Group Works Report

Kohei Arai

Group Works

- Group communication: Face-to-Face, Discord
- Time for Group Works: 9:00~9:30
- Theme: **Comparing the Most Effective Industries for AI Predictive Analytics** (Python coding is not mandatory)
- Final Presentations from the Groups on 17 July: 20 min. for each presentation and 20 min. for Q/A

Major Industries and Key Applications

Industry	Key Use Cases	Notable Impact
Healthcare	Early disease detection, diagnosis support, readmission risk prediction, resource allocation	Improved patient outcomes, cost reduction, higher quality of care
Finance & Insurance	Credit risk assessment, fraud detection, investment risk forecasting, customer churn prediction	Loss prevention, faster credit decisions, operational cost savings
Retail & E-commerce	Demand forecasting, inventory optimization, personalized marketing, customer churn prediction	Increased sales, reduced inventory loss, higher customer satisfaction
Manufacturing	Predictive maintenance, quality control, production scheduling optimization	Reduced downtime, improved quality, cost savings
Energy & Logistics	Demand forecasting, asset maintenance optimization, route optimization	Stable supply, lower operational costs, efficiency gains

- Healthcare
- AI-driven early diagnosis and patient data analysis have a direct impact on patient health and safety. Predictive analytics reduces readmissions and healthcare costs, making its societal impact especially significant.

- Finance & Insurance
- Real-time fraud detection and credit risk assessment directly prevent losses and improve customer experience. AI enables more accurate and faster decisions than traditional methods.

- Retail & E-commerce
  Demand forecasting and personalized marketing powered by AI lead to higher sales, optimal inventory management, and increased customer loyalty. Some companies report sales increases of 10–15% after implementing predictive analytics.
- Manufacturing
  Using sensor and IoT data for predictive maintenance prevents equipment failures and optimizes production lines, resulting in significant reductions in downtime and maintenance costs.

- Energy & Logistics
- AI helps optimize energy demand and supply, as well as logistics routes, leading to greater efficiency and cost savings. This is particularly impactful in renewable energy management and logistics optimization.

Overall Assessment

- Healthcare, Finance, Retail, and Manufacturing are the industries where AI predictive analytics has the highest impact.
- Healthcare stands out for its societal benefits and cost savings, while Finance, Retail, and Manufacturing excel in profit growth, risk reduction, and operational efficiency.
- The effectiveness of AI in each industry depends on the scale of adoption and the maturity of data utilization, but these sectors are expected to continue leading in AI-powered predictive analytics.

Candidates for your report

- The following projects can be tailored to specific business needs and scaled up using cloud platforms or big data frameworks as required.
- Each example provides a starting point for implementation in Python.

1. Sales Forecasting System

- Description: Predict future sales based on historical sales data, seasonality, and external factors.
- Techniques: Time series analysis, machine learning regression models (ARIMA, **Prophet**, XGBoost).

```
from fbprophet import Prophet
import pandas as pd

df = pd.read_csv('sales_data.csv')
df.rename(columns={'date': 'ds', 'sales': 'y'}, inplace=True)
model = Prophet()
model.fit(df)
future = model.make_future_dataframe(periods=30)
forecast = model.predict(future)
```

2. Customer Churn Prediction

- Description: Identify customers likely to leave a service or subscription using historical usage and engagement data.
- Techniques: Classification algorithms (Logistic Regression, **Random Forest**, XGBoost).

```
from sklearn.ensemble import RandomForestClassifier
import pandas as pd

df = pd.read_csv('customer_data.csv')
X = df.drop('churn', axis=1)
y = df['churn']
model = RandomForestClassifier()
model.fit(X, y)
```

3. Dynamic Pricing Engine

- Description: Automatically adjust product prices based on demand, competition, and other market factors.
- Techniques: Reinforcement learning, **regression models**, demand forecasting.

```
from sklearn.linear_model import LinearRegression
import pandas as pd

df = pd.read_csv('pricing_data.csv')
X = df[['demand', 'competitor_price']]
y = df['price']
model = LinearRegression()
model.fit(X, y)
```

4. Fraud Detection System

- Description: Detect anomalous transactions in financial or e-commerce data.
- Techniques: Anomaly detection, unsupervised learning (Isolation Forest, Autoencoders).

```
from sklearn.ensemble import IsolationForest
import pandas as pd

df = pd.read_csv('transactions.csv')
model = IsolationForest(contamination=0.01)
model.fit(df)
anomalies = model.predict(df)
```

5. Customer Segmentation Using Clustering

- Description: Segment customers into groups for targeted marketing using purchase and demographic data.
- Techniques: Clustering algorithms (K-Means, DBSCAN).

```
from sklearn.cluster import KMeans
import pandas as pd

df = pd.read_csv('customer_features.csv')
model = KMeans(n_clusters=5)
df['segment'] = model.fit_predict(df)
```

7. Supply Chain Demand Optimization

- Description: Predict inventory needs and optimize supply chain logistics using historical and real-time data.
- Techniques: Time series forecasting, optimization algorithms.

```
from statsmodels.tsa.statespace.sarimax import SARIMAX
import pandas as pd

df = pd.read_csv('inventory_data.csv')
model = SARIMAX(df['demand'], order=(1, 1, 1), seasonal_order=(1, 1, 1, 12))
results = model.fit()
forecast = results.forecast(steps=12)
```

8. Real-Time Analytics Dashboard

- Description: Visualize and analyze big data streams in real time for business intelligence.
- Techniques: Stream processing (Apache Kafka, Spark Streaming), dashboarding (Dash, Streamlit).

```
import streamlit as st
import pandas as pd

df = pd.read_csv('live_data.csv')
st.line_chart(df['metric'])
```

9. Recommendation System for E-commerce

- Description: Suggest products to users based on their browsing and purchase history.
- Techniques: Collaborative filtering, content-based filtering, matrix factorization.

```
from sklearn.neighbors import NearestNeighbors
import pandas as pd

df = pd.read_csv('user_item_matrix.csv')
model = NearestNeighbors(metric='cosine', algorithm='brute')
model.fit(df.values)
```

10. Big Data ETL Pipeline

- Description: Build scalable Extract, Transform, Load (ETL) pipelines for large-scale business data processing.
- Techniques: Distributed computing (PySpark, Dask), data engineering.

```
from pyspark.sql import SparkSession

spark = SparkSession.builder.appName('ETL').getOrCreate()
df = spark.read.csv('bigdata.csv', header=True, inferSchema=True)
df_clean = df.dropna()
df_clean.write.parquet('cleaned_data.parquet')
```