ระบบสาธิตการบำรุงรักษาเชิงการคาดการณ์

PREDICTIVE MAINTENANCE DEMONSTRATION SYSTEM

กรรณชัย ปัญญาเลิศทิพย์

เกียรติวงศ์ จดแตง

ณัฐพงษ์ ลุนหงส์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2560

ระบบสาธิตการบำรุงรักษาเชิงการคาดการณ์

PREDICTIVE MAINTENANCE DEMONSTRATION SYSTEM

กรรณชัย ปัญญาเลิศทิพย์
เกียรติวงศ์ จดแตง
ณัฐพงษ์ ลุนหงส์

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต
สาขาวิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2560

ปริญญานิพนธ์ปี การศึกษา 2560

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง ระบบสาธิตการบำรุงรักษาเชิงการคาดการณ์

PREDICTIVE MAINTENANCE DEMONSTRATION SYSTEM

ผู้จัดทำ

1. นาย กรรณชัย ปัญญาเลิศทิพย์ รหัสนักศึกษา 57010018
2. นาย เกียรติวงศ์ จดแตง รหัสนักศึกษา 57010119
3. นาย ณัฐพงษ์ ลุนหงส์ รหัสนักศึกษา 57010435

อาจารย์ที่ปรึกษา

(อาจารย์สรยุทธ กลมกล่อม)

ระบบสาธิตการบำรุงรักษาเชิงการคาดการณ์

นายกรรณชัย ปัญญาเลิศทิพย์ 57010018
นายเกียรติวงศ์ จดแตง 57010119
นายณัฐพงษ์ ลุนหงส์ 57010435
อาจารย์สรยุทธ กลมกล่อม อาจารย์ที่ปรึกษา
ปีการศึกษา 2560

บทคัดย่อ

เครื่องจักรนั้นมีส่วนสำคัญอย่างมากกับอุตสาหกรรมการผลิต ซึ่งหากเกิดการชำรุดขึ้นกับเครื่องจักรนั้น จะส่งผลกระทบอย่างมากกับการผลิตทั้งค่าเสียหายในการซ่อมบำรุงและ ค่าเสียหายจาก operation machine downtime โครงการนี้จึงพยายามที่จะจัดทำแพลตฟอร์มสำหรับคาดการณ์การชำรุดของเครื่องจักรล่วงหน้า (predictive maintenance platform) เพื่อให้สามารถจัดการกับการซ่อมบำรุงได้ก่อนที่เครื่องจักรจะเกิดการชำรุด โดยจะทำการเก็บข้อมูลจาก sensor ที่ติดอยู่กับเครื่องจักรส่ง และบันทึกไว้ในระบบ โดยอาศัยการทำงานร่วมกันระหว่าง IoT (Internet of Things) และ Big Data จากนั้นเมื่อรับข้อมูลจาก sensor ในปัจจุบันเข้ามาก็จะนำมาใช้ในการคาดการณ์การชำรุดแบบ real-time โดยอาศัย model machine learning ที่สร้างจากข้อมูลของเครื่องจักรที่เคยได้รับมา ซึ่งจะจำลองการทำงานของแพลตฟอร์ม จากชุด gearbox ซึ่งติด sensor สำหรับจำลองสถานการณ์ต่างๆ แทนเครื่องจักร ซึ่งได้ผลลัพธ์การคาดการณ์การชำรุดอยู่ที่ 95 เปอร์เซ็นต์จากการใช้ neural network algorithm โดยในระบบสามารถทราบสถานะของเครื่องจักรได้แบบ real-time จากที่ไหนก็ได้ผ่าน web application หรือ สามารถใช้การแสดงผลแบบ augmented reality ผ่าน mobile application จากหน้างานจริง ซึ่งง่ายและสะดวกต่อผู้ดูแลระบบในการดูข้อมูลของเครื่องจักรในหน้างานจริงและเป็นข้อมูลประกอบการตัดสินใจให้กับผู้ดูแล

PREDICTIVE MAINTENANCE DEMONSTRATION SYSTEM

Mr. Kunnachai	Punyaleadtip	57010018
Mr. kiadtiwong	jodtaeng	57010119
Mr. Nuttapong	Loonhong	57010435
Mr. Sorayut	Glomglome	Advisor

Academic Year 2560

ABSTRACT

Nowadays, machines are very important components in manufacturing processes. If they become failure during ongoing processes, this will create undesirable effects such as the repair cost and the loss of productivity during downtime. This project proposed the predictive maintenance systems to predict the failure of machines, so the replacement can be arranged before the failure happens. Sensory data of machines are collected, sent, and stored in the system using the combination of IoT and big data technologies. Current collection of machine data is used for real time prediction based on machine learning model built from historical stored data. This project uses toy motors instead of real machines for the demonstration purpose. The real-time prediction gives precision of 95% using neural networks algorithm. In addition, real-time machine status can be monitored using web application from anywhere or augmented reality application on mobile computing devices in front of machines.

กิตติกรรมประกาศ

ปริญญานิพนธ์ฉบับนี้สำเร็จลุล่วงได้ด้วยดีด้วยความช่วยเหลือจากหลายฝ่ายทั้งในทางตรงและทางอ้อม ปริญญานิพนธ์ฉบับนี้จะสำเร็จลงไม่ได้หากปราศจากความช่วยเหลือของบุคคลเหล่านี้ ขอขอบคุณ อาจารย์ที่ปรึกษา คือ อาจารย์สรยุทธ กลมกล่อม เป็นผู้ให้คำแนะนำ คำปรึกษา และให้ความช่วยเหลือตลอดการทำโครงงาน ซึ่งทำให้การทำงานต่าง ๆ เป็นไปได้อย่างราบรื่นและสำเร็จลุล่วงไปได้ด้วยดี

ขอขอบคุณอาจารย์และบุคลากรต่าง ๆ ในสาขาวิชาวิศวกรรมคอมพิวเตอร์ที่ได้ให้คำแนะนำและสั่งสอนความรู้ต่าง ๆ มาโดยตลอด รวมถึงห้องแล็ป HCRL (The Hybrid Computing Research Laboratory) ที่ได้เอื้อเฟื้อสถานที่ในการทำวิจัยและพัฒนาโครงงาน ขอขอบคุณรุ่นพี่และเพื่อนหลาย ๆ คนในภาควิชาวิศวกรรมคอมพิวเตอร์ที่ได้ให้คำแนะนำคำปรึกษาและแบ่งปันความรู้ในทุก ๆ ด้าน

ในสุดท้ายนี้ ขอขอบคุณ บิดา มารดา และครอบครัว ที่ได้เลี้ยงดู สั่งสอน และให้การสนับสนุน พร้อมทั้งให้โอกาสในการศึกษาและให้กำลังใจเสมอมา

กรรณชัย ปัญญาเลิศทิพย์
เกียรติวงศ์ จดแตง
ณัฐพงษ์ ลุนหงส์

สารบัญ

หน้า

บทคัดย่อภาษาไทย I

บทคัดย่อภาษาอังกฤษ II

กิตติกรรมประกาศ III

สารบัญ IV

สารบัญตาราง VI

สารบัญรูป VII

บทที่ 1 บทนำ 1

1.1 ความเป็นมาของปัญหา 1

1.2 วัตถุประสงค์โครงการ 4

1.3 ประโยชน์ที่คาดว่าจะได้รับ 4

1.4 ขอบเขตของโครงงาน 4

1.5 ข้อจำกัดของโปรแกรม 5

1.6 แผนการดำเนินงาน 6

บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง 7

2.1 การทำ Predictive Maintenance 7

2.2 เทคโนโลยีที่เกี่ยวข้องกับการรับข้อมูล 7

2.3 การรับข้อมูลจาก controller 8

2.4 การเก็บข้อมูล 9

2.5 การประมวลผลด้วย machine learning 9

2.6 การแสดงผลด้วย Augmented Reality Technology 13

2.7 งานวิจัยที่เกี่ยวข้อง 13

บทที่ 3 การออกแบบและพัฒนา 15

3.1 ภาพรวมการทำงานภายในระบบ 15

3.2 การออกแบบการทำงาน และสถาปัตยกรรมของระบบ 18

3.3 Model ทดสอบ 33

สารบัญ

หน้า

บทที่ 4 การทดลอง 35
4.1 การทดลองเกี่ยวกับการรับส่งข้อมูล 35
4.2 การทดลองเกี่ยวกับ machine learning 38
4.3 การทดลองเกี่ยวกับการแสดงผล 45
บทที่ 5 บทสรุปและข้อเสนอแนะ 48
5.1 บทสรุป 48
5.2 ปัญหาและอุปสรรค 50
5.3 แนวทางแก้ปัญหา 50
5.4 แนวทางการพัฒนาระบบ 50

สารบัญตาราง

ตาราง หน้า

4.1 การเปรียบเทียบความแม่นยำของการสร้าง model machine learning 43

4.2 การเปรียบเทียบการทดสอบ real-time predict ในแต่ละสถานะ 44

สารบัญรูป

รูป หน้า

1.1 ยุคอุตสหกรรม 4.0 1

1.2 โมเดลตัวอย่างระบบ IoT ในโรงงาน 3

1.3 การใช้ AR ในการ monitor เครื่องจักร 3

1.4 AR Module 5

1.5 แผนการดำเนินงาน 6

2.1 โครงสร้าง node ของ neural network 11

2.2 การทำงานของ Artificial Neuron 11

2.3 Proposed SEMS framework. 14

2.4 ANN-based condition monitoring block 14

3.1 System Diagram 15

3.2 ระบบรับข้อมูลชุด gearbox 16

3.3 ระบบรับส่งข้อมูล 16

3.4 ระบบประมวลผลเพื่อคาดการณ์การชำรุดของเครื่องจักร 17

3.5 ระบบแสดงผล 18

3.6 Flowchart การรับข้อมูลจาก model 19

3.7 Topic ของระบบ 20

3.8 ตัวอย่าง data frame ข้อมูลจาก topic 20

3.9 Diagram การนำข้อมูลชุด gearbox มาเก็บใน Apache Hadoop 21

3.10 Diagram การส่งข้อมูลจาก gearbox มายัง spark streaming 21

3.11 Lambda Architecture 22

3.12 Diagram การทำ Predictive Model Generator 22

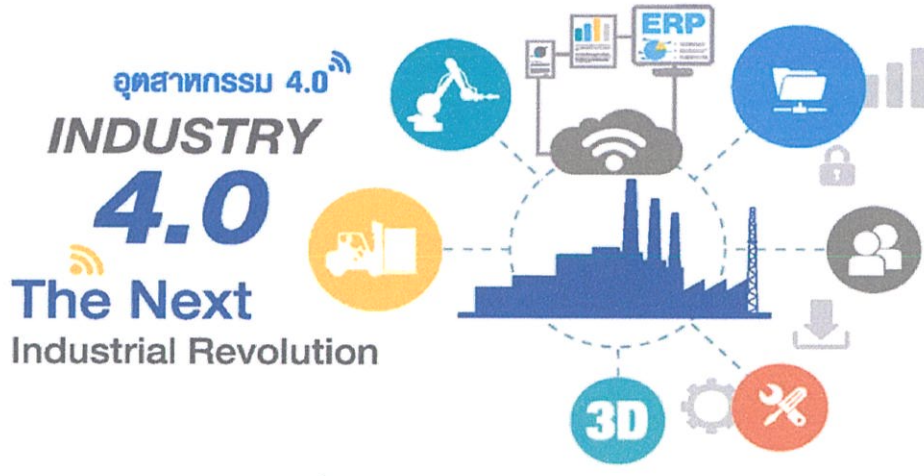
3.13 Diagram การทำ Real-time prediction 23

3.15 โปรแกรมที่เกี่ยวข้องกับการพัฒนาโปรแกรม 24
3.16 การทำงานของ Vuforia 25
3.17 Flowchart การใช้งาน Vuforia ร่วมกับ Unity 26
3.18 Flowchart กระบวนการพัฒนา application 27
3.19 Diagram การรับข้อมูลจาก Apache Hadoop 28
3.20 Flowchart การทำงานของ AR application 28
3.22 Web Use Case Diagram 29
3.23 Diagram การทำงานระหว่าง web application, web server และ HDFS 31
3.24 Dashboard page 32
3.25 Real-time page 32
3.26 Historical data page 32
3.28 ชุดโมเดลจำลอง 33
3.29 แผนผังวงจรที่ 1 33
3.30 แผนผังวงจรที่ 2 34
3.31 โมเดลจำลองทั้ง 3 สถานะ 34
4.1 ข้อมูลที่ส่งไปยัง Apache Kafka 35
4.2 ข้อมูลที่เข้ามายัง Apache Kafka 36
4.3 Apache Kafka ส่งข้อมูลมาเก็บไว้ที่ HDFS 37
4.4 ลักษะการแบ่งสถานะด้วยค่า RPM 39
4.5 Chart ข้อมูลที่ gearbox ทำงานปกติ 40
4.6 Chart ข้อมูลที่ gearbox มีสถานะใกล้ชำรุด 40
4.7 Chart ข้อมูลที่ gearbox มีสถานะชำรุด 41
4.8 การทดลองแสดงผลจาก QR code 45
4.9 การทดลองรับข้อมูล JSON 46
4.10 แสดงข้อมูล JSON ที่ได้รับมาในรูปแบบ AR 47

บทที่ 1
บทนำ

1.1 ความเป็นมาของปัญหา

ในปัจจุบันนี้ประเทศไทยกำลังขับเคลื่อนการพัฒนาประเทศในทุกๆ ด้านให้เข้าสู่ความเป็นดิจิทัล ในด้านอุตสาหกรรมการผลิตก็เช่นกันเริ่มมีการปรับเป็น Industry 4.0 เพิ่มขึ้นเรื่อยๆ โดยแนวคิดหลักของ Industry 4.0 ก็คือ การบูรณาการโลกของการผลิต เข้ากับการเชื่อมต่อทางเครื่อข่ายในรูปแบบ “Internet of Things (IoT)” ทำให้สามารถเชื่อมการผลิตสินค้าเข้ากับเทคโนโลยีดิจิทัล ยกตัวอย่างเช่น การใส่ตัวส่งข้อมูลในเครื่องใช้ไฟฟ้าเพื่อประมวลสถิติการใช้และแจ้งกลับไปยังโรงงานเมื่อเกิดปัญหา โดยจะมีองค์ประกอบหลักๆ 9 ด้าน คือ Autonomous robots, Simulation, System integration, Internet of Things, Cybersecurity, Cloud computing, Additive manufacturing, AR และ Big Data ซึ่งผลลัพธ์ที่ได้ก็คือสามารถผลิตสินค้าหลากหลายรูปแบบแตกต่างกัน (ตามความต้องการเฉพาะของผู้บริโภคแต่ละราย) เป็นจำนวนมากในเวลาอันรวดเร็ว โดยใช้กระบวนการผลิตที่ประหยัดและมีประสิทธิภาพด้วยเทคโนโลยีดิจิทัลครบวงจรแบบ “Smart Factory”



รูปที่ 1.1 ยุคอุตสหกรรม 4.0[1]

ข้อดีอีกอย่างของการที่เครื่องจักรนั้นมีเชื่อมต่อกับ IoT คือทำให้เราสามารถมีการทำการซ่อมบำรุง เครื่องจักรได้อย่างหลากหลายและมีประสิทธิภาพมากขึ้น โดยในด้านการซ่อมบำรุงเครื่องจักรของอุตสาหกรรมการผลิตนั้นมีทั้ง preventive maintenance หรือ การบำรุงรักษา

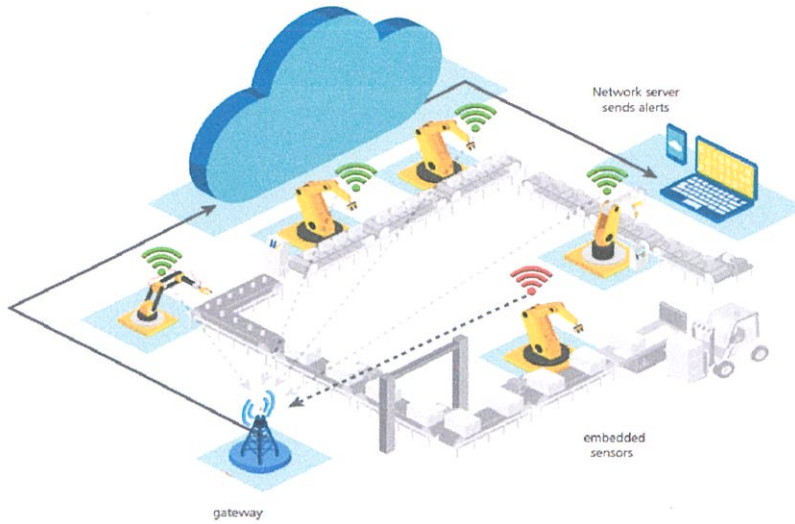
---

[1] https://www.applicadthai.com/tag/industry-4-0/

เชิงป้องกัน ซึ่งมักจะนิยมใช้กันในอุตสาหกรรมการผลิตทั่วไป เป็นการบำรุงรักษาเครื่องจักรตามระยะเวลาที่กำหนดขึ้น โดยอาจได้มาจาก ประสบการณ์ หรือคู่มือการใช้งานของเครื่องจักรนั้น ๆ เพื่อให้เครื่องจักรสามารถทำงานได้ตามอายุการใช้งานของเครื่องจักร แต่ข้อเสียที่ตามมาคือการทำ preventive maintenance ไม่สามารถขจัดปัญหาการชำรุดเครื่องจักรโดยไม่คาดคิดได้ โดยเมื่อเครื่องจักรเสียหาย อาจจะหาแรงงาน หรืออะไหล่มาไม่ได้ในทันที หรือแม้ว่าจะทำการบำรุงรักษาตามแผนแล้วก็ตาม การบำรุงรักษาตามเวลานั้น เราอาจมีการเปลี่ยนชิ้นส่วนบางชิ้นโดยไม่จำเป็น ทั้งนี้ เนื่องจากสภาพการใช้งานมีความแตกต่างในแต่ละเครื่องอาจจะได้รับงานที่หนักไม่เท่ากัน ทำให้อาจมีการชำรุดก่อนเวลาที่กำหนดได้ ดังนั้นจึงเริ่มมีเปลี่ยนไปใช้การซ่อมบำรุง เครื่องจักรแบบ predictive maintenance หรือ การบำรุงรักษาเชิงการคาดการณ์ เป็นการคาดการณ์อัตราการเสื่อมของเครื่องจักร จากผลการตรวจวัดโดยใช้เครื่องมือจะสามารถทำให้ได้ข้อมูลที่สามารถคาดการณ์การเฝ้าระวังปัจจัยเสี่ยงต่างๆ อาทิเช่น ระดับสัญญาณความสั่นสะเทือน (Vibration analysis), สารหล่อลื่น(Oil wear particle analysis), คลื่นความร้อน(Temperature monitoring) และการสึกหรอหรือรอยแตกร้าว (Thickness tester) โดยนำข้อมูลที่เก็บได้มาประมวลผลเพื่อพยากรณ์อาการชำรุดที่จะเกิดขึ้นในอนาคตได้ ในขณะที่ยังสามารถจัดเตรียมล่วงหน้าสำหรับ แรงงาน ชิ้นส่วน อะไหล่ แผนการผลิตที่อาจมีผลกระทบก่อนที่จะเกิดความเสียหายจริงขึ้น เพื่อป้องกันความเสียหายที่เกิดขึ้นทั้งจากเครื่องจักร และระบบการผลิตของโรงงานอุตสาหกรรมนั้น เพราะความเสียหายที่เกิดขึ้นจริงนั้นนอกจากจะต้องคำนวณความเสียหายเป็นค่าซ่อมและ ค่าอะไหล่ของเครื่องจักรแล้ว ยังต้องคำนึงไปถึง operational machine downtime เพราะ เมื่อเครื่องจักรหยุดทำงานเป็นระยะเวลาหนึ่งเพื่อรอการซ่อมแซม ระบบการผลิตอาจเกิดความเสียหายทำให้ไม่สามารถผลิตสินค้าได้ หรือ ผลิตได้ช้าลง ทำให้สูญเสียรายได้ที่ควรจะได้ไป จึงทำให้การทำ predictive maintenance นั้นจำเป็นอย่างยิ่ง เพราะจะทำให้วางแผนจัดการ operational machine downtime ที่จะเกิดขึ้นได้ โดยอาจเตรียม แรงงาน ชิ้นส่วน อะไหล่ แผนการผลิตที่อาจมีผลกระทบเพื่อลดความเสียหาย ที่จะเกิดขึ้นจากเครื่องจักรที่กำลังจะเสีย ทำให้หลีกเลี่ยงความเสียหายที่จะเกิดขึ้นได้ ถึงแม้ predictive maintenance จะสามารถแก้ปัญหาการซ่อมบำรุงเครื่องจักรนี้ได้ แต่การจะทำ predictive maintenance นั้นจำเป็นต้องใช้เครื่องมือวัดที่ทันสมัย ใช้ทักษะของผู้ตรวจสอบและผู้ชำนาญการในวิเคราะห์ข้อมูลขั้นสูง ซึ่งทำให้มีค่าใช้จ่ายในการตรวจวัดและวิเคราะห์สูงทำให้อุตสาหกรรมการผลิตในไทยนั้นยังขาดแคลนการซ่อมบำรุงแบบการบำรุงรักษาเชิงการคาดการณ์นี้อยู่

ทำให้ผู้จัดทำโครงการมีความสนใจที่จะจัดทำระบบสำหรับสาธิตการซ่อมบำรุงด้วย predictive maintenance platform ในโรงงานโดยนำเอาเทคโนโลยีดิจิทัลเข้ามาช่วย โดยแนวคิดคือการสร้าง predictive maintenance platform โดยใช้ IoT ในการรับส่งข้อมูลของ sensor ที่ติดอยู่กับชุด gearbox ซึ่งเราสามารถจำลองสถานะต่างๆของเครื่องจักรได้โดยอ้างอิงจากรูปแบบที่เคยมีคนนำ gearbox

มาทดลอง เพื่อใช้เป็นโมเดลจำลองการส่งข้อมูลจากเครื่องจักรในระบบการผลิตของโรงงานดังในรูปที่ 1.2 ให้กับระบบสาธิตการบำรุงรักษาเชิงการคาดการณ์ จากนั้นนำไปประมวลผลด้วย machine learning เพื่อทำ predictive maintenance คอยตรวจจับการชำรุดของเครื่องจักรจากข้อมูลที่ได้รับแล้วแจ้งเตือนผู้ใช้



รูปที่ 1.2 โมเดลตัวอย่างระบบ IoT ในโรงงาน[2]

จากนั้นนำไปแสดงผลด้วยเทคโนโลยี AR และ web application โดยที่เลือกใช้ AR เนื่องจากในระบบการผลิตจริงนั้นเป็นเรื่องยากที่จะค้นหาเครื่องจักรที่เราต้องการ ดังนั้น AR จะเป็นส่วนช่วยในการ identify เครื่องจักรและยังสามารถแสดงค่าสถานะต่างๆของเครื่องจักรดังในรูป 1.3 ได้อีกด้วย

รูปที่ 1.3 การใช้ AR ในการ monitor เครื่องจักร[3]

---

[2] https://www.semtech.com/.../Semtech_HB_PredictiveMaintenance_AppBrief-FINAL.pdf

[3] ***https***://www.ptc.com/en/test-alias

จากที่กล่าวมาโครงการนี้จึงมีความคิดที่จะนำเสนอ ระบบสาธิตการบำรุงรักษาเชิงการคาดการณ์ที่รวบรวมเทคโนโลยีตามที่กล่าวไว้ข้างต้นไว้ด้วยกัน โดยจะเน้นการพัฒนาไปที่ architecture เพื่อให้ใช้งานได้จริงในโรงงาน โดยสาธิตการทำงานจากโมเดลจำลอง

1.2 วัตถุประสงค์โครงการ

1) ศึกษาและประยุกต์ใช้งาน IoT ในโรงงานอุตสาหกรรม
2) ศึกษาและประยุกต์ใช้งาน Big Data และการประมวลผลแบบ real-time
3) ศึกษาและประยุกต์ใช้งาน machine learning บน Big Data
4) ศึกษาและประยุกต์ใช้งาน AR (Augmented reality)
5) ศึกษาและจำลองระบบการบำรุงรักษาเชิงการคาดการณ์โดยใช้เทคโนโลยีที่กล่าวมาข้างต้น

1.3 ประโยชน์ที่คาดว่าจะได้รับ

1) สามารถคาดการณ์การชำรุดของเครื่องจักร
2) ช่วยให้วางแผนบำรุงรักษาก่อนที่เครื่องจักรจะชำรุด
3) ช่วยลดความเสียหาย ที่เกิดขึ้นจากการรอบำรุงรักษา
4) มีระบบมอนิเตอร์ในการดูสถานะ ระบุตัวเครื่องจักร จากเครื่องจักรจำนวนมาก

1.4 ขอบเขตของโครงงาน

ตัวโครงงานนี้เป็นการจำลองแพลตฟอร์มคาดคะเนการเสื่อมสภาพของอุปกรณ์หรือเครื่องจักร โดยจะสร้างโมเดลทดสอบจากชุด gearbox เพื่อใช้ในการทดลองส่งข้อมูลต่าง ๆ จาก sensor แล้วส่งข้อมูลไปเก็บไว้ จากนั้นนำไปประมวลผลด้วย machine learning พอได้ผลลัพธ์แล้วสามารถนำมาแสดงผ่าน AR application และ web application

โดยสามารถแบ่งระบบได้เป็น 3 ส่วนย่อยได้แก่ IoT, machine learning และ ส่วนแสดงผล

1.4.1 IoT

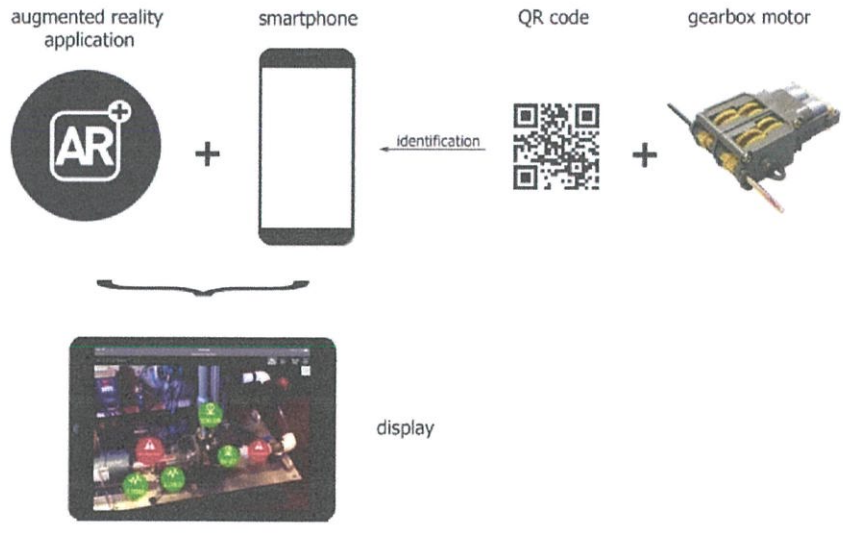
จะมีการรับข้อมูลมาจาก Sensor ต่าง ๆ ซึ่งติดอยู่กับชุด gearbox โดยเก็บข้อมูลจาก sensor ด้วย Raspberry Pi ส่งข้อมูลไปยัง Apache Kafka เพื่อนำไปเก็บที่ storage และส่งข้อมูลแบบ streaming ไปยัง spark streaming เพื่อคาดการณ์การชำรุดแบบ real-time

1.4.2 Machine learning

จะนำข้อมูลจาก storage มาวิเคราะห์การชำรุดของชุด gearbox เพื่อให้สามารถรู้ถึงโอกาสที่จะชำรุดได้ก่อนเหตุการณ์จะเกิดขึ้น โดยใช้ machine learning มาทำการวิเคราะห์ด้วย Neural Network algorithm เพื่อให้ได้ model สำหรับการทำ predictive maintenance สำหรับชุด gearbox และ ระบบ predictive แบบ real-time ที่ใช้ในการตรวจจับ แจ้งเตือนแก่ผู้ใช้ผ่าน platform ต่าง ๆ เช่น Line หรือ Facebook

1.4.3 AR Module

ในส่วนของการแสดงผลจะมี 2 แบบ คือ AR application และ web application โดยการแสดงผลแบบ AR application จะใช้โทรศัพท์ในการสแกน QR code เพื่อนำ identification มาดึงข้อมูลจาก storage นำมาแสดงในรูปแบบ AR บนโทรศัพท์ ส่วน web application จะมี Dashboard ที่แสดงข้อมูลทั้งหมด ของตัวระบบดังรูปที่ 1.4

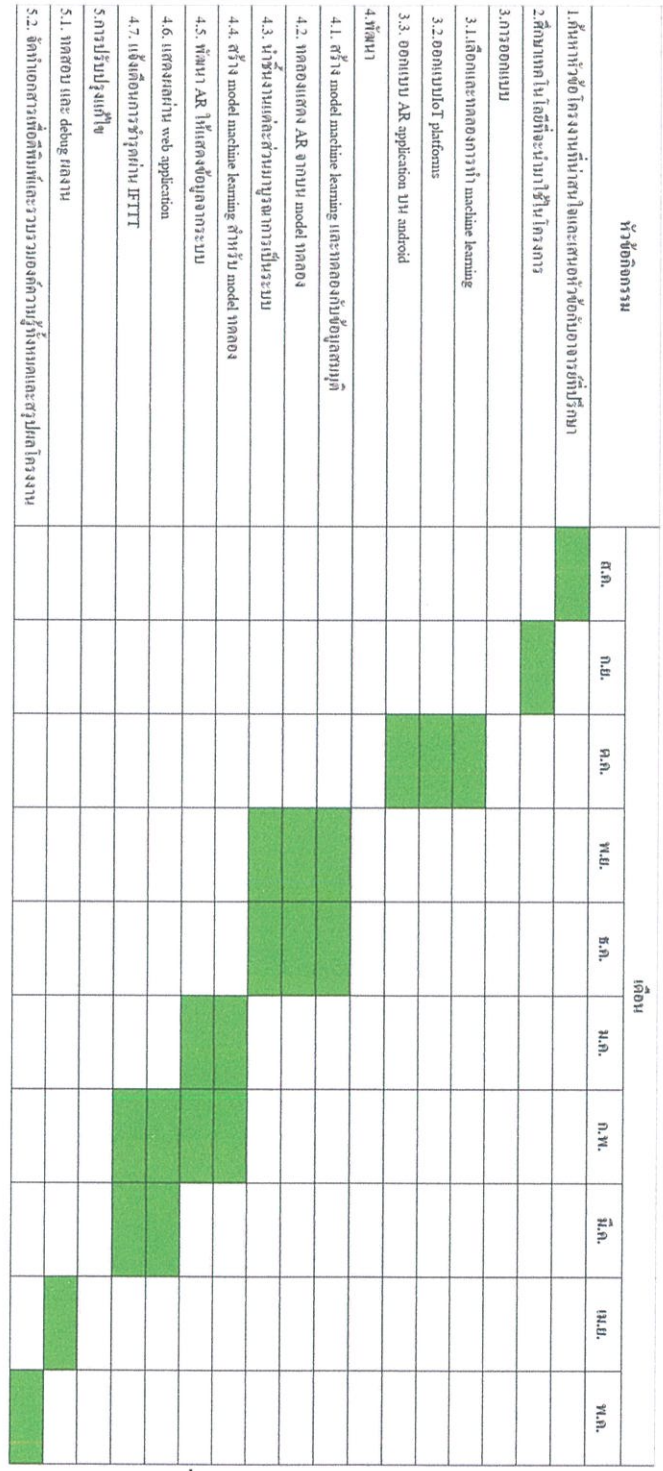


รูปที่ 1.4 AR Module

1.5 ข้อจำกัดของแบบ

เนื่องจากค่าข้อมูลที่ใช้ได้มาจาก model gear box สาธิตจึงได้ค่า parameter ไม่เหมือนกับในโรงงานอุตสาหกรรมเพราะในเครื่องจักรที่ละเอียดอ่อนนั้นจะตรวจวัดค่าได้ทั้ง ความสั่นสะเทือน สารหล่อลื่น ความร้อน และการแตกร้าว ซึ่ง model ไม่สามารถจำลองได้ทั้งหมดแต่วิธีการ และระบบนำเสนอนั้นสามารถนำไปประยุกต์ใช้งานได้จริง

1.6 แผนการดำเนินงาน



รูปที่ 1.5 แผนการดำเนินงาน

บทที่ 2
ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1 การทำ Predictive Maintenance

การบำรุงรักษาเชิงการคาดการณ์ (Predictive Maintenance) เป็นการคาดการณ์อัตราการเสื่อมของเครื่องจักรจากผลการตรวจวัดโดยใช้เครื่องมือตรวจจับต่างๆเพื่อให้ได้ข้อมูลที่สามารถคาดการณ์การเฝ้าระวังปัจจัยเสี่ยงต่างๆ อาทิเช่น ระดับสัญญาณความสั่นสะเทือน (Vibration analysis), สารหล่อลื่น (Oil wear particle analysis), คลื่นความร้อน(Temperature monitoring) และการสึกหรอหรือรอยแตกร้าว (Thickness tester) เพื่อนำมาใช้ process data พยากรณ์อาการชำรุดที่จะเกิดขึ้นในอนาคตได้ ในขณะที่ยังสามารถจัดเตรียมการล่วงหน้าสำหรับ แรงงาน ชิ้นส่วน อะไหล่ แผนการผลิตที่อาจมีผลกระทบก่อนที่จะเกิดความเสียหายจริง

2.2 เทคโนโลยีที่เกี่ยวข้องกับการรับข้อมูล

จากที่ได้ทำการศึกษาข้อมูลเกี่ยวกับการทำ predictive maintenance โดยทั่วไปแล้ว จะมีปัจจัยเสี่ยงที่สามารถช่วยในการทำนายการชำรุดของเครื่องจักรได้ โครงงานนี้จึงได้จัดทำ model จำลองจาก gearbox ขึ้นโดยใช้ controller คอยรับส่งข้อมูล sensor ต่างๆ โดยจะมี sensor ที่ใช้ตรวจวัดปัจจัยเสี่ยงต่างๆ

2.2.1 Controller

Raspberry PI เป็นคอมพิวเตอร์ขนาดเล็กราคาถูก สามารถเขียนโปรแกรมติดต่อกับภายนอกได้ง่าย

2.2.2 Sensor สำหรับวัดค่าพารามิเตอร์ต่างๆ

2.2.2.1 Temperature sensor

Infrared Thermometer (MLX90614) เป็นโมดูลวัดอุณหภูมิโดยใช้สัญญาณ Infrared และติดต่อสื่อสารกับ Raspberry Pi โดย I2C ใช้ไฟเลี้ยง DC 3.3 - 5V

2.2.2.2 Vibration sensor

ใช้ Accelerometer sensor ซึ่งใช้ IC เบอร์ ADXL335 ในการตรวจจับความสั่นไหว ใช้ไฟเลี้ยง DC 3-5 V โดยให้สัญญาออกมาแบบ Analog

2.2.2.3 Motor speed sensor counter

ใช้ infrared ในการตรวจจับวัตถุที่มาบังร่องตรวจจับและส่งสัญญาออกมาเป็น Digital เพื่อนำไปคำนวนความเร็วรอบของมอเตอร์ได้ ใช้ไฟเลี้ยง DC 3.3-5V

2.2.2.4 **Voltage and current sensor module**

โมดูลนี้ใช้ IC เบอร์ MAX471/472 เป็น sensor วัดแรงดันและกระแสไฟฟ้า โดยให้สัญญาณออกมาเป็น Analog ใช้ไฟเลี้ยง DC 3.3 - 5 V

2.2.3 การรับข้อมูลจาก sensor

ใช้ภาษา Python ในการควบคุม controller เพื่อข้อมูลจาก sensor ที่ส่งข้อมูลชนิด Digital จะใช้ library WiringPi2-Python ในการติดต่อระหว่าง sensor และ controller ผ่าน GPIO ส่วน sensor ที่ส่งข้อมูลชนิด Analog จะรับข้อมูลผ่าน IC MCP3008 ซึ่งเป็น IC Analog to Digital 8 Channel 10 Bit และติดต่อกับ controller ผ่าน SPI ได้โดยตรง

2.3 การรับข้อมูลจาก controller

เมื่อ controller ได้รับข้อมูลจาก sensor แล้ว controller จะส่งข้อมูลไปยัง IoT platform เพื่อนำข้อมูลไปจัดเก็บต่อไป

2.3.1 ทฤษฎีเกี่ยวกับสถาปัตยกรรมที่ใช้

2.3.1.1 **Apache Kafka**

เป็น platform ที่ใช้กระจายข้อความแบบ streaming โดยอาศัยหลัก public/subscriber กับ topic ซึ่งเป็นตัวแบ่งแยก massage ต่างๆ เหมือน channel หรือ pipeline โดยหากมองง่ายๆก็คือ topic จะเป็นทางเชื่อ ระหว่าง Producers (public) ซึ่งทำหน้าที่ส่งข้อมูลให้กับ topics ที่กำหนดไว้ คอยแจกเลือก partition ภายใน topic เอง ซึ่งปกติทำงานเป็นแบบ round-robin เพื่อความสมดุล หรือ อาจจะใช้ function อื่นในการกระจายก็ได้ และ Consumer (subscriber) ที่รอรับข้อมูลจาก topic

โดยในโครงการนี้ใช้ Apache Kafka ในการรับข้อมูลจาก controller ที่เชื่อมต่อกับ sensor หลายชนิด และจากหลายเครื่องทำให้การจัดการข้อมูลให้เป็นระเบียบและมีประสิทธิภาพ

2.4 การเก็บข้อมูล

2.4.1 ทฤษฎีเกี่ยวกับสถาปัตยกรรมที่ใช้

2.4.1.1 **Hadoop Data File System(HDFS)**

เป็นตัวเก็บข้อมูลแบบกระจาย คือสามารถแบ่งเก็บข้อมูลขนาดใหญ่เป็น file ย่อยๆลงใน Data Node จำนวนมากโดยจะมี Master Node ทำหน้าที่ระบุว่าข้อมูลอยู่ที่ Data Node ไหน โดย HDFS เป็น fault-tolerant ที่จะเก็บข้อมูลซ้ำๆในหลายๆที่ทำให้เก็บข้อมูลได้ดีไม่สูญหาย และสามารถเป็น Horizontal Scale ได้ ดังนั้น HDFS จึงเหมาะสำหรับโครงการนี้เพราะนอกจากจะเก็บข้อมูลได้คงทนแล้วยังสามารถโหลดข้อมูล data ขนาดใหญ่ที่เก็บไว้มาทำการประมวลผลด้วย machine learning ที่ต้องใช้ข้อมูลจำนวนมากในการคาดการณ์การชำรุดของเครื่องจักร

2.5 การประมวลผลด้วย machine learning

2.5.1 ทฤษฎีเกี่ยวกับ machine learning

Machine learning เป็นสาขาหนึ่งของ ปัญญาประดิษฐ์ (Artificial Intelligence) ซึ่งเป็นศาสตร์ที่สามารถทำให้คอมพิวเตอร์สามารถเรียนรู้ได้โดยไม่ต้องเขียนโปรแกรมลงไปตรงๆ โดย machine learning จะเรียนรู้ที่จะจัดการกับข้อมูลที่รับเข้ามาด้วยตัวเอง ตามความสัมพันธ์ของข้อมูลที่รับเข้ามา โดยเมื่อลักษณะข้อมูลมีการเปลี่ยนแปลง machine learning ก็จะคอยวิเคราะห์ และปรับแก้ไขโปรแกรมด้วยตัวมันเอง เพื่อจัดการกับข้อมูลชุดใหม่ที่มีลักษณะข้อมูลที่ความสัมพันธ์ต่างจากเดิม

โดยพื้นฐานแล้ว machine learning สามารถแบ่งหลักๆออกเป็น 2 ประเภท คือ Supervised learning และ Unsupervised learning

2.5.1.1 **Supervised learning**

Supervised learning หรือ การเรียนรู้แบบได้รับคำแนะนำ สามารถบอกให้เข้าใจได้ง่ายคือ การที่ machine learning ได้รู้ก่อนว่าผลลัพธ์ของข้อมูลแบบนี้คืออะไร เช่น เรานำข้อมูลของคนเป็นมะเร็ง กับ คนไม่เป็นมะเร็ง (training data) มาให้ดู และฝึกสอนว่าคนไข้รายนี้เป็นมะเร็ง (label) โดย machine learning ก็จะเริ่มเรียนรู้ตามที่ label ไว้จากคุณสมบัติข้อมูล (feather)โดยแต่ละกรณีที่ใช้ supervised learning ก็อาจมีความแตกต่างกันของการใช้งานโดยแบ่งออกคร่าวๆได้เป็น

1) Classification เป็นการจัดหมวดหมู่ให้ข้อมูลโดยนิยมใช้กับข้อมูลที่ไม่ต่อเนื่อง (discreate)
2) Regression จะเป็นการใช้วิเคราะห์ข้อมูลที่มีความต่อเนื่อง

โดยทั้งนี้ supervised learning จะสามารถหาคำตอบได้แค่เท่าที่เราสอนมันไว้ทำให้มีการ learning แบบ Unsupervised learning

2.5.1.2 **Unsupervised learning**

เป็นการแบ่งข้อมูลเป็นกลุ่ม (clustering) แต่เราจะไม่สามารถระบบบุได้ว่าปลายทางเป็นอะไรบ้างได้แค่จัดข้อมูลแบ่งออกเป็นกลุ่มๆ เช่น การแบ่งลูกค้าออกเป็นกลุ่มย่อยๆ ตามลักษณะของลูกค้า เป็นต้น

การเรียนรู้ของ machine learning เพื่อสร้างแบบจำลอง (model) นั้นมีอยู่ 3 ขั้นตอน

1) Feature extraction การแปลงลักษณะของข้อมูล ที่อยู่ในรูปแบบของข้อความหรือรูปภาพ ให้กลายเป็นชุดตัวเลข
2) Regularization พิจารณาคุณสมบัติ (feature) ของข้อมูลว่าส่งผลกระทบกับข้อมูลมากแค่ไหน เพื่อตัดหรือลดส่วนที่ไม่จำเป็นออกเพื่อทำให้โมเดลซับซ้อนน้อยลง
3) Cross-validation หลังจากได้ model มาก็นำมาทดสอบว่าโมเดลที่ได้สามารถวิเคราะห์ข้อมูลได้อย่างแม่นยำเพียงพอหรือไม่ โดยอาจนำไปทดสอบกับข้อมูลที่ไม่เคยเป็น input ของ model มาก่อน

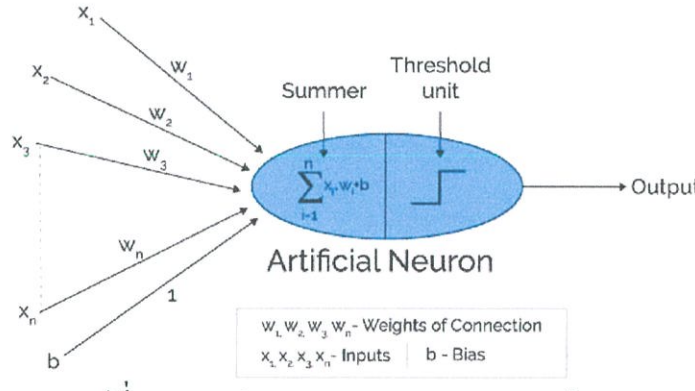
ทั้งนี้ โครงการนี้ได้นำ machine learning มาใช้ในการจะแนกเครื่องจักรว่ามีแนวโน้มจะชำรุดหรือไม่จากข้อมูลของเครื่องจักร

2.5.2 Algorithm ที่ใช้ในการทำ machine learning

Neural network (โครงข่ายประสาทเทียม) เป็นการจำลองการทำงานของสมองมนุษย์ด้วยคณิตศาสตร์ โดยสมองของมนุษย์นั้นจะประกอบไปด้วยตัวประมวลผลที่เรียกว่า นิวรอน (Neuron) โดยตัวประมวลผลนี้จะรับส่งข้อมูลกันด้วยไซแนปส์ (synapses) ซึ่งทุกๆ neuron นั้นจะทำงานไปพร้อมๆกัน ทำให้เกิดเป็นโครงข่ายประมวลผลหรือ neural network โดยการทำงานของแต่ละ neuron นั้นจะขึ้นอยู่กับปริมาณการกระตุ้นจากเซลล์ที่เชื่อมต่อกับมันโดยหากกระแสประสาทที่ส่งมาแรงพอก็จะสามารถส่งต่อไปยังเซลล์อื่นได้

รูปที่ 2.1 โครงสร้าง node ของ neural network [4]

โดยกลไกข้างต้นนั้นถูกนำมาประยุกต์เป็น โครงข่ายประสาทเทียม (Artificial neural network) โดยเปรียบ neuron เป็น node ซึ่งสามารถรับส่งข้อมูลกันได้เหมือนโครงข่ายประสาท



รูปที่ 2.2 การทำงานของ Artificial Neuron[5]

โดยถ้ามองจากรูปด้านบน โครงข่ายประสาทจะประกอบด้วย 5 องค์ประกอบหลักดังนี้

---

[4] https://medium.com/@startuphackers/building-a-deep-learning-neural-network-startup-7032932e09c

[5] https://hackernoon.com/overview-of-artificial-neural-networks-and-its-applications-2525c1addff7:

1) ข้อมูลเข้า (input) เป็นข้อมูลที่อยู่ในรูปตัวเลข หรือในภาพคือ x_1-x_n
2) ข้อมูลส่งออก (output) ผลลัพธ์จากกระบวนการที่เกิดขึ้นในโครงข่ายประสาทเทียม
3) ค่าน้ำหนัก (weights) เป็นเหมือนค่าความรู้ (knowledge) เป็นสิ่งที่ใช้ในการช่วยพิจารณาค่าข้อมูลที่เข้ามา โดยอิงจากทารทดลองที่ผ่านมา
4) ฟังก์ชันผลรวม (summation function) เป็นผลรวมเมื่อนำ input มารวมเข้ากับน้ำหนักแล้ว
5) ฟังก์ชันการแปลง (transfer function) เป็นการคำนวนเพื่อจำลองการทำงานของโครงข่ายประสาท

2.5.3 ทฤษฎีเกี่ยวกับสถาปัตยกรรมที่ใช้

2.5.3.1 **Apache Spark**

เป็นตัวประมวลผลข้อมูลโดยมีจุดเด่นคือสามารถทำงานบน memory และประมวลผลข้อมูลขนาดใหญ่ได้ โดยความสามารถของ spark นั้นมากพอจะประมวลผลข้อมูลจากเครื่องจักรได้มากเกิน 100 เครื่อง ทำให้เหมาะอย่างมากกับโครงการนี้อีกทั้งยังรองรับภาษา Scala, Python, Java และ R โดยโครงการนี้เลือกใช้ python Spark นั้นมี 3 module หลักคือ Spark Core, Spark Streaming และ MLlib โดยในโครงการนี้จะใช้2 ตัวหลักๆคือ Spark Core ในการประมวลผลข้อมูลแบบ batch เพื่อสร้าง module machine learning และ Spark Streaming ในการประมวลผลแบบ real-time

2.5.3.2 **MLlib**

MLlib หรือ Machine Learning Library เป็น library ในการประมวลผล machine learning สำหรับ spark โดยเฉพาะเนื่องจาก ตัว MLlib ถูกออกแบบมาเพื่อการ distributed processing โดยภายในรองรับ Algorithm และ utilities ดังนี้

- Classification
- Regression
- Clustering
- Collaborative filtering
- Dimensionality reduction

โดยในโครงการนี้ได้นำ MLlib มาประยุกต์ในการใช้ทำ machine learning สำหรับคาดการณ์การชำรุดของเครื่องจักรด้วย Multilayer perceptron classifier (neural network) algorithm

2.5.3.3 **Spark Streaming**

Spark Streaming เป็นส่วนเสริมของ core Spark API โดยนอกจากจะทำงานประมวลผลได้เช่นเดียวกับ spark แล้วยังสามารถนำมาใช้ร่วมกับ platform streaming ต่างๆ เช่น Kafka, Flume หรือ Kinesis เป็นต้น โดยตัว spark streaming จะสามารถประมวลผลจากข้อมูลที่ได้รับมาได้แบบ real-time แล้วอาจนำไปแสดงผลใน dashboards, Databases หรือ HDFS ได้ โดยโครงการนี้ได้ใช้ spark streaming เพื่อคาดการณ์การชำรุดของเครื่องจักรในแบบ real-time เพื่อให้สามารถแก้ไขปัญหาได้อย่างทันท่วงที

2.6 การแสดงผลด้วย Augmented Reality Technology

2.6.1 คำนิยามของ AR

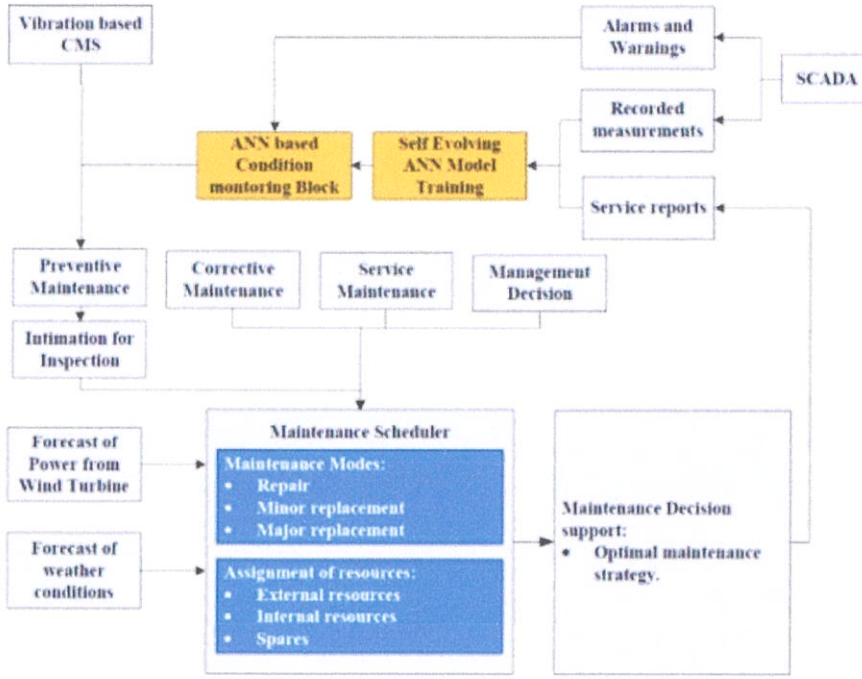
AR (Augmented Reality) หรือเทคโนโลยีความจริงส่วนขยาย เป็นการบูรณาการข้อมูลดิจิตอลเข้ากับสภาพแวดล้อมของผู้ใช้ในเวลาจริง ซึ่งข้อมูลดิจิทอลที่นำมาใช้นั้นมีหลากหลาย เช่น รูปภาพ, วิดีโอ, เสียง, โมเดลสามมิติ เป็นต้น เราสามารถเรียกข้อมูลเหล่านั้นได้ว่าข้อมูลมัลติมีเดีย โดยการนำข้อมูลมัลติมีเดียแหล่านี้เข้าไปรวมกับสภาพแวดล้อม จะมีความแตกต่างกับเทคโนโลยี VR (Virtual Reality) หรือเทคโนโลยีความจริงเสมือนที่มีการนำข้อมูลต่างๆซ้อนทับลงไปในสภาพแวดล้อมจริง

2.6.2 การนำเสนอของ AR

AR มีความต้องการที่แตกต่างขึ้นอยู่กับสภาพแวดล้อมและเงื่อนไข มีสองวิธีสำหรับการแสดงผลของ VR หนึ่ง monitor-based AR ทุกๆฉากที่ถูกแสดงออกมาทางหน้าจอ เป็นการแสดงผลที่ง่ายที่สุด และมีความต้องการ hardware ต่ำ สอง แสดงผลผ่าน head-mounted เป็นการแสดงผลที่มีต้องการ head-mounted เพื่อที่จะแสดงผลและปรับตำแหน่งภาพ โดยที่ภาพจะถูกแสดงบนจอ head-mounted

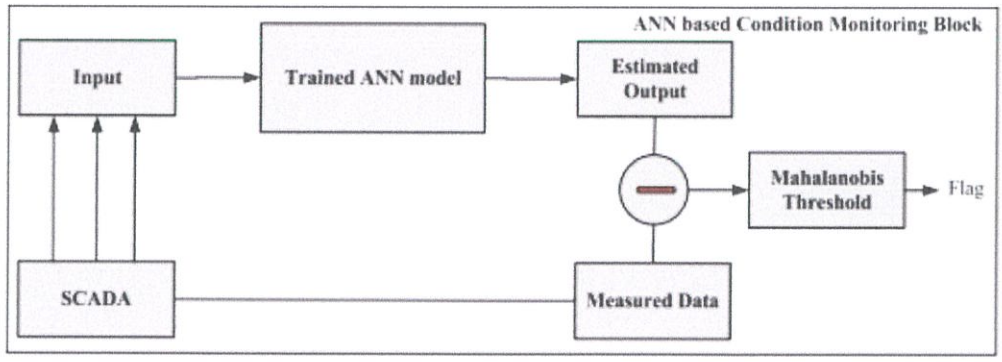
2.7 งานวิจัยที่เกี่ยวข้อง

ในโครงงานนี้ได้เลือกใช้ model gearbox ในการจำลองการทำงานของเครื่องจักรภายในโรงงานโดยอ้างอิงจากงานวิจัย An Artificial Neural Network Approach for Early Fault Detection of Gearbox Bearings โดยงานวิจัยได้พิสูจน์ว่า Gearbox นั้นเป็นปัจจัยหลักในการเกิด downtime ของเครื่องจักร โดยปัจจัยที่สำคัญอยู่ที่ตลับลูกปืนเกียร์ ซึ่งมีผลอย่างมากในการทำ predictive maintenance โดยโครงการนำเสนอ self-evolving maintenance scheduler framework สำหรับจัดการกังหันลมที่ใช้ผลิตไฟฟ้าด้วย artificial neural network (ANN) โดยอ้างอิงข้อมูลจริงจาก gearbox ในกังหันลมที่ริมทะเลในเขตทางใต้ของสวีเดน โดยในโครงการจะใช้ SCADA system



รูปที่ 2.3 Proposed SEMS framework.

ในการบันทึกข้อมูลจาก gearbox ในทุก 10 นาทีซึ่งใช้ framework ตามด้านบน โดยอิงจากงานวิจัยของ the National Renewable Energy Laboratory เกี่ยวกับ gearbox โดยได้ความว่า 70 เปอร์เซนต์ของการชำรุดของ gearbox เกิดจากตลับลูกปืน และอีก 26 เปอร์เซนต์เกิดจากตัวเกียร์ โดยอาศัยปัจจัยในการวิเคราะห์จาก สัญญานการสั่นสะเทือน รอยร้าวภายใน อุณหภูมิ และความดัน จะได้ ตัวแสดงผลดังนี้



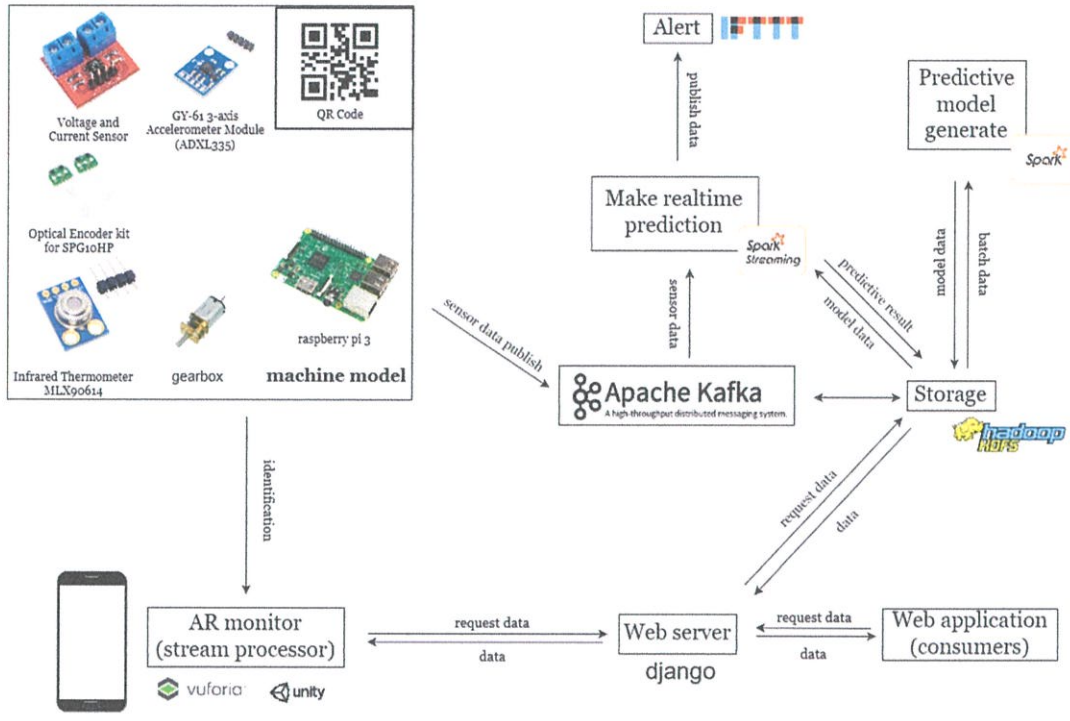
รูปที่ 2.4 ANN-based condition monitoring block

ซึ่งจากงานวิจัยที่กล่าวมาข้างต้นนั้นเป็นกรณีศึกษาให้ประยุกต์ใช้งาน ชุด gearbox เพื่อแทนเครื่องจักรและนอกจากนั้นยังเป็นส่วนช่วยในการเลือก parameter ในการทำ predictive maintenance จนไปถึงการนำ ANN มาทดลองใช้กับโครงการที่จะทำ

บทที่ 3

การออกแบบและพัฒนา

3.1 ภาพรวมการทำงานภายในระบบ



รูปที่ 3.1 System Diagram

โครงการนี้ได้มีความต้องการที่จะจัดทำ predictive maintenance platform ที่ภายในประกอบไปด้วยเทคโนโลยี IoT, Big Data, AR และ machine learning โดยจัดแบ่งจากรูปแบบการทำงานได้ทั้งหมด 4 ระบบหลัก ได้แก่

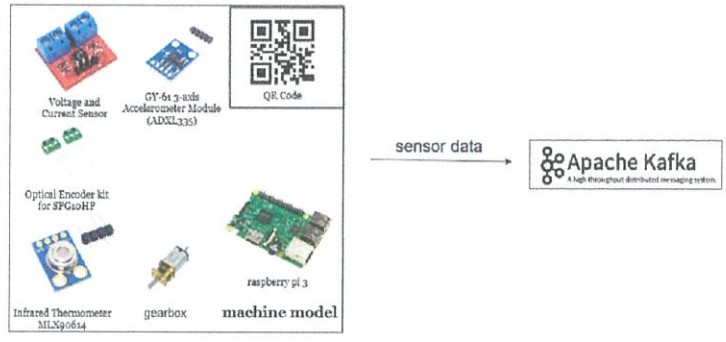
1) ระบบรับข้อมูลจาก gear box
2) ระบบรับส่งข้อมูลภายในระบบ
3) ระบบประมวลผลคาดการณ์การชำรุดของเครื่องจักร
4) ระบบแสดงผลผ่าน AR monitor และ web application

โดยการทำงานเบื้องต้นด้านในแพลตฟอร์มจะเริ่มจาก

3.1.1 ระบบรับข้อมูลจาก gearbox

ซึ่งในส่วนนี้จะประกอบไปด้วย gearbox ซึ่งติด sensor สำหรับตรวจจับปัจจัยเสี่ยงต่างๆไว้เพื่อจำลองการทำงานของเครื่องจักร โดย sensor นั้นจะเชื่อมอยู่กับ Raspberry Pi เพื่อคอยส่งข้อมูลให้กับ Apache Kafka นำข้อมูลไปกระจายในแพลตฟอร์มต่อไปดังรูปที่ 3.2 โดยข้อมูลจะถูกส่งไปในรูปแบบของ JSON โดยภายในจะประกอบไปด้วย

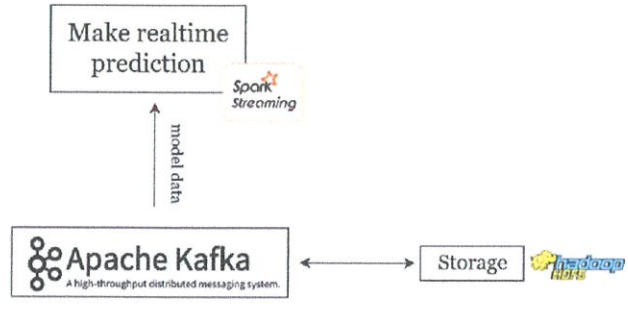
1) Temperature มีหน่วยเป็น องศาเซลเซียส โดยส่งเป็น integer
2) Voltage มีหน่วยเป็นโดยส่งเป็น integer
3) RPM มีหน่วยเป็นรอบ/วินาทีส่งเป็น integer
4) Time ส่งเป็น string
5) ค่าการสั่นสะเทือนแกน x y z รับเป็นตำแหน่งโดยส่งแยก เป็นค่า x ,y และ z โดยส่งเป็น integer



รูปที่ 3.2 ระบบรับข้อมูลชุด gearbox

3.1.2 ระบบรับส่งข้อมูล

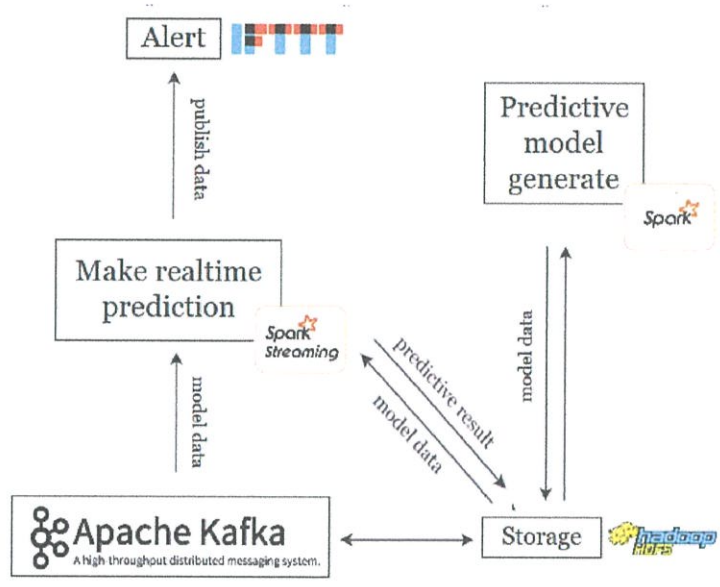
เมื่อ Apache Kafka ได้รับข้อมูลก็จะปฏิบัติตัวเป็นตัวกระจายข้อมูลให้ทั้งระบบโดยจะคอยส่งข้อมูลให้กับระบบเก็บข้อมูล(HDFS) และ ระบบประมวลผลเพื่อคาดการณ์การชำรุดของเครื่องจักรแบบ real-time (Spark Streaming) ดังรูปที่ 3.3



รูปที่ 3.3 ระบบรับส่งข้อมูล

3.1.3 ระบบประมวลผลเพื่อคาดการณ์การชำรุดของเครื่องจักร

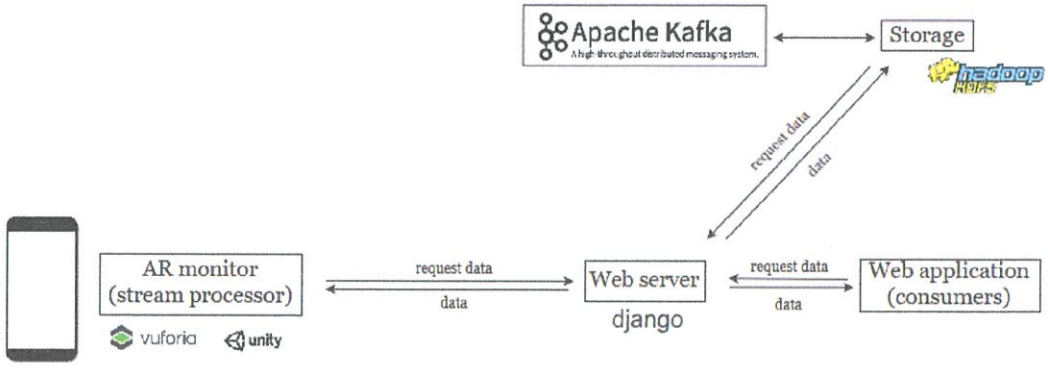
โดยขั้นแรกข้อมูลที่ส่งมาจาก sensor นั้นจะถูกส่งผ่าน Apache Kafka เข้าไปเก็บใน HDFS เพื่อเก็บข้อมูลไว้ใช้ในการทำ machine learning สำหรับสร้าง predictive maintenance model ที่ใช้ในการคาดการณ์การชำรุดของเครื่องจักร ด้วย Spark Core โดยเมื่อได้ model แล้ว จะนำ model ที่ได้เก็บไว้ใน HDFS เพื่อเป็นตัวกลางรับ model ไว้ให้ Spark Streaming สามารถนำ model machine learning ไปทำนายการชำรุดแบบ real-time แล้วจะนำค่าผลลัพธ์ที่ได้ส่งไปเก็บที่ Apache Hadoop และหากเกิดการชำรุดของเครื่องจักรขึ้น ระบบจะส่งข้อมูลเงื่อนไขไปที่ IFTTT เพื่อส่งข้อความไปที่แพลตฟอร์มที่สามารถติดต่อผู้ดูแล เช่น email เพื่อแจ้งเตือนการชำรุดของเครื่องจักร



รูปที่ 3.4 ระบบประมวลผลเพื่อคาดการณ์การชำรุดของเครื่องจักร

3.1.4 ระบบแสดงผล

จะนำข้อมูล sensor และ ผลลัพธ์การคาดการณ์การชำรุดของเครื่องจักรจาก Apache Hadoop ไปแสดงบน web application และ AR application โดยที่ตัว AR application จะใช้งานกับผู้ดูแลระบบที่อยู่ในพื้นที่ เมื่อต้องการตรวจสถานะของเครื่องจักรเครื่องใด ก็สามารถสแกน QR code เพื่อนำ identification มาดึงข้อมูลจาก storage นำมาแสดงในรูปแบบ AR บนโทรศัพท์ ส่วน web application จะมี Dashboard ที่แสดงข้อมูลทั้งหมด ของตัวระบบ



รูปที่ 3.5 ระบบแสดงผล

3.2 การออกแบบการทำงาน และสถาปัตยกรรมของระบบ

3.2.1 ระบบรับข้อมูลจากชุด gearbox

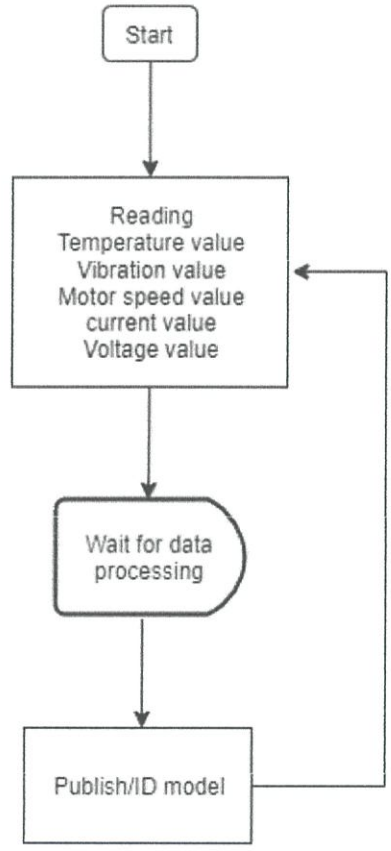
3.2.1.1 ส่วนประกอบ Model gearbox

Model จะประกอบไปด้วย controller เป็นตัวควบคุม ประมวลผล และมี sensor ที่ใช้วัดค่าเพื่อตรวจสอบปัจจัยเสี่ยงที่มีผลต่อ gearbox ดังนี้

- Temperature sensor
- Vibration sensor
- Motor speed sensor counter
- Current sensor module
- Voltage sensor module

โดยเมื่อ ได้รับค่าจาก sensor มาแล้ว Controller ที่ควบคุมด้วยภาษา Python ในการรับข้อมูลต่างๆจาก sensor และส่งต่อข้อมูลต่อไปยัง Apache Kafka ด้วย library Kafka

3.2.1.2 ขั้นตอนการดำเนินงาน



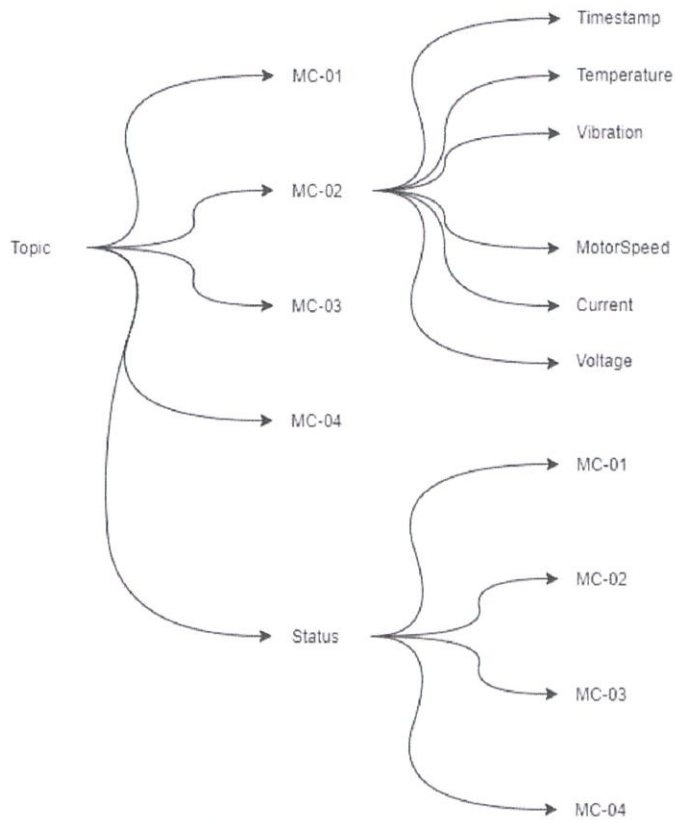
รูปที่ 3.6 Flowchart การรับข้อมูลจาก model

3.2.2 ระบบรับส่งข้อมูลภายในระบบ

3.2.2.1 การออกแบบ Topic

โครงการนี้มี model ทั้งหมด 4 model gearbox จึงได้ออกแบบโดยจะแยก topic ออกเป็นชื่อตาม ID model และมีอีก 1 topic เป็น topic ไว้เก็บข้อมูลสถานะของตัว model ซึ่งภายในประกอบไปด้วยชุดของค่า sensor แล้ว มีเวลาคอยกับกับอยู่สำหรับ และ สำหรับตัวแสดงผลจะเป็น topic status เก็บสถานะจากการคาดการณ์ไว้เพื่อรองรับการคาดการณ์แบบ real-time ด้วย Spark Streaming

3.2.2.2 **Topic ข้อมูลในระบบ**



รูปที่ 3.7 Topic ของระบบ

ซึ่งอธิบายง่ายๆก็คือหากระบบต้องการข้อมูล sensor จาก เครื่องจักร ID MC-02 ก็สามารถ subscribe และได้ data frame ที่มีหน้าตา ดังนี้

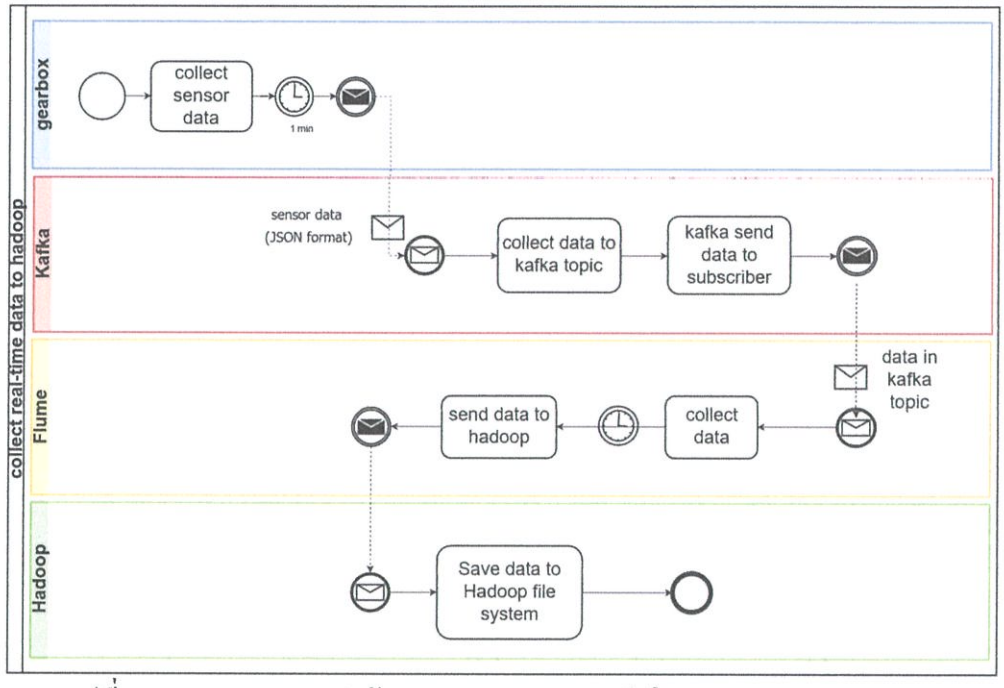
{"Time": "2018-04-29 14:57:57.785971", "RPM": 401, "Temperature": 31, "Xaxis": [513, 515, 512,..., 503], "Yaxis": [503, 496, 509,...496], "Zaxis": [495, 503, 517,... 501], "Volt": 8.85}

รูปที่ 3.8 ตัวอย่าง data frame ข้อมูลจาก topic

3.2.2.3 **Process การทำงาน**

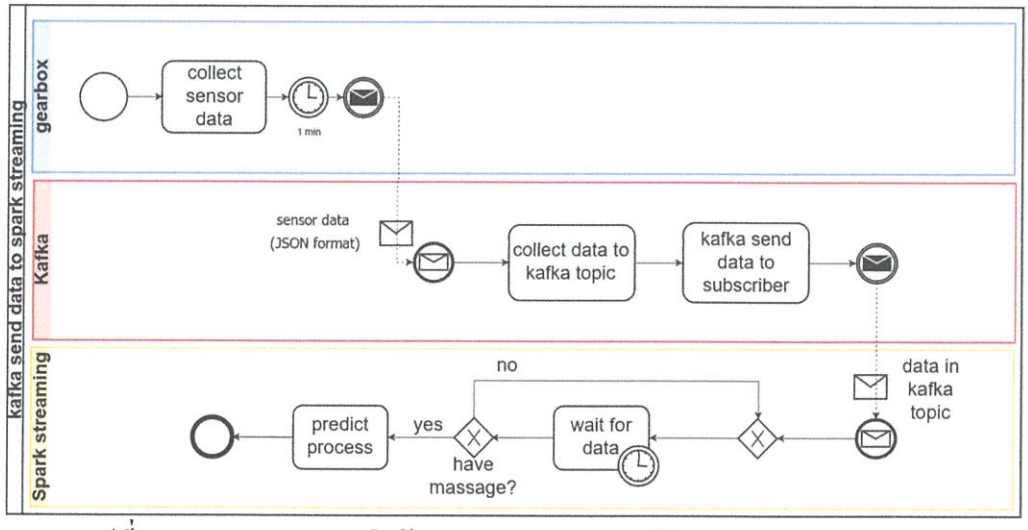
โดยแบ่งการทำงานออกเป็น 2 ส่วนหลักคือ นำข้อมูลที่ได้รับจากชุด gearbox มาเก็บไว้ใน storage (Apache Hadoop) และ นำข้อมูลไปส่งให้ spark streaming โดยจะแสดงขั้นตอนการทำงานให้เห็นดัง BPMN diagram ทั้ง 2 นี้

1) การนำข้อมูลมาเก็บใน HDFS



รูปที่ 3.9 Diagram การนำข้อมูลชุด gearbox มาเก็บใน Apache Hadoop

2) การส่งข้อมูลไปให้ spark streaming

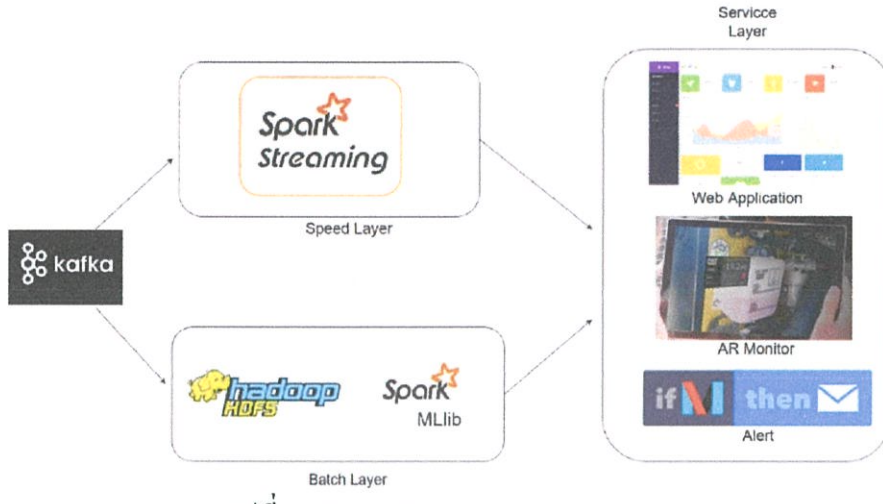


รูปที่ 3.10 Diagram การส่งข้อมูลจาก gearbox มายัง spark streaming

3.2.3 ระบบประมวลผลคาดการณ์การชำรุดของเครื่องจักร

3.2.3.1 รูปแบบสถาปัตยกรรมของระบบ

ระบบประมวลผลคาดการณ์การชำรุดของเครื่องจักรนั้น เป็นการนำ machine learning มาประยุกต์ใช้ในการคาดการณ์การชำรุดของเครื่องจักร โดยใช้ Lambda architecture

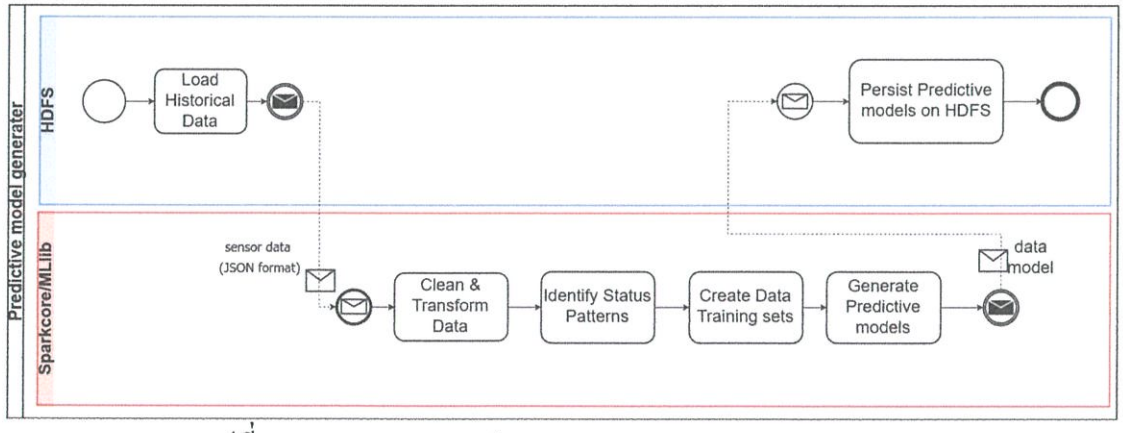


รูปที่ 3.11 Lambda Architecture

โดยจะแบ่งการประมวลผลออกเป็น 2 แบบ คือ

1. **การประมวลผลแบบ Batch Layer**

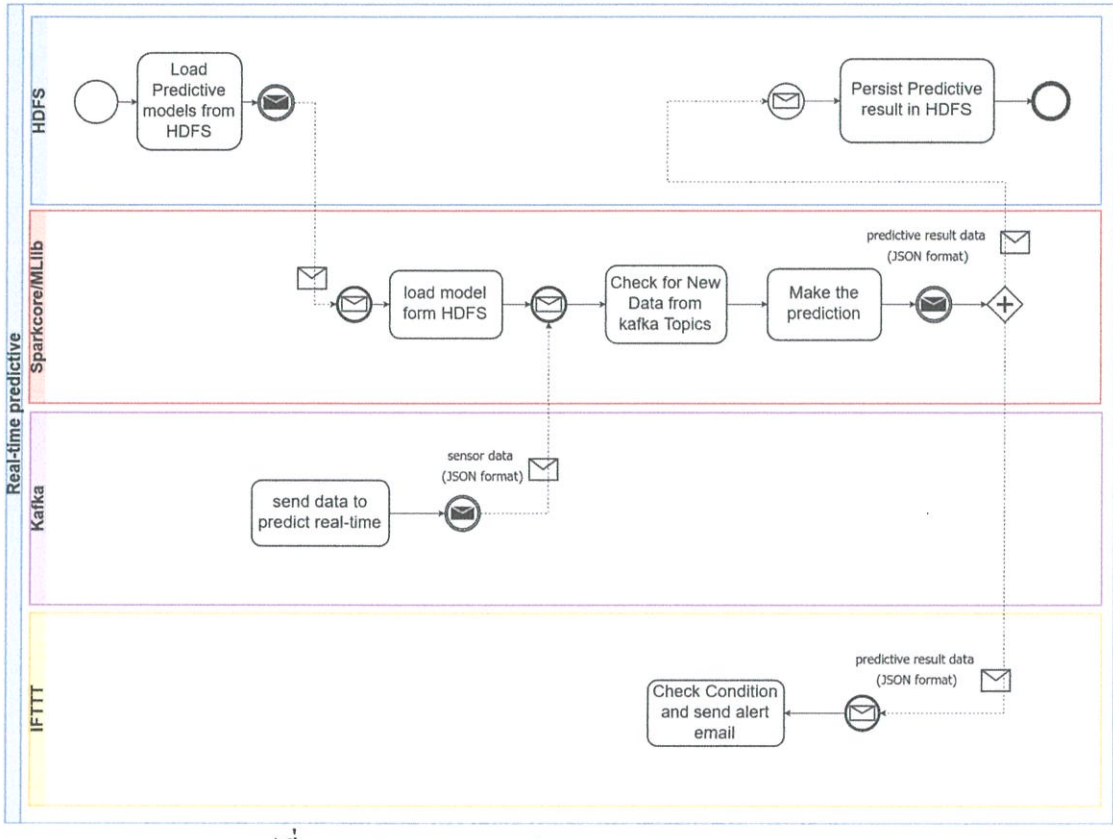
คือการประมวลผล immutable data ที่เป็นรูปแบบ Batch ใช้กับ Data ขนาดใหญ่โดยจะประมวลตามตารางเวลาที่ได้กำหนดไว้โดย นำข้อมูลที่เคยบันทึกไว้จาก sensor ของ model มาสร้าง model machine learning ที่สามารถคาดการณ์การชำรุดของเครื่องจักรได้ล่วงหน้า แล้วนำ model นั้นไปเก็บไว้ยัง HDFS เพื่อใช้วิเคราะห์การชำรุดแบบ Real-time ต่อไป



รูปที่ 3.12 Diagram การทำ Predictive Model Generator

2. การประมวลผลแบบ **Speed Layer**

คือการประมวลผล Data Streams (Real time-Data) โดยจะนำ data ที่ได้จากเครื่องจักรมาประมวลผลจาก model ที่ได้รับจาก batch layer เพื่อคาดการณ์การชำรุดของเครื่องจักรได้ทันก่อนที่มันจะชำรุดขึ้น เมื่อได้ผลลัพธ์แล้วจะไปเก็บไว้ใน HDFS และหากผลลัพธ์ที่ได้มีสถานะเป็นใกล้ชำรุดหรือชำรุดแล้วก็จะส่งแจ้งเตือนไปยังผู้ใช้ผ่าน IFTTT



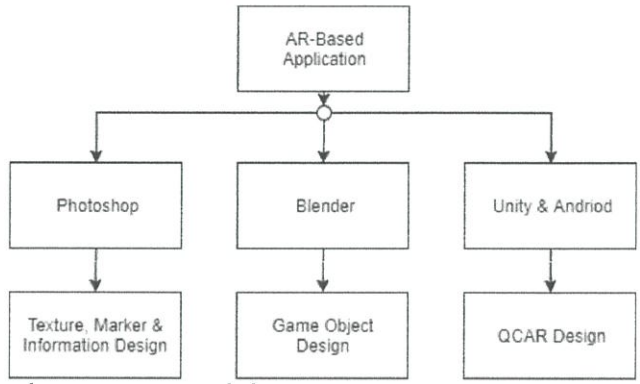
รูปที่ 3.13 Diagram การทำ Real-time prediction

3.2.4 ระบบแสดงผลผ่าน AR application

3.2.4.1 ภาพรวมของ application

ในการพัฒนา application นี้ มีการใช้โปรแกรมเพื่อประกอบการพัฒนาหลักสาม โปรแกรม ได้แก่

1) Photoshop เพื่อการทำ user interface ต่างๆ
2) Blender เพื่อการสร้าง 3D Object
3) Unity เพื่อการพัฒนา application



รูปที่ 3.15 โปรแกรมที่เกี่ยวข้องกับการพัฒนาโปรแกรม

3.2.4.2 **Unity Software**

Unity เป็น game design engine ที่สามารถพัฒนาได้หลากหลาย platform และมีประสิทธิภาพ โดย Unity เหมาะสำหรับการพัฒนา application เกี่ยวกับกราฟิก พัฒนาเกม พัฒนา application ที่มีการโต้ตอบ และยังมีความสามารถในการปรับใช้กับ PC, Mac, Linux, Desktop, Web, iOS, Android, PS4, Xbox One เป็นต้น นอกจากนี้ Unity ยังมีเอกสารเกี่ยวกับโปร เจคและตัวอย่างต่างๆ ให้ผู้ใช้งานศึกษาเพื่อลดเวลา และความซับซ้อนของการพัฒนา

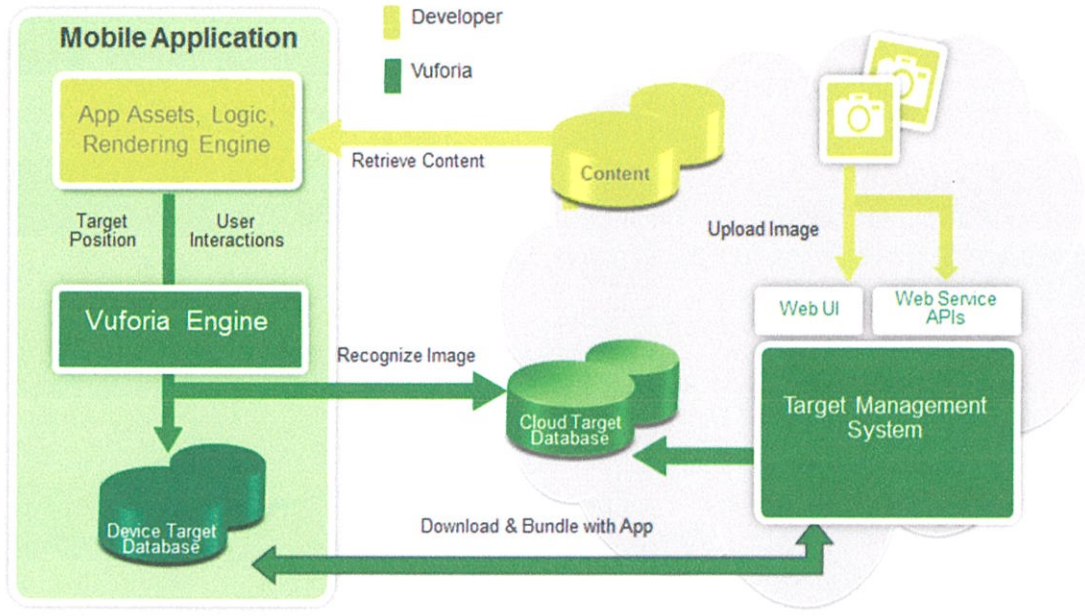
การพัฒนาโปรแกรมด้วย Unity เป็นการใช้งานเพื่อพัฒนาระบบสาธิตการ บำรุงรักษาเชิงการคาดการณ์ ระบบถูกออกแบบให้ใช้สามารถดูสถานะผ่านระบบ AR จึงจำเป็นต้อง ใช้ Vuforia SDK เพื่อเพิ่มความสามารถของ application ทางด้าน AR

3.2.4.3 **Vuforia SDK**

Vuforia SDK (Vuforia Software Development Kit) เป็น SDK ที่ช่วยใน การพัฒนา software ที่ใช้เทคโนโลยี AR ได้ง่ายขึ้น และยังสามารถใช้งานร่วมกับ Unity ได้อีกด้วย

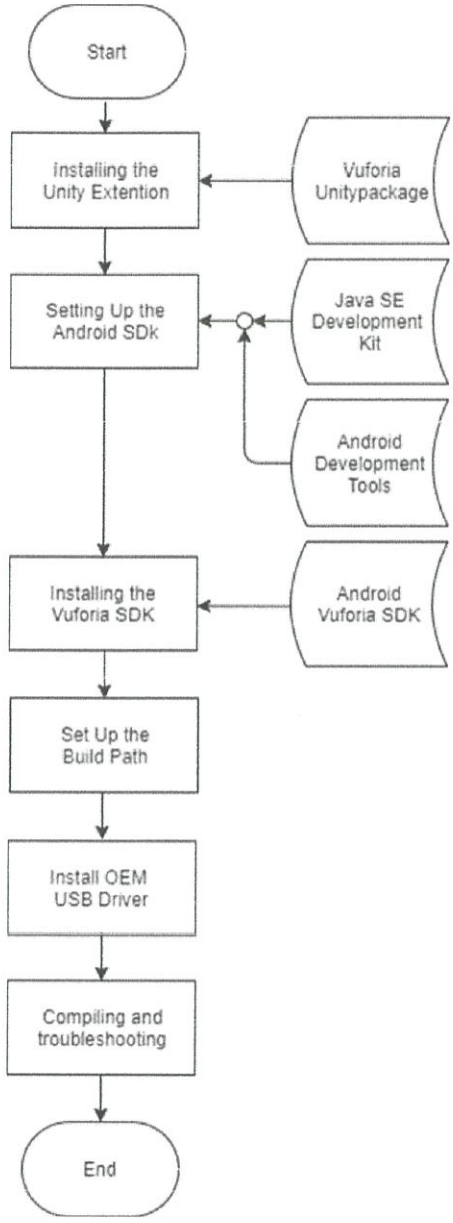
ความสามารถของ Vuforia SDK มีหลายอย่าง เช่น virtual button , multi target , image target , frame marker เป็นต้น โดยในโปรเจคนี้เราจะใช้งาน image target เป็นหลัก

การทำงานของ Vuforia นั้นมีส่วนเกี่ยวข้องสองส่วนได้แก่ หนึ่ง ส่วนของ developer เป็นส่วนที่เกี่ยวข้องกับการกำหนดรูปที่ต้องการตรวจจับ และการทำงานของ application สอง ส่วนของ vuforia เป็นส่วนที่ทำการเก็บข้อมูลรูปที่ต้องการตรวจจับและส่ง package เพื่อมางานการตรวจจับรูปใช้ใน application



รูปที่ 3.16 การทำงานของ Vuforia

ในการใช้งาน Vuforia ร่วมกับ Unity นั้น เราจำเป็นต้องสมัครใช้งาน Vuforia บนเวปไซต์ Qualcomm Vuforia developer portal เพื่อใช้งานฟังก์ชัน target images และการใช้งาน target managerใน web base สำหรับจัดการ target ที่ใช้งาน โดย target image ที่เราจะใช้ในการทำโปรเจคนี้คือ QR code

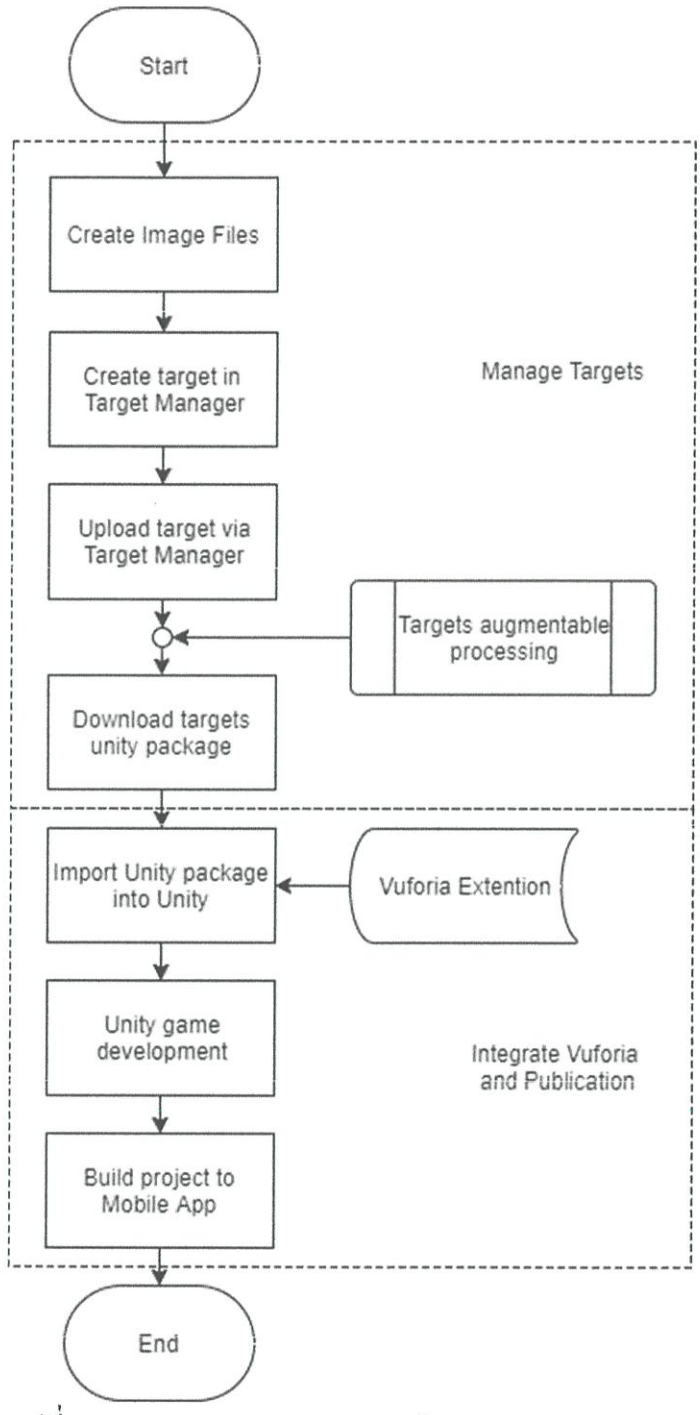


รูปที่ 3.17 Flowchart การใช้งาน Vuforia ร่วมกับ Unity

3.2.4.4 **การใช้งาน unity ร่วมกับ JSON**

การใช้งาน unity ร่วมกับ JSON นั้น unity มีความสามารถที่จะเขียนสคริปต์ลงไปในวัตถุต่างๆใน unity ได้เราจึงสามารถเขียนสคริปต์ที่มีการดึงข้อมูลของ JSON มาเชื่อมโยงกับการใช้งาน Vuforia ที่มีความสามารถตรวจสอบวัตถุเพื่อนำมาระบุเครื่องจักรที่ต้องการรับข้อมูลได้

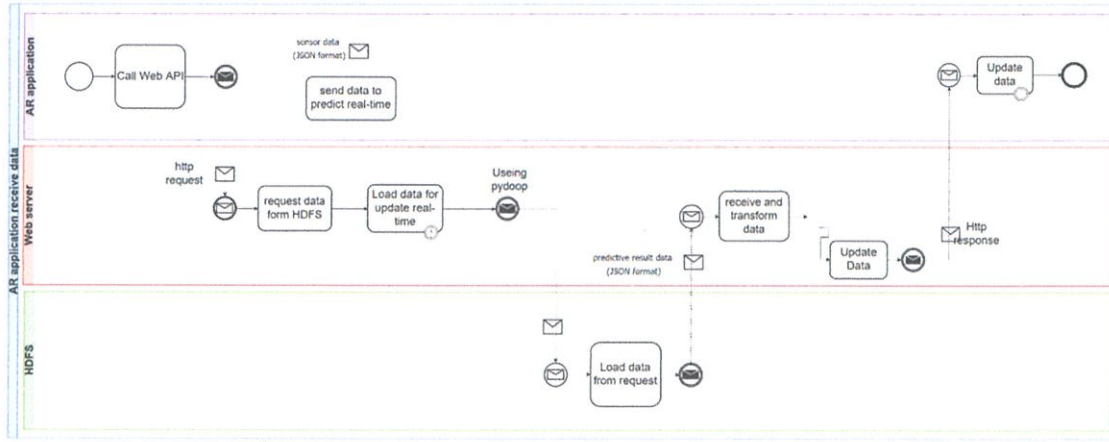
3.2.4.5 ขั้นตอนการพัฒนา application



รูปที่ 3.18 Flowchart กระบวนการพัฒนา application

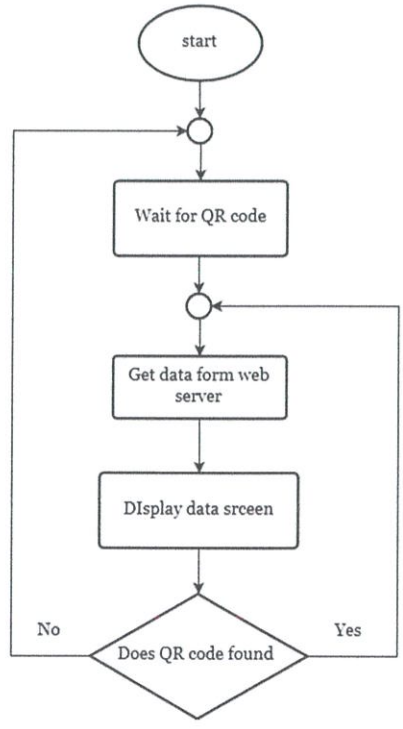
3.2.4.1 การรับข้อมูลจากระบบมาแสดงบน AR application

เนื่องจากข้อมูลที่ต้องนำมาแสดงนั้นต้องดึงมาจากผลลัพธ์การ predict ของ spark streaming ซึ่งถูกเก็บอยู่ใน HDFS ดังนั้น AR จะทำการร้องขอไปยัง web server ถึงข้อมูล sensor และ สถานะของเครื่องจักรที่ต้องการ ดังที่แสดงใน diagram ด้านล่างนี้



รูปที่ 3.19 Diagram การรับข้อมูลจาก Apache Hadoop

3.2.4.1 การทำงานของ AR application

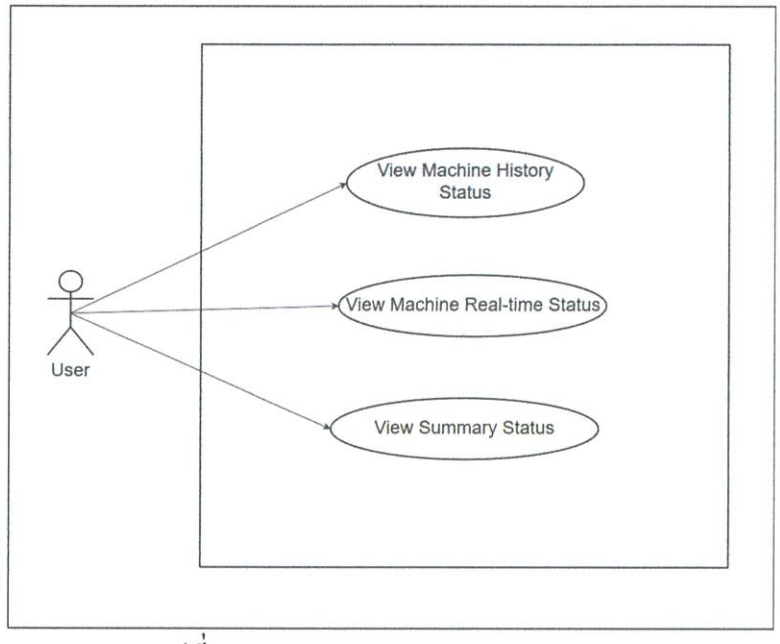


รูปที่ 3.20 Flowchart การทำงานของ AR application

3.2.5 ระบบแสดงผลผ่าน Web Application

3.2.5.1 การออกแบบ web application

โครงการนี้จะมี service web เพื่อใช้ในการเป็น dashboard monitor สิ่งที่เกิดขึ้นในระบบ ทั้งค่าที่ได้จาก sensor, status ของเครื่องจักรจาก machine learning โดยจะคอยนำค่าที่ได้จากการคาดการณ์มาแสดงผลบน web โดยดึงข้อมูลจาก Apache Hadoop



รูปที่ 3.22 Web Use Case Diagram

3.2.5.2 ยูสเคสของเว็บมอนิเตอร์

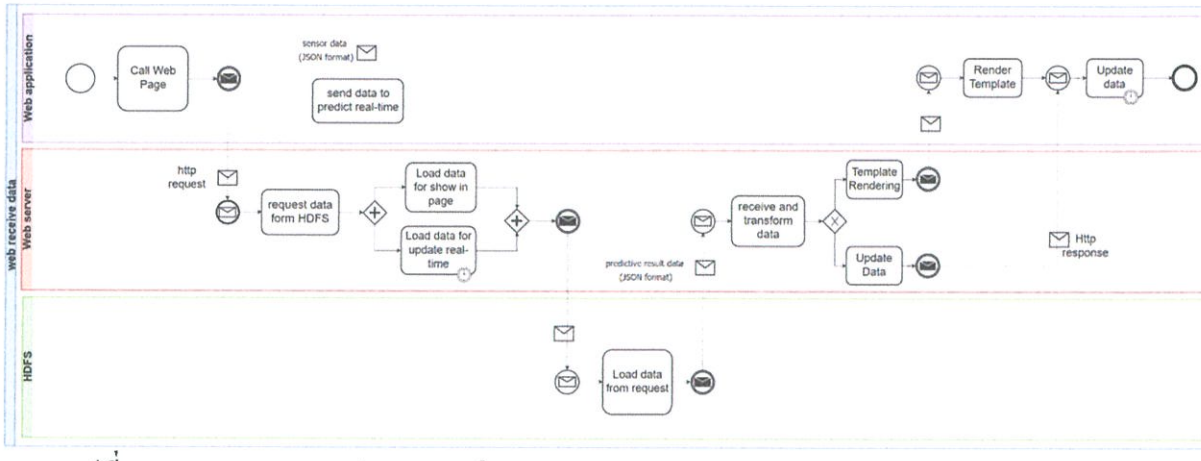
Use Case Title: View Machine History Status	**Use Case ID:** 1
Primary Actor: User	
Stakeholder Actor: -	
Main Flow: เมื่อมีการคาดการณ์สถานะของเครื่องจักร จะมีการบันทึกข้อมูลลง HDFS แล้วหน้าเว็บ Historical Data จะสามารถเลือกดึงข้อมูลในช่วงเวลาที่ user กำหนดมาแสดงเป็นตารางที่แสดงค่าจาก sensor และ status ที่ใช้ machine learning คาดการณ์	
Exceptional Flow ที่ 1: หากไม่มีข้อมูลในช่วงที่ user กำหนด จะแสดงตารางว่างเปล่า	

Use Case Title: View Machine Real-time Status	**Use Case ID: 2**
Primary Actor: User	
Stakeholder Actor: -	
Main Flow: เมื่อมีการคาดการณ์สถานะของเครื่องจักร จะมีการบันทึกข้อมูลลง HDFS แล้วหน้าเว็บ Real-time Monitor จะดึงข้อมูลล่าสุดของแต่ละเครื่องจักรมาแสดงเป็นตารางที่แสดงค่าจาก sensor และ status ที่ใช้ machine learning คาดการณ์ และแสดงสถานะล่าสุดของเครื่องจักรตัวต่างๆในระบบด้วย	
Exceptional Flow ที่ 1: กรณีที่ไม่มีข้อมูลล่าสุดจะแสดงเป็นตารางว่าง	

Use Case Title: View Machine Summary Status	**Use Case ID: 3**
Primary Actor: User	
Stakeholder Actor: -	
Main Flow: เมื่อมีการคาดการณ์สถานะของเครื่องจักร จะมีการบันทึกข้อมูลลง HDFS แล้วหน้าเว็บ Dashboard จะนำข้อมูลที่ได้เก็บไว้มาแสดงเป็น chart รายวัน และสถานะโดยรวมของเครื่องจักรในระบบ	
Exceptional Flow ที่ 1: ในกรณีที่เครื่องจักรไม่ได้เปิดให้ทำงาน จะแสดงสถานะเป็น turn off และไม่ plot chart ในช่วงนั้น	

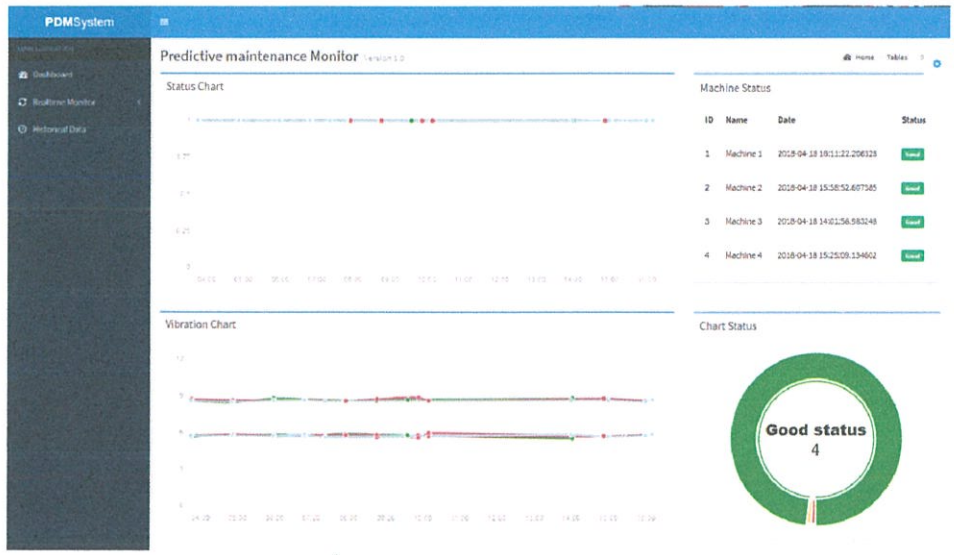
3.2.5.3 การรับข้อมูลจากระบบมาแสดงบนเว็บ

เนื่องจากข้อมูลที่ต้องนำมาแสดงนั้นต้องดึงมาจากผลลัพธ์การ predict ของ spark streaming ซึ่งถูกเก็บอยู่ใน HDFS ดังนั้น webserver (Django)จะใช้ library pydoop ซึ่งเป็นภาษา python ในการดึงข้อมูลจาก HDFS มาแสดงโดยจะมีการดึงข้อมูลในตอนเรียก page ครั้งแรก และ ดึงข้อมูล real time ทุกๆ 10 วินาที ดังที่แสดงใน diagram ด้านล่าง

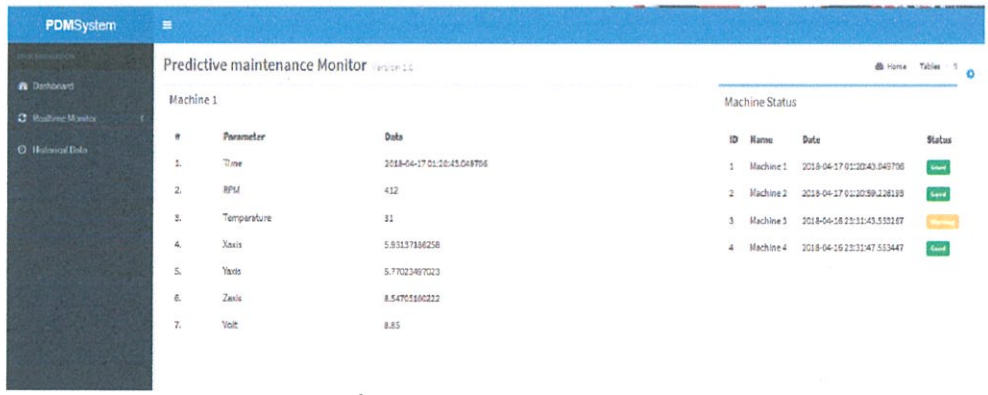


รูปที่ 3.23 Diagram การทำงานระหว่าง web application, web server และ HDFS

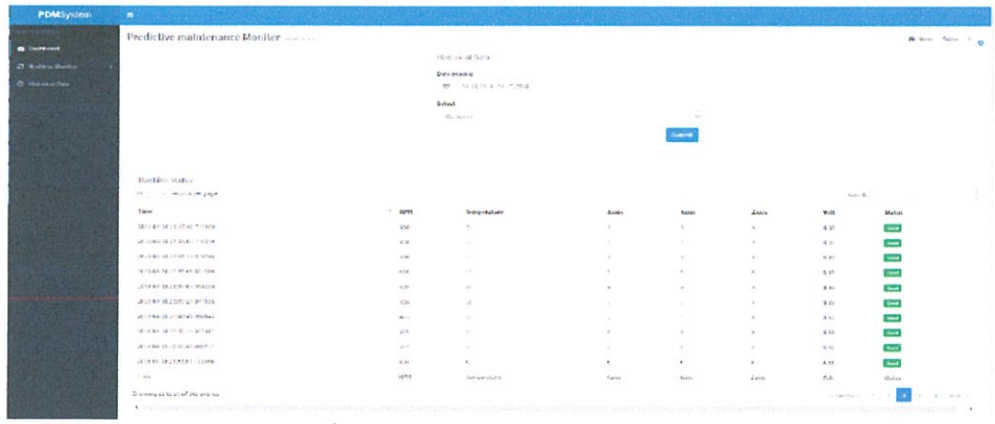
3.2.5.4 UI Design



รูปที่ 3.24 Dashboard page



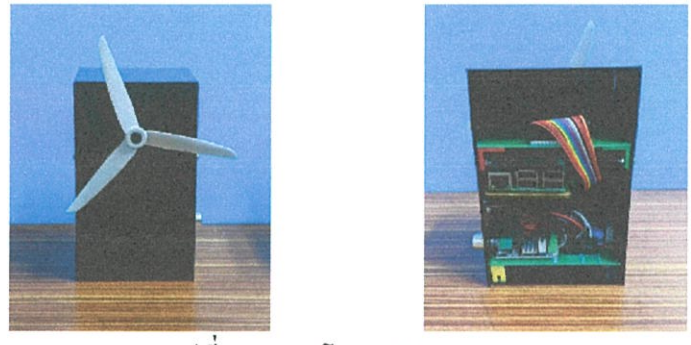
รูปที่ 3.25 Real-time page



รูปที่ 3.26 Historical data page

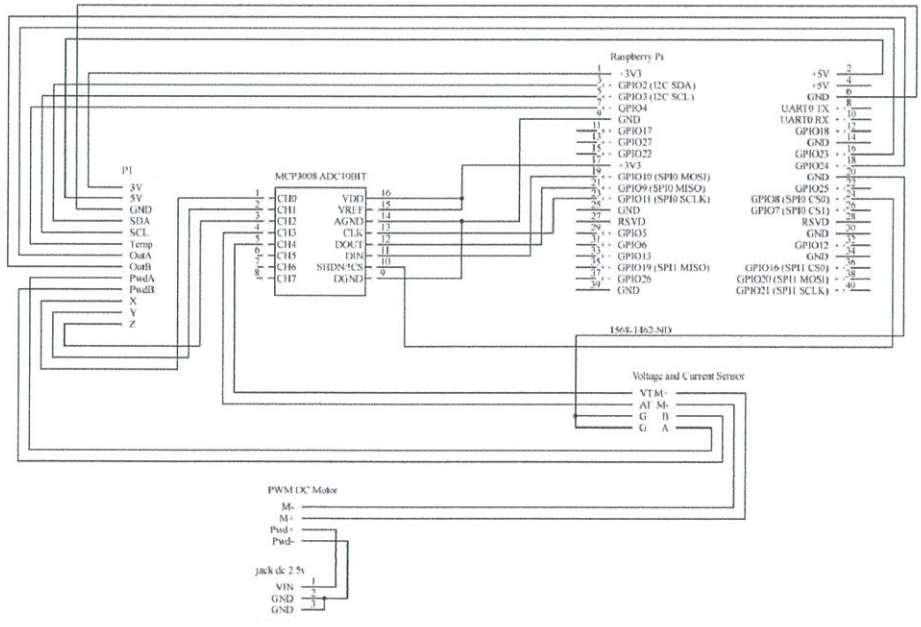
3.3 Model ทดสอบ

เนื่องจากตัวโครงการต้องการที่จะจำลองระบบในการทำ predictive maintenance ขึ้นมาจึงได้สร้างชุดโมเดลจำลองสำหรับใช้ทำงานแทนเครื่องจักรที่ใช้ในการเก็บข้อมูลเพื่อทำ machine learning และจำลองสถานการณ์ต่างๆที่จำเป็นต่อการทดสอบการทำงานของระบบโดยจะใช้ชุด gearbox ต่อเข้ากับ sensor ต่างๆแล้วอ่านข้อมูลด้วย raspberry pi โดยจะประกอบกันเป็น 1 ชุดโมเดลจำลองมีลักษณะดังนี้



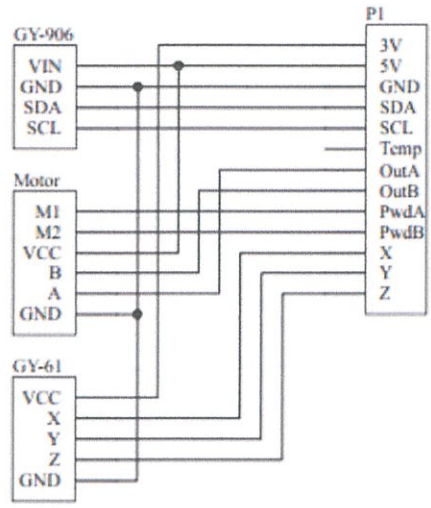
รูปที่ 3.28 ชุดโมเดลจำลอง

โดยภายในจะประกอบไปด้วย 2 วงจรที่ทำหน้าที่ในการเชื่อมต่อการทำงานระหว่าง raspberry pi, sensor และ motor โดยวงจรที่วงจรที่1 จะทำหน้าที่เชื่อมต่อ raspberry pi เข้ากับตัว ADC ที่ใช้แปลงข้อมูลที่ได้จาก voltage and current sensor ซึ่งรับค่าจาก PWM DC motor ที่ควบคุมการจ่ายไฟให้ motor ดังแผนผังวงจรด้านล่าง



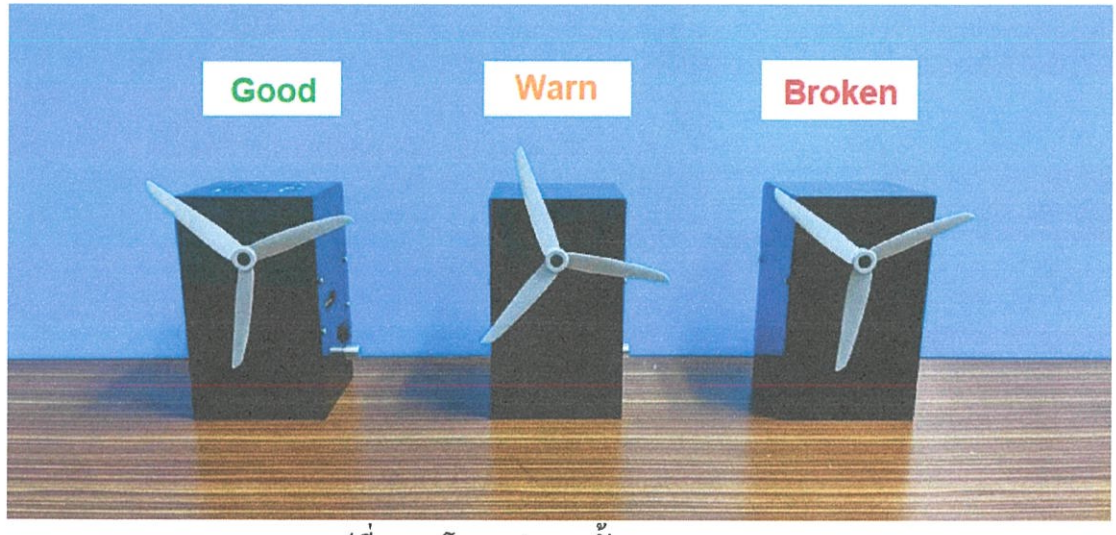
รูปที่ 3.29 แผนผังวงจรที่ 1

วงจรที่ 2 จะทำหน้าที่จ่ายไฟให้กับ motor ที่เป็นตัวขับเคลื่อน load ซึ่งในโมเดลจำลองนี้ใช้เป็นใบพัดซึ่งติดอยู่กับชุด gearbox นอกจากนั้นวงจรที่ 1 จะรับค่าจาก sensor ตรวจจับอุณหภูมิ และ vibration ซึ่งจะติดอยู่กับตัว motor ดังแผนผังวงจรด้านล่าง



รูปที่ 3.30 แผนผังวงจรที่ 2

โดยในการทดลองจะทดลองเปิดใช้งานโมเดลจำลองจนชำรุด และใกล้ชำรุดเพื่อนำมาทดสอบระบบต่อไปโดยจะมี 3 โมเดลจำลองซึ่งจะมีสถานะ good คือทำงานปกติ warn คือใกล้ชำรุด และ broken สำหรับโมเดลที่ไม่สามารถทำงานได้แล้วโดยอ้างอิงสถานะจากตัว motor ที่ทำงานขับเคลื่อน gearbox



รูปที่ 3.31 โมเดลจำลองทั้ง 3 สถานะ

บทที่ 4

การทดลอง

หลักจากที่ได้ออกแบบ predictive maintenance platform แล้ว ก็ได้เริ่มทำการทดลองในระบบหลักๆของโครงงาน โดยประกอบไปด้วย 3 การทดลองคือ การทดลองเกี่ยวกับการรับส่งข้อมูล, การทดลองเกี่ยวกับ machine learning และ การทดลองเกี่ยวกับการแสดงผล

4.1 การทดลองเกี่ยวกับการรับส่งข้อมูล

การทดลองนี้เป็นการแสดงถึงความสามารถของแพลตฟอร์ม ว่าสามารถเชื่อมต่อและ ส่งข้อมูลกัน ผ่านเทคโนโลยีต่างๆ ซึ่งประกอบไปด้วย controller, Apache Kafka, Apache Hadoop เพื่อให้ระบบอื่นๆในแพลตฟอร์มสามารถนำข้อมูลไปใช้ต่อได้ โดยมีการทดลองดังนี้

4.1.1 การทดสอบรับส่งข้อมูลจาก controller ไปยัง Apache Kafka

4.1.1.1 วัตถุประสงค์

ทดสอบการส่งข้อมูลจาก controller ไปยัง Apache Kafka เพื่อทดสอบว่าสามารถเชื่อมต่อระบบเพื่อรับค่าจาก gearbox ได้

4.1.1.2 วิธีการทดลอง

ทดลองส่งข้อมูลโดยใช้ภาษา Python ในการส่งข้อมูลจาก controller ไปยัง Apache Kafka

4.1.1.3 ผลการทดลอง

ภาพด้างล่างนี้เป็นรูปของชุดข้อมูลที่ถูกส่งไปยัง Apache Kafka โดยภายในจะถูกบรรจุไปด้วยค่าที่ตรวจวัดได้จาก sensor ที่เชื่อมอยู่กับ controller โดยจะประกอบไปด้วย

1) เวลาที่ส่ง
2) ค่า RPM
3) ค่าอุณหภูมิ
4) ค่าตำแหน่งแกน x ,y และ z ที่ใช้ในการคำนวณแรงสั่น
5) ค่า voltage

แสดงให้เห็นว่าสามารถส่งข้อมูลที่อ่านได้จาก model จำลองไปยัง Apache Kafka ได้

{"Time": "2018-04-29 14:57:57.785971", "RPM": 401, "Temperature": 31, "Xaxis": [513, 515, 512,..., 503], "Yaxis": [503, 496, 509,...496], "Zaxis": [495, 503, 517,... 501], "Volt": 8.85}

รูปที่ 4.1 ข้อมูลที่ส่งไปยัง Apache Kafka

```
 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

546 packages can be updated.
282 updates are security updates.

Last login: Wed Nov 22 22:27:46 2017 from 10.252.39.136
kafka@kafkaserver:~/Project$ cd cn
-bash: cd: cn: No such file or directory
kafka@kafkaserver:~/Project$ cd confluent-3.3.1/
kafka@kafkaserver:~/Project/confluent-3.3.1$ ./bin/kafka-console-consumer --boot
strap-server localhost:9092 --topic test --from-beginning
{ "Time":"2017-11-22 22:29:37.622183","Speed":"349.00","Temperature":"25.45","Vi
bration":"1020.00","Volt":"3.29","Amp":"0.14" }
{ "Time":"2017-11-22 22:29:43.626688","Speed":"343.00","Temperature":"25.49","Vi
bration":"1050.00","Volt":"3.24","Amp":"0.11" }
{ "Time":"2017-11-22 22:29:49.632799","Speed":"350.00","Temperature":"25.16","Vi
bration":"1019.00","Volt":"3.25","Amp":"0.11" }
{ "Time":"2017-11-22 22:29:55.638678","Speed":"347.00","Temperature":"25.81","Vi
bration":"1001.00","Volt":"3.29","Amp":"0.15" }
{ "Time":"2017-11-22 22:30:01.647153","Speed":"345.00","Temperature":"25.14","Vi
bration":"1009.00","Volt":"3.30","Amp":"0.12" }
```

รูปที่ 4.2 ข้อมูลที่เข้ามายัง Apache Kafka

4.1.2 การทดลองส่งข้อมูลจาก Apache Kafka ไปยัง Apache Hadoop

4.1.2.1 วัตถุประสงค์

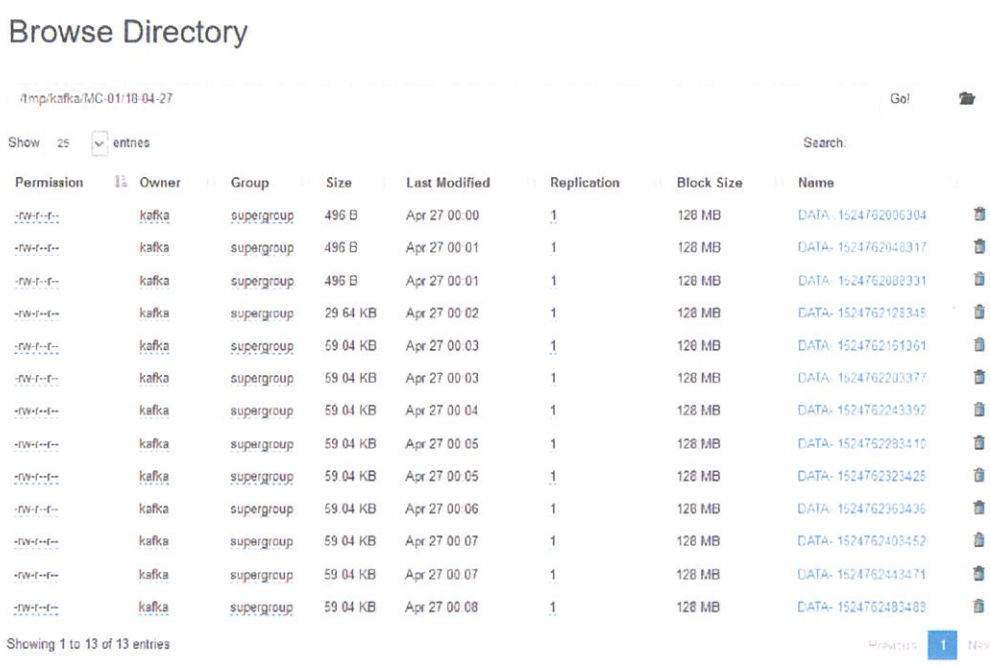
ทดสอบการส่งข้อมูลจาก Apache Kafka ไปยัง Apache Hadoop เพื่อทดสอบว่าสามารถเชื่อมต่อเข้ากับตัวเก็บข้อมูลได้

4.1.2.2 วิธีการทดลอง

เมื่อข้อมูลถูกส่งมายัง Apache Kafka แล้ว Apache Kafka จะทำการเก็บข้อมูลจาก topic นั้นๆ มาเก็บไว้ยัง HDFS

4.1.2.3 ผลการทดลอง

ภาพด้างล่างนี้เป็นรูปของชุดข้อมูลที่ถูกส่งไปยัง HDFS แล้วถูกบันทึกเป็นไฟล์ .txt ไว้



รูปที่ 4.3 Apache Kafka ส่งข้อมูลมาเก็บไว้ที่ HDFS

โดยจะเห็นได้ว่าจากการทดลองที่ผ่านมาตัวแพลตฟอร์มที่ได้ออกแบบไว้นั้นสามารถเชื่อมต่อและรับส่งข้อมูลกันได้จริงโดยทดลองมาจนถึงจุดที่สามารถนำข้อมูลจาก model gearbox มาใช้งานในแพลตฟอร์มได้

4.2 การทดลองเกี่ยวกับ machine learning

เป็นการทดลองใช้งาน machine learning ในการคาดการณ์การชำรุดของเครื่องจักร ด้วย algorithm Neural Network ผ่าน MLlib โดยใช้ Spark Core ในการประมวลผล แล้วนำมาประมวลผลแบบ real-time ต่อด้วย Spark Streaming เพื่อทดสอบว่าแพลตฟอร์มนี้สามารถคาดการณ์การชำรุดของเครื่องจักรได้จริง

4.2.1 การทดสอบความแม่นยำของการสร้าง model machine learning

ในการทำ predictive maintenance นั้นเราได้ใช้ machine learning สร้าง model สำหรับทำการ classification เพื่อจัดกลุ่มสถานะของเครื่องจักร โดยได้แบ่ง label สถานะของเครื่องจักรออกเป็น 3 ชนิด คือ

1) สถานะปกติ(good)
2) สถานะใกล้จะชำรุด(warning)
3) สถานะชำรุดแล้ว(broken)

โดยข้อมูลที่ได้นำมาใช้นั้นจะเป็น model จากชุด gearbox ที่ได้เก็บค่าไว้ตั้งแต่เริ่มใช้งานจนชุด gearbox ชำรุดไม่สามารถทำงานได้เพื่อใช้ข้อมูลในการทำการทดลอง

4.2.1.1 วัตถุประสงค์การทดลอง

เพื่อสังเกตลักษณะของข้อมูลที่ได้เก็บจากชุด gearbox ว่าสามารถนำไปทำการสร้าง predictive maintenance model สำหรับ classification สถานะของเครื่องจักรได้

4.2.1.2 วิธีการทดลอง

นำข้อมูลทั้งหมดที่ได้เก็บไว้จากชุด gearbox ตั้งแต่ทำงานได้ปกติจนชำรุดไม่สามารถทำงานได้จำนวน 8171 ชุดใช้เวลา 9 ชั่วโมง โดยใน 1 ชุดข้อมูลจะประกอบไปด้วย

1) RPM (Revolutions per minute)
2) time
3) voltage
4) temperature
5) ค่า SD พิกัดแกน X
6) ค่า SD พิกัดแกน Y
7) ค่า SD พิกัดแกน Z

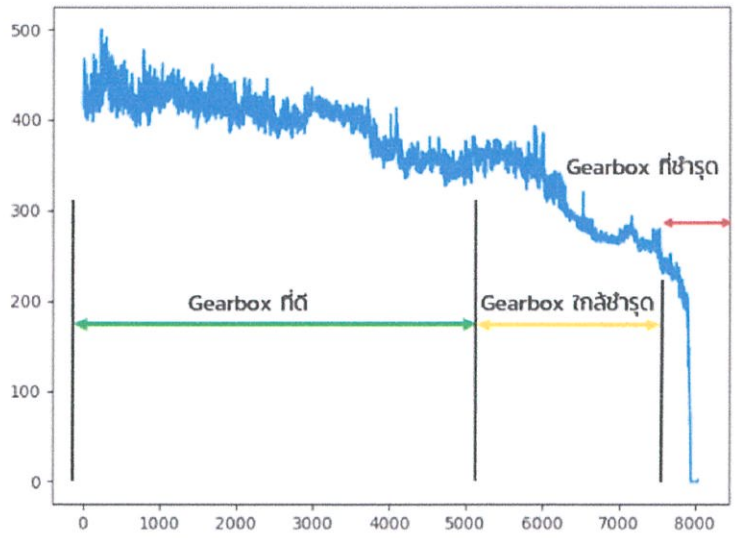
ซึ่งจะนำข้อมูลทั้งหมดมา label เป็นสถานะ 3 สถานะ

1) ตั้งแต่เริ่มทดลองถึงก่อนชุด gearbox ชำรุด 2 ชั่วโมง 30 นาทีเป็นช่วงที่ชุด gearbox สามารถทำงานได้ปกติ

2) ช่วง 2 ชั่วโมง 30 นาทีก่อนเครื่องจักรชำรุดเป็นช่วงที่ ชุด gearbox มีสถานะใกล้ชำรุด

3) ช่วงที่ชุด gearbox หยุดทำงานมีสถานะเป็นชำรุด

ซึ่งอ้างอิงจากลักษณะข้อมูล RPM ที่เคยเก็บได้ซึ่งมีลักษณะดังนี้



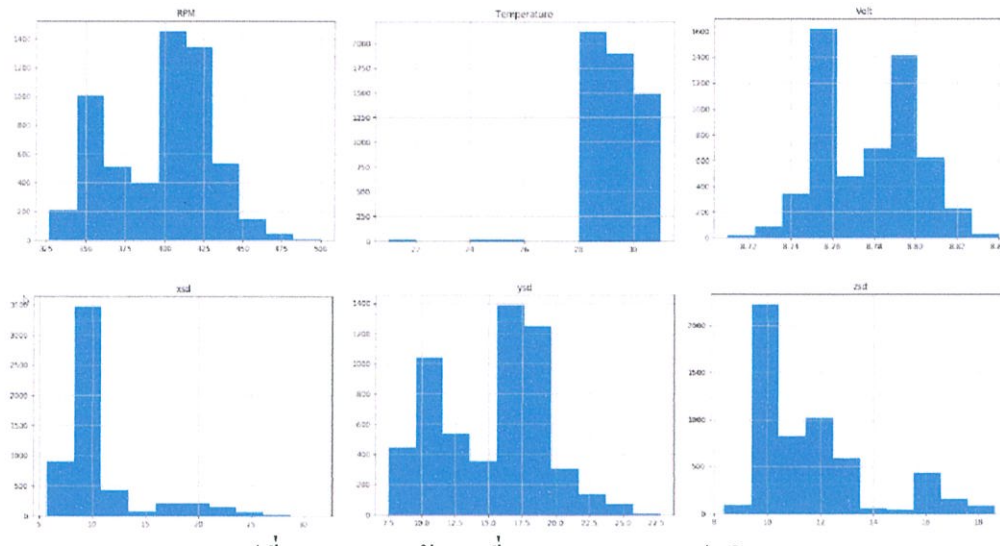
รูปที่ 4.4 ลักษะการแบ่งสถานะด้วยค่า RPM

โดยจะ plot chart ของ ข้อมูลจาก sensor ต่างๆมาทำการวิเคราะห์ว่าสามารถใช้ในการ classification สถานะต่างๆได้

4.2.1.3 **ผลการทดลอง**

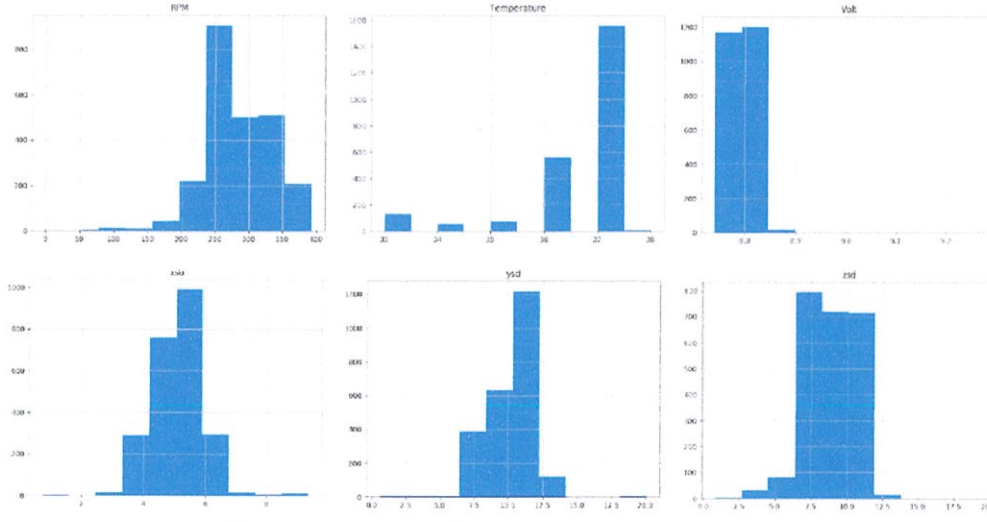
จากการทดลองหลังจาก clean ข้อมูลแล้วนำมาแบ่งตามช่วงที่ได้กำหนดไว้มา plot เป็น histogram ได้ผลดังนี้

1) ลักษณะ chart ที่ได้จากการ plot ข้อมูลที่ label ว่าชุด gearbox ทำงานได้ปกติ



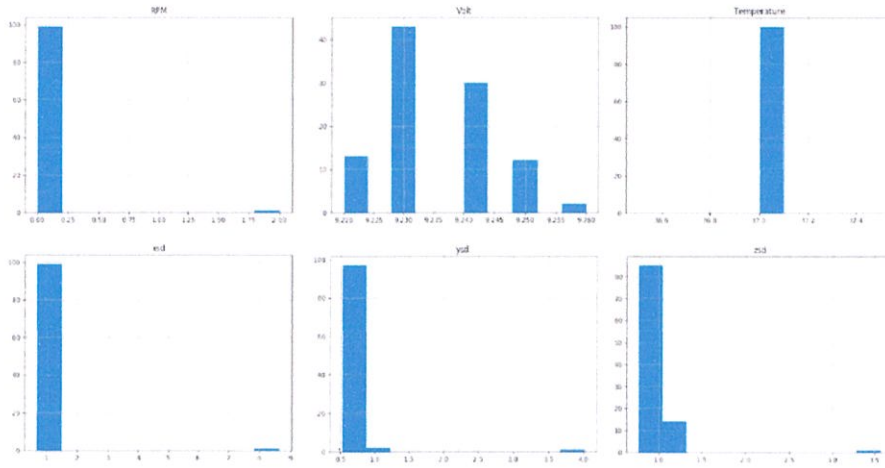
รูปที่ 4.5 Chart ข้อมูลที่ gearbox ทำงานปกติ

2) ลักษณะ chart ที่ได้จากการ plot ข้อมูลที่ label ว่าชุด gearbox มีสถานะใกล้ชำรุด



รูปที่ 4.6 Chart ข้อมูลที่ gearbox มีสถานะใกล้ชำรุด

3) ลักษณะ chart ที่ได้จากการ plot ข้อมูลที่ label ว่าชุด gearbox มีสถานะชำรุด



รูปที่ 4.7 Chart ข้อมูลที่ gearbox มีสถานะชำรุด

จาก histogram chart ที่ได้แสดงให้เห็นว่าค่า RPM และค่าตำแหน่ง x, y และ z มีความแตกต่างกันของข้อมูลอย่างชัดเจนจึงสรุปได้ว่าสามารถนำค่าข้อมูลที่แบ่งการ label แบบที่ได้กล่าวมาสามารถนำไปใช้สร้าง model ได้

4.2.2 ทดสอบความแม่นยำของแต่ละ machine learning algorithm ในการทำ predictive maintenance

4.2.2.1 **วัตถุประสงค์**

ในการสร้าง model machine learning เพื่อทำ classification สามารถเลือกได้หลาย algorithm การทดลองนี้จึงตั้งใจที่จะทดสอบความแม่นยำในแต่ละ algorithm ในการนำมาประยุกต์ใช้กับข้อมูลที่มีในการนำมาทำ predictive maintenance

4.2.2.2 **วิธีการทดลอง**

ทดลองนำข้อมูลที่มีมาแบ่งเป็น 3 ส่วนตามสถานะที่ได้กำหนดไว้ในการทดลอง 2.2.1 โดยนำมาทดลองสร้าง model สำหรับ classification โดยใช้ algorithm พื้นฐาน ซึ่งที่เลือกมาได้แก่

1) Decision tree
2) Random forest
3) Neural network

ซึ่งทำการประมวลผลโดยใช้ spark core ผ่าน library ของ MLlib โดยแบ่งอัตราส่วน train 70 เปอร์เซ็นต์ test 30 เปอร์เซ็นต์

4.2.2.3 ผลการทดลอง

1) Decision Tree จากการทดลองสร้าง classification model ด้วย decision tree classifier ของ MLlibโดยปรับ categories ใน tree เป็นไม่เกิน 4 ได้ผลว่า ได้ model decision tree depth เป็น 3 มีทั้งหมด 7 node โดยได้ accuracy อยู่ที่ 93.80
2) Random forest จากการทดลองสร้าง classification model ด้วย random forest classifier ของ MLlibโดยปรับ categories ใน tree เป็นไม่เกิน 4 และใช้ tree ใน model เป็น 10 ได้ accuracy อยู่ที่ 93.80
3) Neural network จากการทดลองสร้าง classification model ด้วย neural network หรือ multilayer perceptron classifier ของ MLlibโดยปรับ hidden layer 2 ชั้น เป็น 5และ 4 ตามลำดับโดยให้ block size เป็น 128 และ max Iterative อยู่ที่ 300 ได้ accuracy อยู่ที่ 95.8

จากการทดลองสร้าง classification model เพื่อทำ predictive maintenance โดยแบ่งสถานะของเครื่องจักรที่จะทำการ predict ออกเป็น 3 สถานะ คือ สถานะปกติ, สถานะใกล้จะชำรุด และ สถานะชำรุดแล้ว ได้ algorithm ที่สามารถใช้งานได้ดีมีประสิทธิภาพและปรับแก้ให้เหมาะกับข้อมูลได้ที่สุดคือ neural network โดยการทดลองนี้จะเปรียบเทียบจากการใช้ MLlib ซึ่งเป็น library สำหรับ machine learning ที่เหมาะสมกับการทำงานแบบ Big Data เท่านั้น

4.2.3 การทดลองปรับจำนวน node ของ hidden layer ของ neural network

4.2.3.1 วัตถุประสงค์

เป็นการทดลองเพื่อหาจำนวน Node ใน Hidden Layer ที่ดีที่สุดที่เหมาะสมกับการทำ classification สำหรับข้อมูลที่มี

4.2.3.2 วิธีการทดลอง

การทดสอบการ generate model ด้วย Neural network Algorithm โดยทดลองนำข้อมูลที่ได้ มาทดลองสร้าง classification model โดยการนำไฟล์ batch ที่ได้มาทำการแบ่งเป็นตัว train 70 เปอร์เซ็นต์ ตัว test 30 เปอร์เซ็นต์ แล้วทดลองปรับ hidden layer ในแบบต่างๆ เพื่อสามารถปรับเข้ากับข้อมูลในรูปแบบต่างๆได้ โดยจะมี 6 input แทนค่าที่ได้รับจาก sensor ต่างๆ และ 3 output เพื่อบอกว่าเครื่องจักรกำลังอยู่ในสถานะใด โดยมี hidden layer 2 ชั้นผลการทดลอง

ตารางที่ 4.1 การเปรียบเทียบความแม่นยำของการสร้าง model machine learning

ครั้งที่	จำนวน node hidden layer ชั้นที่ 1	จำนวน node hidden layer ชั้นที่ 2	Accuracy
1	2	1	0.521
2	3	2	0.835
3	3	1	0.521
4	4	3	0.521
5	4	2	0.835
6	4	1	0.834
7	5	4	0.525
8	5	3	0.835
9	5	2	0.521
10	5	1	0.522
11	6	5	0.934
12	6	4	0.958
13	6	3	0.835
14	6	2	0.834
15	6	1	0.835

จะเห็นได้ว่าจากจำนวน input output และ data ที่มี จะได้ค่าจำนวน node ของ hidden layer ที่เหมาะสมของระบบนี้อยู่ที่จำนวน node ชั้นแรกประมาน 6 และ จำนวน node ชั้นที่ 2 ประมาณ 5-4 โดยได้ความแม่นยำอยู่ที่ประมาณ 95 เปอร์เซ็นต์สามารถนำไปใช้เป็น model machine learning ที่ใช้ในการประมวลผลต่อได้

4.2.4 การทดสอบความแม่นยำของการทำ machine learning แบบ real-time

4.2.4.1 วัตถุประสงค์

ในการทำงานของระบบจะต้องมีการ predict สถานะของชุด gearbox แบบ real-time ด้วย spark streaming การทดลองนี้จึงต้องการทดสอบส่งข้อมูลที่ได้จาก sensor เข้ามาเพื่อทำการทดสอบ การ classification ข้อมูลเพื่อ predict สถานะและทดสอบเกี่ยวกับการ overfit

4.2.4.2 วิธีการทดลอง

โดยทดลองใช้ spark streaming load model ที่เก็บไว้ขึ้นมาแล้วส่งค่าเข้าไปเป็น data frame โดยใช้ข้อมูลจากชุด gearbox ที่ model ไม่ถูกใช้ในการ train โดยส่งไปจำนวน 30 ครั้ง

4.2.4.3 ผลการทดลอง

ตารางที่ 4.2 การเปรียบเทียบการทดสอบ real-time predict ในแต่ละสถานะ

ข้อมูลจากชุด gearbox ที่สถานะ	จำนวนที่คาดการณ์แบบ real-time ได้ถูก	accuracy
สถานะปกติ	10	1.0
สถานะใกล้ชำรุด	10	1.0
สถานะชำรุด	10	1.0

โดยจากในรูปเมื่อทำการทดลองได้ 30 ครั้งสามารถทำนายได้อย่างแม่นยำถึง 100 เปอร์เซ็นต์จึงสามารถสรุปได้ว่าการทดลองเกี่ยวกับ machine learning นั้นสามารถใช้กับโครงงานนี้ได้จริง

จากการทดลองเกี่ยวกับ machine learning พบว่าแพลตฟอร์มนี้สามารถลองรับการประมวลผลเพื่อคาดการณ์การชำรุดของเครื่องจักรได้จริงและ model ที่ได้ไม่ได้มีปัญหาการ overfitเนื่องจากทดลองส่งข้อมูลจาก ชุด gearbox ตัวอื่นแล้วยังสามารถทำงานได้โดยได้ accuracy ที่ดี

4.3 การทดลองเกี่ยวกับการแสดงผล

เป็นการทดลองการเขียน AR application ผ่าน Vuforia โดยใช้ Unity 3D ในการพัฒนาโปรแกรม โดยการนำข้อมูลจาก web server มาแสดงผลเป็น AR เมื่อมีการตรวจพบภาพที่ใช้ในการระบุเครื่องจักร โดยในการทดลองนี้ใช้เป็น QR code โดยมีการทดลองดังนี้

4.3.1 ทดลองใช้ Vuforia และ Unity ในการพัฒนา application

4.3.1.1 **วัตถุประสงค์**

ทดสอบการแสดง AR โดยการใช้รูปภาพที่ได้กำหนดขึ้นมา ในการระบุเป้าหมาย โดยการใช้งาน Vuforia ในโปรแกรม Unity

4.3.1.2 **วิธีการทดลอง**

ทดลองเขียน android application ด้วยโปรแกรม Unity โดยใช้ Vuforia ช่วยในการระบุเป้าหมาย และแสดงผล AR

4.3.1.3 ผลการทดลอง

ภาพด้านล่างเป็นการแสดงผลการระบุเป้าหมายที่ได้กำหนดขึ้น และแสดงผลเป็น AR ด้วย 3D object

รูปที่ 4.8 การทดลองแสดงผลจาก QR code

4.3.2 ทดสอบการรับข้อมูลจาก web server

4.3.2.1 วัตถุประสงค์

ทดสอบการรับข้อมูลจาก web server เพื่อนำมาแสดงผล ใน android application โดยใช้ Unity

4.3.2.2 วิธีการทดลอง

ทดลองพัฒนา android application ด้วย Unity โดยใช้ JSON ในการส่งข้อมูลจาก web server มายัง android application เพื่อนำมาแสดงผล

4.3.2.3 ผลการทดลอง

ภาพด้านล่างเป็นการแสดงผลข้อมูล JSON ที่ได้รับมาจาก web server

รูปที่ 4.9 การทดลองรับข้อมูล JSON

จากการทดลองพบว่าแพลตฟอร์มสามารถลองรับ application ที่ใช้แสดงผลข้อมูลด้วย AR ได้

4.3.3 ทดสอบรับข้อมูลจาก web server ที่มีการระบุเป้าหมายด้วย QR code และแสดงข้อมูลผ่าน AR application

4.3.3.1 **วัตถุประสงค์การทดลอง**

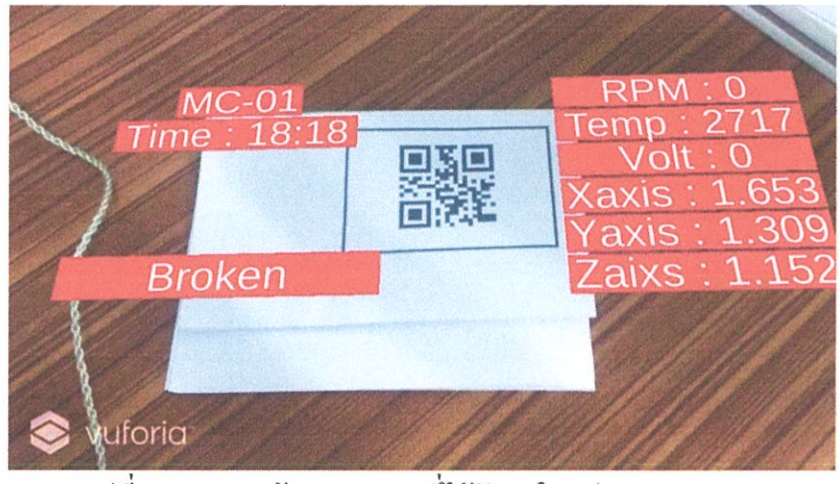
ทดสอบการทำงานร่วมกันระหว่างการระบุเป้าหมายของ Vuforia กับการรับข้อมูล JSON ของเครื่องจักรแต่ละเครื่อง

4.3.3.2 **วิธีการทดลอง**

การทดลองจะทำการ ส่ง get เพื่อรับค่า JSON จาก web sever เมื่อมีการตรวจพบ QR code ของเครื่องจักรใดๆ พร้อมทั้งจะส่ง get ในทุกๆ 5 วินาที เพื่อที่จะอัพเดจข้อมูลให้เป็นข้อมูลปัจจุบัน

4.3.3.3 **ผลการทดลอง**

ภาพด้านล่างเป็นการแสดงผลข้อมูล JSON ที่ได้รับมาจาก web server ในรูปแบบของ AR



รูปที่ 4.10 แสดงข้อมูล JSON ที่ได้รับมาในรูปแบบ AR

จากการทดลองสามารถที่จะแสดงข้อมูล JSON ในรูปแบบของ AR ได้

บทที่ 5
บทสรุปและข้อเสนอแนะ

5.1 บทสรุป

ในปัจจุบันเครื่องจักรนั้นมีส่วนสำคัญอย่างมากกับอุตสาหกรรมการผลิต ซึ่งหากเกิดการชำรุดขึ้นกับเครื่องจักรนั้น จะส่งผลกระทบอย่างมากกับการผลิตทั้งค่าเสียหายในการซ่อมบำรุงและค่าเสียหายจาก operation machine downtime โครงการนี้จึงพยายามที่จะจัดทำแพลตฟอร์มสำหรับคาดการณ์การชำรุดของเครื่องจักรล่วงหน้า (predictive maintenance platform) โดยอาศัยปัจจัยเสี่ยงต่างๆ เช่น ระดับสัญญาณความสั่นสะเทือน (Vibration analysis), คลื่นความร้อน(Temperature monitoring) เพื่อให้สามารถจัดการซ่อมบำรุงเครื่องจักรให้เกิดความเสียหายกับการผลิตน้อยที่สุด

โครงงานนี้จึงต้องการพัฒนาแพลตฟอร์มสำหรับทำ predictive maintenance โดยจำลองระบบจาก gearbox ซึ่งเป็นปัจจัยหลักๆในการชำรุดของเครื่องจักรที่ใช้งานจริงในโรงงาน ซึ่งจะใช้ IoT(internet of things), machine learning, Big Data และ AR (Augmented reality) เป็นเทคโนโลยีที่คอยขับเคลื่อนแพลตฟอร์ม ส่วนประกอบของแพลตฟอร์ม

5.1.1 ส่วนประกอบระบบ

โดยแบ่งระบบภายในได้ออกเป็น 4 ระบบ ประกอบไปด้วย

5.1.1.1 **ระบบการรับข้อมูลเข้ามาในแพลตฟอร์ม**

เป็นระบบที่มีการรับข้อมูลมาจาก Sensor ต่าง ๆ ซึ่งติดอยู่กับตัวโมเดลทดสอบที่ควบคุมด้วย Raspberry Pi แล้วส่งข้อมูลไปยัง Apache Kafka

5.1.1.2 **ระบบการจัดการกับข้อมูลภายในแพลตฟอร์ม**

เป็นระบบที่การทำงานหลักจะอยู่ที่ Apache Kafka โดยจะเป็นตัวกลางการรับส่งข้อมูล ผ่าน topic เพื่อใช้ในการสื่อสารกับส่วนต่างๆของระบบ

5.1.1.3 **ระบบประมวลผลการชำรุดของเครื่องจักร**

เป็นระบบที่จะนำข้อมูลจาก storage มาวิเคราะห์การชำรุดของเครื่องจักร โดยสร้างโมเดลสำหรับทดสอบการชำรุด ให้ได้ก่อนเหตุการณ์จะเกิดขึ้นช่วงเวลานึง โดยใช้ machine learning พื้นฐานเช่น Neural Network เพื่อให้ได้ model สำหรับ predictive maintenance โมเดลทดสอบ และ ระบบ predictive แบบ real time ที่ใช้ในการ แจ้งเตือนแก่ผู้ใช้ผ่าน platform ต่างๆ เช่น email

5.1.1.4 **ระบบแสดงผลผ่านทาง AR application และ web application**

จะนำข้อมูล sensor และ ผลลัพธ์การคาดการณ์การชำรุดของเครื่องจักรจาก Apache Kafka ไปแสดงบน web application และ AR application โดยที่ตัว AR application จะใช้งานกับผู้ดูแลระบบที่อยู่ในพื้นที่ เมื่อต้องการตรวจสถานะของเครื่องจักรเครื่องใด ก็สามารถสแกน QR code เพื่อนำ identification มาดึงข้อมูลจากเว็บ server เพื่อดึงข้อมูลจาก storage มาแสดงในรูปแบบ AR บนโทรศัพท์ ส่วน web application จะมี Dashboard ที่แสดงข้อมูลทั้งหมด ของตัวระบบ

5.1.2 การทดลอง

โดยได้ทำการทดลองในระบบหลักๆของแพลตฟอร์ม ประกอบไปด้วย 3 การทดลองคือ การทดลองเกี่ยวกับการรับส่งข้อมูล, การทดลองเกี่ยวกับ machine learning และ การทดลองเกี่ยวกับการแสดงผล

5.1.2.1 **การทดลองเกี่ยวกับการรับส่งข้อมูล**

เพื่อทดสอบความสามารถของแพลตฟอร์ม ว่าสามารถเชื่อมต่อและ ส่งข้อมูลกัน ผ่านเทคโนโลยีต่างๆ ซึ่งประกอบไปด้วย controller, Apache Kafka, Hadoop เพื่อให้ระบบอื่นๆในแพลตฟอร์มสามารถนำข้อมูลไปใช้ต่อได้ ซึ่งผลที่ได้คือแพลตฟอร์มสามารถเชื่อมต่อและรับส่งข้อมูลกันได้จริงโดยสามารถนำข้อมูลจาก model gearbox มาใช้งานในแพลตฟอร์มได้

5.1.2.2 **การทดลองเกี่ยวกับ machine learning**

เพื่อทดสอบการใช้งาน machine learning ในการคาดการณ์การชำรุดของเครื่องจักร ด้วย algorithm Neural Network ผ่าน MLlib โดยใช้ Spark Core ในการประมวลผล แล้วนำมาประมวลผลแบบ real-time ต่อด้วย Spark Streaming ซึ่งผลที่ได้ออกมาคือสามารถนำ Neural Network มาประยุกต์ใช้กับ machine learning เพื่อใช้คาดการณ์การชำรุดของเครื่องจักรได้จริงและค่อนข้างมีประสิทธิภาพทั้งตอนสร้าง model machine learning และ การคาดการณ์การชำรุดแบบ real-time

5.1.2.3 **การทดลองเกี่ยวกับการแสดงผล**

เป็นการทดสอบ AR application ว่าสามารถที่จะ scan QR code และ generate ข้อมูลออกมาในรูปแบบ AR เพื่อแสดงผล และ การแสดงผลของเว็บสามารถทำงานสอดคล้องกับระบบ

5.2 ปัญหาและอุปสรรค

ปัญหาและอุปสรรคที่เกิดขึ้นในการพัฒนาโครงงานนี้มีหลายประการ ได้แก่

5.2.1 ปัญหาสำหรับการเชื่อมต่อกันของระบบ

- การเชื่อมระหว่าง HDFS และ Apache Kafka ไม่สามารถทำได้ตาม document

5.2.2 ปัญหาสำหรับการทำ machine learning

- การทำ machine learning สำหรับ Big Data นั้นไม่สามารถใช้ library ที่นิยมใช้โดยทั่วไปได้เนื่องจากไม่เหมาะสำหรับการทำงานกับ Big Data
- โมเดล gear box ที่จำลองขึ้นมานั้นมีข้อจำกัดในการจำลองข้อมูลค่อนข้างมาก
- ข้อมูลที่ได้จาก sensor บางครั้งอ่านค่าไม่ได้ เช่น อุณหภูมิบางครั้งได้ 0 องศา

5.2.3 ปัญหาสำหรับการทำ AR application

- Target image ในการทำ AR ต้องมีสีที่แตกต่างชัดเจน

5.3 แนวทางแก้ปัญหา

5.3.1 การแก้ปัญหาสำหรับการเชื่อมต่อกันของระบบ

- ใช้ Apache Flume ในการช่วย stream ข้อมูลจาก Apache Kafka ไปไว้ใน HDFS

5.3.2 การแก้ปัญหาสำหรับการทำ machine learning

- ใช้ MLlib บน Apache Spark ซึ่งเหมาะกับการทำงานแบบ Big Data
- ใช้การ clean data ลด noise

5.3.3 ปัญหาสำหรับการทำ AR application

- จัดทำ QR code เฉพาะสำหรับ identify gearbox

5.4 แนวทางการพัฒนาระบบในอนาคต

ลองใช้แพลตฟอร์มนี้กับ เครื่องจักรแบบอื่นๆ หรือในอุปกรณ์ที่มีความซับซ้อนและข้อมูลที่หลากหลายมากขึ้น รวมไปถึงการ prediction ที่มีความละเอียดมากขึ้นเช่นบอกเวลาที่เหลือได้ ปรับหรือพัฒนาอุปกรณ์ และ sensor ที่ใช้รับข้อมูลจากเครื่องจักรเพื่อข้อมูลที่ดีขึ้นหลากหลายปัจจัยขึ้น ส่วนการแสดงผลอาจจะพัฒนา feature ในการแสดงผลเพิ่มเช่นแสดงเป็น chart หรือ แสดงรายละเอียดต่างๆ ในเครื่องจักรที่กำลัง monitor และในส่วน web application อาจพัฒนาให้ลองรับการ monitor จากหลายโรงงาน

บรรณานุกรม

Pramod Bangalore and Lina Bertling Tjernberg. "An Artificial Neural Network Approach for Early Fault Detection of Gearbox Bearings". IEEE TRANSACTIONS ON SMART GRID, VOL. 6, NO. 2, MARCH 2015

Mikel Canizo, Enrique Onieva, Angel Conde, Santiago Charramendieta and Salvador Trujillo, "Real-time Predictive Maintenance for Wind Turbines Using Big Data Frameworks". IEEE International Conference on Prognostics and Health Management (ICPHM),2017

S. Sheng. "Report on wind turbine subsystem reliability—A survey ofvarious databases". Nat. Renew. Energy Lab., Washington, DC, USA,Tech. Rep. REL/PR-5000-59111, Jun. 2013

A. P. Verma. "Performance monitoring of wind turbines : a dataminingapproach". Ph.D. dissertation, University of Iowa, 2012

G. Cai and S. Mahadevan. "Big data analytics in online structural healthmonitoring". in 2016 Annual Conference of the Prognostics and Health Management Society, vol. 7. International Journal of Prognostics and Health Management, 2016

Lambda architecture. [Online] Available: http://lambda-architecture.net

Spark. [Online] Available: http://spark.apache.org/

Kafka. [Online] Available: http://kafka.apache.org/