**ภาคผนวก**

**ขั้นตอนการติดตั้งไลบราลีและเครื่องมือสำหรับการใช้งานโครงข่ายประสาทเชิงลึกด้วยคอมพิวเตอร์ส่วนบุคคล**

**ส่วนที่ 1 ส่วนประกอบที่จำเป็นในการใช้งานโปรแกรม**

* 1. ส่วนประกอบที่จำเป็นในการติดตั้งโปรแกรม

1.1.1. Windows 10 x64 bits

1.1.2. Python version 3.7

1.1.3. Anaconda Navigator

1.2. ติดตั้งไลบราลีที่จำเป็น โดยแนะนำให้ติดตั้งเวอร์ชั่นต่อไปนี้หรือใหม่กว่า

1.2.1. TensorFlow 2.3.0

1.2.2. Keras 2.4.3

1.2.3. Pip 20.2.3

1.2.4. Scikit Learn 0.23.2

1.2.5. Pandas 1.1.2

1.2.6. Tkinter 8.6.10

1.2.7. CUDA 1.1.0

1.2.8. cuDNN 8.0

**ส่วนที่ 2 ขั้นตอนการใช้งานและการทำงานของโปรแกรมที่เกี่ยวข้อง**

* 1. การติดตั้งสภาพแวดล้อมที่จำเป็นโดยใช้ Anaconda Navigator
     1. เข้าเว็บไซต์ และเลือกดาวน์โหลดแอพพลิเคชั่นสำหรับ Windows 64 bit

Graphical user interface, text, application

Description automatically generated

**รูปที่ ผ.1** การโหลดแอพพลิเคชั่น Anaconda Navigator ผ่านเว็บไซต์

* + 1. สร้างสภาพแวดล้อมใหม่เลือกเป็น Python เวอร์ชั่น 3.7

Graphical user interface

Description automatically generated  
**รูปที่ ผ.2** การสร้าง Environment เพื่อใช้งานโปรแกรมทั้งหมดในการทำวิจัย

2.1.3. ติดตั้งไลบราลีที่จำเป็น

Graphical user interface, text, application, email

Description automatically generated

**รูปที่ ผ.3** การค้นหาเครื่องมือ Tensorflow และ Keras

Graphical user interface, text, application, email

Description automatically generated  
**รูปที่ ผ.4** รูปไลบรารีที่จำเป็นหลังติดตั้งเสร็จสิ้นแล้ว

* + 1. เมื่อติดตั้งเสร็จ ให้เปิดด้วยโปรแกรม Spyder ผ่านสภาพแวดล้อมที่ Anaconda สร้างเอาไว้

Graphical user interface, application

Description automatically generated

**รูปที่ ผ.5** การเปิดแอพพลิเคชั่น Spyder ผ่าน Anaconda Navigator

A picture containing text, monitor, screenshot, screen

Description automatically generated

**รูปที่ ผ.6** หน้าต่างของแอพพลิเคชั่น Spyder

* 1. วิธีการใช้งานโปรแกรม

1. ทำการแตกไฟล์ Train set for AI Firewall
2. เปิด Spyder ผ่าน Tensorflow Enviroment และทำการรัน setup\_aio.py โดยการดำเนินการสร้างชุดแพ็คเกตฝึกสอน เทรนโมเดล และประมวลผลโมเดล จะถูกทำผ่าน Interface นี้

Graphical user interface, text, application

Description automatically generated

**รูปที่ ผ.6** การเปิดแอพพลิเคชั่น Spyder ผ่าน Anaconda Navigator

1. ตรวจสอบกฎไฟร์วอลล์ใน packet\_generator/rule\_assign.py

Text

Description automatically generated

**รูปที่ ผ.7** ช่องทางในการปรับกฎไฟร์วอลล์

วิธีการสร้างชุดฝึกสอน (Source Code: packet\_generator/Function.py)

1. กดปุ่ม generate template และตามด้วยการกดปุ่ม open template
2. Open template จะทำการเปิดเอกสาร excel โดยให้เรากรอกจำนวนข้อมูลฝึกสอนที่เราต้องการแต่ละกฎให้ทำการกรอกจำนวนที่เราต้องการและใส่ชื่อของชุดฝึกสอนตามลำดับ.

Graphical user interface, application, table, Excel

Description automatically generated

**รูปที่ ผ.8** ไฟล์ CSV ที่สร้างขึ้นผ่านการ generate template

Application, table, Excel

Description automatically generated

**รูปที่ ผ.9** วิธีการใส่จำนวนชุดฝึกสอนที่ต้องการในแต่ละกฎ

1. เมื่อเช็คความถูกต้องเรียบร้อยแล้ว ให้ทำการกดปุ่ม template -> data\_info เพื่อเปลี่ยนชื่อไฟล์ ซึ่ง data\_info จะเป็นไฟล์ที่จะนำเข้าตัวโปรแกรม train set generator
2. ทำการกดปุ่ม generate table เพื่อสร้างตารางในการบันทึกผลลัพธ์ที่ได้ในแต่ละขั้นตอน โดยตารางจะถูกเก็บอยู่ใน \_csv/result

Text

Description automatically generated

**รูปที่ ผ.10** ไฟล์ตารางที่บันทึกผลลัพธ์ที่สร้างขึ้นผ่านคำสั่ง generate table

1. จากนั้นให้ทำการกด generate packet โปรแกรมจะทำการสร้างชุดฝึกสอนขึ้นมา โดยชุดฝึกสอนจะเก็บอยู่ใน \_csv/train\_set

Text

Description automatically generated**รูปที่ ผ.11** การดำเนินงานผ่านคำสั่ง generate packet

Graphical user interface, application

Description automatically generated

**รูปที่ ผ.12** ชุดฝึกสอนที่สร้างผ่านคำสั่ง generate packet

Graphical user interface, table

Description automatically generated

**รูปที่ ผ.13** ผลการสร้างชุดฝึกสอนในตาราง generate result

**ขั้นตอนการฝึกสอนโมเดล** (Source Code: dnn\_model/training.py)

1. ตรวจสอบโครงสร้างของโปรแกรมฝึกโมเดลใน dnn\_model/training.csv

Text

Description automatically generated

**รูปที่ ผ.14** โครงสร้างของโมเดลจาก training.csv

1. กดปุ่ม training model โดยโมเดลจะอ่านชุดฝึกสอนที่สร้างทั้งหมดใน \_csv/train\_set

A screenshot of a computer

Description automatically generated

**รูปที่ ผ.15** การดำเนินงานผ่านคำสั่ง training model

1. โมเดลทั้งหมดหลังจากถูกฝึกสอนจะถูกเก็บอยู่ในโฟลเดอร์ model และสรุปผลการดำเนินงานที่ได้จากการฝึกโมเดลจะถูกเก็บใน training\_result

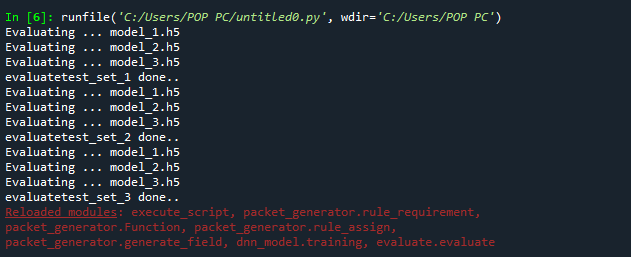
Table

Description automatically generated

**รูปที่ ผ.16** ผลการฝึกโมเดลในตาราง training\_result

**ขั้นตอนการประมวลผลโมเดล** (Source Code: evaluate/evaluate.py)

1. ทำการสร้างชุดทดสอบโดยทำเหมือนกับวิธีสร้างชุดฝึกสอน แต่ให้เปลี่ยนชื่อไฟล์เป็น test\_1, test\_2 เป็นต้น
2. ทำการย้ายไฟล์ชุดทดสอบจาก csv/train\_set ไปอยู่ในโฟลเดอร์ csv/test\_set
3. กดปุ่ม evaluate model โปรแกรมจะนำเข้าโมเดลทั้งหมดที่สร้างจากขั้นตอนก่อนหน้ามาประมวลผลชุดทดสอบทีละโมเดล



**รูปที่ ผ.17** ผลการดำเนินงานผ่านคำสั่ง evaluate model

1. ข้อมูลสรุปผลการดำเนินจะถูกเก็บบันทึกไว้ในตาราง evaluate\_result

Table

Description automatically generated

**รูปที่ ผ.18** ผลการประมวลผลโมเดลในตาราง evaluate\_result

Source code ในงานวิจัย (Github)

https://github.com/Kodashi/AI-Firewall-Training-set-Researching-main