

รายงานความก้าวหน้าวิชา Computer Engineering Project Preparation

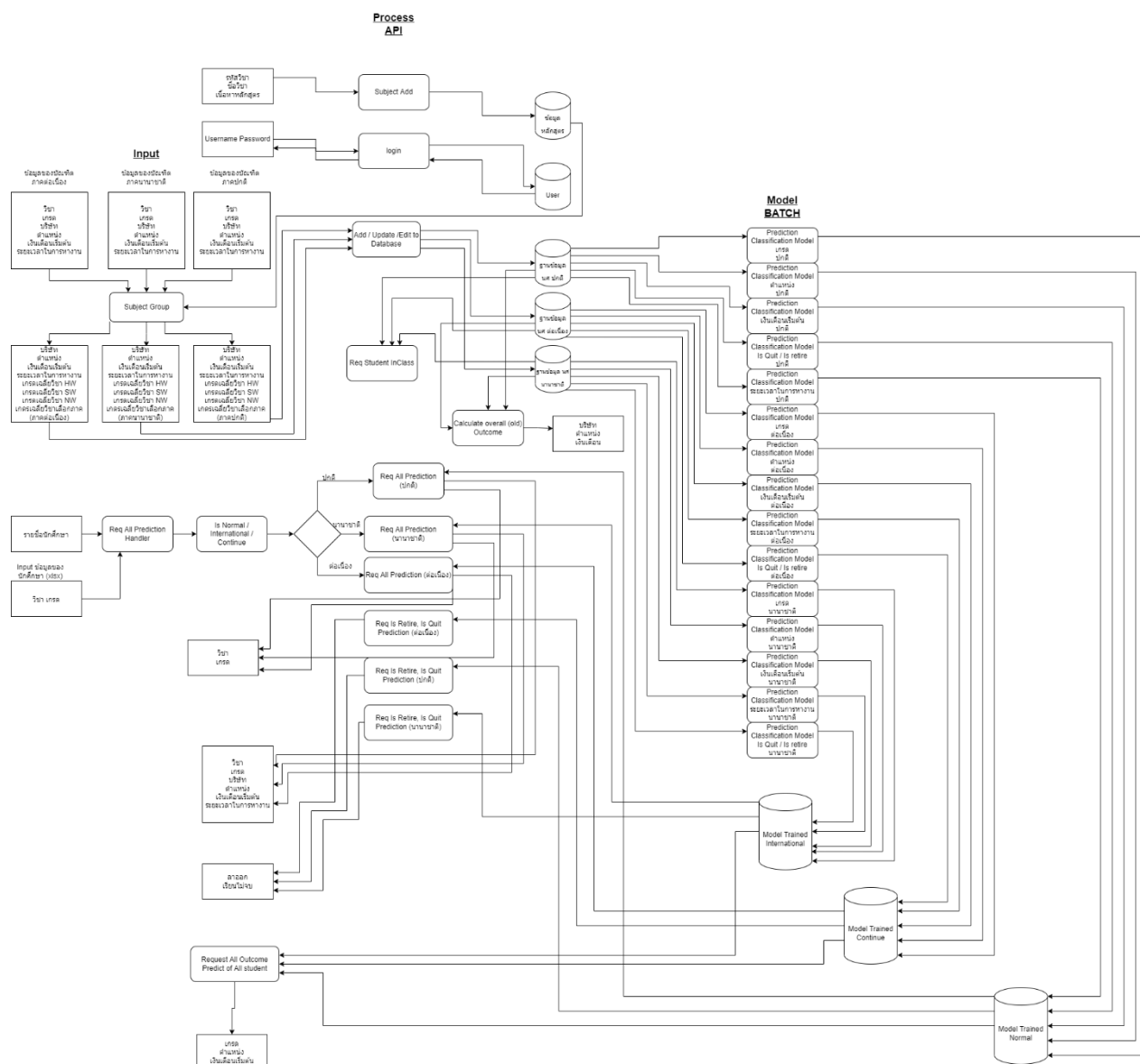
ครั้งที่ 4

1. ชื่อโครงการ (อังกฤษ) ..Recommendation System for CE Curriculum administrators
2. การดำเนินงานมีความก้าวหน้า.....94.....%
3. ความก้าวหน้าระหว่างวันที่ 23 เม.ย. 65 ถึงวันที่ 06 พ.ค. 65
4. รายละเอียดความก้าวหน้า
 - ทำการนัดประชุมกับอาจารย์ที่ปรึกษา เพื่อหาแนวทางการดำเนินการต่อในขั้นตอนต่อไป และแก้ไขปรับเปลี่ยนงานตามความเหมาะสม
 - ปรับแก้เอกสารที่เกี่ยวข้อง และทำการศึกษาเพิ่มเติม
 - ออกแบบ และปรับแก้ไขแผนผัง Data flow ให้มีความสมบูรณ์ ตามรูปที่ 1.1
 - ออกแบบ และปรับแก้ไขแผนผังภาพรวมของระบบ ตามรูปที่ 1.2
 - ได้ศึกษาเอกสารงานวิจัยที่เกี่ยวข้องเพิ่มเติมเกี่ยวกับหัวข้อโครงการ ได้แก่ การพัฒนาระบบแนะนำแหล่งท่องเที่ยวของจังหวัดมหาสารคาม โดยใช้วิธีการกรองข้อมูลแบบผสมผสาน ตามรูปที่ 1.3 และการเปรียบเทียบประสิทธิภาพของแบบจำลองในการพยากรณ์ความสำเร็จการศึกษาของนักเรียนระดับประกาศนียบัตรวิชาชีพ ตามรูปที่ 1.4
 - เพิ่มฟังก์ชันของการทำงานในระบบให้สำหรับคณะกรรมการหลักสูตร
 - เพิ่มเอกสารอ้างอิงใน
 - เพิ่มหน้า UI สำหรับ Web Server ของหน้ากรรมการหลักสูตร
 - เพิ่มฟังก์ชันระบบ Login
 - ลองติดตั้งและศึกษาข้อมูลของเครื่องมือ Metabase เพื่อใช้สำหรับการทำ Data Visualization เริ่มทำการติดตั้งผ่าน java ตามรูปที่ 1.5
 - ทดลองผ่าน local host: 3000 ตามรูปที่ 1.6
 - ทดสอบใช้ function ที่ Metabase เตรียมมาให้ ตามรูปที่ 1.7

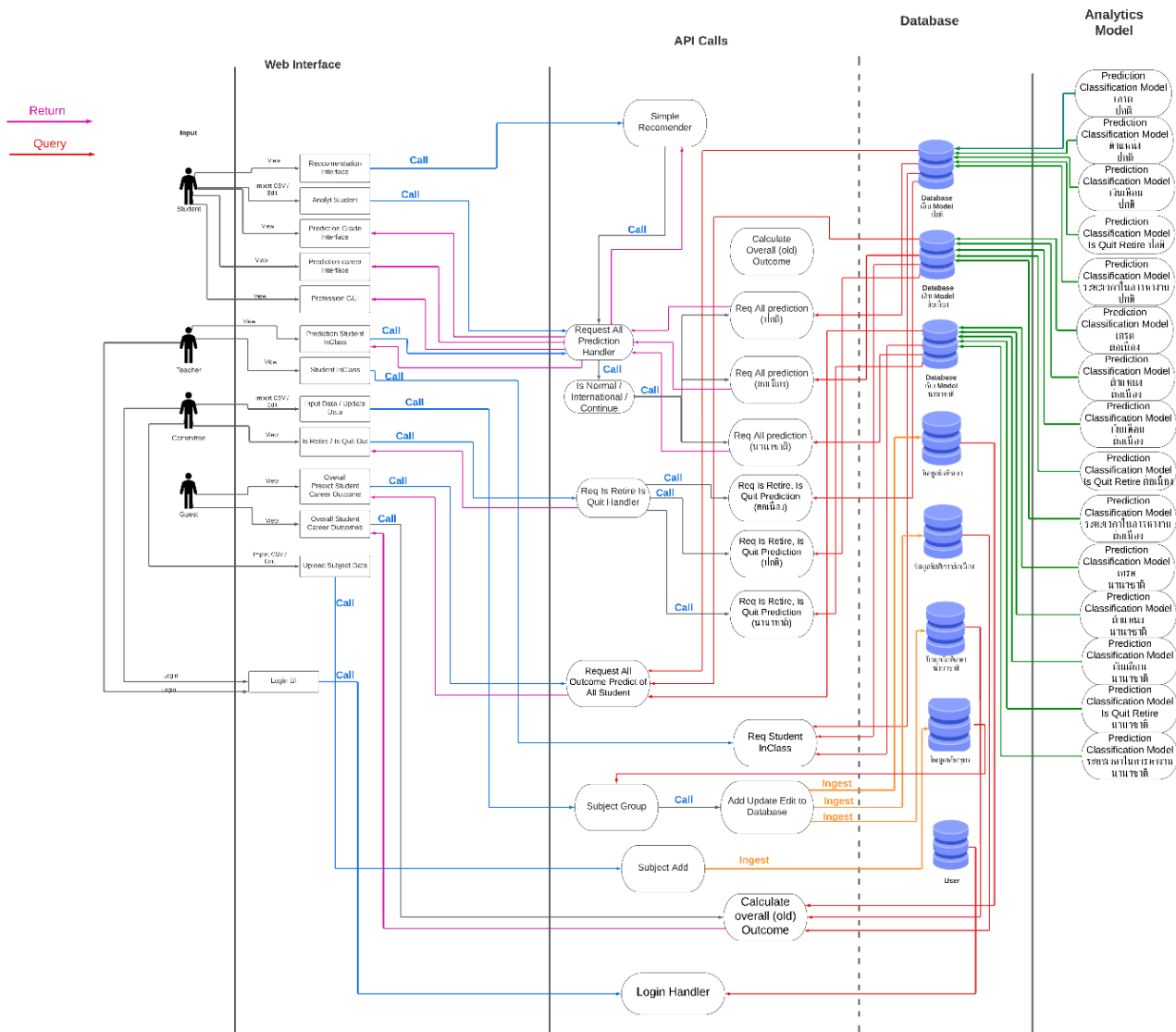
- ทดลองฟังชัน และเขียนผลการศึกษา ได้แก่

ทดลองใช้งาน Pandas : การทดลองใช้ pandas ทดสอบการทำ Extract transform และ load ข้อมูลโดยข้อมูลของรถแต่ละรุ่น โดยมี parameter คือ ชื่อรถ ปีที่ผลิต ราคา และ น้ำมันที่ใช้ โดยการ Demo จะเป็นการ Extract ข้อมูลในหลายรูปแบบ โดยใน lib ของ pandas โดยข้อมูลจะอยู่ในรูปของ CSV JSON และ XML โดยจะทำการ Extract file ชนิดเดียวกันมาก่อนแล้วนำมา join กัน ในรูปของ dataframe หลังจากนั้นจะนำทั้งหมดมารวมกันในรูปของ dataframe เช่นเดียวกัน หลังจากนั้นจะนำมา transform ให้อยู่ในรูปที่สมควร โดยใช้ Function transform แต่ในที่นี้ไม่ได้ทำการ transform อะไร หลังจากนั้นจะนำข้อมูลทั้งหมดไป load ลง file CSV ผ่าน function ของ pandas เพื่อให้ข้อมูลที่อยู่ในรูปของ dataframe อยู่ในรูปแบบที่ต้องการเพื่อนำเอาไปใช้งานในส่วนต่อไป

ทดลองใช้งาน Scikit learn : ตัวอย่างนี้แสดงให้เห็นว่า scikit-learn สามารถใช้ในการจดจำภาพของตัวเลขที่เขียนด้วยลายมือได้อย่างไร ตั้งแต่ 0-9 ชุดข้อมูลตัวเลขประกอบด้วยภาพตัวเลขขนาด 8x8 พิกเซล แอตทริบิวต์ images ของชุดข้อมูลจะจัดเก็บอาร์เรย์ 8x8 ของค่าระดับสีเทาสำหรับแต่ละภาพ ในการใช้ตัวแยกประเภทกับข้อมูลนี้ เราจำเป็นต้องทำให้รูปภาพเรียบ โดยเปลี่ยนอาร์เรย์ 2 มิติของค่าระดับสีเทาจากรูปร่าง (8, 8) ให้เป็นรูปร่าง (64,) ต่อจากนั้น ชุดข้อมูลทั้งหมดจะมีรูปร่าง (n_samples, n_features) โดยที่ n_samples คือจำนวนภาพ และ n_features คือจำนวนพิกเซลทั้งหมดในแต่ละภาพ จากนั้น เราสามารถแบ่งข้อมูลออกเป็นชุดฝึกและทดสอบชุดย่อย และใส่ตัวแยกประเภทเวกเตอร์ที่รองรับบน function train ของ scikit learn



รูปที่ 1.1 แผนผัง Data Flow



รูปที่ 1.2 แผนผังงานรวมของโปรเจก

หัวข้อวิจัย	การพัฒนาระบบแนะนำแหล่งท่องเที่ยวของจังหวัดมหาสารคาม โดยใช้วิธีการกรองข้อมูลแบบผสมผสาน
ผู้ดำเนินการวิจัย	รศ.ดร.สิทธิชัย บุษหมั่น นายทศพล ฤทธิ์เจริญวัตถุ
หน่วยงาน	โปรแกรมวิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยราชภัฏมหาสารคาม
ปี พ.ศ.	2558

บทคัดย่อ

การวิจัยครั้งนี้มีวัตถุประสงค์ เพื่อพัฒนาระบบแนะนำแหล่งท่องเที่ยวของจังหวัดมหาสารคาม โดยใช้วิธีการกรองข้อมูลแบบผสมผสาน ระบบแนะนำนี้ดำเนินการโดยอาศัยกรอบแนวความคิดหลักของการพัฒนาระบบแนะนำ ร่วมกับเทคนิคการกรองแบบผสมผสาน เพื่อให้ได้ผลลัพธ์คือ ระบบแนะนำแหล่งท่องเที่ยวของจังหวัดมหาสารคาม และความพึงพอใจของผู้ใช้ที่มีต่อระบบแนะนำแหล่งท่องเที่ยวของจังหวัดมหาสารคาม ระบบนี้ทำงานในลักษณะของเว็บแอปพลิเคชัน และทำงานบนระบบเครือข่ายอินเทอร์เน็ต การประเมินประสิทธิภาพของระบบแนะนำ โดยผู้เชี่ยวชาญ จำนวน 5 ท่าน พบว่า ประสิทธิภาพของระบบอยู่ในระดับดี ทั้งโดยรวมและรายด้านและทำการประเมินความพึงพอใจของผู้ใช้งาน พบว่า ผู้ใช้งานมีความพึงพอใจในการใช้งานระบบแนะนำแหล่งท่องเที่ยวของจังหวัดมหาสารคาม ในภาพรวมและรายด้าน อยู่ในระดับมาก แสดงให้เห็นว่าระบบแนะนำแหล่งท่องเที่ยวของจังหวัดมหาสารคาม โดยใช้วิธีการกรองข้อมูลแบบผสมผสานสามารถให้คำแนะนำกับผู้ใช้ได้อย่างมีประสิทธิภาพ ใช้งานที่ง่าย สะดวกรวดเร็ว สามารถนำไปประยุกต์ใช้งานได้อย่างมีความเหมาะสม

รูปที่ 1.3 การพัฒนาระบบแนะนำแหล่งท่องเที่ยวของจังหวัดมหาสารคาม โดยใช้วิธีการกรองข้อมูลแบบผสมผสาน

การเปรียบเทียบประสิทธิภาพของแบบจำลองในการพยากรณ์ความสำเร็จการศึกษาของนักเรียนระดับประกาศนียบัตรวิชาชีพ

The Comparison of Performance Models for Predicting Students Success in Vocation Education

พัฒนพงษ์ ดลรัตน์¹, จารี ทองคำ²

Pattanaphong Donrat¹, Jaree Thongkam²

Received: 27 March 2017 ; Accepted: 23 November 2017

บทคัดย่อ

ปัจจุบันประเทศไทยนั้นอยู่สภาวะขาดแคลนแรงงานฝีมือ กระทรวงศึกษาธิการจึงได้มีนโยบายในปีพุทธศักราช 2554 การเปลี่ยนแปลงสัดส่วนของผู้เรียนอาชีวศึกษาต่อผู้เรียนสายสามัญจากเดิม 40:60 เป็น 60:40 ภายในปีพุทธศักราช 2561 ซึ่งเป็นปีสิ้นสุดของการปฏิรูปการศึกษาทศวรรษที่ 2 ซึ่งสถานศึกษาอาชีวศึกษาเอกชนก็เป็นหน่วยงานหนึ่งที่ได้ดำเนินการสนองนโยบายของรัฐบาลเพื่อแก้ไขปัญหาสภาวะขาดแคลนแรงงานฝีมือ เพิ่มจำนวนผู้สำเร็จการศึกษา พัฒนาการศึกษให้เกิดประสิทธิภาพสูงสุด ดังนั้นงานวิจัยนี้จึงมีจุดประสงค์เพื่อเปรียบเทียบประสิทธิภาพของแบบจำลองในการพยากรณ์ความสำเร็จการศึกษาของนักเรียนระดับประกาศนียบัตรวิชาชีพ งานวิจัยนี้ได้ใช้ 6 เทคนิคที่มีประสิทธิภาพในการสร้างแบบจำลอง คือ C4.5, Random Forest, Random Tree, Reduced Error Pruning (REP Tree), k-Nearest Neighbors (k-NN) และ Support Vector Machine (SVM) และวัดประสิทธิภาพการพยากรณ์ของแบบจำลองด้วยค่าความถูกต้อง (Accuracy) ค่าความแม่นยำ (Precision) และค่าความระลึก (Recall) จากการศึกษาพบว่า แบบจำลอง C4.5 มีประสิทธิภาพในการพยากรณ์ความสำเร็จการศึกษาของนักเรียนระดับประกาศนียบัตรวิชาชีพมากที่สุดถึง 95.36%

คำสำคัญ แบบจำลองการพยากรณ์ความสำเร็จการศึกษา ประกาศนียบัตรวิชาชีพ ต้นไม้การตัดสินใจ เหมืองข้อมูล

Abstract

Currently, Thailand has a shortage of skilled labor. Ministry of Education policy requires that between B.E 2554 and 2561, the proportion of vocational students compare with the common line should increase from 40:60 to 60:40, the year of the second decade of education reform education. The private Vocational College was working to meet the government's policy to solve the shortage of skilled labor, increase the number of graduates, and develop the best performance. This research compares performance of models for predicting student success in vocation education. This research using six powerful techniques in modeling is C4.5, Random Forest, Random Tree, Reduced Error Pruning (REP Tree), k-Nearest Neighbors (k-NN) and Support Vector Machine (SVM) A set of test data performance measurement and prediction of models with accuracy, precision and recall found that the C4.5 model was effective in predicting educational success. Most vocational certificate levels reached 95.36%

Keywords: Models predicting graduation, Vocational Education Certificate, Decision tree, Data mining

¹ นิสิต, สาขาเทคโนโลยีสารสนเทศ คณะวิทยาการสารสนเทศ มหาวิทยาลัยมหาสารคาม, มหาสารคาม, 44150.

² ผู้ช่วยศาสตราจารย์, อาจารย์ที่ปรึกษา หน่วยวิจัยสารสนเทศประยุกต์ คณะวิทยาการสารสนเทศ มหาวิทยาลัยมหาสารคาม, มหาสารคาม, 44150.

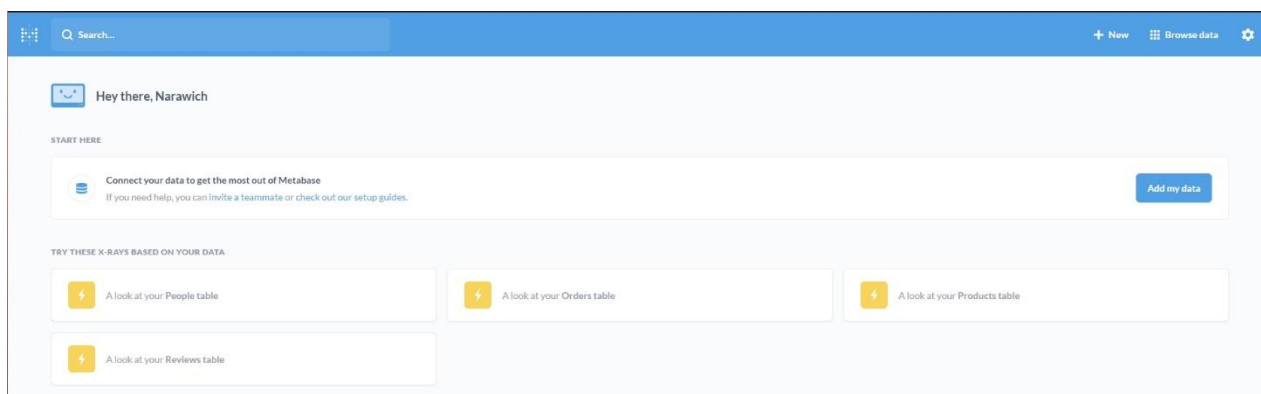
รูปที่ 1.4 การเปรียบเทียบประสิทธิภาพของแบบจำลองในการพยากรณ์ความสำเร็จการศึกษาของนักเรียนระดับประกาศนียบัตรวิชาชีพ

```

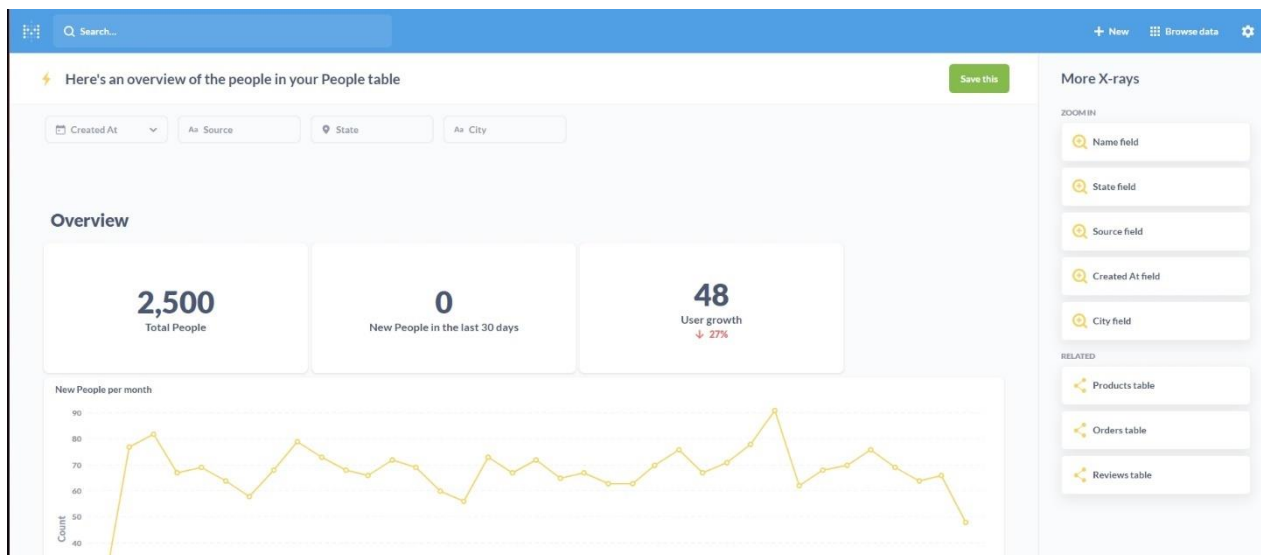
2022-05-05 15:54:05,615 INFO metabase.events :: Starting events listener: :metabase.events.revision/Revisions
2022-05-05 15:54:05,616 INFO metabase.events :: Starting events listener: :metabase.events.view-log/ViewLog
2022-05-05 15:54:05,768 INFO metabase.task :: Initializing task SyncDatabases
2022-05-05 15:54:05,825 INFO metabase.task :: Initializing task CheckForNewVersions
2022-05-05 15:54:05,849 INFO metabase.task :: Initializing task SendAnonymousUsageStats
2022-05-05 15:54:05,854 INFO task.send-pulses :: Sending scheduled pulses...
2022-05-05 15:54:05,869 INFO metabase.task :: Initializing task SendAbandonmentEmails
2022-05-05 15:54:05,876 INFO metabase.task :: Initializing task SendPulses
2022-05-05 15:54:05,885 INFO metabase.task :: Initializing task SendFollowUpEmails
2022-05-05 15:54:05,903 INFO metabase.task :: Initializing task TaskHistoryCleanup
2022-05-05 15:54:05,920 INFO driver.impl :: Initializing driver :sql...
2022-05-05 15:54:05,921 INFO driver.impl :: Initializing driver :sql-jdbc...
2022-05-05 15:54:05,921 INFO driver.impl :: Initializing driver :h2...
2022-05-05 15:54:05,930 INFO metabase.core :: Metabase Initialization COMPLETE

```

รูปที่ 1.5 เครื่องมือ Metabase เพื่อใช้สำหรับการทำ Data Visualization



รูปที่ 1.6 ทดลองผ่าน local host: 3000



รูปที่ 1.7 ทดสอบใช้ function ที่ Metabase เตรียมมาให้


```

import glob                                # this module helps in selecting files
import pandas as pd                        # this module helps in processing CSV files
import xml.etree.ElementTree as ET        # this module helps in processing XML files.
from datetime import datetime

tmpfile = "dealership_temp.tmp"            # file used to store all extracted data
logfile = "dealership_logfile.txt"         # all event logs will be stored in this file
targetfile = "dealership_transformed_data.csv" # file where transformed data is stored

# Add the CSV extract function below
def extract_from_csv(file_to_process):
    dataframe = pd.read_csv(file_to_process)
    return dataframe

# Add the JSON extract function below
def extract_from_json(file_to_process):
    dataframe = pd.read_json(file_to_process, lines=True)
    return dataframe

# Add the XML extract function below, it is the same as the xml extract function above but the column names need to be renamed.
def extract_from_xml(file_to_process):
    dataframe = pd.DataFrame(columns=["car_model", "year_of_manufacture", "price", "fuel"])
    tree = ET.parse(file_to_process)
    root = tree.getroot()
    for car in root:
        car_model = car.find("car_model").text
        year_of_manufacture = int(car.find("year_of_manufacture").text)
        price = float(car.find("price").text)
        fuel = car.find("fuel").text
        dataframe = dataframe.append({"car_model":car_model, "year_of_manufacture":year_of_manufacture, "price":price, "fuel":fuel}, ignore_index=True)
    return dataframe

def extract():
    extracted_data = pd.DataFrame(columns=['car_model', 'year_of_manufacture', 'price', 'fuel']) # create an empty data frame to hold extracted data

    #process all csv files
    for csvfile in glob.glob("dealership_data/*.csv"):
        extracted_data = extracted_data.append(extract_from_csv(csvfile), ignore_index=True)

```

รูปที่ 1.8 Pandas ส่วนของการ Import Library และ Function ที่สำคัญ

```

,car_model,year_of_manufacture,price,fuel
0,ritz,2014,5000.0,Petrol
1,sx4,2013,7089.55223880597,Diesel
2,ciaz,2017,10820.89552238806,Petrol
3,wagon r,2011,4253.731343283582,Petrol
4,swift,2014,6865.671641791044,Diesel
5,vitara brezza,2018,13805.970149253732,Diesel
6,ciaz,2015,10074.626865671642,Petrol
7,s cross,2015,9701.492537313432,Diesel
8,ciaz,2016,13059.701492537311,Diesel
9,ciaz,2015,11119.402985074626,Diesel
10,alto 800,2017,4253.731343283582,Petrol
11,ciaz,2015,10223.880597014926,Diesel
12,ciaz,2015,11194.029850746268,Petrol
13,ertiga,2015,9104.477611940298,Petrol
14,dzire,2009,3358.208955223881,Petrol
15,ertiga,2016,11567.164179104477,Diesel
16,ertiga,2015,10820.89552238806,Diesel
17,ertiga,2016,11567.164179104477,Diesel
18,wagon r,2015,4850.746268656716,CNG
19,sx4,2010,3955.223880597015,Petrol
20,alto k10,2016,4253.731343283582,Petrol
21,ignis,2017,7313.432835820897,Petrol
22,sx4,2011,6567.164179104479,Petrol
23,alto k10,2014,3731.3432835820895,Petrol
24,wagon r,2013,4328.358208955224,Petrol
25,swift,2011,4477.611940298508,Petrol
26,swift,2013,6194.029850746269,Petrol
27,swift,2017,8955.223880597016,Petrol
28,alto k10,2010,2910.4477611940297,Petrol
29,ciaz,2015,11119.402985074626,Diesel
30,ritz,2012,4626.8656716418,Diesel
31,ritz,2011,3507.4626865672,Petrol
32,swift,2014,7388.0597014925,Diesel
33,ertiga,2014,8955.223880597,Diesel

```

รูปที่ 1.9 Pandas ส่วนของผลลัพธ์ของการ load file to CSV


```

import matplotlib.pyplot as plt

# Import datasets, classifiers and performance metrics
from sklearn import datasets, svm, metrics
from sklearn.model_selection import train_test_split

digits = datasets.load_digits()

_, axes = plt.subplots(nrows=1, ncols=4, figsize=(10, 3))
for ax, image, label in zip(axes, digits.images, digits.target):
    ax.set_axis_off()
    ax.imshow(image, cmap=plt.cm.gray_r, interpolation="nearest")
    ax.set_title("Training: %i" % label)

# flatten the images
n_samples = len(digits.images)
data = digits.images.reshape((n_samples, -1))

# Create a classifier: a support vector classifier
clf = svm.SVC(gamma=0.001)

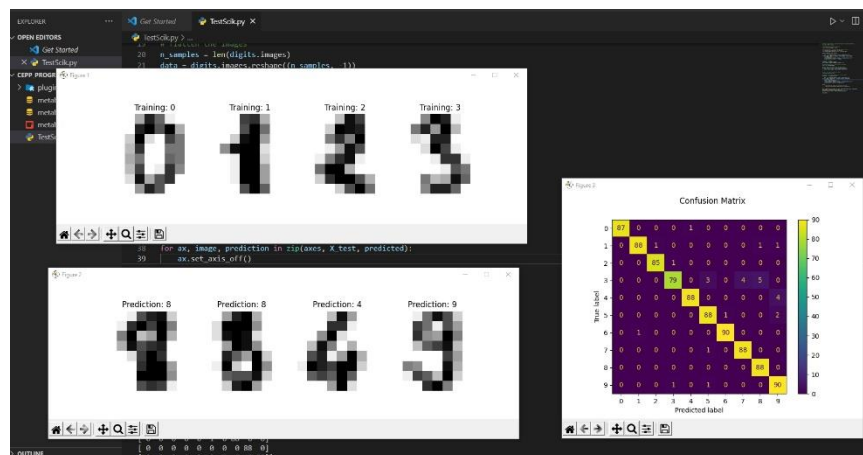
# Split data into 50% train and 50% test subsets
X_train, X_test, y_train, y_test = train_test_split(
    data, digits.target, test_size=0.5, shuffle=False
)

# Learn the digits on the train subset
clf.fit(X_train, y_train)

# Predict the value of the digit on the test subset
predicted = clf.predict(X_test)

```

รูปที่ 1.10 Scikit Learn ส่วนของการ Import Library และ Function ที่สำคัญ



รูปที่ 1.11 Scikit Learn ส่วนของผลลัพธ์

5. ปัญหาที่เกิดขึ้นและแนวทางการแก้ไข

Problem No. 10

พบปัญหาในการรายงานความก้าวหน้า ครั้งที่ 3

สถานะ ☐ กำลังดำเนินการ ☒ แก้ไขสำเร็จ

รายละเอียดปัญหา

- ยังทดลองการใช้ Sklearn ไม่เพียงพอ เนื่องจากเวลา และภาระงาน

แนวทางแก้ไข/การแก้ไข

- แบ่งเวลาการทำงาน
- ศึกษาการใช้ Sklearn เพิ่มเติม
- ทดลองใช้งาน

Problem No. 11

พบปัญหาในการรายงานความก้าวหน้า ครั้งที่ 4

สถานะ ☐ กำลังดำเนินการ ☒ แก้ไขสำเร็จ

รายละเอียดปัญหา

- ฟังก์ชันไม่พอต่อการใช้งาน

แนวทางแก้ไข/การแก้ไข

- คุยปรึกษากับอาจารย์
- เพิ่มฟังก์ชันที่เกี่ยวข้องเพิ่มเติม

Problem No. 12

พบปัญหาในการรายงานความก้าวหน้า ครั้งที่ 4

สถานะ ☐ กำลังดำเนินการ ☒ แก้ไขสำเร็จ

รายละเอียดปัญหา

- Diagram ต่างๆยังไม่สอดคล้องกัน

แนวทางแก้ไข/การแก้ไข

- ปรับแก้ Diagram ต่างๆ ตรวจสอบและวิเคราะห์

Problem No. 13

พบปัญหาในการรายงานความก้าวหน้า ครั้งที่ 4

สถานะ ☐ กำลังดำเนินการ ☒ แก้ไขสำเร็จ

รายละเอียดปัญหา

- Diagram ยังอ่านไม่ค่อยเข้าใจ มีบางจุดต่างๆที่ทำให้ผู้อ่านเข้าใจยาก

แนวทางแก้ไข/การแก้ไข

- ปรึกษาอาจารย์
- ปรับแก้ Diagram ต่างๆ ตรวจสอบและวิเคราะห์

6. สิ่งที่จะดำเนินการต่อไป

- ทำการศึกษาเพิ่มเติมและติดตั้งเครื่องมือ และระบบต่าง ๆ เพื่อเตรียมพร้อมสำหรับการทำโครงงาน
- ทดลองระบบ และเขียนผลการศึกษา อ้างอิง
- ศึกษางานวิจัยที่เกี่ยวข้องเพิ่มเติม
- ลองใช้ Sklearn เพิ่มเติม
- หาข้อมูลเพิ่มเติมเพื่อให้พร้อมทั้งโปรเจก และเอกสารที่เกี่ยวข้อง
- ปรับแก้ไขรายงานตามความเหมาะสม
- ปรับแก้การจัดวางของ Model ให้อ่านง่ายขึ้น
- ปรับแก้ Site Map