การศึกษาวิจัยเพื่อพัฒนาสร้างชุดข้อมูลในการฝึกสอนไฟร์วอลล์ปัญญาประดิษฐ์ด้วยเทคโนโลยีโครงข่ายประสาทเทียมจากกฎของไฟร์วอลล์

Researching for developing training sets with artificial neural network technology based on firewall rules

โดย

ฐิติโชติ ใจเมือง

Thitichote Chaimuang

พิพัฒน์บุญ พุทธคุณ

Pipatboon Buddhakul

อาจารย์ที่ปรึกษา

ผู้ช่วยศาสตราจารย์ อัครินทร์ คุณกิตติ

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรบัณฑิต

สาขาวิชาเทคโนโลยีสารสนเทศ

คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ภาคเรียนที่ 2 ปีการศึกษา 2563

การศึกษาวิจัยเพื่อพัฒนาสร้างชุดข้อมูลในการฝึกสอนไฟร์วอลล์ปัญญาประดิษฐ์ด้วยเทคโนโลยีโครงข่ายประสาทเทียมจากกฎของไฟร์วอลล์

Researching for developing training sets with artificial neural network technology based on firewall rules

โดย

ฐิติโชติ ใจเมือง

พิพัฒน์บุญ พุทธคุณ

อาจารย์ที่ปรึกษา

ผู้ช่วยศาสตราจารย์ อัครินทร์ คุณกิตติ

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรบัณฑิต

สาขาวิชาเทคโนโลยีสารสนเทศ

คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ภาคเรียนที่ 2 ปีการศึกษา 2562

RESEARCHING FOR DEVELOPING TRAINING SETS WITH ARTIFICIAL NEURAL NETWORK TECHNOLOGY BASED ON FIREWALL RULES

THITICHOTE CHAIMUANG

PIPATBOON BUDDHAKUL

A PROJECT SUBMITTED IN PARTIAL FULFILLMENT

OF THE REQUIREMENT FOR THE DEGREE OF

BACHELOR OF SCIENCE PROGRAM IN INFORMATION TECHNOLOGY

FACULTY OF INFORMATION TECHNOLOGY

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

2/2019

COPYRIGHT 2020

FACULTY OF INFORMATION TECHNOLOGY

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

ใบรับรองปริญญานิพนธ์ ประจำปีการศึกษา 2562

คณะเทคโนโลยีสารสนเทศ

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง การศึกษาวิจัยเพื่อพัฒนาสร้างชุดข้อมูลในการฝึกสอนไฟร์วอลล์ปัญญาประดิษฐ์ด้วยเทคโนโลยีโครงข่ายประสาทเทียมจากกฎของไฟร์วอลล์

Researching for developing training set with artificial neural network technology based on firewall rules

ผู้จัดทำ

1. นายฐิติโชติ ใจเมือง รหัสประจำตัว 60070019
2. นายพิพัฒน์บุญ พุทธคุณ รหัสประจำตัว 60070065

..... อาจารย์ที่ปรึกษา

(ผู้ช่วยศาสตราจารย์ อัครินทร์ คุณกิตติ)

ใบรับรองโครงงาน (Project)

เรื่อง

การศึกษาวิจัยเพื่อพัฒนาสร้างชุดข้อมูลในการฝึกสอนไฟร์วอลล์ปัญญาประดิษฐ์ด้วยเทคโนโลยีโครงข่ายประสาทเทียมจากกฎของไฟร์วอลล์

Researching for developing training sets with artificial neural network technology based on firewall rules

นายฐิติโชติ ใจเมือง รหัสประจำตัว 60070019

นายพิพัฒน์บุญ พุทธคุณ รหัสประจำตัว 60070065

ขอรับรองว่ารายงานฉบับนี้ ข้าพเจ้าไม่ได้คัดลอกมาจากที่ใด

รายงานฉบับนี้ได้รับการตรวจสอบและอนุมัติให้เป็นส่วนหนึ่งของ

การศึกษาวิชาโครงงาน หลักสูตรวิทยาศาสตรบัณฑิต (เทคโนโลยีสารสนเทศ)

ภาคเรียนที่ 2 ปีการศึกษา 2562

.....

(นายฐิติโชติ ใจเมือง)

.....

(นายพิพัฒน์บุญ พุทธคุณ)

หัวข้อโครงงาน	การศึกษาวิจัยเพื่อพัฒนาสร้างชุดข้อมูลในการฝึกสอนไฟร์วอลล์ปัญญาประดิษฐ์ด้วยเทคโนโลยีโครงข่ายประสาทเทียมจากกฎของไฟร์วอลล์
นักศึกษา	ฐิติโชติ ใจเมือง รหัสนักศึกษา 60070019 พิพัฒน์บุญ พุทธคุณ รหัสนักศึกษา 60070065
ปริญญา	วิทยาศาสตรบัณฑิต
สาขาวิชา	เทคโนโลยีสารสนเทศ
ปีการศึกษา	2563
อาจารย์ที่ปรึกษา	ผู้ช่วยศาสตราจารย์ อัครินทร์ คุณกิตติ

บทคัดย่อ

Project Title	Researching for developing training set with artificial neural network technology based on firewall rules			
Student	Thitichote	Chaimuang	Student ID	60070019
	Pipatboon	Buddhakul	Student ID	60070065
Degree	วิทยาศาสตรบัณฑิต			
Program	เทคโนโลยีสารสนเทศ			
Academic Year	2020			
Advisor	ผู้ช่วยศาสตราจารย์ อัครินทร์ คุณกิตติ			

ABSTRACT

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สาเร็จลุล่วงไปได้ด้วยดี ทางผู้จัดทาขอขอบพระคุณเป็นอย่างสูงกับความกรุณาช่วยเหลือและการให้คาปรึกษาของ ผู้ช่วยศาสตราจารย์อัครินทร์ คุณกิตติ ที่ช่วยชี้แนะแนวทางตั้งแต่วันแรกถึงวันสุดท้าย และขอบพระคุณอาจารย์ คณะเทคโนโลยีสารสนเทศ สถาบันเทคโนโลยีพระ จอมเกล้าเจ้าคุณทหารลาดกระบังทุกๆท่าน ที่ให้ความรู้อันเป็นประโยชน์ยิ่ง ต่อการพัฒนาต่อยอดองค์ความรู้

ขอขอบคุณครอบครัวที่ให้การสนับสนุนอย่างดีเสมอมา

ขอขอบคุณคู่โครงงานที่อดทนและร่วมแรงร่วมใจช่วยกันมาจนถึงทุกวันนี้

ฐิติโชติ ใจเมือง

พิพัฒน์บุญ พุทธคุณ

บทที่ 1

บทนำ

1.1 ความเป็นมาของโครงงาน

Firewall ถูกสร้างขึ้นเพื่อจุดประสงค์ทางด้านความปลอดภัยทางเครือข่าย มีหน้าที่เปรียบเสมือนยามเฝ้าประตู โดยข้อมูลภายในเครือข่ายจะผ่านการคัดกรองข้อมูลด้วยหลักการของ Packet Filtering เมื่อเวลาผ่านไป การพัฒนาของเทคโนโลยีใหม่ๆและรูปแบบการโจมตีทางเครือข่ายที่มีมากขึ้น Firewall แบบเก่าที่กำหนดโดยผู้ควบคุมระบบเพียงอย่างเดียว ไม่สามารถตอบโจทย์ทางด้านความปลอดภัยได้ ทำให้มีการนำปัญญาประดิษฐ์หรือ AI มาประยุกต์ใช้งานกับ Firewall ให้มีความคิดและตัดสินใจเลือกคัดกรอง Packet ได้เอง ผู้จัดทำมีความคิดที่จะพัฒนา AI Firewall ให้มีประสิทธิภาพมากยิ่งขึ้น โดยนำมาประยุกต์ใช้กับกระบวนการเรียนรู้แบบ Deep Neural Network และมีชุดข้อมูล Packet ฝึกสอนที่สร้างขึ้นอ้างอิงตามนโยบายข้อกำหนดจาก Firewall Rules เพื่อแก้ปัญหาข้อมูลฝึกสอนที่ไม่ได้เป็นไปตามนโยบายข้อกำหนด ที่แต่เดิมต้องเอาข้อมูลการโจมตีที่เคยเกิดขึ้นมาก่อนเป็นข้อมูลฝึกสอน

1.2 วัตถุประสงค์

1. เพื่อให้เข้าใจหลักการทำงานของ Neural Network ที่จะใช้พัฒนาปัญญาประดิษฐ์
2. เพื่อพัฒนาสร้างชุดข้อมูลฝึกสอนให้เป็นไปตามนโยบายข้อกำหนดตาม Firewall Rules
3. เพื่อให้ชุดข้อมูล Network Packet ที่สร้างขึ้นสามารถฝึกสอนได้อย่างถูกต้องและแม่นยำ เมื่อนำไปใช้กับ AI ที่มีการเรียนรู้แบบ Deep Neural Network Model
4. เพื่อแก้ไขข้อจำกัดของข้อมูลฝึกสอน Firewall ให้ผ่านเงื่อนไขที่กำหนด เช่น เวลาที่ใช้ หรือปริมาณของข้อมูล Packet

1.3 วิธีการดำเนินงาน

พัฒนาสร้างชุดข้อมูลฝึกสอน Network Packet ที่สร้างขึ้นโดยมีการอ้างอิงจาก Firewall Rules ไปใช้กับ AI Firewall ที่มีการเรียนรู้แบบ Neural Network Model และทำการตรวจสอบความถูกต้อง ความผิดพลาดที่ได้เปรียบเทียบกับ Firewall Rules ที่กำหนด โดยทำการทดลองหลายๆครั้ง เปลี่ยนตัวแปรและปัจจัยต่างๆ เพื่อหาวิธีการที่ทาให้ระบบสามารถทำงานได้อย่างถูกต้องและมีประสิทธิภาพมากที่สุด

1.4 ขอบเขตของโครงงาน

พัฒนา Neural Network Model และชุดข้อมูลฝึกสอน Network Packet ที่สร้างขึ้นโดยอ้างอิงจาก Firewall Rules นำไปผ่านการเรียนรู้และทำการทดสอบ ลองเปลี่ยนปัจจัยและค่าตัวแปรต่างๆ เปรียบเทียบผลลัพธ์ในแต่ละรูปแบบ ใช้ความถูกต้อง ความผิดพลาดที่อิงจากกฎของ Firewall Rules เป็นเกณฑ์ในการวัดผล ศึกษาหาวิธีการและผลลัพธ์ที่ดีที่สุดภายใต้การทำงานของโปรแกรมคอมพิวเตอร์

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1. พัฒนาทักษะการเขียนโปรแกรมที่เขียนด้วยภาษา Python
2. เรียนรู้วิธีการสร้างชุดข้อมูลฝึกสอน สามารถประยุกต์ใช้กับปัญญาประดิษฐ์ได้
3. เรียนรู้วิธีการพัฒนาอัลกอริทึ่มที่ช่วยลดเวลา เพิ่มประสิทธิภาพในการคำนวณข้อมูลได้
4. สามารถประยุกต์ learning model ไปใช้กับปัญญาประดิษฐ์รูปแบบอื่น เช่น การทำแชทบอท โปรแกรมวิเคราะห์ข้อมูล หรือ ระบบปฏิบัติการตอบโต้อัตโนมัติ

บทที่ 2

ทฤษฎีการนำโครงข่ายระบบประสาทเชิงลึก มาใช้ในการทำงานของไฟร์วอลล์

2.1 เทคโนโลยีของไฟร์วอลล์และโครงข่ายระบบประสาทเชิงลึก

2.1.1 Firewall

Firewall เป็นระบบควบคุมและรักษาความปลอดภัยของระบบเครือข่าย คัดกรองข้อมูลเข้าออกในช่องทางอินเตอร์เน็ต เปรียบเสมือนยามเฝ้าประตู คอยป้องกันการโจมตี สแปม ผู้บุกรุกต่างๆที่ไม่หวังดีต่อระบบ และยังสามารถใช้ควบคุมการใช้งานของโปรแกรมที่ต้องการ ในปัจจุบันมีการใช้งานได้ทั้งระบบ Hardware และ Software ขึ้นอยู่กับความเหมาะสม ผลลัพธ์ที่ออกมาจาก Firewall จะพิจารณาการกระทำของ Packet ออกมาเป็น Allow หรือ Deny

2.1.1.1 Packet Filtering

ระบบการทำงานของ Firewall ทำงานในระบบ Internet Layer และ Transport Layer ตรวจสอบและคั่นกรอง Packet ที่เข้ามาในเครือข่าย โดยพิจารณาจาก Packet Header ตัดสินใจว่าจะทำการ Allow หรือ Deny โดยใช้กฎของ Firewall ในการอ้างอิง ซึ่ง Firewall แบ่งประเภทตามลักษณะการทำงาน ได้แก่

2.1.1.2 Stateful Filtering

Stateful Filtering จะมีเก็บสถานะ Packet ใดที่เคยถูกปล่อยผ่านและเก็บบันทึกไว้ใน State Table ทำให้การทำงานของ Firewall นี้จะถูกตรวจสอบเริ่มจากที่ State Table ก่อน ถ้าหาก Packet ที่กำลังถูกตรวจสอบอยู่ยังไม่เคยถูกปล่อยผ่านยังไม่มีการเก็บสถานะเอาไว้ถึงจะไปพิจารณากฎของไฟรวอลล์เป็นอันดับถัดไป กลไกนี้จะช่วยไฟร์วอลล์ทำงานได้เร็วขึ้น เพราะช่วยลดระยะเวลาในการทำงานไม่ต้องเสียเวลาพิจารณาทุก Packet Header ในกลไก Packet Filtering

2.1.1.3 Application Firewall

มีชื่อเรียกได้อีกอย่างหนึ่งว่า “Application-level Firewall” หรือ “Application Gateway” เป็น Firewall ชนิดที่ติดตั้งบนเครื่องคอมพิวเตอร์แยกต่างหาก ทำให้คอมพิวเตอร์เครื่องดังกล่าวทำหน้าที่เป็น Firewall โดยเฉพาะ อย่างไรก็ตามApplication Firewall สามารถกรอง Packet ที่จะผ่านเข้ามาในเครือข่าย อีกทั้งยังตรวจสอบเนื้อหาใน Packet ได้เช่นเดียวกับ Stateful Filtering Firewall นอกจากนี้ Application Firewall ยังทำหน้าที่คล้ายกับ Proxy Server ในการให้บริการคำร้องขอของผู้ใช้ได้อีกด้วย โดยความสามารถของ Application Firewall สามารถแบ่งทำได้ดังนี้

Security

การยืนยันตัวตนด้วย AAA คือ Authentication, Authorization และ Audit โดยการสร้าง Token ไปให้ทั้งผู้รับ และผู้ส่ง มีการกำหนด Policy เพื่อการเข้าถึงข้อมูล และยังทำการเก็บข้อมูลการเข้าออกของ Policyนั้นๆ อีกทั้งยังมีการป้องกันด้วยการตรวจสอบข้อมูลที่ได้รับก่อนว่าถูกต้องตามโครงสร้างที่ได้กำหนดไว้หรือไม่

Integration

การสร้างการเชื่อมต่อเข้ากับระบบต่างๆให้สามารถทำงานร่วมกันได้ เช่น ถ้าหากระบบที่ใช้มีโปรโตคอลที่แตกต่างกัน มันจะทำการแปลงโครงสร้างข้อความโดยการจับคู่ข้อมูล

Control and Managing

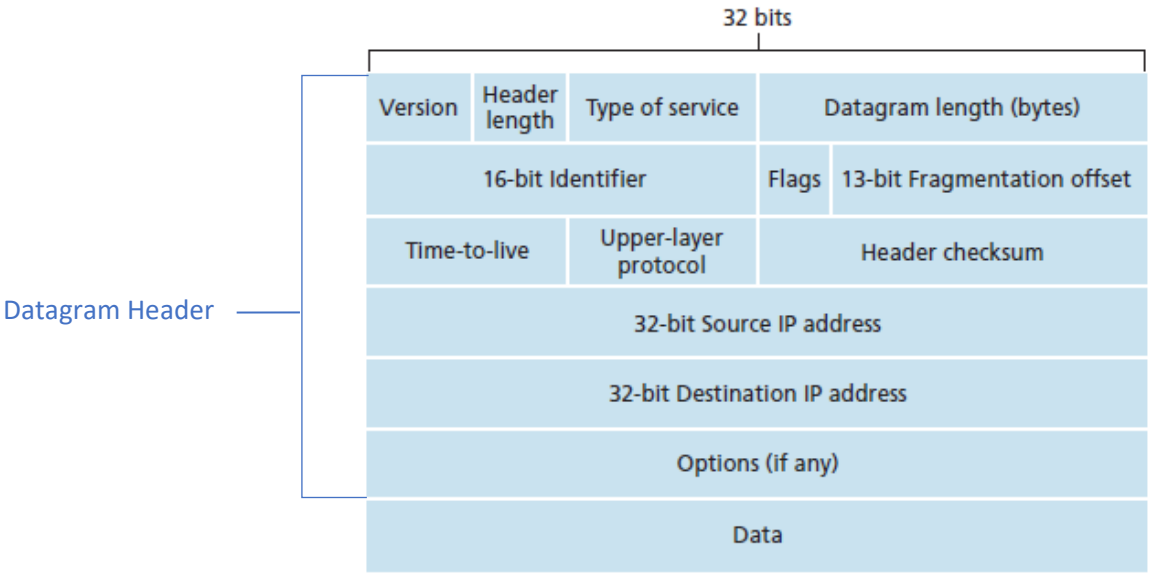
การควบคุมปริมาณของข้อความที่จะวิ่งเข้าไปหา Server โดยการกำหนด Policy แยกตามประเภทของ API และประเภทของข้อมูล สำหรับการควบคุมปริมาณข้อความนี้จะเป็นการป้องกันการถูกผู้ไม่หวังดีโจมตีจากช่องโหว่ของระบบได้ เช่น เรามี API ที่เปิดให้ลูกค้าหรือบุคคลอื่นๆเข้ามาใช้งานได้ ถ้าหากไม่มีการกำหนดปริมาณการเรียกใช้ API หรือเส้นทางของข้อมูล ก็จะเกิดช่องโหว่ของระบบที่ผู้ไม่หวังดีสามารถทำการ DOS ได้

Optimizing

การลดภาระการทำงาน ของ Server โดยการทำ SSL และนำภาระงานจากการถอดรหัสที่ Server ไปให้ไฟร์วอลล์ทำงานแทน จะทำให้ Server มีทรัพยากรเหลือพอที่จะรองรับการทำงานมากขึ้น

2.1.2 Packet Header

Packet Header เป็นโปรโตคอลอินเตอร์เน็ต มาตรฐานที่ทำให้อินเตอร์เน็ตสามารถเชื่อมต่อเข้าหากัน ติดต่อสื่อสารข้อมูลได้ด้วยการกำหนดวิธีการติดต่อสื่อสารร่วมกัน ในส่วนของ Packet Header จะเป็นลำดับชั้นโครงสร้างประกอบไปด้วย Field ข้อมูลที่บ่งบอกถึงวัตถุประสงค์และลักษณะการทำงานของ Packet โดยองค์ประกอบของ Packet Header มีดังนี้



Version ส่วนที่ระบุเวอร์ชั่นโปรโตคอลของ Datagram

Header length ส่วนที่ระบุขนาดของ Datagram Header

Type of service ส่วนที่ระบุประเภทของ Datagram เช่น low delay high throughput หรือ Reliability

Datagram length ส่วนที่ระบุขนาดของ Datagram ทั้งหมดรวมถึง Datagram Header

Identifier ส่วนที่มีไว้เพื่อยืนยันตัว หากมีการทำ Fragmention

Flags ส่วนที่ระบุว่า Datagram นี้จะทำการ Fragmention หรือไม่

Fragmentation offset ส่วนที่แสดงให้เห็นถึงจำนวนของข้อมูลก่อนทำการ Fragmention

Time-to-live ส่วนที่กำหนดวงจรชีวิตของ Datagram เพื่อป้องกันไม่เกิด Loop ในเครื่อข่าย

Protocol ส่วนที่ระบุโปรโตคอลที่ใช้ใน Datagram นี้

Header checksum ส่วนที่ใช้สำหรับตรวจสอบความถูกต้อง Datagram Header

Source and destination IP addresses ส่วนที่ระบุที่อยู่ของ IP ต้นทางกับ IP ปลายทาง

Options ส่วนเพิ่มเติมที่คอยเก็บข้อมูลเช่น เส้นทางต้นทางของ Datagram เส้นทางที่ใช้โดยเก็บไว้เพื่อตรวจสอบการทำงาน เป็นต้น

2.1.3 Artificial Intelligent

Artificial Intelligence คือ เครื่องจักรอัจฉริยะที่มีความสามารถในการทำความเข้าใจ เรียนรู้องค์ความรู้ต่างๆ เช่น การรับรู้ การให้เหตุผล ในการแก้ไขปัญหาต่างๆเพื่อปฏิบัติงานตามความต้องการของมนุษย์ เครื่องจักรที่มีความสามารถนี้ถูกเรียกอีกชื่อหนึ่งว่า “ปัญญาประดิษฐ์”

AI ถูกจำแนกเป็น 3 ระดับตามความสามารถดังนี้

Narrow Artificial Intelligent ปัญญาประดิษฐ์เชิงแคบ คือ AI ที่มีความสามารถเฉพาะทางได้ดีกว่ามนุษย์ เช่น เครื่องจักรที่ใช้ในการผ่าตัด

General Artificial Intelligent ปัญญาประดิษฐ์ทั่วไป คือ AI ที่มีความสามารถระดับเดียวกับมนุษย์ สามารถทำทุกอย่างในประประสิทธิภาพที่ใกล้เคียงกับมนุษย์

Strong Artificial Intelligent ปัญญาประดิษฐ์แบบเข้ม คือ AI ที่มีความสามารถมากกว่ามนุษย์ในหลายๆด้าน

และจากการนำปัญญาประดิษฐ์มาประยุกต์ใช้ในการแก้ไขปัญหา มุมมองต่อ AI ที่แต่ละคนมีอาจไม่เหมือนกัน ขึ้นอยู่กับว่า เราต้องการความฉลาดโดย คำนึงถึงพฤติกรรมที่มีต่อสิ่งแวดล้อมหรือคำนึงการคิดได้ของผลผลิต AI ดังนั้นจึงมีคำนิยาม AI ตามความสามารถที่มนุษย์ต้องการ ให้มันแบ่งได้ 4 กลุ่ม ดังนี้

Thinking humanly (การคิดคล้ายมนุษย์)

natural language processing สื่อสารกับ มนุษย์ได้ด้วยภาษาที่มนุษย์ใช้ เช่น ภาษาอังกฤษ เป็นการประมวลผลภาษาธรรมชาติ

computer vision มีประสาทรับสัมผัสคล้ายมนุษย์ เช่นคอมพิวเตอร์วิทัศน์ รับภาพได้โดยใช้อุปกรณ์รับสัญญาณภาพ

machine learning เพื่อปรับให้เข้ากับสถานการณ์ใหม่และ ตรวจจับและคาดการณ์รูปแบบ

Thinking rationally (คิดอย่างมีเหตุผล)

คิดอย่างมี เหตุผล หรือคิดถูกต้อง โดยใช้หลักตรรกศาสตร์ในการคิดหาคำตอบอย่างมีเหตุผล เช่น ระบบผู้เชี่ยวชาญ

Acting humanly (การกระทำคล้ายมนุษย์)

การคิดคล้าย มนุษย์ ก่อนที่จะทำให้เครื่องคิดอย่างมนุษย์ได้ ต้องรู้ก่อนว่ามนุษย์มีกระบวนการคิดอย่างไร ซึ่งการวิเคราะห์ลักษณะการคิดของมนุษย์เป็นศาสตร์ด้าน cognitive science เช่น ศึกษาโครงสร้างสามมิติของเซลล์สมอง การแลกเปลี่ยนประจุไฟฟ้าระหว่างเซลล์สมอง วิเคราะห์การเปลี่ยนแปลงทางเคมีไฟฟ้าในร่างกายระหว่างการคิด ซึ่งจนถึงปัจจุบันเราก็ยังไม่รู้แน่ชัดว่า มนุษย์เรา คิดได้อย่างไร

Acting rationally (การกระทำอย่างมีเหตุผล)

กระทำอย่างมีเหตุผล เช่น agent (agent เป็นโปรแกรมที่มีความสามารถในการกระทำ หรือเป็นตัวแทนในระบบอัตโนมัติต่าง ๆ) สามารถกระทำอย่างมีเหตุผลคือ agent ที่กระทำการเพื่อบรรลุเป้าหมายที่ได้ตั้งไว้ เช่น agent ใน ระบบขับรถอัตโนมัติที่มีเป้าหมายว่าต้องไปถึงเป้าหมายในระยะทางที่สั้นที่ สุด ต้องเลือกเส้นทางที่ไปยังเป้าหมายที่สั้นที่สุดที่เป็นไปได้จึงจะเรียกได้ ว่า agent กระทำอย่างมีเหตุผล อีกตัวอย่างเช่น agent ใน เกมหมากรุกมีเป้าหมายว่าต้องเอาชนะคู่ต่อสู้ ต้องเลือกเดินหมากที่จะทำให้คู่ต่อสู้แพ้ให้ได้ เป็นต้น

2.1.4 Machine Learning

Machine Learning คือ ส่วนการเรียนรู้ของเครื่อง ถูกใช้งานเสมือนเป็นสมองของปัญญาประดิษฐ์ในการสร้างความฉลาด มักจะใช้เรียกโมเดลที่เกิดจากการเรียนรู้ของปัญญาประดิษฐ์ โดยมนุษย์มีหน้าที่เขียนโปรแกรมให้เรียนรู้จากชุดข้อมูลฝึกสอนหรือ Training set และอาศัยกลไกที่เป็นโปรแกรม หรือเรียกว่า Algorithm ที่มีหลากหลายแบบ โดยมี Data Scientist เป็นผู้ออกแบบ หนึ่งใน Algorithm ที่ได้รับความนิยมสูง คือ Deep Learning ซึ่งถูกออกแบบมาให้ใช้งานได้ง่าย และประยุกต์ใช้ได้หลายลักษณะงาน อย่างไรก็ตาม ในการทำงานจริง Data Scientist จำเป็นต้องออกแบบตัวแปรต่างๆ ทั้งในตัวของ Deep Learning เอง และต้องหา Algorithm

อื่นๆ มาเป็นคู่เปรียบเทียบ เพื่อมองหา Algorithm ที่เหมาะสมที่สุดในการใช้งานจริง โดยตามหลักแล้วจะแบ่งประเภทของ Machine Learning ได้ดังนี้

2.1.4.1 Supervised

การทำให้เครื่องคอมพิวเตอร์สามารถเรียนรู้ได้จากชุดข้อมูลฝึกสอนหรือ Training set ก่อนที่จะประมวลผล โดยมนุษย์จะเป็นผู้กำหนดคุณลักษณะ ความสัมพันธ์เฉพาะของข้อมูลที่ต้องการให้เครื่องคอมพิวเตอร์เรียนรู้ หรือที่เรียกว่า Label และเมื่อโมเดลผ่านการเรียนรู้แล้ว จะสามารถแยกแยะประเภท มีวิธีการคิดที่เริ่มมีเหตุผล เมื่อข้อมูลที่ต้องการวิเคราะห์มีจำนวนที่มากขึ้นจำเป็นต้องมีข้อมูลที่เป็น Training set มากขึ้นเช่นเดียวกัน โดยการเรียนรู้แบบ Supervised Learning นี้จะประกอบไปด้วยดังนี้

2.1.4.1.1 Classification

คือการสอนโมเดลให้สามารถแบ่งหรือแยกประเภทกลุ่มข้อมูลได้ โดยอ้างอิงจากความสัมพันธ์และน้ำหนักของข้อมูลแต่ละ Label ตัวอย่างเช่น การแยกกลุ่มผู้ป่วยว่าเป็นเนื้องอกในสมอง ซึ่งจะมีปัจจัยต่างๆมากมายไม่ว่าจะเป็น ขนาด, รูปร่าง, ตำแหน่ง หรือ สีผิว ซึ่งถ้าหากมีข้อมูลเพียงแค่ Label เดียว ไม่สามารถพิสูจน์หรือแบ่งกลุ่มได้

2.1.4.1.2 Regression

การสอนโมเดลโดยอิงจากผลลัพธ์ที่ผ่านมา โดยผลลัพธ์จะเป็นการประมาณค่าความเป็นไปได้ที่จะเกิดขึ้นต่อ ทำให้เหมาะแก่การวิเคราะห์ความสัมพันธ์ของตัวแปรที่อยู่ในรูปกราฟ เช่น การหาความสัมพันธ์ระหว่างขนาดของบ้านและราคา การประเมินราคาหุ้น

2.1.4.2 Unsupervised

รูปแบบการเรียนรู้ที่ไม่จำเป็นต้องใช้ชุดข้อมูลฝึกสอน แต่เป็นการป้อนข้อมูล Test set ไปประมวลผลเพียงอย่างเดียว ทำให้ผลลัพธ์ที่ออกมาไม่รู้ผลลัพธ์แน่ชัด ซึ่งอัลกอริทึ่มจะวิเคราะห์และหาโครงสร้างของข้อมูลเอง

2.1.4.2.1 Clustering

เป็นการกำหนดให้เครื่องคอมพิวเคอร์หาวิธีแบ่งกลุ่มหรือจัดกลุ่มข้อมูลเอง เปรียบเสมือนการลด Label ของข้อมูลที่มีปริมาณมาก จัดกลุ่มหาข้อมูลที่มีความสัมพันธ์ใกล้เคียงกัน ผลลัพธ์ที่ได้ออกมาจะมีปริมาณ Label ที่น้อยลงเป็นอย่างมาก

2.1.4.2.2 Dimensionality Reduction

เป็นการกลไกการบีบอัดและลดมิติข้อมูลจำนวนมากให้มีจำนวนลดลงโดยที่ข้อมูลยังครบถ้วน และยังสามารถนำไปจำแนกข้อมูลได้เหมือนเดิม

2.1.4.3 Reinforcement Learning

เป็นการเรียนรู้ด้านหนึ่งของ Machine Learning มักใช้พัฒนาหุ่นยนต์หรือการเรียนรู้ที่อยู่ภายในเกมคอมพิวเตอร์ เช่น การลองผิดลองถูกไปเรื่อยเพื่อหาผลลัพธ์ที่ดีที่สุดประเมินออกมาเป็นคะแนน โดยชุดข้อมูลทดสอบจะเป็นสภาพแวดล้อมโดยรอบขึ้นอยู่กับความต้องการของผู้พัฒนา

2.1.5 Deep Learning

Deep learning คือ อัลกอริทึมการเรียนรู้เชิงลึกโดยใช้หลักการ Artificial Neural Networks ที่มีรูปแบบการทำงานคล้ายคลึงกับเซลล์ประสาทที่เชื่อมต่อกันเป็นโครงข่ายประสาทในสมองมนุษย์ เหมาะกับการวิเคราะห์ข้อมูลขนาดใหญ่ที่มีความซับซ้อน เช่น การจำแนกรูปภาพ การจำแนกใบหน้า ประกอบไปด้วยโครงสร้างของหน่วยประมวลผลจำนวนมากคือเซลล์ประสาท หรือ Neuron โดยอัลกอริทึ่มนี้จะประกอบไปด้วยชั้นต่างๆ ดังนี้

Input Layer มีหน้าที่รับข้อมูลเข้ามาประมวลผลและส่งต่อให้ Hidden Layer

Hidden Layer มีหน้าที่คำนวณและประมวลผลข้อมูลโดยสามารถมีได้หลายชั้น หลายขนาดขึ้นอยู่กับความซับซ้อนของข้อมูล

Output Layer มีหน้าที่ส่งผลลัพธ์ข้อมูลที่ผ่านการประมวลผลแล้วออกมา

เมื่อเริ่มการฝึกฝนจะเริ่มจากการสุ่มค่าถ่วงน้ำหนัก (Weight) และจะเริ่มปรับผลลัพธ์เอามาคูณกับค่าถ่วงน้ำหนักแล้วบวกด้วยค่าความเอนเอียงของข้อมูล (Bias) หลังจากนั้นจะนำผลลัพธ์ที่ได้มาในแต่ละขาของ Neural Network มารวมกันแล้วมาผ่านฟังก์ชั่นส่งต่อให้ลำดับชั้นถัดไปประมวลผลมีการใช้วิธีการประมวลผลทางคณิตศาสตร์ (Activation Function) โดยทุกวันนี้มีการะประยุกต์ใช้อย่างแพร่หลาย แบ่งชนิดโครงข่ายประสาทออกเป็นดังนี้

2.1.5.1 โครงข่ายประสาทแบบป้อนไปข้างหน้า (Forward Propagation)

Feed-forward neural networks ถือเป็นโมเดลที่มีโครงสร้างที่เรียบง่ายที่สุด เพราะว่า การดำเนินการของข้อมูลจะเป็นไปในทิศทางเดียว ก็คือ รับข้อมูลจาก input layer แล้วส่งไปต่อไปยัง hidden layer เลื่อยๆ จนกระทั้งถึง output layer ก็จะหยุด สังเกตุได้ว่าจะไม่มีวงวน หรือ loop เกิดขึ้นเลย

2.1.5.2 โครงข่ายแบบวนซ้ำ (Recurrent neural networks : RNN)

Recurrent neural networks คือ neural networks หลายเลเยอร์ที่สามารถเก็บข้อมูล information ไว้ที่ node จึงทำให้มันสามารถรับข้อมูลเป็นแบบลำดับ (data sequences) และให้ผลลัพธ์ออกเป็นลำดับของข้อมูลได้ อธิบายอย่างง่ายๆ RNN ก็คือ neural network เชื่อมต่อกันหลายๆอันและยังสามารถต่อกันเป็นวงวนหรือ loop ได้นั่นเอง เพราะฉะนั้น RNN จึงเหมาะสมในการประมวลผลข้อมูลที่เป็นลำดับอย่างมาก

2.2 ทบทวนวรรณกรรม

2.2.1 การนำเอาความสามารถของ GPU มาใช้ในการคำนวณ

การที่เราเลือกใช้ GPU ในการทำ Machine Learning เนื่องจากตัว GPU นั้นมีหน่วยความจำที่ให้ค่าแบนด์วิดธ์ที่สูง และตัว GPU เองยังออกแบบให้สามารถแก้สมการทางคณิตศาสตร์ได้อย่างรวดเร็ว นอกจากนี้ยังมีจำนวนหน่วยประมวลผลที่มีมากกว่า CPU หลายเท่าตัว จึงทำให้มีอัตราการประมวลผลที่สูงกว่า CPU และยังมีแพลตฟอร์มของ Nvidia ที่รองรับอย่าง CUDA ซึ่งเป็น Parallel Computing แพลตฟอร์มเพื่อช่วยให้นักพัฒนาสร้าง Tools ในการเรียกใช้การประมวลผลของ GPU และยังมี library อย่าง NVDIA cuDNN ซึ่งรองรับการทำ Deep Neural Network โดยตัว cuDNN ได้อำนวยการปรับแต่งขั้นสูงสำหรับการทำงานของ DNN เช่น forward และ backward convolution pooling normalization activation layers เป็นต้น

2.2.2 ทฤษฎี Rule of Thumb ในการหาจำนวนของ Hidden Layer

การตัดสินใจเลือกจำนวน Neurons ใน Hidden Layers นั้นถือเป็นส่วนสำคัญในการตัดสินภาพรวมของสถาปัตยกรรมโครงข่ายประสาทเทียม โดย Hidden Layers นั้นจะไม่ค่อยมีผลกับองค์ประกอบภายนอกแต่จะมีผลอย่างมากกับผลลัทธ์ที่จะออกมา จึงทำให้การกำหนดจำนวน Hidden Layers และ จำนวน Neurons ใน Hidden Layers นั้นต้องพิจาราณาอย่างระมัดระวัง เพราะถ้าเราใช้จำนวน Neurons น้อยเกินไปผลลัพท์ก็จะเกิดปัญหาที่เรียกว่า Underfitting โดยจะเกิดขึ้นเมื่อมีจำนวน Neurons ใน Hidden Layers น้อยเกินไปจนไม่สามารถตรวจจับสัญญาณในข้อมูลที่ซับซ้อนได้อย่างเพียงพอ แต่ในทางกลับกันหากเราใช้จำนวน Neurons มากเกินก็จะเกิดปัญหาหลายอย่างตามมาโดยอย่างแรกก็คือ Ovefitting โดยจะเกิดขึ้นเมื่อความจุของข้อมูลที่จะประมวลผลมีมากเกินไป ซึ่งจะไปจำกัดข้อมูลที่จะอยู่ในชุดฝึกสอนทำให้ไม่เพียงพอต่อการเรียนรู้ของ Neurons ใน Hidden Layer ปัญหาที่สองนั้นก็สามารถเกิดขึ้นมาได้แม้จะมีการเรียนรู้ของข้อมูลเพียงพอแล้วก็ตาม เนื่องด้วยจำนวน Neurons ที่มากเกินไปนั้นจะทำใช้เวลาในการเรียนรู้เพิ่มขึ้น ซึ่งเวลาในการเรียนรู้ที่เพิ่มขึ้นนั้นสามารถเพิ่มไปถึงจุดที่ทำให้การเรียนรู้ไม่สามารถทำได้อย่างเพียงพอ ดังนั้นทำให้ต้องการกำหนดจำนวน Neurons ที่ไม่น้อยเกินไปหรือมากเกินไป โดยมีหลักการอย่างง่ายในการกำหนดจำนวน Neurons ตามนี้

จำนวน Neurons ควรอยู่ในช่วงขนาดของ Input Layer และ Output Layer

จำนวน Neurons ควรมีขนาดเป็น 2 : 3 ของขนาด Input layer รวมกับ Output layer

จำนวน Neurons ควรมีขนาดน้อยกว่า 2 เท่าของขนาด Input Layer

โดยกฎทั้งสามที่ยกมานั้นเป็นเพียงส่วนหนึ่งในตัวเลือกให้สามารถนำไปใช้เพื่อให้ไม่ต้องมาสุ่มจำนวน Neurons ใหม่ซึ่งเท่าทำให้ไม่เสียเวลาที่ต้องนำไปทดลองกับจำนวน Neurons ที่สุ่มขึ้นใหม่

บทที่ 3

วิธีการดำเนินการวิจัย

การดำเนินการวิจัยการสร้างชุดข้อมูลในการฝึกสอนไฟร์วอลล์ปัญญาประดิษฐ์ด้วยเทคโนโลยีโครงข่ายประสาทเทียมจากกฎของไฟร์วอลล์ มีจุดประสงค์เพื่อพัฒนาชุดข้อมูลฝึกสอนที่สร้างจากกฎของไฟร์วอลล์ เพื่อให้ชุดข้อมูลฝึกสอนสามารถสอนโมเดลได้ถูกต้องและแม่นยำอย่างมีประสิทธิภาพ

3.1 การศึกษาค้นคว้าเทคโนโลยีและเครื่องมือที่ใช้ในการพัฒนาโมเดล

ในการดำเนินการวิจัย ผู้จัดทำเลือกที่จะใช้ Python เป็นภาษาหลักในการพัฒนาโปรแกรมสร้างชุดข้อมูลฝึกสอนและส่วนของโมเดลเรียนรู้ ดังนั้นเพื่อให้การทำงานและการใช้งานเป็นไปตามที่งานวิจัยต้องการ จึงจำเป็นต้องศึกษาความเข้ากันได้ของเครื่องมือและไลบรารีที่เกี่ยวข้องในการพัฒนา

- Anaconda3 โปรแกรมจัดการแพ็คเกจและสร้าง Environment ที่จำเป็นในการเขียนซอฟต์แวร์ภาษา Python เหมาะแก่งาน Data Visualization, Machine Learning, Neural Network และยังสามารถใช้งานร่วมกันกับ IDE ได้หลากหลาย

  Version: Anaconda 3.8 64-Bit
- Spyder โปรแกรมพัฒนาซอฟต์แวร์ด้วยภาษา Python สามารถตรวจสอบตัวแปรได้ง่าย

  Version: Spyder 4.1.4
- TensorFlow ไลบราลีพื้นฐานในการพัฒนา Neural Network Model

  Version: TensorFlow 2.3.0 สามารถใช้ได้กับ Python 64-Bit เท่านั้น
- Sklearn เป็นเครื่องมือสำคัญในการทำ Model Selection และ Data Preprocessing ทำงานโดยพื้นฐานของ Numpy

  Version: Scikit-learn 0.23.2
- Keras เป็น Deep Learning Framework ที่สำคัญ อีกทั้งสามารถประมวลผลได้ทั้ง CPU และ GPU

  Version: Keras 2.4.3
- Pandas เป็นไลบรารีช่วยในการจัดกลุ่ม แยกประเภทข้อมูลกลุ่มโครงสร้างเช่น ไฟล์นามสกุล csv

  Version: Pandas 1.1.2
- Pip เครื่องมือที่ช่วยในการติดตั้งแพ็คเกจในภาษา Python

  Version: pip 20.2.3

- Tkinter ไลบรารีพัฒนาการสร้าง GUI ด้วยภาษา Python

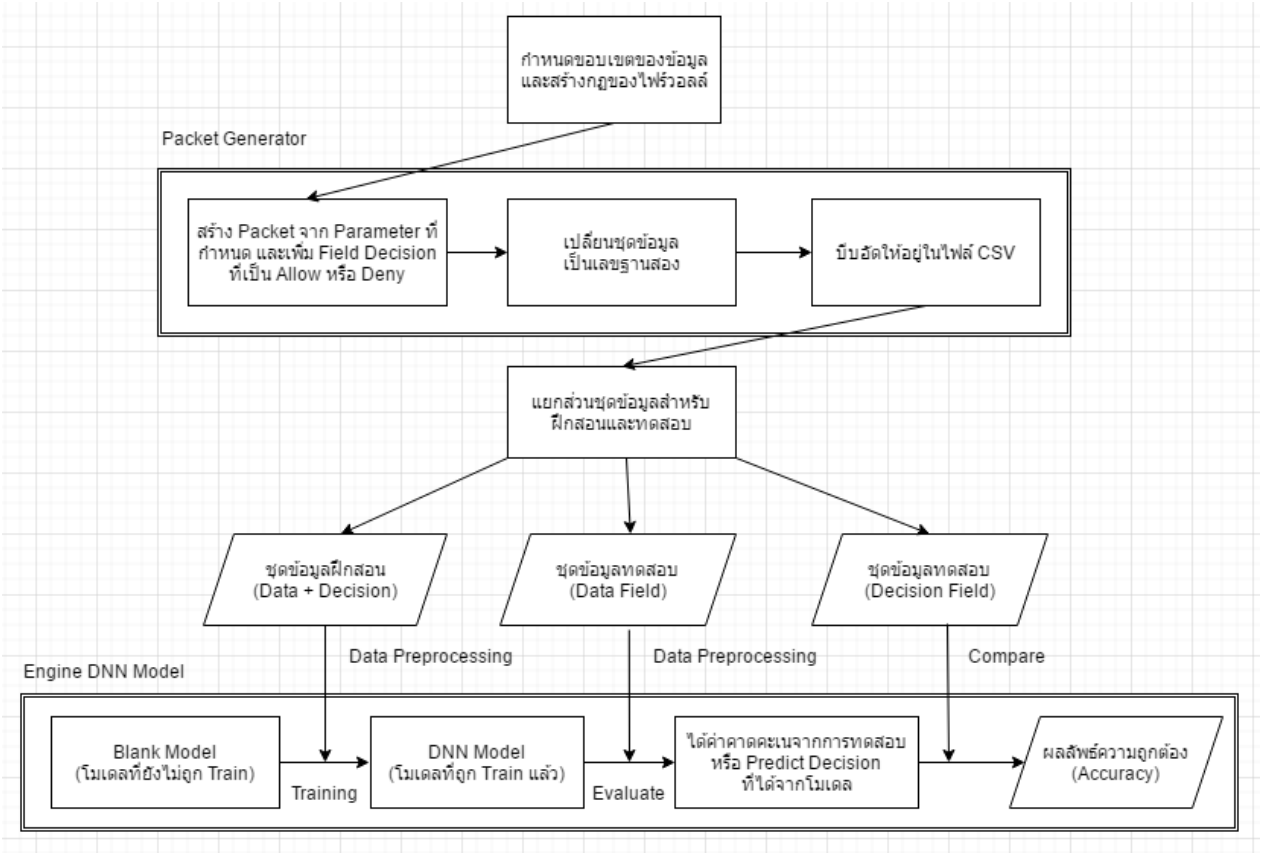
  Version: Tk 8.6.10
- NVIDIA CUDA เครื่องมือช่วยให้คอมพิวเตอร์สามารถประมวลผลผ่าน GPU ได้

  Version: CUDA 11.1.0
- NVIDIA cuDNN เครื่องมือช่วยในการประมวลผล DNN ผ่าน GPU

  Version: cuDNN 8.0

3.2 กระบวนการพัฒนาชุดข้อมูลฝึกสอน Training model

ในการวิจัยจะมุ่งเน้นไปที่การพัฒนาชุดข้อมูลฝึกสอน โดยการเปรียบเทียบหาผลลัพธ์จากการนำชุดฝึกสอนไปผ่านโมเดล DNN และได้ผลลัพธ์ออกมาที่มีความแม่นยำมากที่สุด ซึ่งการทดลองดังกล่าวจำเป็นต้องทำด้วยกันหลายครั้ง ซึ่งในแต่ละครั้งการทดลองก่อนการนำมาเปรียบเทียบจะมีกระบวนการดำเนินงาน ดังนี้



จากรูปภาพ ทำให้แบ่งขั้นตอนการทดลองหลักๆได้เป็น 6 ส่วนหลักตามการทำงานของโปรแกรม ได้แก่

- การกำหนดกฎของไฟร์วอลล์และความเป็นไปได้ทั้งหมดของชุดข้อมูล Packet ในเครือข่าย
- การสร้างชุดข้อมูลสำหรับการฝึกสอนและชุดข้อมูลสำหรับการทดสอบ

- การนำชุดข้อมูลฝึกสอนผ่านโมเดลเพื่อเริ่มทำการเรียนรู้
- การนำชุดข้อมูลทดสอบประมวลผลด้วยโมเดลที่ผ่านการเรียนรู้แล้ว
- การเปรียบเทียบผลลัพธ์ค่าความถูกต้องของโมเดลที่ทดสอบกับชุดข้อมูลทดสอบ
- การนำผลการเปรียบเทียบแต่ละครั้งมาสรุปเพื่อหาผลลัพธ์ที่ออกมาดีที่สุด

ส่วนที่ 1 การกำหนดกฎของไฟร์วอลล์และความเป็นไปได้ทั้งหมดของชุดข้อมูล Packet ในเครือข่าย

เงื่อนไขหลักของการวิจัยคือการสร้างชุดข้อมูลฝึกสอนจากกฎของไฟร์วอลล์ เพื่อให้ได้ระบบการทำงานคัดกรองข้อมูล Packet ที่ได้มาตรฐานและเรียนรู้ได้เองอย่างมีประสิทธิภาพ ความแม่นยำสูง สิ่งที่ต้องทำในส่วนแรกคือการกำหนดขอบเขตความเป็นไปได้ที่ข้อมูลจะสามารถเกิดขึ้นในเครือข่าย และการกำหนดกฎของไฟร์วอลล์เพื่อให้สามารถสร้างชุดข้อมูล Packet ที่จะนำไปฝึกสอนให้กับโมเดล สร้างชุดข้อมูลทดสอบโมเดล ที่สามารถเปรียบเทียบความถูกต้องของผลลัพธ์ที่ได้จากโมเดลหลังผ่านการเรียนรู้แล้ว

การกำหนดขอบเขตความเป็นไปได้ที่จะเกิดชุดข้อมูล Packet ใดๆ จำเป็นต้องรู้ส่วนประกอบทั้งหมดและค่าความเป็นไปได้ของแต่ละ Label ที่จะนำมาพิจารณา เพื่อมาคำนวณต่อหา Sample Space หรือโอกาสที่เกิดขึ้น ถ้าหากมีข้อมูลภายใน Field เพียงชุดเดียวที่แตกต่างกัน ชุดข้อมูล Packet นั้นจะเหมือนเป็นชุดข้อมูลใหม่ แต่ถึงกระนั้นจะต้องดูความเข้ากันได้ของข้อมูลด้วย และข้อมูลนั้นจะต้องสามารถเกิดขึ้นได้จริง ยกตัวอย่างเช่น ผู้รับและส่งไม่สามารถเป็น IP Address เดียวกันได้ หรือโปรโตคอล FTP จะต้องจับคู่กันระหว่าง Port 21 และ Port 22 เท่านั้น เป็นต้น

จากการแจกแจงความเป็นไปได้ของข้อมูลใน Field ทำให้ได้ส่วนประกอบของ Packet ดังนี้

- Source Address
  ความเป็นไปได้ขึ้นอยู่กับ mask เช่น /24 จะเป็นไปได้ทั้งหมด 2^(32-24) ความเป็นไปได้
- Source Port
  ความเป็นได้ขึ้นอยู่กับจำนวน port ใน pull ที่กำหนดไว้
- Destination Address
  ความเป็นไปได้ขึ้นอยู่กับ mask เช่น /24 จะเป็นไปได้ทั้งหมด 2^(32-24) ความเป็นไปได้
- Destination Port
  ความเป็นได้ขึ้นอยู่กับจำนวน port ใน pull ที่กำหนดไว้
- Protocol
  ประกอบไปด้วย TCP และ UDP

ขั้นตอนต่อมาคือการสร้างกฎของไฟร์วอลล์ ในขั้นตอนนี้จะเป็นการกำหนดกระบวนการทำ Packet Filtering ที่จะเป็นการตัดสินว่า ข้อมูล Packet ชุดดังกล่าวจะสามารถถูกตัดสินให้ผ่านหรือไม่ ซึ่ง Packet ทุกชุดจะถูกตรวจสอบในทุกกฎของไฟร์วอลล์โดยมี 2 คำสั่งหลัก ได้แก่ Allow ปล่อยผ่านหรือ Deny ไม่ปล่อยให้ผ่าน ในขั้นตอนนี้จะสำคัญมากในขั้นตอนการสร้างชุดข้อมูลที่ใช้ในการฝึกสอนและการทดสอบ

ตัวอย่างของการสร้างกฎของไฟร์วอลล์

Allow 192.168.0.0 192.168.1.0 any

เมื่อสร้างกฎของไฟร์วอลล์เป็นที่เรียบร้อยแล้ว จะต้องนำค่าความเป็นไปได้และกฎของไฟร์วอลล์ที่ตั้งไว้ไปเป็น Parameter ในโปรแกรม Packet Generator

ส่วนที่ 2 การสร้างชุดข้อมูลสำหรับการฝึกสอนและชุดข้อมูลสำหรับการทดสอบ

การสร้างชุดข้อมูลฝึกสอนและชุดข้อมูลทดสอบจะถูกสร้างโดยโปรแกรม Packet Generator ที่สร้างขึ้นเอง ข้อมูลที่ถูกสร้างขึ้นจะถูกจัดระเบียบอยู่ใน Cell ของไฟล์นามสกุล CSV ทำให้ง่ายแก่การดึงข้อมูลกลับมาใช้ต่อในขั้นตอนถัดไป แต่ก่อนที่จะสร้างชุดข้อมูล Packet นั้นจะต้องทราบความต้องการและจุดประสงค์ของโมเดล ว่าโมเดลดังกล่าวมีการต้องการชุดข้อมูลที่มีความสัมพันธ์และมีจำนวน Input และ Output อย่างไร การสร้างชุดข้อมูล Packet จะถูกคำนวณจากความเป็นไปได้ทั้งหมดของชุดข้อมูล Packet ทั้งหมด และหลังจากนั้นจะเป็นการเพิ่ม Decision Field เข้าไปในชุดข้อมูล Packet แต่ละชุด เพื่อให้โมเดลนำไปเข้ากระบวนการเรียนรู้ และเปรียบเทียบผลลัพธ์ในขั้นตอนหลังการทดสอบ (Evaluate) หาค่า Reference Variant Set โดยค่าภายใน Decision Field จะถูกสร้างอ้างอิงกับกฎของไฟร์วอลล์ในขั้นตอนแรก

- Decision Field

  Allow แทนค่า เป็น 1

  Deny แทนค่า เป็น 0

ในขั้นตอนนี้จะได้ผลลัพธ์ออกมาเป็นไฟล์นามสกุล CSV ที่ประกอบด้วย Packet จำนวนมาก ที่มี Decision Field ในการตัดสินใจว่าชุดข้อมูล Packet นั้นจะสามารถถูกตัดสินให้ผ่านไปได้หรือไม่

ส่วนที่ 3 การนำชุดข้อมูลฝึกสอนผ่านโมเดลเพื่อเริ่มทำการเรียนรู้

เป็นส่วนที่ทำให้โมเดลเกิดการเรียนรู้จากชุดข้อมูลฝึกสอน Packet ที่สร้างขึ้นจากกฎของไฟร์วอลล์ แบ่งส่วนข้อมูลที่จะนำมาพิจารณาและผลลัพธ์การตัดสินใจ ทำการจัดข้อมูลให้อยู่ในรูปของ Matrix ตัวโมเดลจะทำการเลือกรูปแบบที่ให้ผลลัพธ์ที่ดีที่สุดโดยวัดผลจากค่าความแม่นยำและอัตราการสูญเสียข้อมูล เมื่อวิเคราะห์จากความต้องการและจุดประสงค์การเลือกใช้ของโมเดลแล้ว ทำให้สรุปได้ว่า Sequential Model ที่มีการ Compile แบบ Binary Classification Problem สามารถตอบโจทย์ได้ดีที่สุด เนื่องจากผลลัพธ์ Output สุดท้ายจะเข้าข่ายการตัดสินใจแบบ Two-Class-Label หมายความว่าที่โมเดลจะทำการตัดสินใจจะมีเพียง 2 ตัวเลือก ซึ่งในงานวิจัยจะมีเพียง Allow หรือ Deny เท่านั้นภายในการทดสอบ

กระบวนการทำงานในขั้นตอนนี้ จะเป็นการแยกส่วนข้อมูลที่จะใช้พิจารณาแยกกันในไฟล์นามสกุล CSV ที่สร้างจากขั้นตอนที่แล้ว โดยแบ่งออกเป็นชุดข้อมูลฝึกสอนและชุดข้อมูลทดสอบในอัตราส่วนที่ได้จาก Rule of Thumb คือ 8:2 และแบ่งชุดข้อมูลดังกล่าวออกอีก ได้แก่

- ชุดข้อมูลฝึกสอน ที่ประกอบไปด้วย Data Field ภายใน Packet ทั้งหมด
- ชุดข้อมูลฝึกสอน ที่ประกอบไปด้วย Decision ที่เป็นผลลัพธ์ตัดสินใจว่าจะปล่อยผ่าน
- ชุดข้อมูลทดสอบ ที่ประกอบไปด้วย Data Field ภายใน Packet ทั้งหมด
- ชุดข้อมูลทดสอบ ที่ประกอบไปด้วย Decision ที่เป็นผลลัพธ์ตัดสินใจว่าจะปล่อยผ่าน

ต่อมาจะเป็นการทำ Data Preprocessing หรือการจัดข้อมูลชุดให้อยู่ในรูป Matrix เปลี่ยนค่าภายในในกลายเป็นค่าถ่วงน้ำหนัก เป็นค่าที่โมเดลจะนำไปเรียนรู้ต่อและหาค่าความสัมพันธ์ว่าชุดข้อมูลดังกล่าวจะถูกตัดสินว่าเป็น Allow หรือ Deny โดยชุดข้อมูลที่จะต้องนำไปทำ Data Preprocessing ได้แก่

- ชุดข้อมูลฝึกสอน ที่ประกอบไปด้วย Data Field ภายใน Packet ทั้งหมด
- ชุดข้อมูลทดสอบ ที่ประกอบไปด้วย Data Field ภายใน Packet ทั้งหมด

ต่อมาจะเป็นการทำ Compile และการเลือกสร้าง Model (กลไกการทำงานในชั้น Code จะเขียนในส่วนของโปรแกรม) ซึ่งค่า Parameter ที่เราต้องเป็นผู้กำหนดคือ จำนวน Layer, จำนวน Node ในแต่ละ Layer, ขนาดของ Batch, และจำนวนรอบ Epoch

ซึ่ง Parameter ที่เรากำหนด มีดังนี้ –

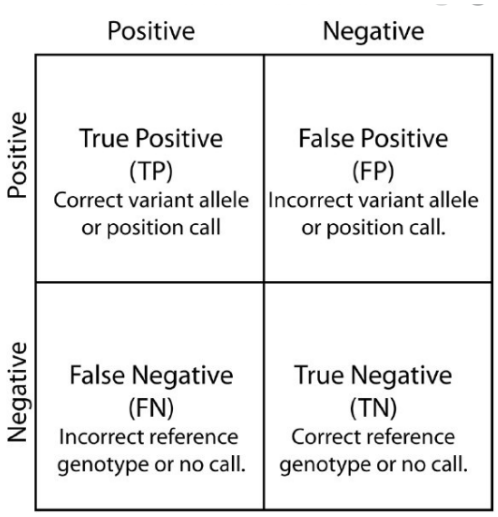
เมื่อได้โมเดลที่ผ่านการเรียนรู้แล้ว บันทึกโมเดลถือเป็นอันเสร็จสิ้น

ส่วนที่ 4 การนำชุดข้อมูลทดสอบประมวลผลด้วยโมเดลที่ผ่านการเรียนรู้แล้ว

การทดสอบหรือ Evaluate เป็นส่วนที่โมเดลที่ผ่านการเรียนรู้แล้วเริ่มทดสอบกับชุดข้อมูลทดสอบที่ประกอบไปด้วย Data Field ภายใน Packet เข้าฟังก์ชั่นการทำงาน model predict จะได้เป็นค่าคาดคะเนว่าจะเกิดขึ้นระหว่าง Allow หรือ Deny มากกว่ากัน ค่าที่เห็นจากตัวแปรจะเป็นตัวเลขทศนิยมที่อยู่ระหว่าง 0 ถึง 1 และเมื่อเข้าฟังก์ชั่น model.predict_class จะเป็นการให้โมเดลอ่านค่าผลลัพธ์ออกมาเป็นแค่ 0 หรือ 1 ในขั้นตอนนี้จะต้องมีการจับเวลาเพื่อหาความสัมพันธ์ระหว่างเวลาและจำนวนของข้อมูลด้วย

ส่วนที่ 5 การเปรียบเทียบผลลัพธ์ค่าความถูกต้องของโมเดลที่ทดสอบกับชุดข้อมูลทดสอบ

เมื่อการขั้นตอนของการทดสอบเสร็จสิ้น ให้นำค่าที่ได้จากขั้นตอนที่ 4 มาเปรียบเทียบกับชุดข้อมูลทดสอบที่มีเพียง Decision Field กระบวนการนี้จะเป็นการเปรียบเทียบว่ามีค่าตรงกันหรือไม่ และเมื่อเปรียบเทียบผลลัพธ์มีโอกาสออกมา 4 รูปแบบด้วยกัน ซึ่งทำให้ไปอ้างอิงกับการคำนวณผลลัพธ์ต่อได้ว่า มีการตัดสินใจออกมาเป็นอย่างไรตามหลัก Reference Variant Set



ผลลัพธ์ที่ได้จะประกอบไปทั้งหมด 4 ค่า ได้แก่

True Positive โมเดลอนุญาติให้ข้อมูลผ่านตรงตามกฎของไฟร์วอลล์ ให้ Allow ถูกต้อง
True Negative โมเดลไม่อนุญาติให้ข้อมูลผ่านตรงตามกฎของไฟร์วอลล์ ให้ Deny ถูกต้อง
False Positive โมเดลอนุญาติให้ข้อมูลผ่าน ไม่ตรงตามกฎของไฟร์วอลล์ ให้ Allow ผิดพลาด
False Negative โมเดลไม่อนุญาติให้ข้อมูลผ่าน ไม่ตรงตามกฎของไฟร์วอลล์ ให้ Deny ผิดพลาด

ผลลัพธ์ที่ได้จะเป็นไปตามสูตร

ความแม่นยำ (Accuracy) = SUM(TP, TN) / SUM(TP, TN, FP, FN)

ส่วนที่ 6 การนำผลการเปรียบเทียบแต่ละครั้งมาสรุปเพื่อหาผลลัพธ์ที่ออกมาดีที่สุด

เป็นการนำผลลัพธ์ของการทดสอบในแต่ละครั้งของการทดลองมาบันทึกผล แล้วสรุปให้อยู่ในรูปกราฟที่ประกอบไปด้วยผลลัพธ์จากการทดลองภายใต้สภาพแวดล้อมเดียวกัน

ตัวแปรที่มีการเปลี่ยนค่าไปตามการทดลอง

- จำนวนของ Packet ที่นำเข้าระบบ หรือ Sample(N)
- จำนวน Node ของแต่ละ Layer
- จำนวนรอบการทดสอบ หรือ Epoch

ผลลัพธ์ที่ค่าจะต้องเปลี่ยนแปลงไปตามการทดสอบแต่ละครั้ง

- เวลาที่โมเดลใช้ในการเรียนรู้จากชุดข้อมูลฝึกสอน หรือ Training
- เวลาที่โมเดลใช้ในการตัดสินใจจากชุดข้อมูลทดสอบ หรือ Predict
- ค่าความแม่นยำ หรือ Accuracy
- ค่าอัตราการสูญเสีย หรือ Loss
- อัตราความผิดพลาดที่อ้างอิงจาก Reference Variant Set

3.3 กระบวนการสร้างโปรแกรมและเครื่องมือที่เกี่ยวข้อง

Packet Generator

เป็นโปรแกรมที่ใช้ในการสร้างชุดข้อมูล Packet โดยสุ่มจากพารามิเตอร์ที่กำหนดจากกฎของไฟร์วอลล์โดยชุดข้อมูลที่ได้จากการสุ่มจะถูกนำไปแปลงค่าข้อมูลเป็นเลขฐานสอง บันทึกเก็บไว้ในไฟล์นามสกุล CSV ก่อนจะนำไปเรียกใช้ต่อในโมเดล Depp Neural Network โดยโปรแกรมนี้จะถูกแบ่งไปใช้ในการทำงาน 2 ส่วน ได้แก่ ส่วนที่ใช้ในการสร้างชุดข้อมูลฝึกสอน และ ส่วนที่ใช้ในการสร้างชุดข้อมูลทดสอบ

Deep Neural Network Model Engine

เป็นเครื่องมือสร้าง Artificial Intelligent ที่พัฒนาขึ้นเอง โดยพัฒนาและประยุกต์โมเดลให้สามารถเรียนรู้กับชุดข้อมูลฝึกสอนที่ป้อนเข้าไป นำไปประมวลผล ตัดสินใจได้ว่าจะชุดข้อมูลที่ป้อนค่าเข้าไปนั้นเป็น Allow หรือ Deny และทำการตรวจสอบผลลัพธ์ที่ได้

บทที่ 4

ผลการวิเคราะห์ข้อมูล

บทที่ 5

ผลการดำเนินงานวิจัย

บทที่ 6

บทสรุปและข้อเสนอแนะ

บรรณานุกรม

[1] TensorFlow Teams. “Essential Documentation” [Online]. เข้าถึงได้จาก : https://www.tensorflow.org/guide. 2560

[2] สมาคมโปรแกรมเมอร์แห่งประเทศไทย. “Artificial Intelligent” [Online]. เข้าถึงได้จาก : https://www.thaiprogrammer.org/2018/12/whatisai/

[3] Garry Fairhurst. “IPv4 Packet header Datagram” [Online]. เข้าถึงได้จาก : https://networklessons.com/cisco/ccna-routing-switching-icnd1-100-105/ipv4-packet-header

[4] Sci-kit learn developers. “scikit classification model” [Online]. เข้าถึงได้จาก : https://scikit-learn.org/stable/search.html?q=classification

[5] พื้นฐาน Deep Learning. [Online]. เข้าถึงได้จาก : https://www.tensorflow.org/guide. 2560

[6] Sinlapachai Lorpaiboon. “การใช้ Pandas ในการจัดระเบียบข้อมูลใน Python” [Online]. เข้าถึงได้จาก : https://medium.com/@sinlapachai.hon/เรียนรู้วิธีการใช้งาน-Pandas-ใน-Python

ภาคผนวก

ขั้นตอนการใช้งานโปรแกรม Packet Generator

ขั้นตอนการติดตั้งและการเรียกใช้งานโมเดล

ประวัติผู้เขียน