawi, S.A. 2019. Sistem Pengurusan Stok (PTA-FTSM-2019-048). Fakulti Teknologi & Sains Maklumat, Universiti Kebangsaan Malaysia.
- Setiyawan, N.F., Priyadi, Y. & Astuti, W. 2023. Development of class diagrams based on use case and sequence diagrams using a text mining approach in SRS Penguin. *IEEE World AI IoT Congress (AIIoT)*: 70–76.
- Shearer, C. 2000. The CRISP-DM model: The new blueprint for data mining. *Journal of Data Warehousing*, 5(4): 13–22.

- Silva, C.V. & Villarroel, R. 2014. JPI UML: UML class and sequence diagrams proposal for aspect-oriented JPI applications. 33rd International Conference of the Chilean Computer Science Society (SCCC).
- Sungkawa, I. & Megasari, R.T. 2011. Penerapan ukuran ketepatan nilai ramalan data deret waktu dalam seleksi model peramalan volume penjualan PT Satriamandiri Citramulia. Mathematics & Statistics Department, School of Computer Science, Binus University.
- Tanenbaum, A.S. 2003. Computer Networks (4th ed.). Pearson Education.
- Uzun, E. & Buluş, H.N. 2017. Object-based flowchart drawing library. International Conference on Computer Science and Engineering (UBMK): 110–115.
- Xiong, R., Yu, L. & Li, Y. 2024. Research on sales forecasting model based on linear regression algorithm. *Proceedings of the 2024 International Conference on Data Science and Network Security (ICDSNS)*: 1–5.
- Xu B. & Lin, C. 2013. An extended practical three-tier architecture based on middleware. 4th International Conference on Software Engineering and Service Science (ICSESS): 243–246.
- Yong, K.T., & Abu Bakar, A. 2020. Model ramalan dalam kebolehpasaran graduan UKM. Fakulti Teknologi & Sains Maklumat, Universiti Kebangsaan Malaysia.

Danish Dean Bin Noor Sallahuddin (A192095)

TS. Dr. Nor Samsiah Binti Sani

Fakulti Teknologi & Sains Maklumat

Universiti Kebangsaan Malaysia