Technical documentation

|  |  |  |
| --- | --- | --- |
| เทคโนโลยี หรือ ซอฟแวร์ | ฟังก์ชัน | รายละเอียด |
| VSCODE | IDE | โปรแกรมประยุกต์ซอฟต์แวร์ที่ช่วยให้โปรแกรมเมอร์พัฒนาซอฟต์แวร์ได้อย่างมีประสิทธิภาพ ในที่นี่ใช้ในการทำทุกๆอย่าง |
| XAMPP | Frameworks & Library | XAMPP Version: 7.4.29 |
| PYTHON | Languages | ใช้สำหรับการ Coding ทั้งทางด้าน AI |
| Javascript | Languages | ภาษาที่ใช้ในการทำ Backend ของเว็บไซต์ |
| react | Languages | ภาษาที่ใช้ในการตกแต่งหน้าเว็บให้มีความสวยงามมากยิ่งขึ้น |
| jupyter | Frameworks & Library | เครื่องมือที่ใช้เขียนcodeในการที่เราจะสร้าง machine learning model |
| conda | Frameworks & Library | Conda เป็นที่ setting Environment library ที่จำเป็นใน Python |

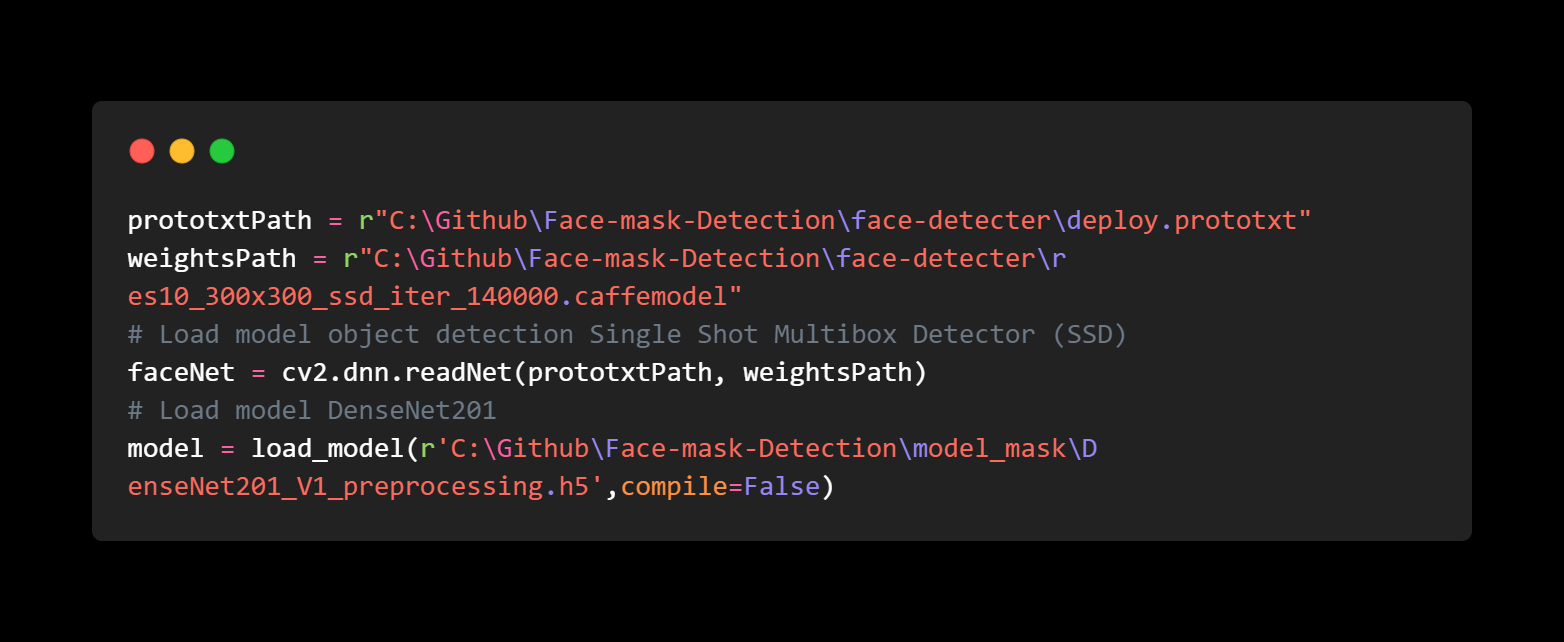
การ setup Environment สำหรับโปรเจค

1. การลง library ที่ใช้ในการทำโปรเจค
2. pip install opencv-python
3. pip install numpy
4. pip install sqlite3
5. pip install pymysql
6. pip install argparse
7. pip install tensorflow
8. pip install imutils
9. pip install mysql-connector-python
10. pip install centroidtracker
11. pip install pandas
12. ทำการ clone ตัว Repository
13. การนำฐานข้อมูลลงในตัว XAMPP
14. ทำการสร้างฐานข้อมูลใน http://localhost/phpmyadmin/ ตั้งชื่อ database ว่า mit ใช้ utf8\_general\_ci
15. Import: Face-mask-Detection\database\mit.sql
16. ทำการเปลี่ยน path ตามความต้องการใน File “Run\_camera.py”
17. Run file “Run\_camera.py”

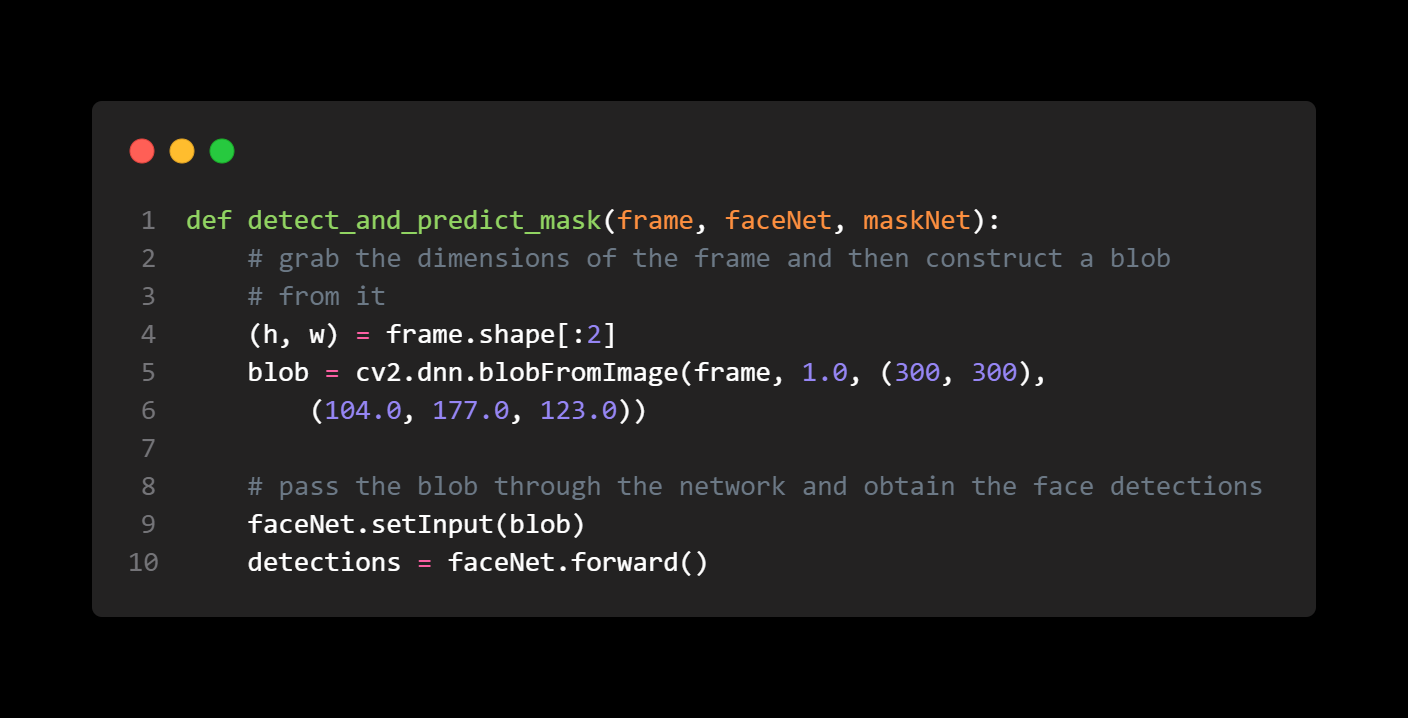
การทำงานของโค้ดในไฟล์ “Run\_camera.py” จะแบ่งเป็น 3 ส่วนคือ

1. การตีกรอบเพื่อตรวจจับใบหน้า
2. การตรวจจับบุคคลใส่หน้ากากอนามัย
3. การนำข้อมูลที่ได้จาก Model เข้าฐานข้อมูล

การตีกรอบเพื่อตรวจจับใบหน้า



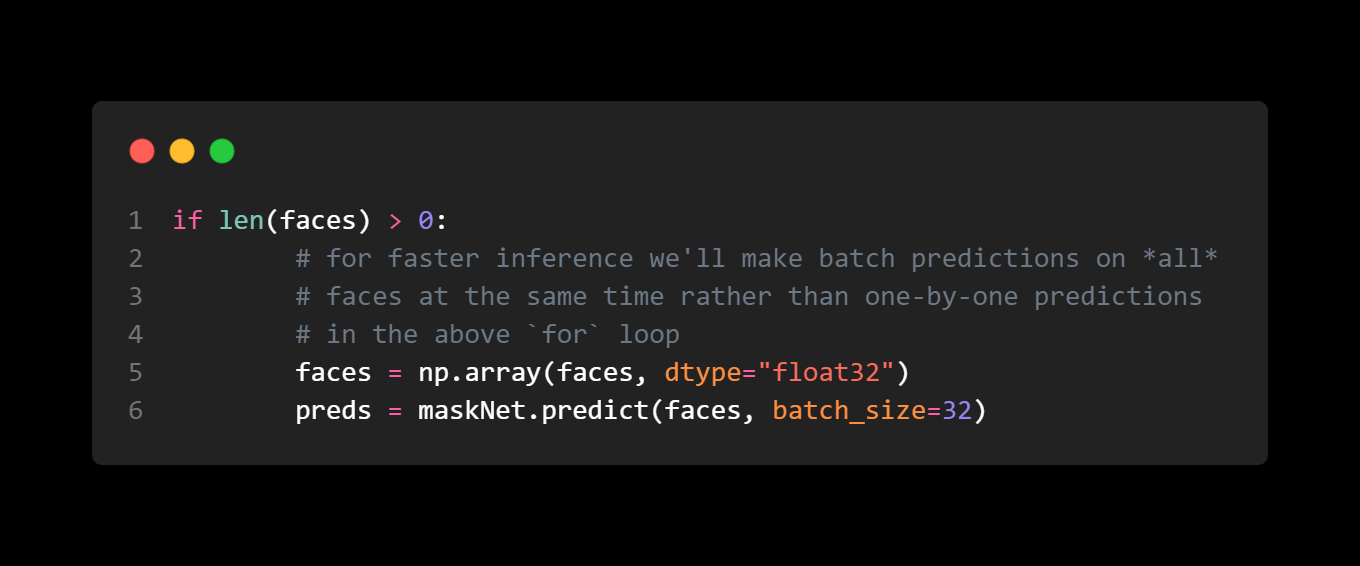
* การตรวจจับใบหน้าจะใช้ deploy.prototxt และ res10\_300x300\_ssd\_iter\_140000.caffemodel เป็นโมเดลตรวจจับวัตถุแบบ Single Shot Multibox Detector (SSD) ซึ่งเป็นโมเดลที่ถูกพัฒนาขึ้นโดย Google เพื่อตรวจจับวัตถุในภาพ โมเดลนี้สามารถใช้สำหรับตรวจจับหลายประเภทของวัตถุได้ เช่น ใบหน้า รถยนต์ และสัตว์ เป็นต้น
* นำไฟล์ทั้งสองมาทำการอ่านด้วย Method ของ OpenCV เพื่อสร้างตัวแปรที่เก็บโมเดลการตรวจจับใบหน้า โดย
* faceNet คือ Model ที่ใช้สำหรับการตรวจจับใบหน้า
* Model คือ Model ที่ใช้สำหรับการตรวจจับหน้ากากอนามัย

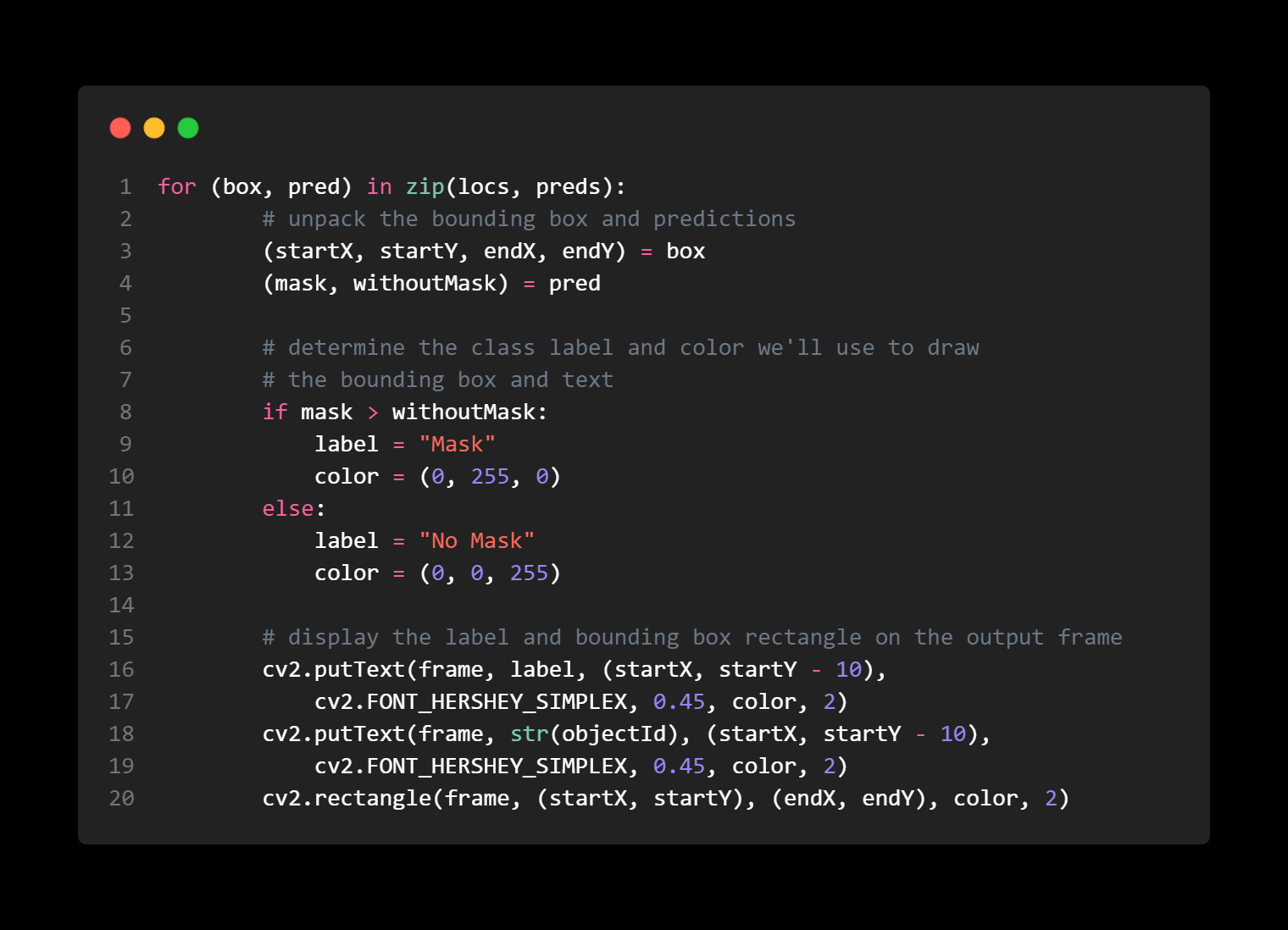


* ใน Function detect\_and\_predict\_mask ทำการเรียกใช้ Model faceNet เพื่อทำการตีกรอบใบหน้า



* เมื่อ Model ตรวจพบใบหน้าให้ทำการ preprocessing บริเวณที่ตรวจพบใบหน้าเพื่อเตรียมพร้อมสำหรับการ predicted ของ Model ที่ตรวจจับหน้ากากอนามัย ลงในตัวแปร faces และ rerturn ค่า locs ที่ระบุตำปหน่งของใบหน้าที่พบบน Frame video

การตรวจจับบุคคลใส่หน้ากากอนามัย

* เมื่อพบใบหน้าบน Frame video ให้ทำการเรียกโมเดล maskNet เพื่อทำการ predict ว่าใบหน้าที่พบใส่แมสหรือไม่ใส่แมส แล้ว return preds ผลที่ predicted ได้
* เมื่อ Model ตรวจจับบุคคลใส่แมสได้แล้ว ระบบจะทำการตีกรอบบริเวณใบหน้าตามค่าของ locs ที่ได้จากการตรวจจับใบหน้า และทำการ Label ตามผล predicted ของโมเดล

การนำข้อมูลที่ได้จากการทำนายลง Database

Text

Description automatically generated

Text

Description automatically generated

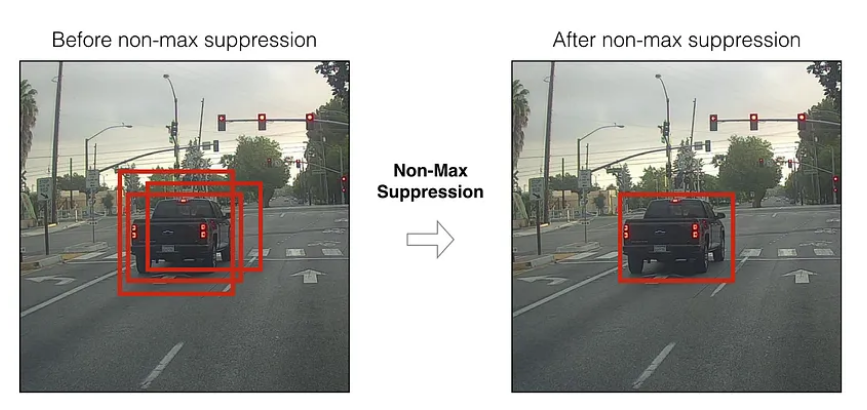
* ใช้ centroidtracker.py ช่วยในการ Track จุดศูนย์กลางของ Object และกำหนด Object Id ในแต่ละ Object

Text

Description automatically generated

* กำหนดเส้นเพื่อตรวจสอบการข้ามก่อนนำข้อมูลลง Database
* ตรวจสอบการข้ามเส้นโดยใช้ จุดซ้ายสุดของกรอบหน้าเทียบกับตำแหน่งเส้น และทำการ Insert ลง Database โดย Throw Exception เพื่อให้ระบบยังทำงานต่อไปไดGraphical user interface, text

  Description automatically generated



* Non-Maximum Suppression อัลกอริทึมสำหรับการนำเอาผลลัพธ์ที่เป็น True Positive ทั้งหมดแล้วลบ Duplicate ออกให้เหลือเป็นผลลัพธ์ ในกรณีที่มีการจับ Object เดิมหลายครั้ง

Dataset

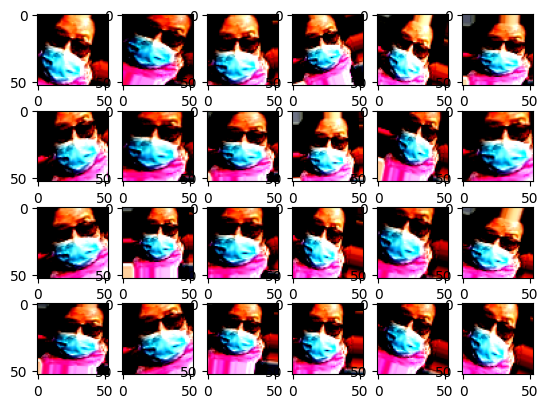
ในส่วนของ Dataset ที่ใช้สำหรับการ Train model จะใช้ Dataset จากเว็บ Kaggle

<https://www.kaggle.com/datasets/ashishjangra27/face-mask-12k-images-dataset>

มีจำนวนทั้งหมด 11,792 ภาพ โดยแบ่งสัดส่วนของ Dataset ได้ดังนี้

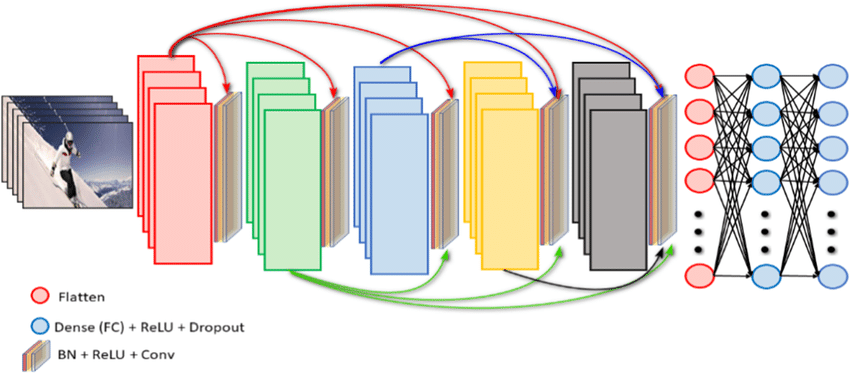
|  |  |  |
| --- | --- | --- |
|  | With Mask | Without Mask |
| Train | 5000 | 5000 |
| Test | 483 | 509 |
| validation | 400 | 400 |

Preprocessing and Augmentation data



ใช้วิธีการ Preprocess ทั้งหมด 7 วิธีดังนี้ Shearing, contrasting, flipping horizontally, rotating, zooming and blurring โดยขนาดรูปภาพที่ใช้ในการ Train model คือ 128x128

Model

DenseNet 201

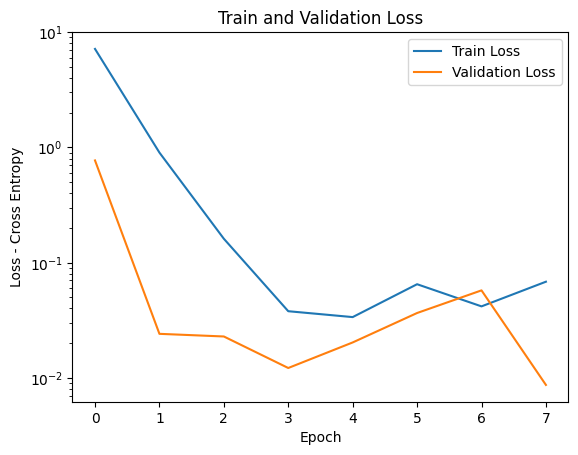
DenseNet201 เป็นโมเดลโครงข่ายประสาทเทียม (neural network) ที่ถูกพัฒนาขึ้นโดยทีมวิจัยของ Facebook AI Research (FAIR) โดยมีจำนวนชั้นประมวลผลทั้งหมด 201 ชั้น (hence the name DenseNet201) โดยใช้โครงสร้างที่เรียกว่า "Densely Connected Convolutional Networks" (DenseNets) ที่เป็นแบบ CNN (Convolutional Neural Networks) ที่มีรูปแบบการเชื่อมต่อชั้นใหม่ ๆ แบบแต่ละชั้นมีการเชื่อมต่อกับชั้นก่อนหน้า และยังสามารถใช้พารามิเตอร์ที่น้อยกว่าเมื่อเทียบกับโมเดลอื่น ๆ ที่มีประสิทธิภาพใกล้เคียงกัน

Text

Description automatically generated with medium confidence

โดย Parameter ที่ปรับจะมี Batch size = 32 และ Epoch = 8

Chart, line chart

Description automatically generatedEvaluate Model

Accuracy: 97.88%

