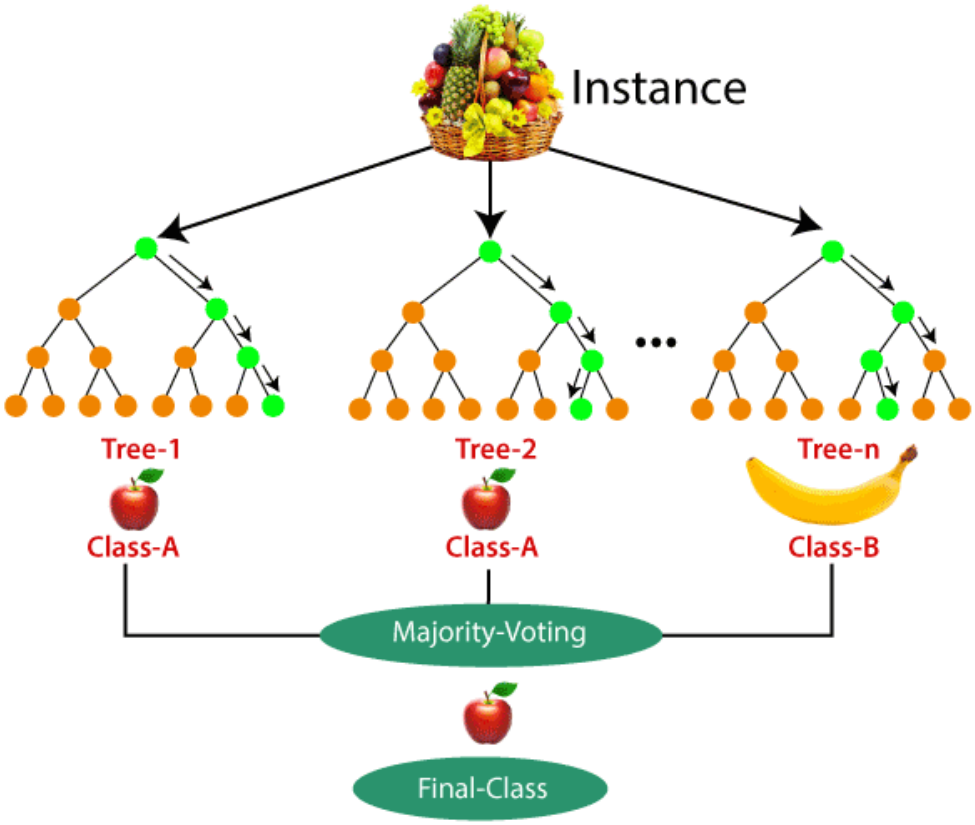
1.รับข้อมูลตะกร้าผลไม้จากผู้ใช้งาน

2.นำข้อมูลตะกร้าผลไม้จากผู้ใช้งานส่งให้อัลกอริทึม Decision Tree ที่อยู่ภายใน

3.อัลกอริทึม Decision Tree ประมวลผลลัพธ์ที่ดีที่สุดออกมาจากชุดข้อมูลตะกร้าผลไม้

4.อัลกอริทึม Random Forest ทำการโหวตผลลัพธ์ที่อลักอริทึม Decision Tree แสดงออกมามากที่สุด

5.แสดงผลลัพธ์ผลไม้ที่เลือกออกมาตามที่ข้อมูลที่ได้รับเข้าไป



รูปที่ 2.12 การแสดงการเลือกผลลัพธ์ของอัลกอริทึม Random Forest [5]

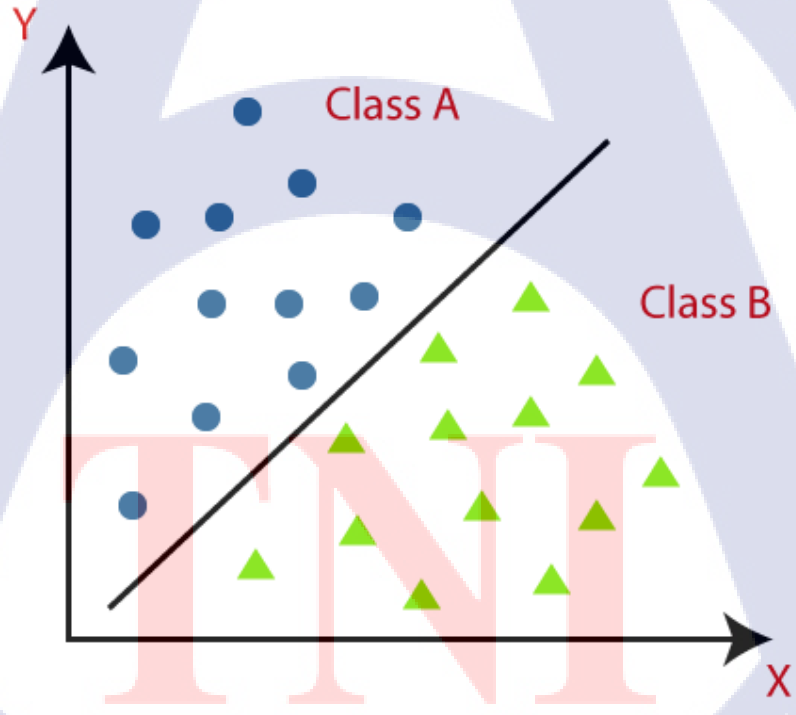
2.4.3 ประเภทของอัลกอริทึม Random Forest

อัลกอริทึม Random Forrest นั้นถูกใช้ในงานต่าง ๆ อย่างแพร่หลายไม่ว่าจะเป็นทางคณิตศาสตร์, วิทยาศาสตร์, สังคมศาสตร์, ภูมิศาสตร์ และ อาจจะใช้ในงานอื่นๆที่สอดคล้องกัน แต่เนื่องจากว่าในงานแต่ละประเภทนั้นมีรายละเอียดที่แตกต่างกันอย่างสิ้นเชิง ดังนั้นแล้วอัลกอริทึม

Random Forest จึงถูกแบ่งแยกประเภทออกมา 2 ประเภท เพื่อที่จะทำให้สามารถใช้ประโยชน์ในงานต่างๆได้อย่างสอดคล้องและทำให้เกิดประสิทธิภาพสูงสุด ซึ่งประภาพที่อัลกอริทึม Random Forest ถูกแบ่งมานั้นมีอยู่ 2 ประเภทหลักๆ ได้แก่ 1. Random Forest Classifier และ 2. Random Forest Regression โดยที่ผู้วิจัยจะอธิบายอัลกอริทึม Random Forest ทั้งสองชนิดออกมา ดังนี้

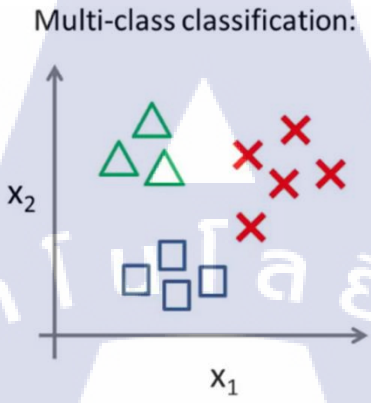
2.4.3.1 Random Forest Classifier

อัลกอริทึม Random Forrest Classifier นั้นถูกใช้ในงาน machine learning ประเภท classification โดยที่ machine learning classifier คือการที่เรานำตัวปัญญาประดิษฐ์ที่ได้พัฒนาขึ้นมาทำการ classifier วัตถุต่างๆ ไม่ว่าจะเป็น การจำแนกสี, การจำแนกกลิ่น, การจำแนกรสชาติอาหาร หรือแม้กระทั่งการจำแนกข้อมูลต่าง ๆ ที่อยู่ในคอมพิวเตอร์ว่าข้อมูลนั้นเป็นข้อมูลประเภทใดเป็นต้น ซึ่งตัวอัลกอริทึม Random Forest Classifier นั้นก็ทำงานโดยการใช้วิธีการจำแนกวัตถุ หรือข้อมูลต่าง ๆ โดยผ่านอัลกอริทึม Random Forest นั่นเอง



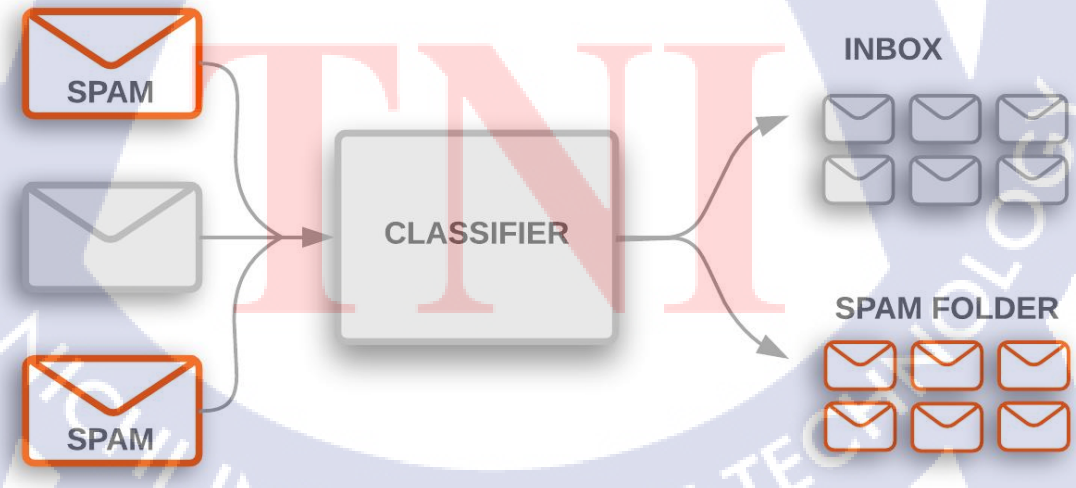
รูปที่ 2.13 การแสดงถึงการ classifier รูปเรขาคณิตจำนวน 2 ชนิด [5]

จาก รูปที่ 2.13 นั้นได้แสดงถึงการใช้ machine learning ในการระบุประเภทของรูปทรงเลขาคณิตโดยใช้หลักการ machine learning classifier มาช่วยในการแบ่งรูปวงกลมและรูปสามเหลี่ยมออกจากกัน โดยใช้วิธีการแบ่งคลาสแบบ Multi-Class Classification



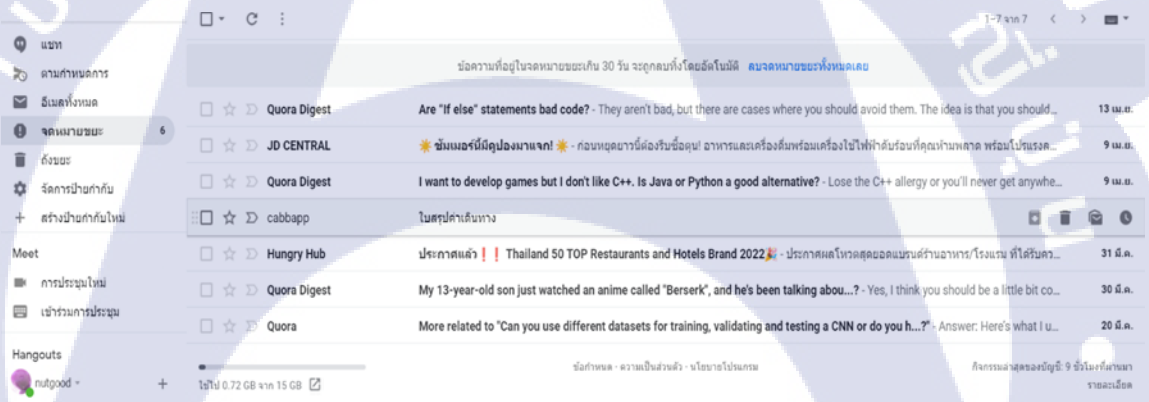
รูปที่ 2.14 การแสดงถึงการ classifier วัตถุสามชนิด [5]

จาก รูปที่ 2.14 จะเห็นได้ว่า machine learning นั้นสามารถ classifier วัตถุต่างๆที่อยู่คนละหมวดหมู่กันได้อย่างถูกต้อง โดยมันสามารถแยก สามเหลี่ยม, สี่เหลี่ยม, และตัวอักษร X ออกจากกันโดยวัตถุเหล่านี้นั้นไม่ได้อยู่ในหมวดหมู่เดียวกัน



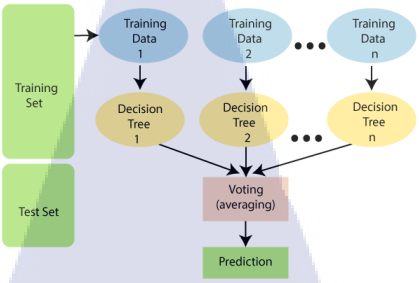
รูปที่ 2.15 การใช้ classifier คัดแยก Spam E-mail [5]

จาก รูปที่ 2.15 นั้นแสดงถึงการใช้ Machine Learning ในการทำการ classified ตัว Spam E-mail ในกล่องขาเข้าของผู้ใช้งานออกจาก E-mail ปกติที่ผู้ใช้งานได้รับเข้ามาโดยใช้ Machine Learning มาช่วยในการคัดแยก E-mail ที่ต้องสงสัยว่าเป็น Spam ออกจาก E-mail ที่เป็น Email ทั่วไป โดยหลักการของการแยกประเภทของ E-mail นั้น ผู้พัฒนาได้ทำการออกแบบปัญญาประดิษฐ์และนำปัญญาประดิษฐ์ที่ออกแบบขึ้น มาพัฒนาเป็นปัญญาประดิษฐ์ที่มีความสามารถในการตรวจจับ Spam E-mail หรือ ใช้ตรวจจับ E-mail ที่น่าสงสัยที่อาจจะทำให้ข้อมูลของผู้ใช้งานนั้นเสียหายโดยการรับ E-mail ที่เป็นการโจมตีทางไซเบอร์เป็นต้น โดยปัญญาประดิษฐ์ตัวนี้จะทำหน้าที่เป็นตัวกรอง E-mail บนเครือข่ายอินเทอร์เน็ตของผู้ให้บริการโดยที่มันจะถูกติดตั้งไว้ที่ Firewall ของเครือข่ายของผู้ให้บริการ และจากนั้นมันจะทำหน้าที่คัดกรอง E-mail ที่เป็นปกติลงในกล่องขาเข้าและคัดกรอง E-mail ที่ผิดปกติหรือที่อาจจะเป็น Spam ลงใน Spam Folder ดัง รูปที่ 2.16

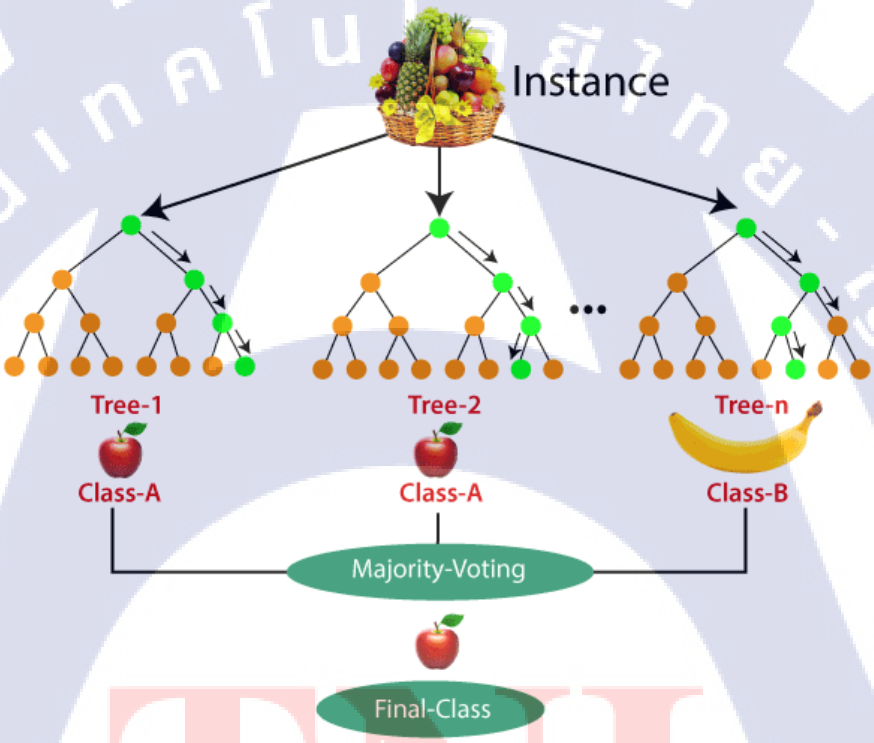


รูปที่ 2.16 ตัวอย่าง Spam E-mail ที่ถูกคัดกรองลงในจดหมายขยะ

จากที่ได้อธิบายหลักการของการ Classifier นั้น อัลกอริทึม Random Forrest Classifier ก็มีหลักการเช่นเดียวกันคือการรับเข้ามูลเข้ามา โดยข้อมูลนั้นเป็นข้อมูลชนิดที่สามารถทำการแยกประเภทหรือจัดหมวดหมู่ได้ไม่ต่างจากข้อมูลที่อัลกอริทึม Machine Learning จะได้รับ



รูปที่ 2.17 แบบจำลองอัลกอริทึม Random Forest Classifier [5]



รูปที่ 2.18 แบบจำลองการทำงานของ Random Forest Classifier [6]

จาก รูปที่ 2.17 และ 2.18 จะแสดงถึงการทำงานของอัลกอริทึม Random Forrest Classifier โดยขั้นตอนการทำงานนั้นจะแบ่งออกเป็นขั้นตอนดังนี้คือ

1.ผู้ใช้งานจะนำชุดข้อมูลที่เป็นชุดข้อมูลประเภทที่อัลกอริทึมนั้นสามารถ Classified ได้มาใช้ทำการฝึกสอน

2.อัลกอริทึมจะทำการรับข้อมูลจากผู้ใช้งานเพื่อนำไปส่งต่อให้กับอัลกอริทึม Decision Tree

3.อัลกอริทึม Decision Tree จะทำการเรียนรู้จากข้อมูลที่ได้รับ

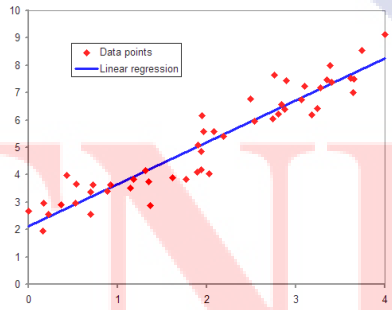
4.เมื่ออัลกอริทึม Decision Tree เรียนรู้ข้อมูลเสร็จแล้วมันจะทำการทดสอบและส่งผลลัพธ์ของข้อมูลออกมา

5.เมื่อเหล่าอัลกอริทึม Decision Tree ส่งผลลัพธ์ของข้อมูลออกมาแล้ว อัลกอริทึม Random Forrest จะทำการโหวตผลลัพธ์ที่จะแสดง โดยใช้วิธีการนำผลลัพธ์ที่เหล่าอัลกอริทึม Decision Tree แสดงออกมามากที่สุด

กล่าวโดยสรุปคือ อัลกอริทึม Random Forest Classifier นั้นเป็นกลุ่มของต้นไม้ที่มีความสามารถในการทำนายผลลัพธ์ โดยที่ผลลัพธ์ของกลุ่มต้นไม้เหล่านั้นจะขึ้นอยู่กับเวกเตอร์สุ่มตัวอย่างแบบอิสระซึ่งเวกเตอร์เหล่านี้จะกระจายตัวอยู่ในต้นไม้ทุกต้นในอัลกอริทึม Random Forest โดยที่อัลกอริธึม Random Forrest จะทำการเลือกค่าตัวอย่างจากชุดข้อมูลที่ได้รับการโหวตสูงสุดจากต้นไม้ที่อยู่ภายในอัลกอริทึมว่าเป็นค่าที่มีความถูกต้องมากที่สุดที่ถูกเลือกออกมา

2.4.3.2 Random Forest Regressor

อัลกอริทึม Random Forest Regressor นั้น เป็นอัลกอริทึม Random Forest ที่ถูกใช้ส่วนของงาน Regression ต่าง ๆ โดยอัลกอริทึม Random Forest จะใช้วิธีการ Ensemble method เข้ามาช่วยในการทำงานในด้านของการ Regression แต่ว่าอัลกอริทึม Random Forest Regressor นั้นก็มีจุดด้อยจนทำให้ไม่ค่อยเป็นที่นิยมเท่าไรนักในด้านของงาน Regression แต่ว่าก่อนจะมากล่าวถึงอัลกอริทึม Random Forest Regressor ผู้วิจัยจะอธิบายเกี่ยวกับการ Regression อย่างคร่าวๆเสียก่อน

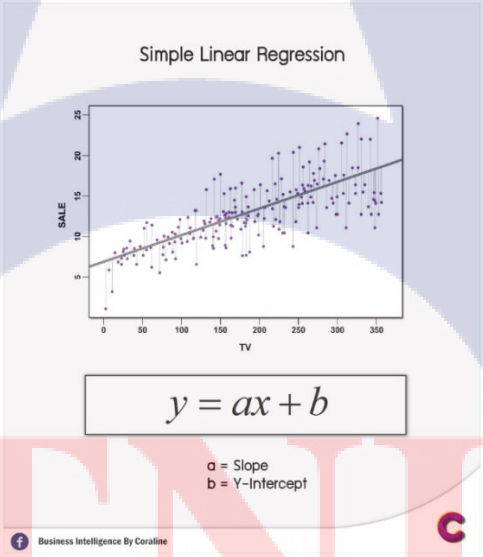


รูปที่ 2.19 การทำ Regression โดยใช้จุดพิกัด 50 จุด [6]

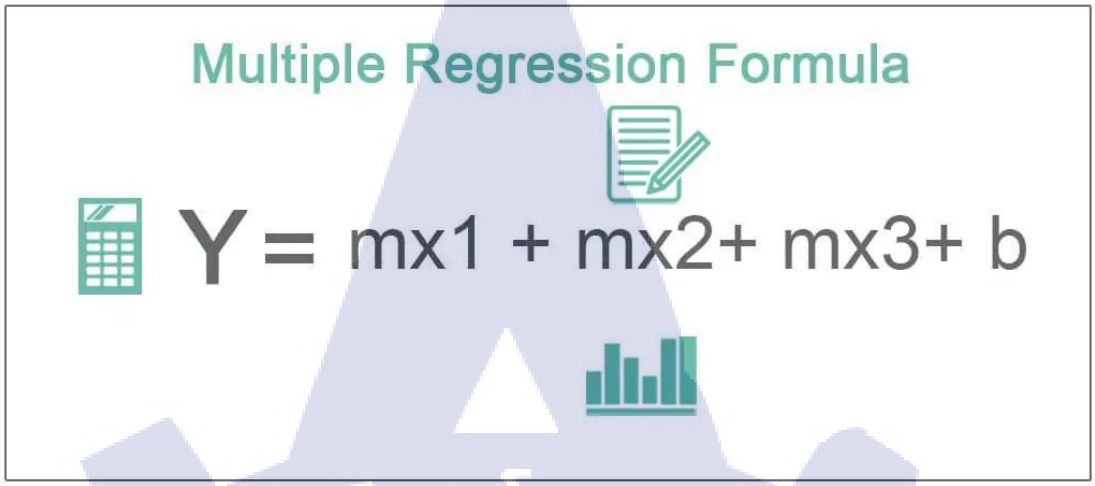
จาก รูปที่ 2.19 นั้นจะแสดงถึงการทำ Regression โดยใช้จุดพิกัดของคู่อันดับแบบสุ่มมา 50 จุด และใช้อัลกอริทึม Linear Regression เป็นตัวทำ Regressor ซึ่งงานในส่วนขของ Regression นั้นจะประกอบไปด้วยตัวแปรที่เป็นอิสระ และ ตัวแปรตาม หรือที่เราเรียกตัวแปรอิสระ

ว่า Independent Variable และตัวแปรตามเรียกว่า Dependent Variable โดยส่วนประกอบนี้จะเป็นส่วนประกอบหลักสำคัญที่สุดในการทำ Regression เพราะถ้าหากขาดตัวแปรสองชนิดนี้ไปก็ไม่สามารถสร้างจุดพิกัดเพื่อนำมาใช้ทำงาน Regression ได้ แต่ว่าองค์ประกอบของการทำ Regression ไม่ได้มีเพียงแค่ตัวแปรอิสระและตัวแปรตามเท่านั้น แต่ยังมีค่าพารามิเตอร์ที่ไม่รู้จัก หรือที่เราเรียกกันว่า Unknown Parameter ซึ่ง Parameter ตัวนี้จะถูกใช้ควบคู่ไปกับค่าตัวแปรอิสระเพื่อกำหนดผลลัพธ์ของค่าตัวแปรตาม และสุดท้ายค่าความคลาดเคลื่อนของการ Regression หรือที่เราเรียกกันว่า Error Term นั่นเอง

จาก รูปที่ 2.20 จะเห็นสมการของ Simple Linear Regression หรือที่เราเรียกว่าสมการเส้นตรงโดยที่สมการตัวนี้นั้นได้สอดคล้องกับองค์ประกอบของ Regression ซึ่งส่วนที่เป็นตัวแปรอิสระนั้นได้ถูกกำหนดไว้ให้เป็นค่า x ส่วนตัวแปรตามได้ถูกกำหนดไว้ให้เป็นค่า y และตัว Unknown Parameter กับ ตัว Error Term นั้นจะเป็นตัว a กับ b ตามลำดับซึ่ง a และ b ในรูปของสมการเส้นตรงที่อยู่ในภาพนั้นจริง ๆ แล้วถูกแทนด้วยค่าความชันและค่า Y-intercept ตามลำดับ



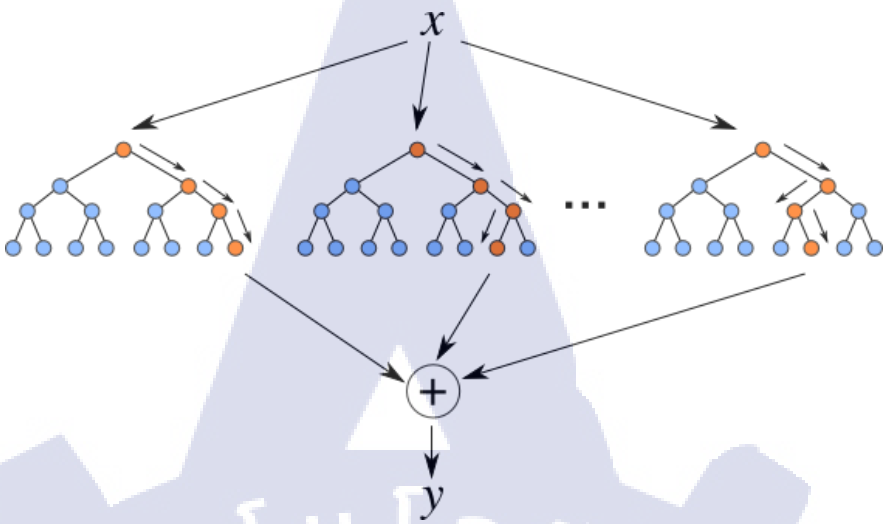
รูปที่ 2.20 Simple Linear Regression Formula [6]



รูปที่ 2.21 Multiple Regression Formula [6]

จาก รูปที่ 2.21 จะเห็นสมการของ Multiple Regression หรือที่เราเรียกว่าสมการเส้นตรงแบบทวีคูณโดยที่สมการตัวนี้นั้นได้สอดคล้องกับองค์ประกอบของ Regression ซึ่งส่วนที่เป็นตัวแปรอิสระนั้นได้ถูกกำหนดไว้ให้เป็นค่า x และตัวแปรอิสระนั้นมีอยู่หลายค่ามาก แต่ว่าตัวแปรอิสระนั้นสามารถกำหนดค่าตัวแปรตามได้ออกมาเพียงค่าเดียวเท่านั้น ส่วนตัวแปรตามได้ถูกกำหนดไว้ให้เป็นค่า y และตัว Unknown Parameter กับ ตัว Error Term นั้นจะเป็นตัว m กับ b ตามลำดับซึ่ง m และ b ในรูปของสมการเส้นตรงที่อยู่ในภาพนั้นจริง ๆ แล้วถูกแทนด้วยค่าความชันและค่า Y-intercept ตามลำดับ

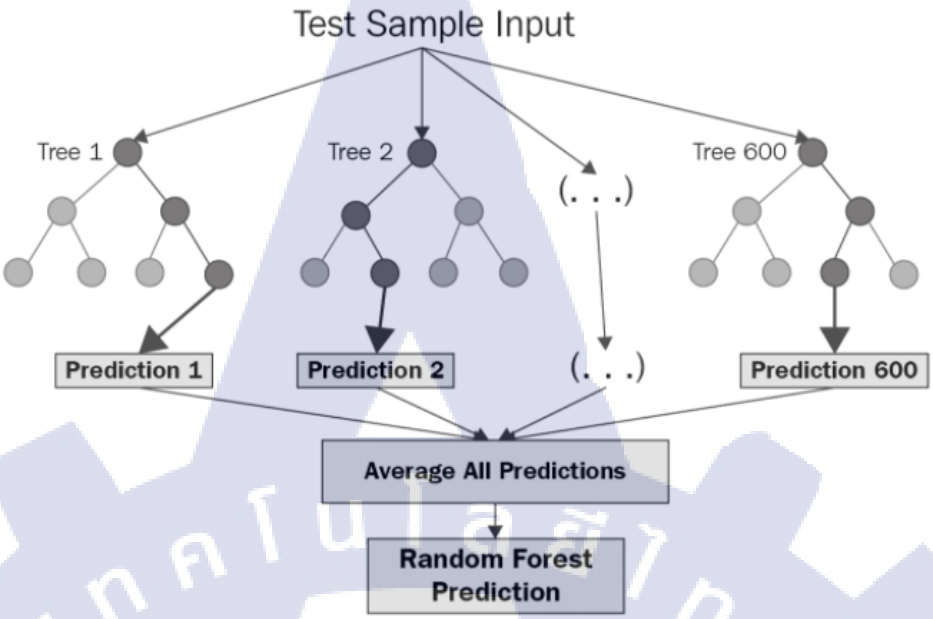
จากการอธิบายในเรื่อง Regression ของผู้วิจัยนั้นสามารถกล่าวโดยสรุปได้ว่าองค์ประกอบของ Regression นั้นมีอยู่ 4 องค์ประกอบหลักด้วยกันคือ 1. Independent Variable 2. Dependent Variable 3. Unknown Parameter 4. Error Term โดยที่องค์ประกอบเหล่านี้นั้นหากปราศจากองค์ประกอบที่ 1 และ 2 ไปแล้วนั่นก็ไม่สามารถทำให้งาน Regression เกิดขึ้นได้



รูปที่ 2.22 โครงสร้างของอัลกอริทึม Random Forest Regressor [6]

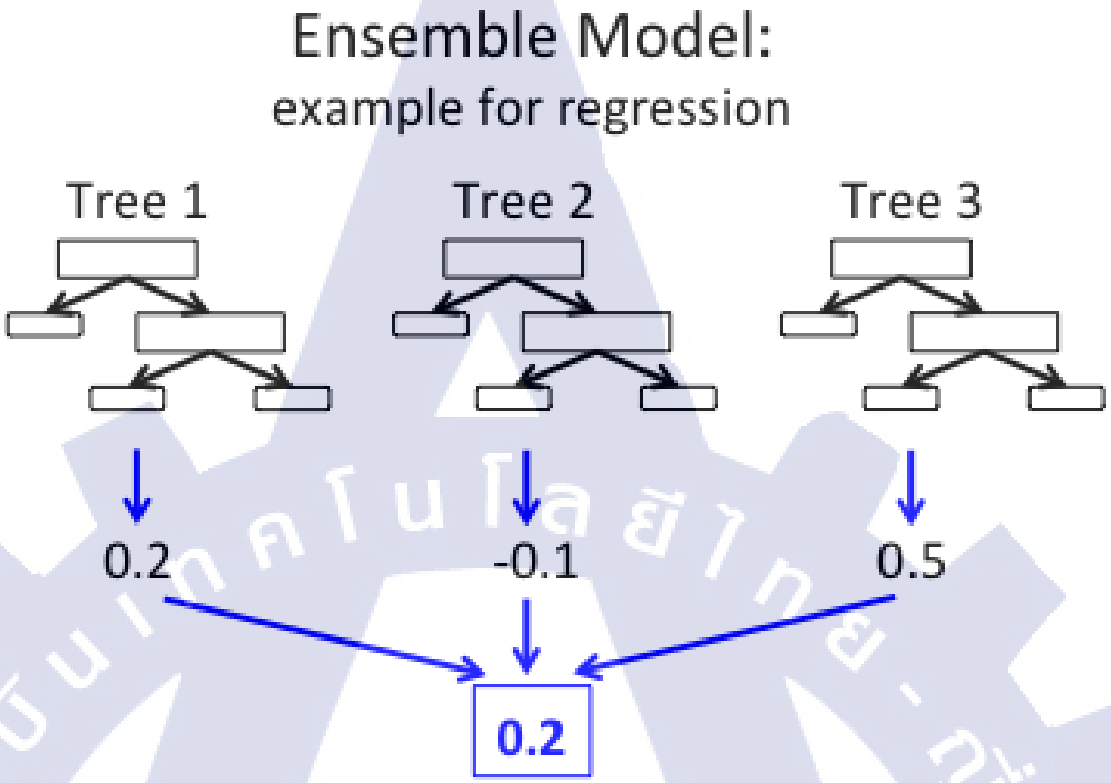
จาก รูปที่ 2.22 ที่แสดงถึงโครงสร้างของอัลกอริทึม Random Forest Regressor นั้น ได้แสดงให้เห็นถึงการกระทำในการทำงานในส่วนของ Regression ว่าเป็นอย่างไรบ้างโดยที่ตัวอัลกอริทึม Random Forest Regressor นั้นจะประกอบไปด้วยชุดข้อมูลที่เป็นตัวแปรอิสระหรือ Independent Variable จากนั้นในส่วนที่ใช้ประมวลผลจะใช้โครงสร้างของอัลกอริทึม Decision Tree เป็นหลักในการประมวลผลหรือการรับข้อมูลฝึกสอนจากชุดข้อมูลตัวแปรอิสระ และจากนั้นในส่วนประมวลผลส่วนต่อมาจะเป็นฟังก์ชั่นที่ใช้เลือกผลลัพธ์ที่ได้รับการโหวตสูงสุดจากอัลกอริทึม Decision Tree โดยผลลัพธ์นี้จะเป็นค่าตัวแปรตามหรือที่เราเรียกกันว่า Dependent Variable และส่วนสุดท้ายจะเป็นผลลัพธ์ที่ได้ถูกสรุปออกมาในรูปแบบของ Dependent Variable ว่ามีค่าออกมาเป็นอย่างไร

ในส่วนย่อหน้าต่อไปนี้ทางผู้วิจัยจะอธิบายถึงการทำงานของอัลกอริทึม Random Forest Regressor ว่าหลักการทำงานของอัลกอริทึมนี้ทำงานเป็นขั้นตอนอย่างไรโดยจะอธิบายผ่านภรูปประกอบ รูปที่ 2.23 และ รูปที่ 2.24 เพื่อทำให้ผู้อ่านนั้นมีความเข้าใจที่มากขึ้นเกี่ยวกับอัลกอริทึม Random Forest Regressor



รูปที่ 2.23 การทำงานของ Random Forest Regressor [6]

จาก รูปที่ 2.23 นั้นจะเห็นได้ว่าอัลกอริทึม Random Forest Regressor ได้รับข้อมูลมาจากชุดข้อมูลที่เป็นข้อมูลชนิด Independent Variable มาทำการสุ่มข้อมูลเพื่อส่งต่อให้กับกลุ่มของอัลกอริทึม Decision Tree มาใช้ในการฝึกสอนและทดสอบ เมื่ออัลกอริทึม Decision Tree นั้นได้รับข้อมูลแบบสุ่มจากชุดฝึกสอนแล้ว อัลกอริทึม Decision Tree จะทำการฝึกฝนและทดสอบ เมื่อเสร็จจากขั้นตอนฝึกสอนและทดสอบของอัลกอริทึม Decision Tree แล้ว อัลกอริทึม Decision Tree แต่ละต้นที่เป็นส่วนประกอบของอัลกอริทึม Random Forest Regressor จะทำการแสดงค่า Dependent Variable ออกมาเพื่อนำไปหาค่าเฉลี่ยโดยรวม และเมื่อหาค่าเฉลี่ยเสร็จอัลกอริทึม Random Forest Regressor นั้นจะทำการนำผลลัพธ์ Dependent Variable แสดงออกมา ตามตัวอย่างจาก รูปที่ 2.20



รูปที่ 2.24 ตัวอย่างการประมวลผลของอัลกอริทึม Random Forest Regressor [6]

จาก รูปที่ 2.24 จะเป็นการแสดงตัวอย่างการทำ Regression โดยการที่อัลกอริทึม Random Forest Regressor จะเป็นอัลกอริทึมที่ถูกใช้ดำเนินการในการประมวลผลข้อมูลเพื่อให้ผลลัพธ์ของการวิเคราะห์การ Regression ออกมา โดยมีขั้นตอนดังต่อไปนี้

1. อัลกอริทึม Random Forest Regressor รับข้อมูล Independent Variable มาจากชุดข้อมูลที่ทางผู้ใช้งานมอบให้

2. อัลกอริทึม Random Forest Regressor จะทำการสุ่มค่า Independent Variable

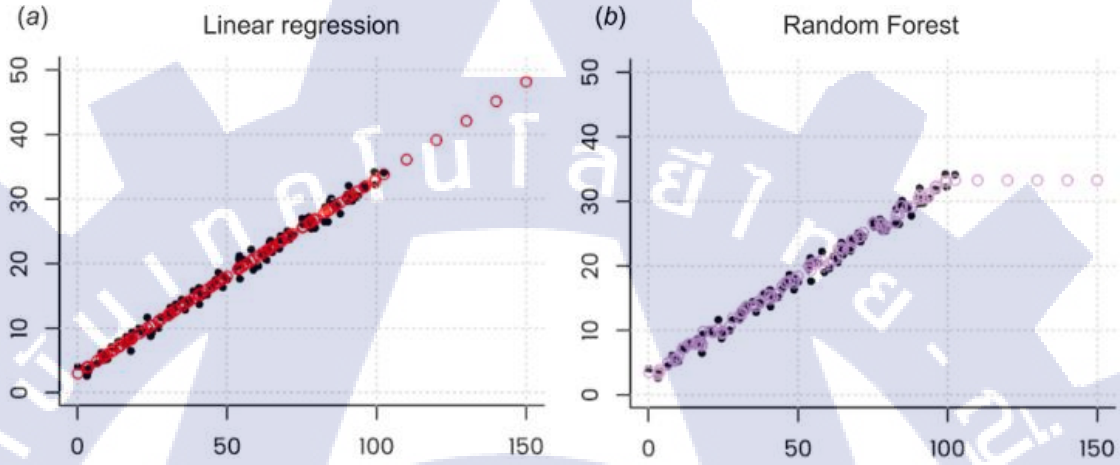
3. อัลกอริทึม Random Forest Regressor จะทำการส่งข้อมูลที่ได้ทำการสุ่มค่าให้กับอัลกอริทึม Decision Tree ที่อยู่ภายในโครงสร้างของอัลกอริทึม Random Forest Regressor

4. อัลกอริทึม Decision Tree จะนำข้อมูลที่ได้รับไปใช้ในการฝึกสอนและทดสอบเพื่อหาค่า Dependent Variable ที่เหมาะสม

5. จากนั้นกลุ่มอัลกอริทึม Decision Tree แต่ละต้นจะทำการแสดงผลลัพธ์ค่า Dependent Variable ออกมา

6. จากนั้นอัลกอริทึม Random Forest Regressor จะทำการหาค่าเฉลี่ยจากผลรวมของค่า Dependent Variable ต่อจำนวนต้นไม้ของอัลกอริทึม Decision Tree และแสดงผลลัพธ์ของค่าเฉลี่ยออกมาเป็นค่า Dependent Variable

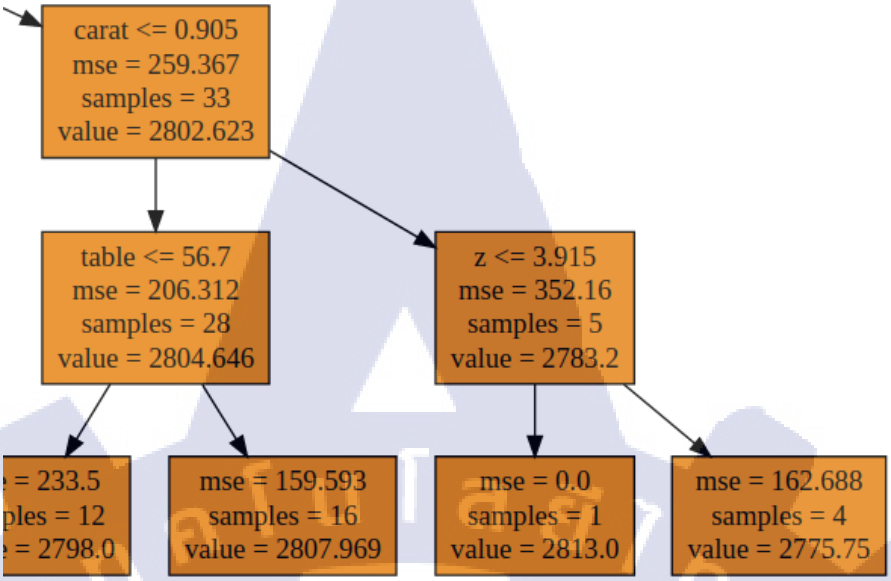
2.3.3.3 ข้อบกพร่องของอัลกอริทึม Random Forest Regressor



รูปที่ 2.25 การเปรียบเทียบผลการทำนายระหว่าง Linear Regression กับ Random Forest [6]

จาก รูปที่ 2.25 จะเห็นได้ว่าผลของการทำนาย Regression ของอัลกอริทึม Random Forest Regressor นั้นค่า Dependent Variable ที่อัลกอริทึม Random Forest นั้นได้ทำนายออกมาโดยรับค่า Independent Variable กลับเป็นค่าคงที่เสมอ โดยทางผู้วิจัยจะอธิบายเหตุผลนี้จากภาพประกอบ ภาพที่ 2.26

รูปที่ 2.26 ภาพอัลกอริทึม Decision Tree ส่วนหนึ่งจากอัลกอริทึม Random Forest [6]

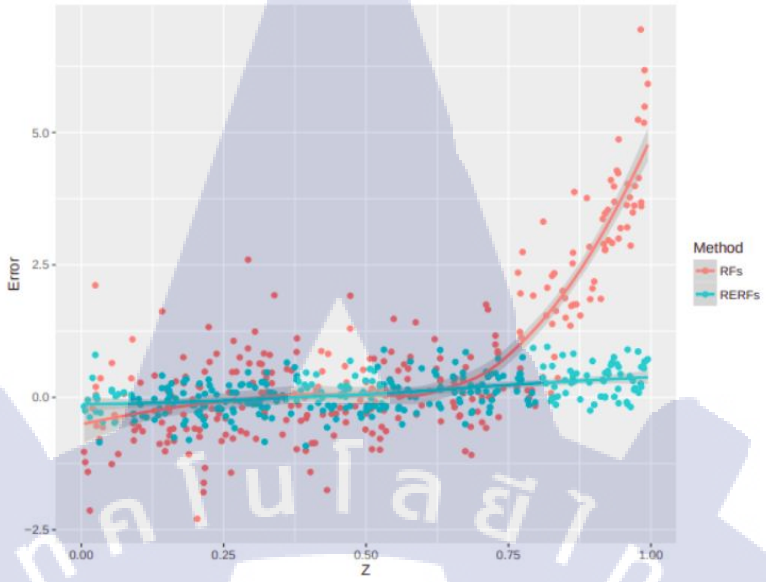


รูปที่ 2.27 รูปขยายบางส่วนจากภาพที่ 2.26 [6]

จากการขยาย รูปที่ 2.26 เป็น รูปที่ 2.27 นั้นจะมีการยกตัวอย่างขึ้นมาอย่างน้อย 4 ตัวอย่างโดยที่ค่า Depth มีค่าไม่มากกว่า 62.75 ค่า x ไม่มากกว่า 5.545 ค่า carat ไม่มากกว่า 0.905 และค่า z ไม่มากไปกว่า 3.915 โดยที่ค่าที่ทำนายออกมานั้นจะมีค่าเท่ากับ 2775.75 โดยจากภาพนั้นจะนำเสนอถึงค่าเฉลี่ยจากตัวอย่างเพียงแค่ 4 ตัวอย่างเพียงเท่านั้น ดังนั้นแล้วไม่ว่าค่าใดที่ถูกนำไปเข้ากระบวนการทำนายของอัลกอริทึม Random Forest Regressor จะทำให้อัลกอริทึมทำนายออกมาได้เพียง 2775.75 เท่านั้น

ดังนั้นแล้วจึงกล่าวได้ว่าเมื่อ Random Forest Regressor ได้รับการมอบมายจากผู้ใช้งานให้ทำนายค่าที่ไม่เคยเห็นมาก่อน อัลกอริทึม Random Forest Regressor นั้นจะนำค่าเฉลี่ยที่มีอยู่ก่อนหน้านั้นออกมาใช้ในการทำนายเสมอ จึงส่งผลทำให้ข้อมูลของกลุ่มตัวอย่างที่นำมาทำนายนั้นไม่สามารถอยู่นอกค่าสูงสุดหรือต่ำสุดจากกลุ่มตัวอย่างที่นำมาทำนายได้

จึงกล่าวสรุปได้ว่าอัลกอริทึม Random Forest Regressor นั้นจะไม่มีความสามารถในการทำนายข้อมูลที่นอกเหนือจากชุดข้อมูลที่ใช้ในการฝึกสอนนั้นได้ ดังนั้นแล้วเมื่อต้องเจอสถานการณ์ที่มีชุดข้อมูลมาจากแหล่งที่ไม่รู้จัก อัลกอริทึม Random Forest Regressor นั้นจะทำการนำค่าสูงสุดในชุดข้อมูลที่ใช้ในการฝึกสอนออกมาเป็นผลลัพธ์การทำนายของชุดข้อมูลที่อัลกอริทึม Random Forest Regressor ไม่รู้จักแทน



รูปที่ 2.28 การแสดงค่าความคลาดเคลื่อนการทำนายของอัลกอริทึม Random Forest ที่ถูกปรับปรุง [6]

จากที่อธิบายมาข้างต้นเราจะสังเกตเห็นได้ว่าอัลกอริทึม Random Forest Regressor นั้นมีปัญหาในการทำนายผลลัพธ์จากชุดข้อมูลที่ไม่ได้ผ่านการฝึกสอนมาก่อนดังนั้นแล้วผู้วิจัย จึงได้นำเสนอวิธีการแก้ไขปัญหา 4 วิธี คือ 1. เปลี่ยนไปใช้อัลกอริทึมแบบเส้นตรงแทน อาทิ SVM Regression, Linear Regression เป็นต้น, 2.ใช้วิธีการสร้างโครงข่ายประสาทเทียมขึ้นมาเพราะโครงข่ายประสาทเทียมนั้นมีความสามารถในการคาดการณ์ค่าที่ได้รับเข้ามาได้, 3. ใช้วิธีการผสมอัลกอริทึมอื่นเข้ากับ Random Forest Regressor เช่น อัลกอริทึมสมการเส้นตรง เป็นต้น, 4. ใช้อัลกอริทึม Random Forest Regressor ที่ได้ถูกปรับปรุงจากผู้พัฒนาแล้ว

2.3.4 ข้อดีและข้อเสียของอัลกอริทึม Random Forest

ในหัวข้อที่ 2.3.4 นั้นทางผู้วิจัยจะทำการเปรียบเทียบข้อดีและข้อเสียของอัลกอริทึม Random Forest แบบภาพรวมทั้งหมดว่าอัลกอริทึม Random Forest นั้นมีข้อดีอะไรบ้าง และมีข้อเสียอะไรบ้าง เพื่อที่จะทำให้ผู้ที่อ่านวิทยานิพนธ์เล่มนี้นั้นสามารถเห็นข้อดีและข้อเสียของอัลกอริทึม Random Forest ได้อย่างชัดเจนมากยิ่งขึ้น

ข้อดีของอัลกอริทึม Random Forest

1.อัลกอริทึม Random Forest นั้นสามารถนำมาใช้ในงาน Classification และงาน Regression ได้

2.อัลกอริทึม Random Forest นั้นสามารถทำงานร่วมกับข้อมูลในเชิงหมวดหมู่และข้อมูลในเชิงตัวเลขได้อย่างมีประสิทธิภาพ และไม่จำเป็นต้องแปลงขนาดและตัวแปรของข้อมูลใดๆ

3.อัลกอริทึม Random Forest นั้นมีความสามารถในการเลือกคุณลักษณะและสามารถสร้างอัลกอริทึม Decision Tree ที่มีองค์ประกอบที่ไม่ซ้ำกันได้ โดยใช้วิธีการสุ่มข้อมูลจากชุดข้อมูลเพื่อนำมาสร้างอัลกอริทึม Decision Tree

4.อัลกอริทึม Random Forest นั้นไม่มีทางที่จะได้รับความผิดปกติจากตัวแปรที่มีความผิดปกติใด ๆ เพราะอัลกอริทึม Random Forest นั้นใช้วิธีการควบรวมตัวแปรทั้งหมด

5.อัลกอริทึม Random Forest นั้นมีความสามารถในการจัดการข้อมูลแบบที่เป็นเชิงเส้นและไม่เป็นเชิงเส้น

6.อัลกอริทึม Random Forest นั้นมักจะให้ค่าความแม่นยำที่สูงและสามารถปรับสมดุลของความแปรปรวนที่มีอคติได้เป็นอย่างดี เนื่องจากหลักการของอัลกอริทึม Random Forest นั้นจะทำงานโดยทำจากหาค่าเฉลี่ยของผลลัพธ์การทำนายจากอัลกอริทึม Decision Tree ในหลาย ๆ รูปแบบ ดังนั้นแล้วอัลกอริทึม Random Forest นั้นจึงทำการหาค่าเฉลี่ยของความแปรปรวนร่วมด้วย

ข้อเสียของอัลกอริทึม Random Forest

1.อัลกอริทึม Random Forest นั้นเป็นอัลกอริทึมตีความได้ยากมาก เพราะว่าอัลกอริทึม Random Forest เป็นอัลกอริทึมที่เน้นไปในงานเชิงคุณลักษณะ แต่ไม่ได้เน้นในเชิงของการหาผลสมบูรณ์ของงาน Regression

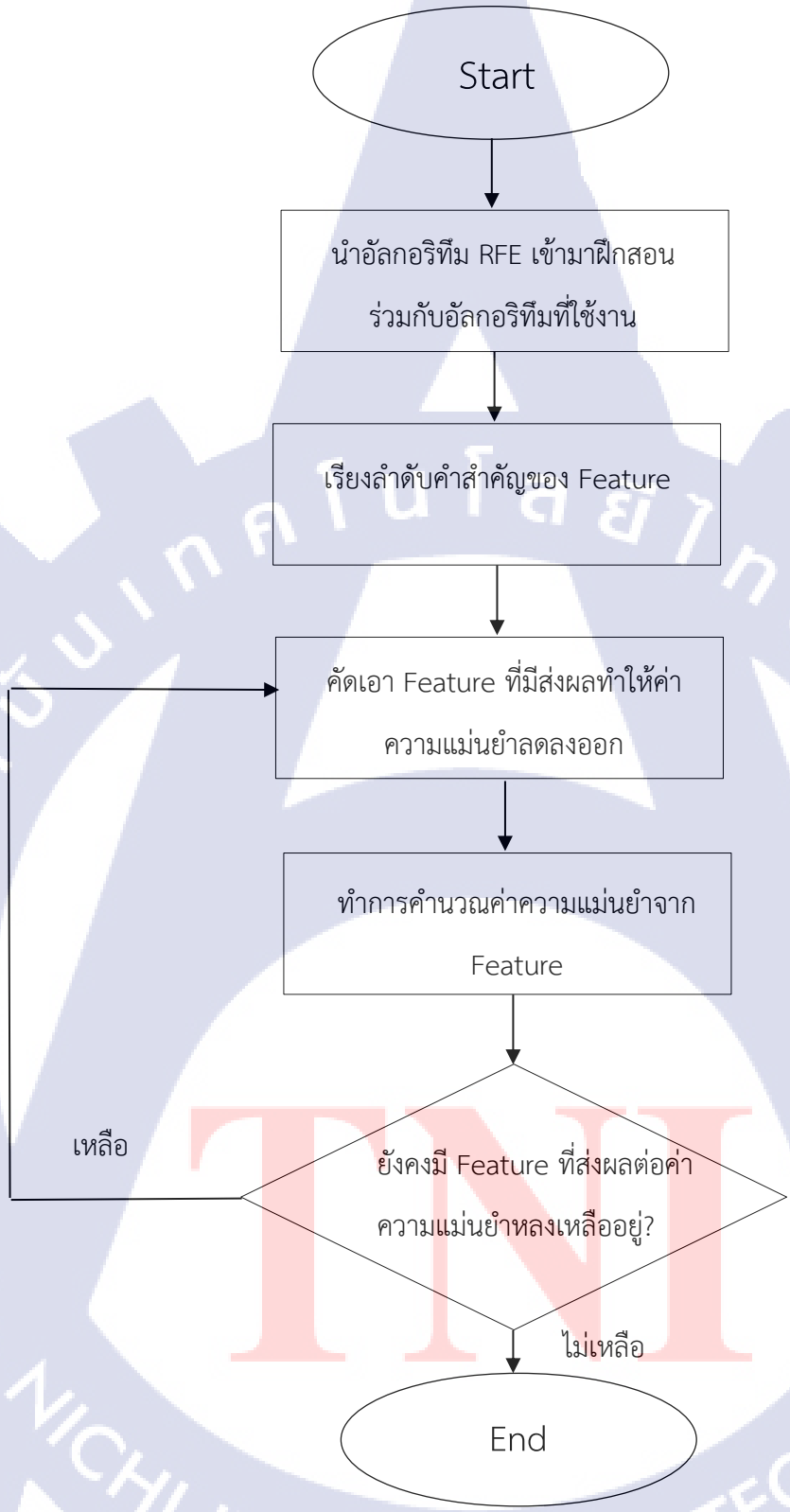
2.อัลกอริทึม Random Forest นั้นจำเป็นต้องใช้ชุดข้อมูลขนาดใหญ่เท่านั้นในการฝึกสอนเพราะชุดข้อมูลขนาดใหญ่จะทำให้การทำงานอัลกอริทึม Random Forest นั้นทำงานได้อย่างมีประสิทธิภาพ

3.อัลกอริทึม Random Forest นั้นเป็นอัลกอริทึมประเภท Black Box อัลกอริทึมจึงทำให้ผู้ใช้งานนั้นไม่สามารถควบคุมในการคำนวณหรือการทำนายผลของอัลกอริทึม Random Forest ได้ หรือบางทีอาจจะควบคุมการทำงานของอัลกอริทึม Random Forest ได้ แต่ว่าอาจจะไม่สามารถควบคุมการทำงานองอัลกอริทึม Random Forest ได้ทั้งหมด

4.อัลกอริทึม Random Forest นั้นใช้เวลาในการฝึกสอนที่นานกว่าเมื่อเทียบกับอัลกอริทึมในเชิงเดี่ยว อาทิ Linear Regression หรือ Decision Tree

2.5 คำอธิบายเกี่ยวกับอัลกอริทึม RFE

อัลกอริทึม Recursive Feature Elimination (RFE Algorithm) คือ อัลกอริทึมที่ใช้ในส่วนของงาน Feature Selection โดยที่มันจะทำงานโดยการหา Feature ที่เหมาะสมกับการเรียนรู้ของ Machine Learning และคัด Feature ที่ส่งผลต่อความแม่นยำที่ลดลงของ Machine Learning ออกไปในระหว่างช่วงการฝึกฝน Model ส่งผลให้ Machine Learning Model นั้นมีความแม่นยำที่มากขึ้น



รูปที่ 2.29 Flowchart การทำงานของอัลกอริทึม RFE

2.6 คำอธิบายเกี่ยวกับชุดข้อมูล (Dataset)

2.6.1 CIC-IDS 2017

เป็นชุดข้อมูลที่มีคำอธิบายโดยละเอียดเกี่ยวกับการคุกคามและการกระจายรูปแบบที่เป็นนามธรรมสำหรับ Application Protocols หรือ Lower Level Network Entities โดยที่ Profile เหล่านี้สามารถใช้ตัวแทนที่เป็นมนุษย์ในการสร้างกิจกรรมบนเครือข่ายที่หลากหลายพร้อมทั้ง topologies ที่แตกต่างกันออกไป และยังสามารถใช้ Profile ร่วมกันเพื่อสร้างชุดข้อมูลสำหรับความต้องการเฉพาะและมี class ที่แตกต่างกันออกไป โดย Feature ของชุดข้อมูลนี้จะมีด้วยกันทั้งหมด 84 Feature ได้แก่ 1.Flow ID, 2.Source IP, 3.Source Port, 4.Destination IP, 5.Destination Port, 6.Protocol, 7.Time stamp, 8.Flow Duration, 9.Total Fwd Packets, 10.Total Backward Packets, 11.Total Length of Fwd Pck, 12.Total Length of Bwd Pck, 13.Fwd Packet Length Max, 14. Fwd Packet Length Min, 15.Fwd Pck Length Mean, 16.Fwd Packet Length Std, 17.Bwd Packet Length Max, 18.Bwd Packet Length Min, 19.Bwd Packet Length Mean, 20.Bwd Packet Length Std, 21.Flow Bytes / s, 22.Flow Packets / s, 23.Flow IAT Mean, 24.Flow IAT Std, 25.Flow IAT Max, 26.Flow IAT Min, 27.Fwd IAT Total, 28.Fwd IAT Mean, 29.Fwd IAT Std , 30.Fwd IAT Max, 31.Fwd IAT Min, 32.Bwd IAT Total, 33.Bwd IAT Mean, 34.Bwd IAT Std, 35.Bwd IAT Max, 36.Bwd IAT Min, 37.Fwd PSH Flags, 38.Bwd PSH Flags, 39.Fwd URG Flags, 40.Bwd URG Flags, 41.Fwd Header Length, 42.Bwd Header Length, 43.Fwd Packets / s, 44.Bwd Packets / s , 45.Min Packet Length, 46.Max Packet Length, 47.Packet Length Mean, 48.Packet Length Std, 49. Packet Len . Variance, 50.FIN Flag Count, 51.SYN Flag Count, 52.RST Flag Count, 53.PSH Flag Count, 54.ACK Flag Count, 55.URG Flag Count, 56.CWE Flag Count, 57.ECE Flag Count, 58.Down / Up Ratio, 59.Average Packet Size, 60.Avg Fwd Segment Size, 61.Avg Bwd Segment Size, 62.Fwd Avg Bytes / Bulk, 63.Fwd Avg Packets / Bulk, 64.Fwd Avg Bulk Rate, 65.Bwd Avg Bytes / Bulk, 66.Bwd Avg Packets / Bulk, 67.Bwd Avg Bulk Rate, 68.Subflow Fwd Packets, 69.Subflow Fwd Bytes, 70.Subflow Bwd Packets, 71.Subflow Bwd Bytes, 72.Init_Win_bytes_fwd, 73.Act_data_pkt_fwd, 74.Min_seg_size_fwd, 75.Active Mean, 76.Active Std, 77.Active Max, 78. Active Min, 79.Idle Mean, 80.Idle Packet, 81.Idle Std, 82.Idle Max, 83.Idle Min, 84.Label โดยที่ Feature โดยชุดข้อมูลนี้ได้มี Label อยู่ในชุดข้อมูลจึงไม่จำเป็นต้องกำหนด Label ใหม่เพื่อใช้ในการฝึกสอนอัลกอริทึมของการเรียนรู้ของเครื่อง ชุดข้อมูลชนิดนี้ได้ทำการเก็บข้อมูลการโจมตีแบบเป็นรายวัน 2 ช่วงเวลาคือ ตอนเช้า และ ตอนบ่าย โดยจะมี

อุปกรณ์ที่ทำหน้าที่เป็น Attacker และ อุปกรณ์หรือเครือข่ายที่ถูกโจมตีนั้นจะเรียกว่า Victim โดยในแต่ละวันและเวลานั้นอาจจะมีรูปแบบการโจมตี DDoS Attack ที่แตกต่างกันออกไป

Attacker	Victim	Attack Name	Date	Attack Start Time	Attack Finish Time
172.31.70.4 (Valid IP:18.221.219.4)	172.31.69.25 (Valid IP:18.217.21.148)	FTP-BruteForce	Wed-14-02-2018	10:32	12:09
172.31.70.6 (Valid IP:13.58.98.64)	18.217.21.148-172.31.69.25	SSH-Bruteforce	Wed-14-02-2018	14:01	15:31
172.31.70.46 (Valid IP:18.219.211.138)	18.217.21.148-172.31.69.25	DoS-GoldenEye	Thurs-15-02-2018	9:26	10:09
172.31.70.8 (Vazlid IP:18.217.165.70)	18.217.21.148-172.31.69.25	DoS-Slowloris	Thurs-15-02-2018	10:59	11:40
172.31.70.23 (Valid IP: 13.59.126.31)	18.217.21.148-172.31.69.25	DoS-SlowHTTPTest	Fri-16-02-2018	10:12	11:08
172.31.70.16 (Valid IP:18.219.193.20)	18.217.21.148-172.31.69.25	DoS-Hulk	Fri-16-02-2018	13:45	14:19

รูปที่ 2.30 ตัวอย่างของการโจมตีเครือข่ายประจำวัน [7]

2.7 งานวิจัยที่เกี่ยวข้อง

ในงานวิจัยชิ้นนี้ ผู้วิจัยจะนำปัญญาประดิษฐ์มาพัฒนาเพื่อตรวจจับการโจมตีแบบ DDoS Attack ดังนั้นแล้วในหัวข้อนี้จะเป็นการกล่าวถึงงานวิจัยที่เกี่ยวข้องทั้งหมดว่าในแต่ละงานนั้นมีข้อดีข้อเสียอย่างไรบ้าง ซึ่งในปัจจุบัน โดยในปัจจุบันนั้นการโจมตีแบบ DDoS Attack สามารถพบเห็นได้ทั่วไปในอินเทอร์เน็ต ดังนั้นทางผู้ให้บริการเครือข่ายจำนวนมากจึงคิดหาวิธีการป้องกันการโจมตีแบบ DDoS Attack ด้วยการใช้ปัญญาประดิษฐ์มาช่วยตรวจจับ Packet ที่เข้ามาสู่เครือข่ายของตนว่า Packet ที่เข้ามานั้นเป็นการทำ DDoS Attack หรือไม่ โดยปัญญาประดิษฐ์ที่ใช้นั้นจะมีอัลกอริทึมอยู่ภายในเปรียบเสมือนกับสมองของปัญญาประดิษฐ์ ซึ่งอัลกอริทึมที่มีความซับซ้อนและสามารถคัดกรองหรือแยกแยะข้อมูลต่าง ๆ ได้ในปัจจุบันจะเป็นอัลกอริทึมแบบโครงข่ายประสาทเทียมซึ่งเมื่อ

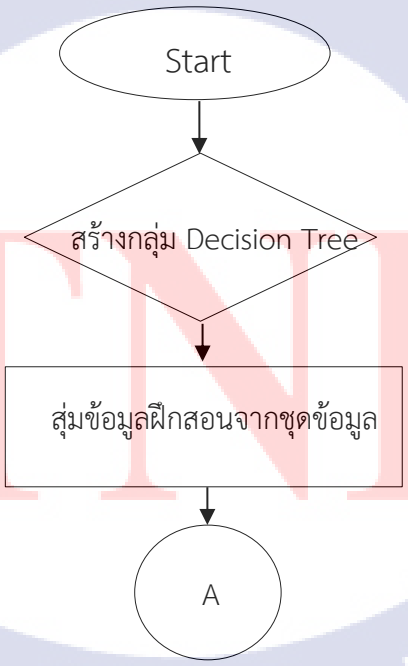
นำมาใช้ในการตรวจจับการโจมตีแบบ DDoS Attack แล้วนั้นจึงส่างผลให้มีความแม่นยำในการแยกแยะ Packet ที่ส่งมาจากภายนอกได้ว่าเป็น Packet แบบไหน แต่ว่าอัลกอริทึมแบบโครงข่ายประสาทเทียมก็มีข้อเสียเพราะเนื่องจากอัลกอริทึมชนิดนี้มีความซับซ้อนสูงจึงทำให้ส่งผลในด้านของเวลาที่ใช้ในการตรวจจับ Packet และใช้ทรัพยากรของฮาร์ดแวร์ที่ปัญญประดิษฐ์นั้นได้ถูกติดตั้งอยู่มาก [1]-[3] และนั่นจึงส่งผลให้ในยุคปัจจุบันที่มีฮาร์ดแวร์ประสิทธิภาพที่ไม่สูงพอที่จะรองรับการตรวจจับการโจมตีแบบ DDoS Attack ได้แล้ว ดังนั้นจึงควรไปเน้นใช้ปัญญาประดิษฐ์ที่ใช้อัลกอริทึมแบบปกติที่มีความซับซ้อนต่ำแต่มีความแม่นยำในการแยกแยะข้อมูลที่สูงเช่น K-Mean, SVM, Random Forest เป็นต้น และยิ่งนำอัลกอริทึมธรรมดามาใช้กับฮาร์ดแวร์ที่มีประสิทธิภาพต่ำก็จะสามารถทำให้เจ้าของระบบเครือข่ายไม่จำเป็นต้องไปเสียงบประมาณกับการปรับปรุงฮาร์ดแวร์ให้มีประสิทธิภาพมากกว่าเดิม แต่ว่าข้อเสียของอัลกอริทึมธรรมดานั้นก็มีเช่นกันคือ อัลกอริทึมแบบธรรมดานั้นมีความสามารถในการแยกแยะหรือคัดกรองข้อมูลที่ต่ำกว่าอัลกอริทึมแบบโครงข่ายประสาทเทียม ดังนั้นแล้วจึงจำเป็นต้องใช้ชุดข้อมูลที่มีคุณภาพและทันสมัยในการฝึกสอนปัญญาประดิษฐ์ที่มีอัลกอริทึมแบบธรรมดา เพื่อให้อัลกอริทึมแบบธรรมดามีประสิทธิภาพในการทำงานที่มากขึ้น [4] – [9] แต่เนื่องจากว่าอัลกอริทึมแบบธรรมดานั้นมีความซับซ้อนที่ต่ำและเมื่อเจอกับชุดข้อมูลที่มี Feature ที่มากจนเกินไปก็อาจจะทำให้เกิดปัญหา overfitting ได้ [10] ดังนั้นจึงจำเป็นต้องมีการจัดการชุดข้อมูลที่ดีพอที่จะให้ปัญญาประดิษฐ์ได้เรียนรู้และไม่เกิดปัญหา overfitting ได้ แต่ในบางทีนั้นก็ได้มีการปรับปรุงอัลกอริทึมที่อยู่ภายในปัญญาประดิษฐ์ให้มีความสามารถในการเรียนรู้และมีประสิทธิภาพในการแยกแยะข้อมูลที่มากขึ้น [11]-[13] แต่ในความเห็นของผู้วิจัยนั้นได้มีความเห็นว่าในบางกรณีนั้นการปรับปรุงอัลกอริทึมเป็นเรื่องที่มีความซับซ้อนและต้องใช้ความรู้เฉพาะทางที่สูง ดังนั้นผู้วิจัยจึงเสนอปัญญาประดิษฐ์ที่ทำงานโดยอัลกอริทึมแบบธรรมดาซึ่งจะมีอัลกอริทึมที่ช่วยเพิ่มประสิทธิภาพในการตรวจจับการโจมตีแบบ DDoS Attack อยู่ภายใน ส่วนอัลกอริทึมแบบธรรมดานั้นจะใช้เป็นอัลกอริทึมชนิดการเรียนรู้ของเครื่องแบบผสมเพื่อช่วยในการยกระดับความแม่นยำในการตรวจจับให้ใกล้เคียงหรือมีประสิทธิภาพที่ดีกว่าอัลกอริทึมแบบโครงข่ายประสาทเทียมและต้องมีประสิทธิภาพในด้านของเวลาที่ดีกว่า พร้อมทั้งไม่ใช้ทรัพยากรของฮาร์ดแวร์ที่ติดตั้งเกินกว่าที่อัลกอริทึมแบบโครงข่ายประสาทเทียมใช้

บทที่ 3
ระเบียบวิธีการวิจัย

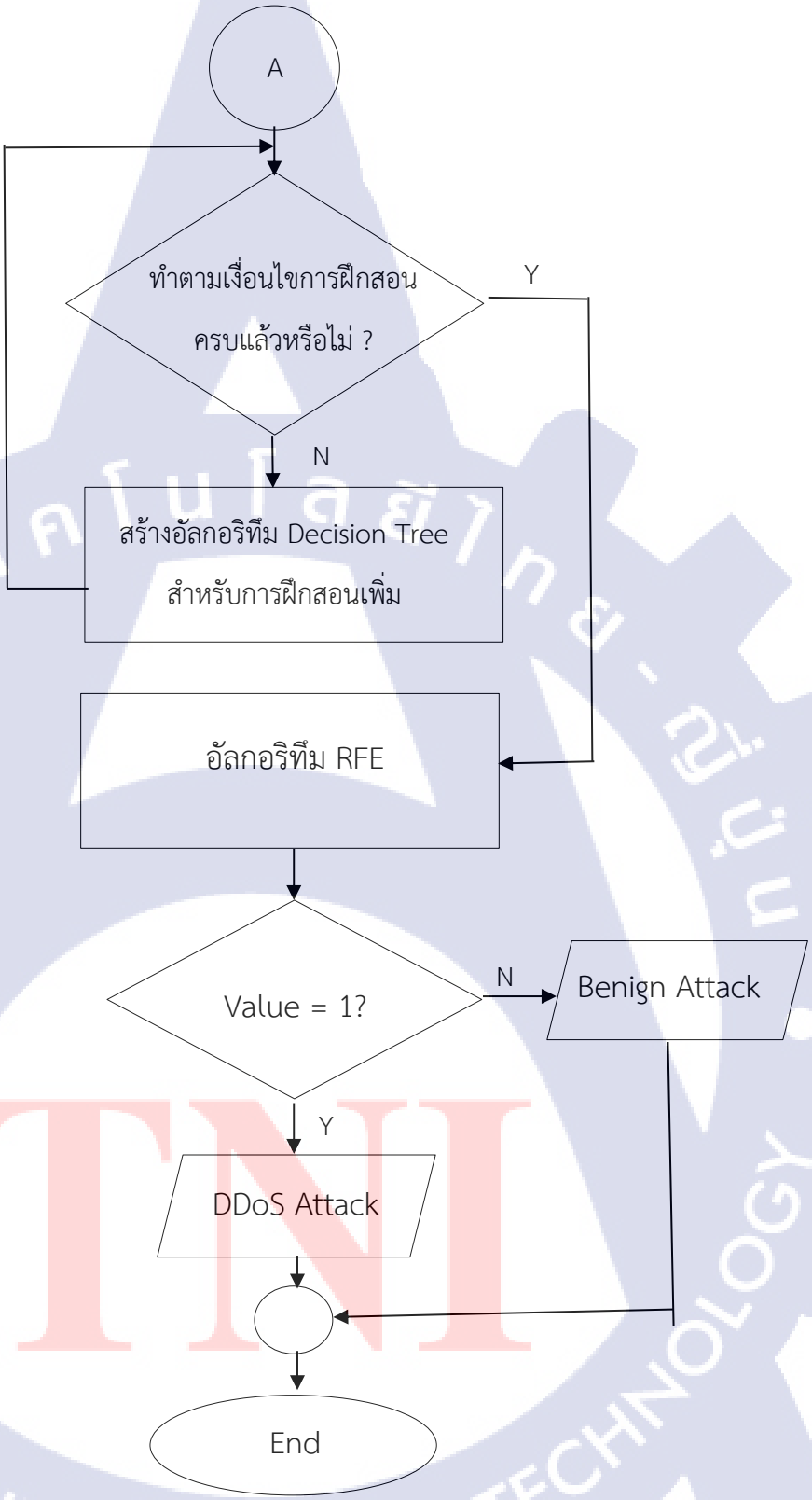
ในบทนี้ผู้วิจัยจะอธิบายถึงขั้นตอนของการดำเนินการวิจัยในขั้นตอนการพัฒนาลักษณะแบบจำลองของ Machine Learning ว่ามีขั้นตอนการพัฒนาอย่างไรเก็บข้อมูลแบบไหนและใช้อะไรเป็นตัววัดผลประสิทธิภาพของงานวิจัยที่ได้ทำ โดยผู้จัดทำจะแบ่งออกเป็นหัวข้อใหญ่ ๆ ดังนี้ คือ กรอบแนวคิดการวิจัย, ขั้นตอนการทดลอง, การดำเนินการวิจัย และ การทำการทดลอง

3.1 กรอบแนวคิดการวิจัย

กรอบแนวคิดการวิจัยผู้จัดทำได้ดำเนินการวิธีการจัดเตรียมข้อมูลและนำผลการทดลองมาเปรียบเทียบบทความวิจัยที่ [1] โดยใช้หลักการ Ensemble Random Forest ผสมกับอัลกอริทึม RFE เพื่อเพิ่มประสิทธิภาพในการตรวจจับการทำ DDoS Attack อย่างมีประสิทธิภาพ โดยการออกแบบและทดสอบโมเดลจะใช้ JupyterLab รุ่น 3.0.14, ภาษา Python รุ่น 3.6.14, Module NumPy รุ่น 1.21.5, Module Pandas รุ่น 1.3.5 และ Module Scikit-learn รุ่น 1.0.2 ในการทำการทดลอง



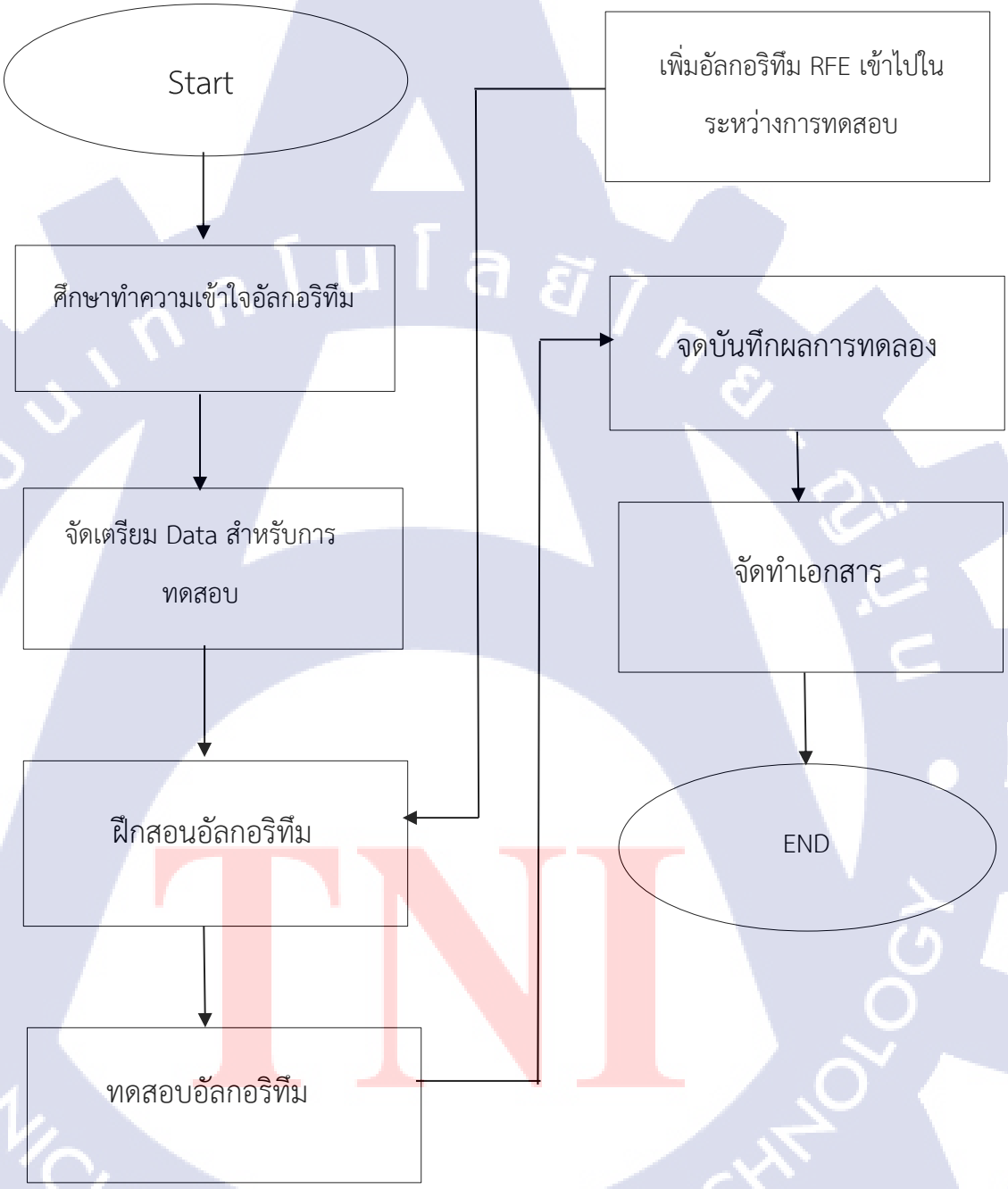
รูปที่ 3.1 Flowchart ของอัลกอริทึม Ensemble Random Forest ผสมกับอัลกอริทึม RFE (1)



รูปที่ 3.2 Flowchart ของอัลกอริทึม Ensemble Random Forest ผสมกับอัลกอริทึม RFE (2)

3.2 ขั้นตอนการทดลอง

ในการทดลองนั้นผู้จัดทำจะเริ่มการทดลองโดยการศึกษาอัลกอริทึมแต่ละชนิดว่ามีคุณสมบัติอย่างไร สามารถนำมาผสมกันได้หรือไม่ ตาม Flowchart ดังนี้



รูปที่ 3.3 Flowchart การทดลอง

3.3. การดำเนินการวิจัย

ในการดำเนินงานวิจัยนี้ทางผู้จัดทำได้ทำการเตรียมชุดข้อมูลของ CICIDS-2017 แบบสุ่ม จำนวนข้อมูล 140,000 ข้อมูล ประกอบด้วยข้อมูล Benign Attack ร้อยละ 60 และ DDoS Attack ร้อยละ 40 มาเพื่อใช้ในการฝึกสอนและทดสอบ ซึ่งแบ่งอัตราส่วนการฝึกสอนและการทดสอบอยู่ที่ 80 : 20 คือ ข้อมูลจากชุดข้อมูล 80% จะถูกใช้เพื่อการฝึกสอนและอีก 20 % ที่เหลือจะถูกใช้ในการทดสอบ โดยที่การเก็บข้อมูลนั้นถ้าเป็น Model เดี่ยว ๆ จะดูที่ค่า accuracy เพียงอย่างเดียว ส่วน Model แบบผสมนั้นจะพิจารณาจากค่า Precision, Recall, F1-score, เวลาที่ใช้ในการฝึกสอนโมเดล, เวลาที่ใช้ทดสอบโมเดล และรวมถึง การใช้ทรัพยากรของ CPU ร่วมอีกด้วย

3.3.1 Accuracy

Accuracy เป็นค่าความแม่นยำจากการทดสอบซึ่งเป็นที่นิยมใช้กันมากที่สุดในปัจจุบันในการทดสอบ Model ของ Machine Learning ต่าง ๆ บนโลกนี้ ต่างก็จะนิยมใช้ค่านี้เป็นค่าหลักเพื่อบ่งชี้ถึงประสิทธิภาพของ Model ที่ได้พัฒนาขึ้น

$$Accuracy(A) = \frac{Accurately\ classified\ records}{Total\ Sample} \times 100 \quad (1)$$

3.3.2 Precision

Precision เป็นค่าความแม่นยำที่ใช้ในกรณีที่เกิดเหตุการณ์ False Positive มีปริมาณมาก โดยเฉพาะกับเหตุการณ์ที่มี traffic บน server สูง แต่เป็น traffic แบบปกติธรรมดาทั่วไปไม่ใช่การกระทำ DDoS ATTACK ตัว Precision ก็จะสามารถนำมาเป็นตัวชี้วัดว่า model นี้มีความสามารถในการตรวจจับค่า False Positive มากแค่ไหน

$$Precision(\mathrm{Pr}) = \frac{True\ Positive}{True\ Positive + False\ Positive} \times 100 \quad (2)$$

3.3.3 Recall

Recall เป็นค่าความแม่นยำที่ใช้ในกรณีที่เกิด False Negative มีปริมาณจำนวนมาก ๆ โดย Recall จะวัดว่า Model ที่พัฒนาสามารถตรวจจับ True Positive ที่เป็น False Negative ได้มากเพียงใด

$$Recall(Rc) = \frac{True\ Positive}{True\ Positive + False\ Negative} \times 100 \quad (3)$$

3.3.4 F1-score

F1-score เป็นค่าที่นำ Recall และ Precision มาทำงานร่วมกันโดยที่จะเน้นการวัดค่าความแม่นยำที่เป็น True-Negative เป็นหลัก

$$F1\ Score = \frac{2 \cdot (Recall \times Precision)}{Recall + Precision} \times 100 \quad (4)$$

3.4 การทำการทดลอง

ในหัวข้อ 3.4 ผู้จัดทำจะอธิบายรายละเอียดเพิ่มเติมถึงการวัดค่าและเปรียบเทียบค่าต่าง ๆ เพื่อที่ว่าจะให้สามารถระบุได้ว่าค่านั้น ๆ ที่ออกมาจะสามารถดำเนินการต่อไปได้หรือไม่

1.การเตรียมข้อมูลของการทดลอง

ในขั้นตอนนี้ผู้จัดทำจะต้องศึกษาอัลกอริทึมของ Machine Learning ในส่วนของ Supervise Learning และ Unsupervised Learning ว่า สามารถนำมาใช้งานในการตรวจจับ DDOS ATTACK ได้หรือไม่ และสามารถนำมาพัฒนาเป็น Ensemble Machine Learning Model ได้หรือไม่

2.การสร้าง Model

ในขั้นตอนนี้ผู้จัดทำจะต้องทำการใช้การ Coding Machine Learning Model ขึ้นมา โดย Model นั้นจะต้องเป็น Ensemble Machine Learning Model และสามารถรองรับการประมวลผลของชุดข้อมูลที่นำมาได้ด้วย

3.การทดสอบ Model

ในขั้นตอนนี้ผู้จัดทำจะทำการทดสอบ Model กับชุดข้อมูลว่ามีความแม่นยำมากน้อยเพียงใดในการตรวจจับการโจมตีแบบ DDoS Attack ซึ่งค่า Accuracy ที่รับได้จะต้องไม่ต่ำกว่า 85%

4.การรวม Model เข้ากับอัลกอริทึม RFE

ในขั้นตอนนี้ผู้จัดทำจะทำการนำ Model ที่ได้พัฒนาขึ้นมารวมเข้ากับอัลกอริทึม RFE เพื่อทำให้ Model นั้นมีประสิทธิภาพในการตรวจจับ DDoS Attack สูงกว่า Model เดิมที่ได้พัฒนาขึ้น

5.การทดสอบ Model ที่รวมกับอัลกอริทึม RFE

ในขั้นตอนนี้จะเป็นการทดสอบ Model ที่รวมเข้ากับอัลกอริทึม RFE โดยที่ Model ที่พัฒนาขึ้นจะต้องมีประสิทธิภาพใกล้เคียงกับ Model ในงานวิจัยที่ [1] ในด้าน Accuracy และจะต้องมีประสิทธิภาพเหนือกว่าในด้านการลดการใช้ทรัพยากรของ CPU

[9] M. I. W. Pramana et al., "DDoS detection using modified K-means clustering with chain initialization over landmark window," *Conference on Control, Electronics, Renewable Energy and Communications (ICCEREC)*, vol. 6, no. 3, pp. 7-11, May 2015.

[10] R. Latif et al., "Distributed denial of service (DDoS) attack in cloud-assisted wireless body area networks: a systematic literature review," *Journal of Medical Systems*, vol. 38, no. 6, pp. 1-10, June 2014.

[11] J. Ye et al., "A DDoS attack detection method based on SVM in software defined network," *Security and Communication Networks*, vol. 21, no. 4, pp. 1-2, August 2018.

[12] A. Akhunzada et al., "Secure and dependable software defined networks," *Journal of Network and Computer Applications*, vol. 61, no. 11, pp. 199-221, July 2016.

[13] I. Sharafaldin et al., "Toward generating a new intrusion detection dataset and intrusion traffic characterization," *The International Conference on Information Systems Security and Privacy (ICISSp)*, vol. 18, no. 10, pp. 108-116, May 2018.