

รายงานสรุปผลการศึกษา

วิชา 2110391 เอกซ์ตศึกษาทางวิศวกรรมคอมพิวเตอร์ 3

(Individual Study in Computer Engineering III)

เรื่อง

Tuberculosis Chest X-rays

จัดทำโดย

นายพีรณัฐ ชีระวัฒนชัย

รหัสประจำตัวนิสิต 6231343821

เสนอ

อาจารย์ เอกพล ช่างสุวนิช

รายงานฉบับนี้เป็นส่วนหนึ่งของการศึกษารายวิชาเอกซ์ตศึกษาทางวิศวกรรมคอมพิวเตอร์ 3

จุฬาลงกรณ์มหาวิทยาลัย

ภาคเรียนที่ 1 ปีการศึกษา 2564

คำนำ

รายงานฉบับนี้เป็นส่วนหนึ่งของการศึกษารายวิชาเอกศึกษาทางวิศวกรรมคอมพิวเตอร์ 3 (Individual Study in Computer Engineering III) จัดทำขึ้นเพื่อสรุปสิ่งที่ได้เรียนรู้จากวิชานี้ โดยได้ร่วมทำโปรเจกต์เรื่อง Machine Learning เกี่ยวกับการทำนายการเป็นวัณโรคจาก Chest X-ray image ด้วยการใช้ Deep Learning โดยในรายงานเล่มนี้ประกอบด้วยเนื้อหา ดังนี้

1. ความเป็นมาและวัตถุประสงค์
2. ขั้นตอนการทำงาน สิ่งที่ทำ
3. ผลลัพธ์ของงาน
4. อุปสรรคที่เกิดขึ้น วิธีแก้ปัญหา และสิ่งที่ได้รับจาก Individual Study นี้

ทั้งนี้ข้าพเจ้าหวังเป็นอย่างยิ่งว่ารายงานเล่มนี้จะเป็นประโยชน์ต่อผู้ที่ได้มาศึกษาเป็นอย่างดี และหากมีข้อผิดพลาดประการใดขออภัยมา ณ โอกาสนี้ด้วย

นายพีรณัฐ ธีระวัฒนชัย

ผู้จัดทำ

สารบัญ

รายละเอียด	หน้า
1. ความเป็นมาและวัตถุประสงค์	4
2. ขั้นตอนการทำงาน สิ่งที่ทำ	4
3. ผลลัพธ์ของงาน	5
4. อุปสรรคที่เกิดขึ้น วิธีแก้ปัญหา และสิ่งที่ได้รับจาก Individual Study นี้	9

1. ความเป็นมาและวัตถุประสงค์

- ความเป็นมา

ในปัจจุบันการตรวจหาความผิดปกติที่บริเวณต่างๆภายในร่างกายมนุษย์ เช่น ส่วนนอก ส่วนหัวใจ ส่วนปอด เป็นต้น อาจจะต้องใช้ค่าใช้จ่ายเป็นจำนวนมากและอาจต้องใช้เวลาานพอสมควร แต่ในบางครั้งเราก็สามารถใช้เทคโนโลยีมาประยุกต์ใช้เพื่อทำนายว่าบริเวณเหล่านั้นมีสิ่งผิดปกติอยู่หรือไม่ เพื่อนำไปใช้วิเคราะห์ทางการแพทย์ต่อไป

โดยในงานนี้ผมได้รับมอบหมายให้นำข้อมูลภาพ Chest X-ray มาใช้กระบวนการ Machine Learning ด้วยการนำ Convolutional Neural Network (CNN) เพื่อทำนายการเป็นวัณโรค โดยเป้าหมายหลัก คือ ต้องการให้ Model ที่ Train ออกมามีความแม่นยำมากที่สุดกับ Testing set (ได้ model ที่ไม่ Overfit เกินไป) โดยในระหว่างการทำได้พบว่าเราควรนำภาพที่ได้มา Normalized ก่อน ด้วยการหาค่าเฉลี่ยแล้วหารด้วย Standard deviation เพื่อให้ได้ model ที่มีความแม่นยำมากขึ้น

- วัตถุประสงค์

1. เพื่อทำนายการเป็นวัณโรคจากภาพ Chest X-ray
2. เพื่อศึกษาการทำ Machine Learning เรื่อง Deep learning for Computer Vision โดยเน้นที่เรื่อง Convolutional Neural Network (CNN) เป็นหลัก
3. เพื่อศึกษาการเขียนโปรแกรมภาษา Python ใน Google Colaboratory

2. ขั้นตอนการทำงาน สิ่งที่ทำ

- ขั้นตอนการทำงาน

1. ศึกษาการทำ Machine Learning เรื่อง Deep learning for Computer Vision จาก <https://www.youtube.com/playlist?list=PL5-TkOAFzFbxjBHtzdVCWE0Zbhong7r>
2. วางแผน และ แนวทางในการทำงาน
3. เริ่มทำงานใน Google Colaboratory และรายงานความคืบหน้ากับอาจารย์ทุกสัปดาห์จนเสร็จ
4. ตรวจสอบความถูกต้องของงาน
5. ทำรูปเล่มสรุปการทำงาน

- สิ่งที่ทำ

1. Google Colaboratory: ภายในเป็นการเขียนโปรแกรมด้วย Python เพื่อตอบสนองวัตถุประสงค์ที่จะทำนายการเป็นวัณโรคจาก Chest X-ray image

2. รูปเล่มสรุปการทำงาน: ภายในเป็นการสรุปผลงานที่ทำ อุปสรรคที่เกิดขึ้นระหว่างการทำงาน แนวทางแก้ไข และ สิ่งที่ได้จากการศึกษาครั้งนี้

3. ผลลัพธ์ของงาน

3.1 Download data from Kaggle

- Download dataset จาก Kaggle (<https://www.kaggle.com/raddar/tuberculosis-chest-xrays-shenzhen>) มาเก็บไว้ใน Google Drive

3.2 Create dataframe

- สร้าง dataframe จากไฟล์ shenzhen_metadata.csv โดยการเพิ่ม column label ที่เป็นการดูตาม column finding โดย 0:normal, 1:abnormal (เป็นวัณโรค)
- Reshape Chest X-ray image ให้เป็นขนาด 256*256 กับ 224*224 pixels เพื่อนำไปใช้ Train Machine Learning model

3.3 Preprocessing data

- Normalize image ทุกรูปด้วยการลบค่า mean และ หาค่า standard deviance ของรูปภาพทุกรูป เพื่อให้ train model และ test ได้ดีขึ้น



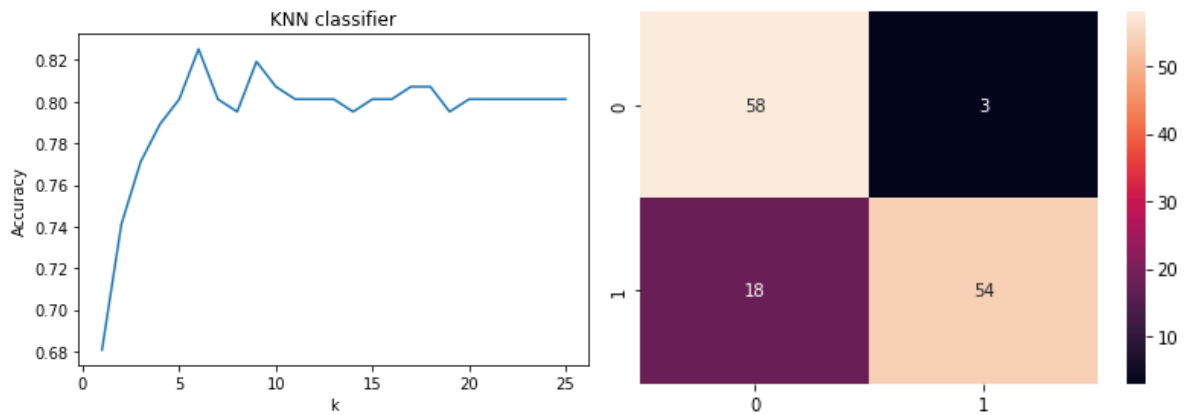
- ใช้ train-test split เพื่อแบ่งรูปภาพไว้ใช้ train validate กับ test สุดท้ายจะได้รูปที่ใช้ train 477 รูป ใช้ validate 85 รูป และใช้ test 100 รูป
- สร้าง Class XRayDataset ใช้เก็บ image pixels กับ labels
- ใช้ Dataloader ในการแบ่ง batch เพื่อใช้ในการ train validate และ test โดยจะใช้ batch size = 16 กับ training set และ batch size = 32 กับ validate และ testing set

3.4 Machine learning model

3.4.1 KNN Classifier

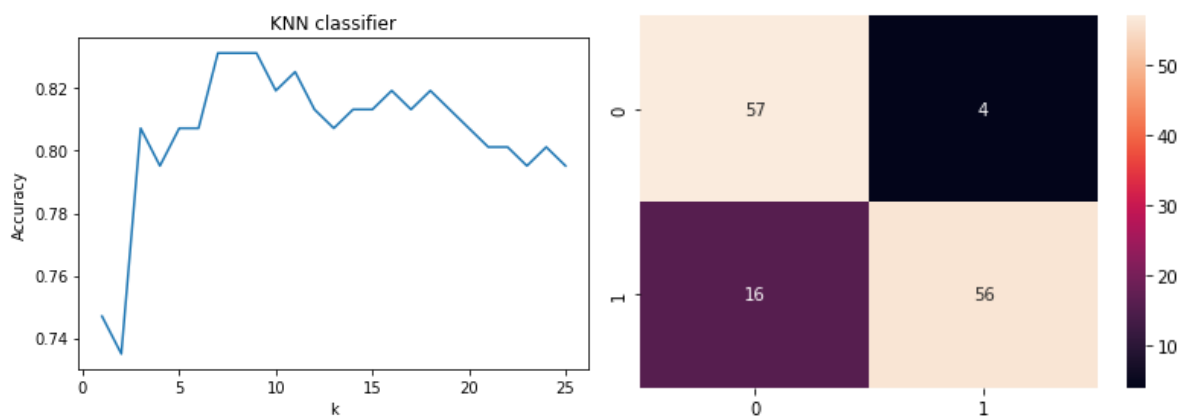
3.4.1.1 ใช้ Minkowski distance (l_p)

- จะพบว่าค่า k ที่ให้ความแม่นยำมากที่สุดคือ k=6 ให้ Accuracy: 0.8253012048192772 และ F1-score: 0.84



3.4.1.2 ใช้ Manhattan distance (l1) และ Euclidean distance (l2)

- จะพบว่าค่า k ที่ให้ความแม่นยำมากที่สุดคือ k=7 ให้ Accuracy: 0.8313253012048193 และ F1-score: 0.85

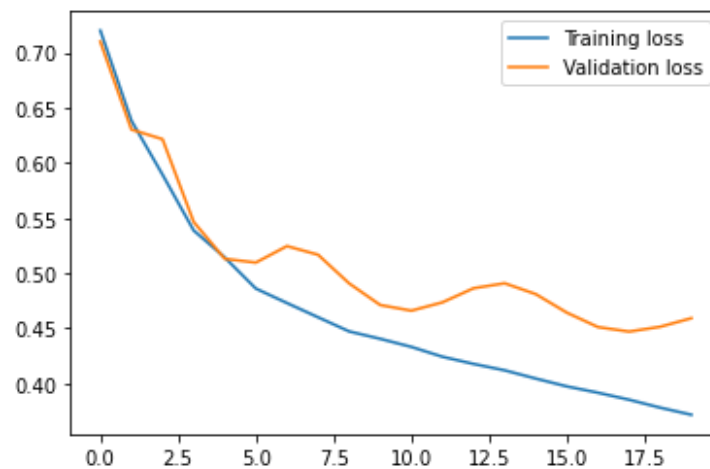


3.4.2 CNN

3.4.2.1 CNN-1 (2 layers)

```
Net(
  (cnn_layers): Sequential(
    (0): Conv2d(1, 4, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
    (1): BatchNorm2d(4, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
    (2): ReLU(inplace=True)
    (3): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)
    (4): Conv2d(4, 4, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
    (5): BatchNorm2d(4, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
    (6): ReLU(inplace=True)
    (7): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)
  )
  (linear_layers): Sequential(
    (0): Linear(in_features=16384, out_features=2, bias=True)
  )
)
```

- **Training 1:** num_epochs = 20, opt_func = torch.optim.Adam, lr = 0.0001, criterion = CrossEntropyLoss()

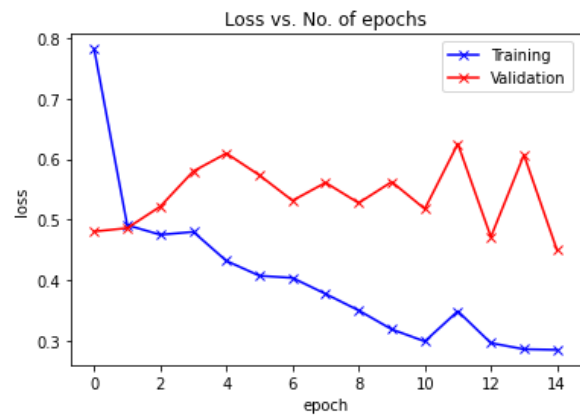
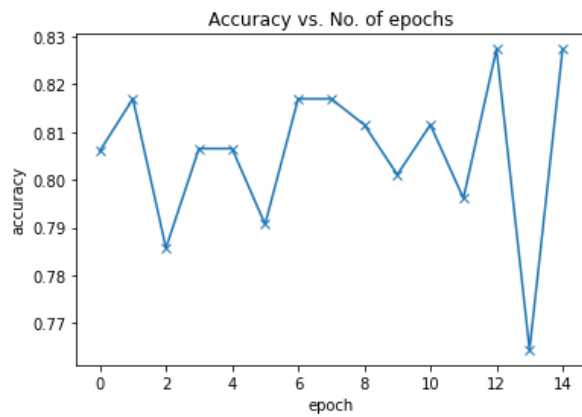


- Training accuracy: 0.8469601677148847
- Validate accuracy: 0.8117647058823529
- Testing accuracy: 0.79

3.4.2.2 CNN-2 (4 layers)

```
XrayClassification(
  (network): Sequential(
    (0): Conv2d(1, 32, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
    (1): ReLU()
    (2): Conv2d(32, 64, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
    (3): ReLU()
    (4): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)
    (5): Conv2d(64, 128, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
    (6): ReLU()
    (7): Conv2d(128, 128, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
    (8): ReLU()
    (9): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)
    (10): Conv2d(128, 256, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
    (11): ReLU()
    (12): Conv2d(256, 256, kernel_size=(3, 3), stride=(1, 1), padding=(1, 1))
    (13): ReLU()
    (14): MaxPool2d(kernel_size=2, stride=2, padding=0, dilation=1, ceil_mode=False)
    (15): Flatten(start_dim=1, end_dim=-1)
    (16): Linear(in_features=262144, out_features=1024, bias=True)
    (17): ReLU()
    (18): Linear(in_features=1024, out_features=512, bias=True)
    (19): ReLU()
    (20): Linear(in_features=512, out_features=2, bias=True)
  )
)
```

- **Training 1:** num_epochs = 15, opt_func = torch.optim.Adam, lr = 0.001, loss = cross_entropy

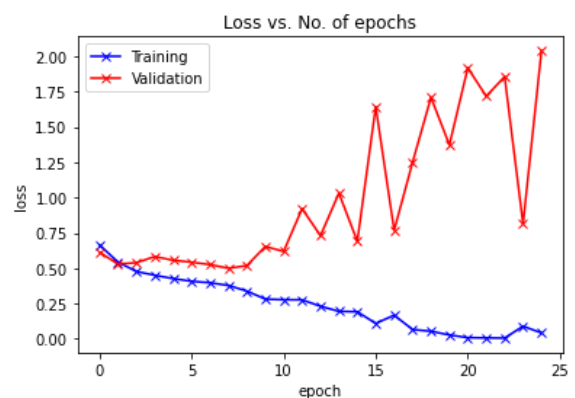
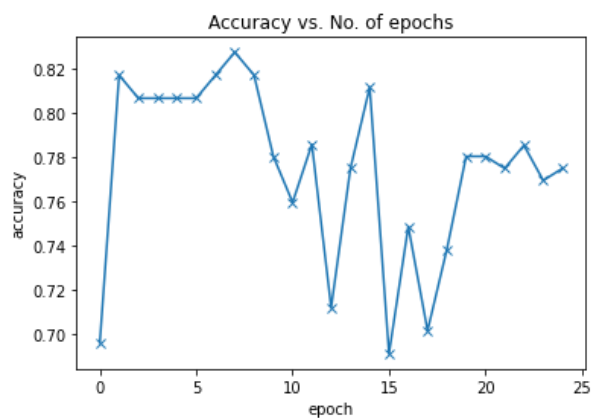


Training accuracy: 0.8844550848007202

Validate accuracy: 0.8273809552192688

Testing accuracy: 0.859375

- **Training 2:** num_epochs = 25, opt_func = torch.optim.Adam, lr = 0.0005, loss = cross_entropy



Training accuracy: 0.9854166507720947

Validate accuracy: 0.7748015522956848 (Overfit)

Testing accuracy: 0.8515625

4. อุปสรรคที่เกิดขึ้น วิธีแก้ปัญหา และสิ่งที่ได้จาก Individual Study นี้

- อุปสรรคที่เกิดขึ้น และ วิธีแก้ปัญหา

1. **อุปสรรค:** ไม่ทราบโครงสร้างของโค้ด และ model บางส่วนที่อยู่นอกเหนือจากการไปศึกษามา และ จำเป็นต้องใช้ในงานนี้

วิธีแก้: ไปค้นหาเพิ่มเติมในอินเทอร์เน็ต ปรึกษาอาจารย์ในการรายงานประจำสัปดาห์ หรือ ถามเพื่อนที่ทำงานด้วยกัน

2. **อุปสรรค:** ภาพ Chest X-ray ที่ได้รับมามีขนาดไม่เท่ากันและใหญ่เกินกว่าที่จะใช้ใน CNN model

วิธีแก้: ใช้ library ใน python ช่วย resize รูปให้เป็นขนาด 256*256 กับ 224*224 pixels

3. **อุปสรรค:** การใช้ GPU บน Google Colaboratory สามารถใช้ได้อย่างจำกัด

วิธีแก้: Save model ไว้ แล้ว reset runtime แล้วค่อย train model ต่อ หรือ มีการลดขนาด batch size เพื่อให้สามารถ train ได้หลาย epoch มากขึ้น

4. **อุปสรรค:** เวลา Mount Google Drive บางครั้งขึ้นว่า Mount เกิน Quota

วิธีแก้: ลองกด Mount ในวันถัดไปเมื่อ Quota reset แล้ว

- สิ่งที่ได้รับจาก Individual Study นี้

1. สรุปผลได้ว่าเมื่อมีการใช้ Machine Learning มาช่วยทำนายการเป็นวัณโรคจาก Chest X-ray image จะพบว่าการใช้ KNN Classifier จะมีความแม่นยำกับ Testing set ที่ 82-83% แต่ถ้าใช้ Convolutional Neural Network (CNN) ด้วย model ที่เหมาะสม จะมีความแม่นยำที่มากที่สุดกับ Testing set ที่ 85.9375%

2. ได้เรียนรู้เกี่ยวกับการทำ Machine Learning ในเรื่อง Deep Learning for Computer Vision

3. ได้เรียนรู้เกี่ยวกับการใช้ Convolutional Neural Network (CNN) มาทำนายข้อมูล

4. ได้ทบทวนการเขียนโปรแกรมด้วย Python และ การใช้ Google Colaboratory

Reference

- <https://www.youtube.com/playlist?list=PL5-TkOAfAZFbzxjBHtzdVCWE0Zbhomg7r>
 - <https://www.kaggle.com/pranjalsoni17/natural-scene-classification>
 - <https://www.analyticsvidhya.com/blog/2021/09/convolutional-neural-network-pytorch-implementation-on-cifar10-dataset/>
 - https://pytorch.org/tutorials/beginner/basics/buildmodel_tutorial.html
 - <https://www.kaggle.com/raddar/tuberculosis-chest-xrays-shenzhen>
-