การพัฒนาระบบแนะนำรายวิชาสำหรับรายวิชาออนไลน์แบบเปิดสู่มวลชนด้วยเทคนิคข้อมูลขนาดใหญ่

โดย
นางศิริพร ศักดิ์บุญญารัตน์

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปรัชญาดุษฎีบัณฑิต
สาขาวิชาเทคโนโลยีสารสนเทศ แบบ 1.1 ระดับปริญญาดุษฎีบัณฑิต
ภาควิชาคอมพิวเตอร์
บัณฑิตวิทยาลัย มหาวิทยาลัยศิลปากร
ปีการศึกษา 2565
ลิขสิทธิ์ของมหาวิทยาลัยศิลปากร

การพัฒนาระบบแนะนำรายวิชาสำหรับรายวิชาออนไลน์แบบเปิดสู่มวลชนด้วยเทคนิคข้อมูลขนาดใหญ่

โดย
นางศิริพร ศักดิ์บุญญารัตน์

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปรัชญาดุษฎีบัณฑิต
สาขาวิชาเทคโนโลยีสารสนเทศ แบบ 1.1 ระดับปริญญาดุษฎีบัณฑิต
ภาควิชาคอมพิวเตอร์
บัณฑิตวิทยาลัย มหาวิทยาลัยศิลปากร
ปีการศึกษา 2565
ลิขสิทธิ์ของมหาวิทยาลัยศิลปากร

APPLIED BIG DATA TECHNIQUE FOR MASSIVE OPEN ONLINE COURSES (MOOCS) RECOMMENDATION SYSTEM

By

MRS. Siriporn SAKBOONYARAT

A Thesis Submitted in Partial Fulfillment of the Requirements

for Doctor of Philosophy INFORMATION TECHNOLOGY

Department of COMPUTER SCIENCE

Graduate School, Silpakorn University

Academic Year 2022

Copyright of Silpakorn University

หัวข้อ	การพัฒนาระบบแนะนำรายวิชาสำหรับรายวิชาออนไลน์แบบเปิดสู่มวลชนด้วยเทคนิคข้อมูลขนาดใหญ่
โดย	นางศิริพร ศักดิ์บุญญารัตน์
สาขาวิชา	เทคโนโลยีสารสนเทศ แบบ 1.1 ระดับปริญญาดุษฎีบัณฑิต
อาจารย์ที่ปรึกษาหลัก	รองศาสตราจารย์ ดร. ปานใจ ธารทัศนวงศ์

บัณฑิตวิทยาลัย มหาวิทยาลัยศิลปากร ได้รับพิจารณาอนุมัติให้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปรัชญาดุษฎีบัณฑิต

.....คณบดีบัณฑิตวิทยาลัย (ผู้รักษาการแทน)
(ผู้ช่วยศาสตราจารย์ ดร.สาธิต นิรัติศัย)

พิจารณาเห็นชอบโดย

.....ประธานกรรมการ
(ผู้ช่วยศาสตราจารย์ ดร. สุนีย์ พงษ์พินิจภิญโญ)

.....อาจารย์ที่ปรึกษาหลัก
(รองศาสตราจารย์ ดร. ปานใจ ธารทัศนวงศ์)

.....ผู้ทรงคุณวุฒิภายใน
(ผู้ช่วยศาสตราจารย์ ดร. รัชดาพร คณาวงษ์)

.....ผู้ทรงคุณวุฒิภายนอก
(รองศาสตราจารย์ ดร. อนุชัย ธีระเรืองไชยศรี)

60309802 : เทคโนโลยีสารสนเทศ แบบ 1.1 ระดับปริญญาดุษฎีบัณฑิต
คำสำคัญ : บิ๊กดาต้า, ระบบแนะนำรายวิชา, รายวิชาออนไลน์แบบเปิดสู่มวลชน, หน่วยความจำสั้นระยะยาว, โครงข่ายประสาทเทียมเชิงลึก

นาง ศิริพร ศักดิ์บุญญารัตน์: การพัฒนาระบบแนะนำรายวิชาสำหรับรายวิชาออนไลน์แบบเปิดสู่มวลชนด้วยเทคนิคข้อมูลขนาดใหญ่ อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก : รองศาสตราจารย์ ดร. ปานใจ ธารทัศนวงศ์

ระบบคำแนะนำแบบดั้งเดิม (Traditional recommendation techniques) ประสบปัญหาในการขยายขนาด (Scale-out) ส่งผลให้ระบบคำแนะนำมีคุณภาพไม่ดีเท่าที่ควร ไม่สามารถรองรับกับข้อมูลที่มีขนาดใหญ่ โดยเฉพาะอย่างยิ่งเมื่อมีตัวเลือกให้เลือกมากมาย กลายเป็นเรื่องยากสำหรับผู้คนที่จะค้นหาสิ่งที่ต้องการหรือสิ่งที่พวกเขาสนใจ เช่นเดียวกับระบบคำแนะนำรายวิชาที่ต้องเผชิญกับปัญหาดังกล่าว ยิ่งไปกว่านั้นระบบแนะนำรายวิชามักจะอยู่ในสภาพแวดล้อมแบบปิดใช้งานแค่ในส่วนของตนเองเท่านั้น. และระบบยังคงใช้เทคโนโลยีในการจัดเก็บข้อมูลและประมวลผลข้อมูลแบบเดิมทำให้ไม่สามารถใช้งานได้จริงตามสถานการณ์ที่ข้อมูลมีการเปลี่ยนแปลงและเพิ่มขึ้นอย่างรวดเร็ว เมื่อมุ่งเป้าไปที่ปัญหาดังกล่าวผู้วิจัยขอเสนอสถาปัตยกรรมตัวแบบระบบแนะนำรายวิชาสำหรับสภาพแวดล้อม MOOCs ชื่อว่า MCR-CG ด้วยแพลตฟอร์มอาปาเช (Apache platform) สำหรับบริหารจัดการกับข้อมูลขนาดใหญ่ และใช้โครงข่ายประสาทเทียมเชิงลึก (Deep Neural Network) ร่วมกับ Long Short-Term Memory (LSTM) กับการทำงานแบบคลัสเตอริ่ง (Clustering) ที่ฝึกสอนด้วยวิธี fit-generator สำหรับการพัฒนาตัวแบบของระบบ MCR-CG ได้นำไปทดสอบกับชุดข้อมูลจริงของ Harvard and MIT published the edX ผลการทดสอบแสดงให้เห็นว่าทั้งในขั้นตอนการฝึกสอนตัวแบบ (Model training) และขั้นตอนการทดสอบตัวแบบ (Model testing) ให้ผลความแม่นยำสูงถึง 75% และมีประสิทธิภาพเชิงเวลาในขั้นตอนการฝึกสอนตัวแบบ (model training) ที่เร็วกว่าถึง 35% เมื่อเทียบกับการทำงานแบบไม่ใช่คลัสเตอริ่ง (non-clustering) ยิ่งไปกว่านั้น MCR-CG สามารถรองรับการปรับขนาดของพื้นที่จัดเก็บข้อมูล (Storage) ในรูปแบบแนวนอน (Scale-out) และรองรับกับข้อมูลที่มีขนาดใหญ่ได้

60309802 : Major INFORMATION TECHNOLOGY
Keyword : Big Data, Course Recommendation, Massive open online courses (MOOCs), Long short-term memory (LSTM), Deep neural network (DNN)

MRS. SIRIPORN SAKBOONYARAT : APPLIED BIG DATA TECHNIQUE FOR MASSIVE OPEN ONLINE COURSES (MOOCS) RECOMMENDATION SYSTEM THESIS ADVISOR : ASSOCIATE PROFESSOR DR. PANJAI TANTATSANAWONG

Traditional recommendation techniques have difficulty scaling out, resulting in poor quality of the recommendation system. Can't support large data Especially when there are so many options to choose from. It becomes difficult for people to find what they want or what they are interested in and the course guidance system that faces such problems. Moreover, the course recommendation system is often in a disabled environment only on its own. And the system still uses the same technology to store and process the data, making it practically unusable in situations where the data changes and increases rapidly. when aiming at such problems The researchers propose a course guide system architecture for MOOCs environment called MCR-CG with an Apache platform for administration. deal with big data and use a deep neural network and Long Short-Term Memory (LSTM) to work with a trained cluster. fit-generator method for developing system models. MCR-CG was tested on real datasets from Harvard and MIT published the edX. The results showed that both the model training and the model testing steps were up to 75% accurate. and has a time efficiency in the model training process that is up to 35% faster compared to non-clustering models. Moreover, the MCR-CG can support the scaling of storage in a horizontal format. (scale-out) and support large data.

กิตติกรรมประกาศ

ขอกราบขอบพระคุณ รองศาสตราจารย์ ดร. ปานใจ ธารทัศนวงศ์ อาจารย์ที่ปรึกษาวิทยานิพนธ์ ที่ได้กรุณาสละเวลา ดูแลให้คำแนะนำเกี่ยวกับการทำวิจัย รวมทั้งให้คำปรึกษาชี้แนะแนวทางการแก้ไขปัญหาต่าง ๆ ที่เกิดขึ้นระหว่างการทำวิจัย จนวิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงไปได้ด้วยดี ผู้วิจัยรู้สึกซาบซึ้งในความกรุณาครั้งนี้เป็นอย่างยิ่ง จึงขอกราบขอบพระคุณไว้ ณ โอกาสนี้

ขอกราบขอบพระคุณคณะกรรมการสอบวิทยานิพนธ์ทุกท่านเป็นอย่างสูง ได้แก่ ผู้ช่วยศาสตราจารย์ ดร.สุนีย์ พงษ์พินิจภิญโญ หัวหน้าภาควิชาคอมพิวเตอร์ มหาวิทยาลัยศิลปากร ที่กรุณาให้เกียรติเป็นประธาน รองศาสตราจารย์ ดร.อนุชัย ธีระเรืองไชยศรี คณะเภสัชศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย และผู้ช่วยศาสตราจารย์ ดร.รัชดาพร คณาวงษ์ ภาควิชาคอมพิวเตอร์ มหาวิทยาลัยศิลปากร กรุณาให้เกียรติเป็นกรรมการในการสอบวิทยานิพนธ์ครั้งนี้ อีกทั้งได้ให้ข้อคิดเห็นรวมถึงคำแนะนำอันเป็นประโยชน์อย่างยิ่งต่องานวิจัย และทำให้งานวิจัยมีความสมบูรณ์มากยิ่งขึ้น

ขอกราบขอบพระคุณคณะผู้บริหารโรงเรียนมหิดลวิทยานุสรณ์ที่เล็งเห็นความสำคัญของการพัฒนาบุคลากรโดยได้มอบโอกาสและทุนการศึกษาในครั้งนี้

ขอขอบคุณสมาชิกทุกคนในครอบครัว และเพื่อน ๆ พี่ ๆ น้อง ๆ ทุกคนที่คอยช่วยเหลือและให้กำลังใจในทุก ๆ ด้านด้วยดีเสมอมา

และในลำดับสุดท้าย ผู้วิจัยขอน้อมรำลึกถึงอำนาจบารมีของคุณพระศรีรัตนตรัย และสิ่งศักดิ์สิทธิ์ทั้งหลายที่อยู่ในสากลโลก อันเป็นที่พึ่งให้ผู้วิจัยมีสติปัญญาในการจัดทำวิทยานิพนธ์ให้สำเร็จลุล่วงไปด้วยดี ผู้วิจัยขอให้เป็นกตเวทิตาแด่บิดา มารดา ครอบครัวของผู้วิจัย ตลอดจนผู้เขียนหนังสือและบทความต่าง ๆ ที่ให้ความรู้แก่ผู้วิจัยจนสามารถให้วิทยานิพนธ์ฉบับนี้สำเร็จได้ด้วยดี

นาง ศิริพร ศักดิ์บุญญารัตน์

สารบัญ

หน้า

บทคัดย่อภาษาไทย ..... ง

บทคัดย่อภาษาอังกฤษ ..... จ

กิตติกรรมประกาศ ..... ฉ

สารบัญ ..... ช

สารบัญตาราง ..... ญ

สารบัญภาพ ..... ฏ

บทที่ 1 บทนำ 1

1.1 ที่มาและความสำคัญของปัญหา 1

1.2 วัตถุประสงค์ของการวิจัย 5

1.3 ขั้นตอนการวิจัย 5

1.4 ขอบเขตการวิจัย 6

1.5 ประโยชน์ที่คาดว่าจะได้รับ 6

1.6 นิยามศัพท์ 7

1.7 โครงสร้างของเนื้อหาวิทยานิพนธ์ 7

บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง 8

2.1 ทฤษฎีที่เกี่ยวข้อง 8

2.1.1 ระบบแนะนำ (Recommendation system, RS) 8

1) วิธีการกรองเนื้อหา (Content-based filtering) 8

2) วิธีการกรองข้อมูลร่วม (Collaborative filtering) 9

3) วิธีการกรองแบบผสมผสาน (Hybrid filtering) 11

2.1.2 การเรียนรู้เชิงลึก (Deep learning) 11

2.1.3 หน่วยความจำสั้นระยะยาว (Long short term memory, LSTM).....14
2.1.4 เทคโนโลยีสำหรับข้อมูลขนาดใหญ่ (Big data).....19
2.2 งานวิจัยที่เกี่ยวข้อง.....22
บทที่ 3 วิธีดำเนินการวิจัย.....29
3.1 การกำหนดปัญหา การตั้งโจทย์ สมมุติฐานงานวิจัย.....29
3.2 เครื่องมือและอุปกรณ์ที่ใช้ในการพัฒนางานวิจัย.....30
3.3 ข้อมูลที่ใช้ในการวิจัย (Data Overviewing).....31
3.4 สำรวจข้อมูลด้วย Exploratory Data Analysis (EDA).....33
3.5 การกำหนดตัวแบบ (Model Creating).....34
3.5.1 ขั้นตอนวิธีแบบกฎความสัมพันธ์.....34
3.5.2 ขั้นตอนวิธีแบบโครงข่ายประสาทเทียมเชิงลึก.....37
3.5.3 ขั้นตอนวิธีแบบโครงข่ายประสาทเทียมเชิงลึกร่วมกับหน่วยความจำสั้นระยะยาว.....42
3.6 การกำหนดสถาปัตยกรรมระบบแนะนำรายวิชา.....47
บทที่ 4 ผลการดำเนินการวิจัย.....54
4.1 ผลการศึกษาขั้นตอนวิธีที่เหมาะสมสำหรับนำไปสร้างตัวแบบระบบแนะนำรายวิชา.....54
4.1.1 ผลการใช้ขั้นตอนวิธีกฎความสัมพันธ์.....54
4.1.2 ผลการใช้ขั้นตอนวิธีแบบแบบโครงข่ายประสาทเทียมเชิงลึก.....58
4.1.3 สรุปผลการใช้ขั้นตอนวิธีโครงข่ายประสาทเทียมเชิงลึกร่วมกับหน่วยความจำสั้นระยะยาว.....62
4.2 ผลการปรับปรุงเทคนิคระบบแนะนำรายวิชาสำหรับข้อมูลขนาดใหญ่.....70
4.3 ผลการพัฒนาตัวแบบระบบแนะนำรายวิชา.....73
บทที่ 5 สรุปผลการดำเนินการวิจัย.....80
5.1 สรุปผลการวิจัย.....80
5.2 การเตรียมความพร้อมสำหรับนำระบบไปใช้งาน.....83

5.3 การนำสถาปัตยกรรมระบบและตัวแบบระบบแนะนำรายวิชาไปใช้ 84
5.4 ข้อเสนอแนะ 85
5.5 ข้อจำกัดของตัวแบบระบบแนะนำรายวิชา.....85
5.6 ปัญหา และอุปสรรคที่เกิดขึ้นกับจากการวิจัย 86
5.7 แนวทางการวิจัยในอนาคต 86
รายการอ้างอิง 88
ภาคผนวก 91
ประวัติผู้เขียน 92

สารบัญตาราง

หน้า

ตารางที่ 1 จำนวนรายวิชาใน MOOCs แยกตามสถาบันการศึกษา ข้อมูลปี พ.ศ. 2561 [2].....3

ตารางที่ 2 รายชื่อผู้ให้บริการรายวิชาใน MOOCs ห้าอันดับแรกจัดลำดับตามผู้ใช้ที่ลงทะเบียน ข้อมูลปี พ.ศ. 2561 [2] 4

ตารางที่ 3 ความแตกต่างระหว่าง Machine Learning และ Deep Learning 13

ตารางที่ 4 สรุปข้อมูลงานวิจัยเกี่ยวข้องกับรายวิชาออนไลน์แบบเปิดสู่มวลชน (MOOCs) 25

ตารางที่ 5 รายละเอียดของแอตทริบิวต์.....31

ตารางที่ 6 ค่าสัมประสิทธิ์สหสัมพันธ์ระหว่างตัวแปร 34

ตารางที่ 7 รายละเอียดแอตทริบิวต์ (User profile).....37

ตารางที่ 8 รายละเอียดแอตทริบิวต์ (User behavior) 37

ตารางที่ 9 แสดงจำนวนรายวิชาและจำนวนผู้เรียนในแต่ละรายวิชา.....39

ตารางที่ 10 รายละเอียดแอตทริบิวต์ User profile 42

ตารางที่ 11 รายละเอียดแอตทริบิวต์ Course learning 43

ตารางที่ 12 ค่าสนับสนุน (Support) 55

ตารางที่ 13 ค่าความเชื่อมั่น (Confidence).....56

ตารางที่ 14 ประสิทธิภาพของการแนะนำรายวิชาด้วยขั้นตอนวิธีแบบ Apriori 57

ตารางที่ 15 รายละเอียดโครงข่ายประสาทเทียมเชิงลึก 58

ตารางที่ 16 ผลการเปรียบเทียบเชิงเวลาในขั้นตอนการเรียนรู้ เมื่อกำหนดค่าพารามิเตอร์ดังนี้ หนึ่ง) กำหนดค่าการเรียนรู้แบบ 1e-3 และ 5e-4 สอง) กำหนดจำนวนชั้นซ่อนเป็นห้ารูปแบบคือจำนวนชั้นซ่อนแบบ 1, 3, 5, 7 และ 9 ชั้น และ สาม) กำหนดจำนวนรอบในการประมวลผลเป็น 250, 300 Epoch 61

ตารางที่ 17 เทคโนโลยีและขั้นตอนวิธีที่ใช้ในการสร้างตัวแบบระบบแนะนำรายวิชา 62

ตารางที่ 18 ค่าความความแม่นยำของตัวแบบในขั้นตอนการฝึกสอนและขั้นตอนการทดสอบ.....68

ตารางที่ 19 ประสิทธิภาพเชิงเวลาของตัวแบบในขั้นตอนการฝึกสอน 69

ตารางที่ 20 สรุปผลประสิทธิภาพของการแนะนำรายวิชาด้วยขั้นตอนวิธีของกฎความสัมพันธ์ (Association rule) แบบ Apriori 70

ตารางที่ 21 สรุปผลประสิทธิภาพของการแนะนำรายวิชาด้วยขั้นตอนโครงข่ายประสาทเทียมเชิงลึก (Deep neural network, DNN).....70

ตารางที่ 22 สรุปผลประสิทธิภาพของการแนะนำรายวิชาด้วยขั้นตอนโครงข่ายประสาทเทียมเชิงลึก (Deep neural network, DNN) ร่วมกับหน่วยความจำสั้นระยะยาว Long Short-Term Memory (LSTM) 70

ตารางที่ 23 คุณลักษณะของ Hadoop cluster 71

มหาวิทยาลัยศิลปากร

สารบัญภาพ

หน้า

ภาพที่ 1 การเติบโตของรายวิชาออนไลน์แบบเปิดสู่มวลชน (MOOCs) สำรวจเมื่อปี พ.ศ. 2563 [2].2

ภาพที่ 2 สัดส่วนของรายวิชาในรายวิชาออนไลน์แบบเปิดสู่มวลชน (MOOCs) สำรวจเมื่อปี พ.ศ. 2563 [2].....3

ภาพที่ 3 โครงสร้างของโครงข่ายประสาทเทียมเชิงลึกแบบ Feedforward.....12

ภาพที่ 4 โครงข่ายประสาทเทียมแบบ RNN แบบคลี่ [10] 14

ภาพที่ 5 โมดูลที่ทำซ้ำใน RNN มาตรฐานมีเลเยอร์เดียว [10] 15

ภาพที่ 6 โมดูลที่ทำซ้ำใน RNN มาตรฐานมีสี่เลเยอร์ [10] 15

ภาพที่ 7 โครงสร้าง Forget gate layer [12] 16

ภาพที่ 8 โครงสร้าง Input gate layer [12] 17

ภาพที่ 9 โครงสร้าง Output gate layer [12].....18

ภาพที่ 10 การกระจายข้อมูลไปเก็บไว้ในแต่ละบล็อก 20

ภาพที่ 11 Apache Kafka แพลตฟอร์มสตรีมมิ่งแบบกระจาย.....21

ภาพที่ 12 โครงข่ายประสาทเทียมแบบเพอร์เซปตรอนหลายชั้น (Multilayer Perceptron, MLP) โดยแต่ละชั้นซ่อน (Hidden layer) ใช้ฟังก์ชันการกระตุ้น (Activation Function) แบบ Sigmoid และใช้ Softmax ในชั้นสุดท้าย.....39

ภาพที่ 13 ตัวแบบสำหรับระบบแนะนำรายวิชาจำนวนชั้นซ่อน 9 ชั้น โดยชั้นซ่อนลำดับที่ 1 มีโหนด 26 โหนด ชั้นซ่อนลำดับที่ 2 มีโหนด 52 โหนด ชั้นซ่อนลำดับที่ 9 มีโหนด 6656 โหนด รวมมีพารามิเตอร์ที่ใช้ทั้งสิ้นกว่า 29 ล้านพารามิเตอร์.....41

ภาพที่ 14 โครงข่ายประสาทเทียมเชิงลึก (Deep neural network, DNN) ร่วมกับหน่วยความจำสั้นระยะยาว Long Short-Term Memory (LSTM).....44

ภาพที่ 15 กรอบสถาปัตยกรรมระบบแนะนำรายวิชาโดยภาพรวมที่ผู้วิจัยนำเสนอ 47

ภาพที่ 16 กรอบสถาปัตยกรรมระบบแนะนำรายวิชารูปแบบที่ 1 เรียกว่าแบบ Non-clustering.....48

ภาพที่ 17 กรอบสถาปัตยกรรมระบบแนะนำรายวิชารูปแบบที่ 2 เรียกว่า แบบ Clustering 49

ภาพที่ 18 กรอบสถาปัตยกรรมระบบแนะนำรายวิชารูปแบบที่ 1 เรียกว่า แบบ Non-clustering (a) ฝึกสอนด้วย fit method เรียกว่า “MCR-NF” (b) ฝึกสอนด้วย fit-generator method เรียกว่า “MCR-NG” 49

ภาพที่ 19 กรอบสถาปัตยกรรมระบบแนะนำรายวิชารูปแบบที่ 2 เรียกว่า แบบ Clustering) ฝึกสอนด้วย fit-generator method เรียกว่า “MCR-CG” 50

ภาพที่ 20 สรุปกรอบสถาปัตยกรรมระบบแนะนำรายวิชาทั้ง 2 รูปแบบ (a) การทำงานแบบ Non-clustering ฝึกสอนด้วย fit method เรียกว่า “MCR-NF” (b) การทำงานแบบ Non-clustering ฝึกสอนด้วย fit-generator method เรียกว่า “MCR-NG” และ (c) การทำงานแบบ Clustering ฝึกสอนด้วย fit-generator method เรียกว่า “MCR-CG” 53

ภาพที่ 21 ค่าความถูกต้องในขั้นตอนการฝึกสอน (Training) ตัวแบบการทำนายรายวิชาด้วยค่าการเรียนรู้เป็น 1e-3 จำแนกตามจำนวนชั้นซ่อน โดยชั้นซ่อนที่ให้ค่าความถูกต้องมากที่สุดคือชั้นซ่อน (Hidden layer) จำนวน 9 ชั้น ให้ผลความถูกต้องเป็น 70%.....59

ภาพที่ 22 ค่าความถูกต้องในขั้นตอนการฝึกสอน (Training) ตัวแบบการทำนายรายวิชาด้วยค่าการเรียนรู้เป็น 5e-4 จำแนกตามจำนวนชั้นซ่อน โดยชั้นซ่อนที่ให้ค่าความถูกต้องมากที่สุดคือชั้นซ่อน (Hidden layer) จำนวน 9 ชั้น ให้ผลความถูกต้องเป็น 72%.....60

ภาพที่ 23 โครงข่ายประสาทเทียมเชิงลึก (Deep neural network, DNN) ร่วมกับหน่วยความจำสั้นระยะยาว Long Short-Term Memory (LSTM).....63

ภาพที่ 24 ประสิทธิภาพเชิงเวลาและค่าความแม่นยำของตัวแบบ MCR-NF.....64

ภาพที่ 25 ประสิทธิภาพเชิงเวลาและค่าความแม่นยำของตัวแบบ MCR-NG.....65

ภาพที่ 26 ประสิทธิภาพเชิงเวลาและค่าความแม่นยำของตัวแบบ MCR-CG 66

ภาพที่ 27 สถาปัตยกรรมตัวแบบระบบแนะนำรายวิชาที่ผู้วิจัยนำเสนอ.....71

ภาพที่ 28 ตัวแบบระบบแนะนำรายวิชาโดยใช้โครงข่ายประสาทเทียมเชิงลึก (Deep Neural Network) ร่วมกับ Long Short-Term Memory (LSTM) 73

ภาพที่ 29 สถาปัตยกรรมและขั้นตอนการประมวลผลข้อมูลตัวแบบระบบแนะนำรายวิชาในรายวิชาออนไลน์แบบเปิดสู่มวลชน (MOOCs).....74

ภาพที่ 30 โครงสร้างตารางข้อมูล User Profile และ Course Learning บน HDFS สำหรับการประมวลผลข้อมูลเพื่อส่งต่อขั้นตอนฝึกสอนและทดสอบตัวแบบระบบแนะนำรายวิชา 77

ภาพที่ 31 ชุดคำสั่งโปรแกรมภาษา Python ใช้ Keras framework สร้างตัวแบบระบบแนะนำรายวิชา 78

ภาพที่ 32 ชุดคำสั่งโปรแกรมฝึกสอนและทดสอบตัวแบบระบบแนะนำรายวิชาด้วย TensorFlow..79

บทที่ 1
บทนำ

1.1 ที่มาและความสำคัญของปัญหา

ศตวรรษที่ 21 ถือเป็นช่วงเวลาที่ท้าท้ายความสามารถของมนุษยชาติ เพราะเป็นยุคที่โลกต้องเผชิญกับความเปลี่ยนแปลงที่เกิดขึ้นอย่างรวดเร็ว ข้อมูลข่าวสารทุกอย่างไม่ได้จำกัดอยู่เพียงรอบตัวเราอีกต่อไป การหาความรู้สามารถทำผ่านโลกดิจิตอลได้หมดแล้วทำให้สามารถก้าวข้ามพรมแดนไปได้ทุกมุมโลก เพราะฉะนั้นรูปแบบการศึกษาที่เราใช้อยู่ในปัจจุบันก็ต้องเปลี่ยนแปลงเพื่อที่จะได้รองรับทั้งผู้เรียนและผู้สอนในยุคใหม่ ต้องเตรียมคนออกไปเพื่อใช้ความรู้และพร้อมเรียนรู้ตลอดชีวิต (Lifelong Learning) ในการดำรงชีวิตและการทำงานในอนาคตภายใต้สิ่งแวดล้อมทางการเรียนรู้ใหม่ๆ ซึ่งสอดคล้องกับทักษะแห่งศตวรรษที่ 21 (21st Century Skills) [1] ที่จะสร้างคนให้เป็นคนที่มีคุณภาพ (Character Qualities) เพื่อให้สามารถใช้ชีวิตให้เหมาะกับบริบทของโลก สามารถจัดการกับความท้าทายในชีวิต และสามารถรับมือกับโลกที่เปลี่ยนไปอย่างรวดเร็ว การศึกษาเป็นพื้นฐานที่สำคัญที่สุดในการพัฒนาสังคมให้คนซึ่งเป็นสมาชิกของสังคม เป็นคนมีคุณภาพ คุณธรรม การศึกษาช่วยสร้างจิตสำนึกในการเป็นมนุษย์มีจิตวิญญาณของผู้มีอารยะธรรมทางปัญญาและความงดงามทางจิตใจ การศึกษาสร้างให้คนมีความรู้ในการดำรงชีวิต การประกอบอาชีพ มีความอดทนในการต่อสู้กับอุปสรรคของชีวิต การศึกษาเป็นสิ่งจำเป็นสำหรับคนทุกวัย การศึกษาสามารถเปลี่ยนแปลงโลกได้ ดังคำกล่าวของบุคคลผู้ซึ่งได้รับรางวัลโนเบลสาขาสันติภาพ[1] ประจำปี พ.ศ.2536 เนลสัน มันเดลา (Nelson Mandela) ความว่า *“Education is the most powerful weapon which you can use to change the world.”*

ปัจจุบันการเรียนการสอนแบบออนไลน์รูปแบบใหม่ที่มีชื่อเรียกว่า “รายวิชาออนไลน์แบบเปิดสู่มวลชน” (Massive Open Online Courses, MOOCs) ได้รับความนิยมแพร่หลายอย่างรวดเร็วไปทั่วโลก โดยมีระบบจัดการรายวิชา (MOOC Platform) ที่มีมหาวิทยาลัยที่มีชื่อเสียงจากหลายมหาวิทยาลัยเป็นผู้จัดทำรายวิชาที่หลากหลายและเผยแพร่สู่สาธารณะแสดงดังภาพที่ 1 และ ภาพที่ 2 [2] เช่น “Coursera”[2] ก่อตั้งโดยอาจารย์จากมหาวิทยาลัยสแตนฟอร์ด (Stanford University) “edX”[3] ก่อตั้งโดยมหาวิทยาลัยฮาร์วาด (Harvard University) และMIT (Massachusetts Institute

---

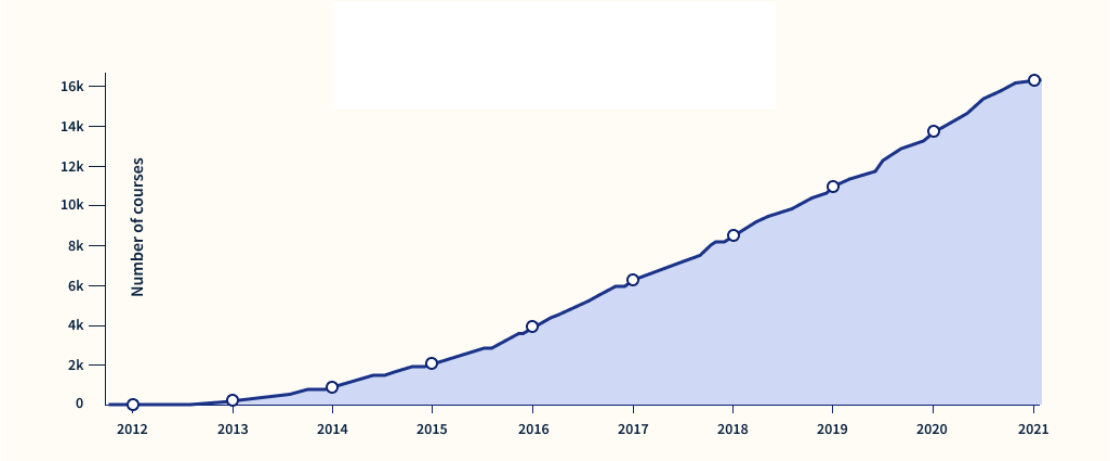
[1] https://www.nobelprize.org/prizes/peace/1993/mandela/facts/

[2] https://www.coursera.org/

[3] https://www.edx.org/

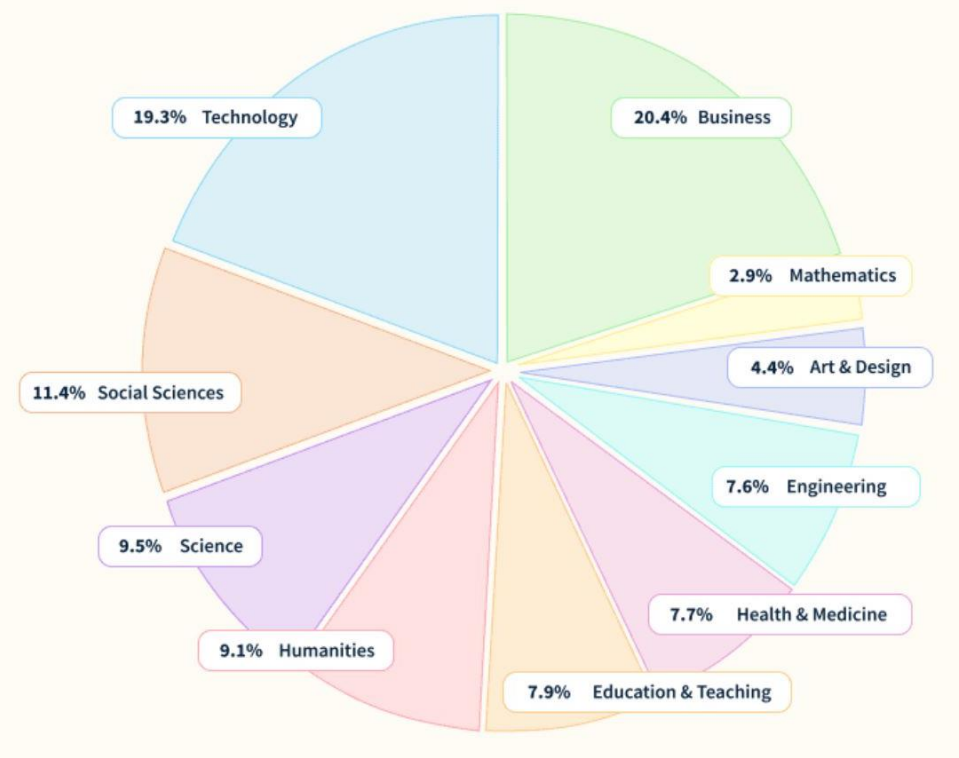
of Technology) และ “Udacity”[4] ก่อตั้งโดยมหาวิทยาลัยสแตนฟอร์ด (Stanford University) แสดงดังตารางที่ 1และตารางที่ 2

โดยในปัจจุบันรายวิชาใน MOOCs มีจำนวนผู้เรียนหลายล้านคน มีรายวิชามากกว่าหนึ่งหมื่นรายวิชา [2] เมื่อรายวิชามีจำนวนเพิ่มมากขึ้นก่อให้เกิดปัญหาการเผชิญกับข้อมูลมหาศาล ซึ่งไม่สามารถจัดการกับข้อมูลและใช้ข้อมูลเหล่านั้นให้เกิดประโยชน์ได้ ผู้ใช้งานบางส่วนไม่เข้าใจข้อมูลรายวิชา เนื่องจากประสบการณ์ในการเรียนหรือระดับความรู้ความเข้าใจที่ไม่มากพอ และปัญหาการเรียนการสอนใน MOOCs คือมีอัตราการหยุดเรียนสูงมากส่วนหนึ่งอาจเนื่องจากเนื้อหาที่นำเสนอไม่ตรงกับที่ผู้เรียนต้องการหรือปัญหาในการกำกับตนเองของผู้เรียน หรือแม้กระทั่งการเรียนการสอนในระบบชั้นเรียนปกติผู้เรียนต้องเผชิญกับข้อมูลรายวิชาที่มากเกินไปทำให้ยากในการตัดสินใจเลือกเรียน



ภาพที่ 1 การเติบโตของรายวิชาออนไลน์แบบเปิดสู่มวลชน (MOOCs) สำรวจเมื่อปี พ.ศ. 2563 [2]

[4] https://www.udacity.com/



ภาพที่ 2 สัดส่วนของรายวิชาในรายวิชาออนไลน์แบบเปิดสู่มวลชน (MOOCs) สำรวจเมื่อปี พ.ศ. 2563 [2]

ตารางที่ 1 จำนวนรายวิชาใน MOOCs แยกตามสถาบันการศึกษา ข้อมูลปี พ.ศ. 2561 [2]

สถาบันการศึกษา	จำนวนรายวิชา (รายวิชา)
1. Massachusetts Institute of Technology	202
2. Indian Institute of Technology, Kharagpur	198
3. Stanford University	184
4. University of Michigan	174
5. University of Pennsylvania	163
6. Indian Institute of Technology Madras	158
7. Harvard University	157
8. University of Naples Federico II	154

สถาบันการศึกษา	จำนวนรายวิชา (รายวิชา)
9. University of Illinois at Urbana-Champaign	150
10. Indian Institute of Technology Kanpur	141

ตารางที่ 2 รายชื่อผู้ให้บริการรายวิชาใน MOOCs ห้าอันดับแรกจัดลำดับตามผู้ใช้ที่ลงทะเบียน ข้อมูลปี พ.ศ. 2561 [2]

ผู้ให้บริการ	จำนวนผู้เรียนที่ลงทะเบียน (ล้านคน)
1. Coursera	37
2. edX	18
3. XuetangX[5]	14
4. Udacity	10
5. FutureLearn[6]	8.7

ความก้าวหน้าและพัฒนาการของเทคโนโลยีสารสนเทศก้าวไปอย่างรวดเร็ว ข้อมูลและสารสนเทศมีมากมายบนโลกอินเทอร์เน็ตซึ่งนับเป็นความท้าทายรูปแบบหนึ่งที่มนุษย์ต้องเผชิญ รายวิชาใน MOOCs เป็นหนึ่งในความท้าทายบนข้อมูลขนาดใหญ่ (Big Data) เช่นเดียวกัน เนื่องด้วยคุณลักษณะดังต่อไปนี้คือ หนึ่ง) ข้อมูลมีปริมาณมาก (Volume) ทั้งจำนวนรายวิชาดังตารางที่ 1 จำนวนผู้เรียนดังตารางที่ 2 สอง) ข้อมูลมีรูปแบบที่หลากหลาย (Variety) ซึ่งอาจเป็นข้อมูลเชิงโครงสร้าง (Structured data)[7] ข้อมูลกึ่งโครงสร้าง (Semi-structured data)[8] ข้อมูลที่ไม่มีโครงสร้างแน่นอน (Unstructured data)[9] และสาม) ข้อมูลมีการเปลี่ยนแปลงตลอดเวลาและรวดเร็ว (Velocity) [3] เช่น ข้อมูลการคลิกเล่นยกเลิกหรือหยุดวิดีโอของผู้เรียน ดังนั้นการประมวลผลบน

[5] https://www.xuetangx.com/

[6] https://www.futurelearn.com/

[7] คือ ข้อมูลที่จัดเก็บในตารางข้อมูลที่มีการจัดเรียงอย่างมีรูปแบบชัดเจนและเป็นระเบียบ สามารถนำมาใช้วิเคราะห์ได้เลย เช่น ข้อมูลที่เก็บไว้ในโปรแกรม spreadsheet อย่าง Microsoft Excel

[8] คือ ข้อมูลที่ถูกจัดเก็บอย่างมีรูปแบบในระดับหนึ่งและข้อมูลสามารถค้นหา (Search) หรือแท็ก (Tag) ได้ เช่น เว็บเพจที่มีการระบุชื่อเพจ คำสำคัญในเพจ และวันที่อัพเดทข้อมูล

[9] คือ ข้อมูลที่ไม่มีโครงสร้างแน่นอน (Unstructured data) เช่น ข้อความเสียง รูปภาพ และวิดีโอ

ข้อมูลขนาดใหญ่ (Big Data) เป็นงานที่ท้าทาย เพราะด้วยคุณลักษณะของข้อมูลที่กล่าวมาส่งผลให้ซอฟต์แวร์หรือฮาร์ดแวร์ธรรมดานั้นไม่สามารถที่จะจัดการหรือวิเคราะห์ได้อย่างมีประสิทธิภาพ จากอัตราการเพิ่มขึ้นของข้อมูลเป็นไปอย่างรวดเร็วเป็นสาเหตุทำให้การประมวลผลเป็นไปได้อย่างยากลำบากจึงต้องหาวิธีประมวลผลข้อมูลให้มีผลดีและถูกต้องเพื่อหาผลลัพธ์ที่ต้องการ โดยเฉพาะการตัดสินใจเลือกสิ่งของต่าง ๆ เป็นสิ่งที่ต้องอาศัยประสบการณ์ ความรู้ เวลา หรือปัจจัยอื่น ๆ เข้ามาช่วยพิจารณา ซึ่งจะเห็นได้ว่าระบบแนะนำ (RS) เข้ามามีบทบาทในชีวิตของมนุษย์มากยิ่งขึ้นช่วยอำนวยความสะดวกให้การค้นหาแคบลงและสะดวกรวดเร็ว เช่น ระบบแนะนำหนังสือ บริการ เพลง สินค้าบน Amazon[10] วิดีโอบน YouTube[11] หรือ Netflix[12] ร้านอาหารบน Wongnai[13] ข้อมูลข่าวสาร เป็นต้น เมื่อข้อมูลมีปริมาณมากขึ้นและข้อมูลมีความหลากหลายมากขึ้นทำให้การเก็บข้อมูลในฐานข้อมูลหรือคลังข้อมูลแบบเดิมไม่ได้มีประสิทธิภาพเท่าที่ควร ดังนั้นงานวิจัยฉบับนี้จึงมุ่งเน้นไปที่การปรับปรุงเทคนิคระบบแนะนำรายวิชาบนข้อมูลขนาดใหญ่ (Big Data) ให้มีประสิทธิภาพ เพื่อรองรับกับโลกยุคดิจิทัล (Digital Age)

1.2 วัตถุประสงค์ของการวิจัย

ในการวิจัยครั้งนี้ผู้วิจัยมีวัตถุประสงค์ ดังนี้

1) เพื่อศึกษาขั้นตอนวิธีที่เหมาะสมสำหรับนำไปสร้างตัวแบบระบบแนะนำรายวิชาในรายวิชาออนไลน์แบบเปิดสู่มวลชน (MOOCs) ด้วยเทคนิคข้อมูลขนาดใหญ่ (Big Data)

2) เพื่อปรับปรุงเทคนิคระบบแนะนำรายวิชาสำหรับข้อมูลขนาดใหญ่ (Big Data) ให้เหมาะสมและมีประสิทธิภาพ เช่น ลดเวลาในการทำงาน หรือมีความแม่นยำ

3) เพื่อพัฒนาตัวแบบระบบแนะนำรายวิชาในรายวิชาออนไลน์แบบเปิดสู่มวลชน (MOOCs) ให้กับผู้เรียน

1.3 ขั้นตอนการวิจัย

ในการวิจัยครั้งนี้ดำเนินการตามขั้นตอนการวิจัย ดังนี้

1) ศึกษาหัวข้องานวิจัย

2) ทบทวนวรรณกรรมที่เกี่ยวข้อง

---

[10] https://www.amazon.com/

[11] https://www.youtube.com/

[12] https://www.netflix.com/

[13] https://www.wongnai.com/

3) กำหนดขอบเขตหัวข้องานวิจัย

4) ออกแบบแนวทางการวิจัย

5) พัฒนางานวิจัยตามที่ได้ออกแบบไว้

6) สรุปผลงานวิจัย

1.4 ขอบเขตการวิจัย

จากวัตถุประสงค์ของการวิจัยดังที่กล่าวแล้วข้างต้น ผู้วิจัยกำหนดขอบเขตของการวิจัย ดังนี้

1) ข้อมูลที่นำมาใช้ คือ ข้อมูลจากรายวิชาออนไลน์แบบเปิดสู่มวลชน (MOOCs) ประกอบด้วยข้อมูลประวัติผู้เรียน (User profile) และข้อมูลเชิงพฤติกรรมของผู้เรียน (User behavior) เท่านั้น

2) ผลลัพธ์ของระบบแนะนำรายวิชาในรายวิชาออนไลน์แบบเปิดสู่มวลชน (MOOCs) สามารถอยู่ในรูปแบบรายวิชาเดี่ยวหรือกลุ่มรายวิชาก็ได้

1.5 ประโยชน์ที่คาดว่าจะได้รับ

1.5.1 ประโยชน์ต่อวงการคอมพิวเตอร์และเทคโนโลยี

1) ได้เทคนิคระบบแนะนำรายวิชาสำหรับข้อมูลขนาดใหญ่ (Big Data) ที่มีประสิทธิภาพ

2) เป็นแนวทางสำหรับการประมวลผลข้อมูลขนาดใหญ่ (Big Data) ซึ่งเหมาะสมกับยุคปัจจุบันที่เป็นยุคของข้อมูล

1.5.2 ประโยชน์ทางการศึกษา

1) ผู้เรียนสามารถเข้าถึงรายวิชาได้ตรงกับความต้องการ

2) ผู้เรียนได้รับคำแนะนำหรือกระตุ้นให้เกิดการเรียนรู้

3) ผู้สร้างรายวิชาสามารถที่จะสร้างรายวิชาซึ่งตรงกับความต้องการของผู้เรียนได้มากขึ้น ตรงกับความต้องการของคนรุ่นใหม่ที่ต้องการเรียนรู้สิ่งต่าง ๆ ที่เฉพาะเจาะจง

1.5.3 ประโยชน์ต่อสังคมและประเทศ

1) เมื่อประชากรมีทักษะที่ดีขึ้น พวกเขาก็สามารถที่จะทำอาชีพได้ดีขึ้น นำความรู้ที่ได้ไปประกอบสัมมาอาชีพได้ถูกต้องมากขึ้น

1.5.4 ประโยชน์ต่อพลโลก

1) พัฒนาทักษะการเรียนรู้ของคนทั้งโลก ไม่ใช่แค่คนไทยเท่านั้น

1.6 นิยามศัพท์

1) รายวิชาออนไลน์แบบเปิดสู่มวลชน (Massive Open Online Courses, MOOCs) หมายถึง การจัดการเรียนการสอนในรูปแบบออนไลน์ที่รองรับผู้เรียนจำนวนมาก ไม่จำกัดเพศ วัย การศึกษา โดยให้บริการฟรีและเป็นการเรียนรู้แบบเรียนได้ทุกที่ทุกเวลา ผ่านอุปกรณ์ที่หลากหลาย เช่น สมาร์ตโฟน คอมพิวเตอร์ แท็บเล็ต โดยเรียนผ่านวิดีโอ บรรยายโดยอาจารย์เจ้าของวิชา มีแบบฝึกหัดหลังบทเรียน บางแพลตฟอร์มมีกระดานสนทนาหรือฟอรั่ม (Forum) ให้ผู้เรียนได้เข้าไปสนทนาแลกเปลี่ยนความรู้กับผู้สอน หรือผู้เรียนด้วยกัน

2) ระบบจัดการรายวิชา (MOOC Platform) หมายถึง ระบบจัดการรายวิชาออนไลน์ที่รองรับผู้เรียนจำนวนมาก มุ่งเน้นเป็นเครื่องมือส่งเนื้อหาความรู้แก่ผู้เรียน และมีเครื่องมือสำคัญที่ใช้สำหรับกิจกรรมการเรียนรู้และการประเมิน เช่น Coursera, Open edX

3) ระบบแนะนำ (Recommender Systems, RS) คือ ระบบที่แนะนำสินค้า ข้อมูล บริการ ซึ่งเรียกว่าไอเทม (Item) ให้กับผู้ใช้ ไอเทมในความหมายของระบบแนะนำ หมายถึง สิ่งที่ต้องการจะแนะนำ โดยระบบจะเรียนรู้ข้อมูลความชอบหรือความต้องการของผู้ใช้แล้วจึงแนะนำไอเทมให้กับผู้ใช้ต่อไป

1.7 โครงสร้างของเนื้อหาวิทยานิพนธ์

โครงสร้างของเนื้อหาวิทยานิพนธ์ประกอบด้วย 5 บท ซึ่งมีรายละเอียดดังต่อไปนี้

บทที่ 1 กล่าวถึง บทนำ ประกอบด้วย ที่มาและความสำคัญของปัญหา วัตถุประสงค์ของการวิจัย ขั้นตอนการวิจัย ขอบเขตการวิจัย ประโยชน์ที่คาดว่าจะได้รับ และนิยามศัพท์

บทที่ 2 กล่าวถึง ทฤษฎีที่เกี่ยวข้องประกอบด้วยบระบบแนะนำ การเรียนรู้เชิงลึก หน่วยความจำสั้นระยะยาว เทคโนโลยีสำหรับข้อมูลขนาดใหญ่ และงานวิจัยที่เกี่ยวข้อง

บทที่ 3 กล่าวถึง การกำหนดปัญหา การตั้งโจทย์ สมมุติฐานงานวิจัย เครื่องมือและอุปกรณ์ที่ใช้ในการพัฒนางานวิจัย ข้อมูลที่ใช้ในการวิจัย การสำรวจข้อมูล การกำหนดตัวแบบ และการกำหนดสถาปัตยกรรมระบบแนะนำรายวิชา

บทที่ 4 กล่าวถึง ผลการดำเนินงานวิจัย

บทที่ 5 กล่าวถึง สรุปผลการดำเนินงานวิจัย

บทที่ 2
ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

การศึกษาและวิจัยในครั้งนี้เพื่อศึกษาขั้นตอนวิธีที่เหมาะสมสำหรับนำไปสร้างตัวแบบระบบแนะนำรายวิชาในรายวิชาออนไลน์แบบเปิดสู่มวลชน (MOOCs) ด้วยเทคนิคข้อมูลขนาดใหญ่ (Big Data) เพื่อนำไปปรับปรุงเทคนิคระบบแนะนำรายวิชาสำหรับข้อมูลขนาดใหญ่ (Big Data) ให้เหมาะสมและมีประสิทธิภาพ และนำไปพัฒนาเป็นตัวแบบระบบแนะนำรายวิชาในรายวิชาออนไลน์แบบเปิดสู่มวลชน (MOOCs) เพื่อแนะนำรายวิชาให้กับผู้เรียนต่อไป ดังนั้นงานวิจัยฉบับนี้จึงได้รวบรวมทฤษฎีและงานวิจัยที่เกี่ยวข้องไว้ดังต่อไปนี้

2.1 ทฤษฎีที่เกี่ยวข้อง

2.1.1 ระบบแนะนำ (Recommendation system, RS)

2.1.2 การเรียนรู้เชิงลึก (Deep learning)

2.1.3 หน่วยความจำสั้นระยะยาว (Long short term memory)

2.1.4 เทคโนโลยีสำหรับข้อมูลขนาดใหญ่ (Big data)

2.2 งานวิจัยที่เกี่ยวข้อง

2.1 ทฤษฎีที่เกี่ยวข้อง

2.1.1 ระบบแนะนำ (Recommendation system, RS)

ระบบแนะนำเป็นระบบที่ใช้ในการเก็บรวบรวมสารสนเทศจากความพึงพอใจของผู้ใช้ในตัวสินค้าและบริการที่สนใจ เพื่อทำนายสิ่งที่ผู้ใช้สนใจในตัวสินค้าหรือบริการจากข้อมูล เทคนิคพื้นฐานที่ใช้ในการสร้างรายการแนะนำสำหรับระบบแนะนำ เช่น วิธีการกรองเนื้อหา (Content-based filtering) วิธีการกรองข้อมูลร่วม (Collaborative filtering) และวิธีการกรองแบบผสมผสาน (Hybrid filtering) [4] รายละเอียดดังนี้

1) วิธีการกรองเนื้อหา (Content-based filtering)

การกรองสารสนเทศโดยดูเนื้อหา (Content-based filtering) ให้ความสนใจกับเนื้อหาตามคุณสมบัติของข้อมูลเป็นหลัก วิธีนี้จะตรวจสอบว่าข้อมูลหรือเนื้อหาส่วนใดที่ตรงกันหรือมีความคล้ายคลึงกับข้อมูลส่วนบุคคลของผู้ใช้ (User profile) ถ้าตรงกันก็จะนำมาแสดงผลเป็นรายการแนะนำให้แก่ผู้ใช้ แต่ถ้าไม่ตรงกันก็จะไม่มีการแสดงผลรายการแนะนำแม้ว่าข้อมูลหรือเนื้อหาอาจจะมีลักษณะใกล้เคียงกับที่ผู้ใช้ต้องการก็ตาม ดังนั้นวิธีการนี้จะเป็นการ

คำนวณหาค่าความคล้ายคลึงระหว่างข้อมูลหรือเนื้อหากับข้อมูลส่วนบุคคลของผู้ใช้ โดยการจับคู่ข้อมูลหรือเนื้อหากับข้อมูลส่วนบุคคลของผู้ใช้เพื่อใช้ในระบบสารสนเทศตามความสนใจของผู้ใช้ เทคนิคที่ใช้ในการกรองสารสนเทศโดยดูเนื้อหา เช่น การจัดกลุ่มข้อมูล (Clustering) การแบ่งกลุ่มข้อมูลแบบเบย์เซียน (Bayesian classifiers) การจำแนกข้อมูลโดยใช้ต้นไม้ตัดสินใจ (Decision trees) โครงข่ายประสาทเทียมที่มีการเลียนแบบการทำงานของสมองของมนุษย์ (Artificial neural networks) ข้อดีของการกรองสารสนเทศโดยดูเนื้อหาคือสามารถประมวลผลบนข้อมูลจำนวนน้อยได้ และจะไม่พบกับปัญหากรณีที่ไอเทมใหม่ไม่ถูกแนะนำไปยังผู้ใช้หรือผู้ใช้ที่เข้าสู่ระบบครั้งแรกจะไม่ได้รับการแนะนำซึ่งปัญหาลักษณะนี้เรียกว่า “Cold start problem” สำหรับข้อเสียของการกรองสารสนเทศโดยดูเนื้อหาคือจะได้รายการไอเทมที่ไม่หลากหลาย (Overspecialization) ระบบแนะนำสินค้าหรือบริการของที่มีการกรองสารสนเทศโดยดูเนื้อหา เช่น IMDB[14] และ Rotten Tomatoes[15]

2) วิธีการกรองข้อมูลร่วม (Collaborative filtering)

เป็นการแนะนำสินค้าหรือบริการที่ได้มาจากการพิจารณาความคล้ายคลึงกันหรือความชอบของผู้ใช้งานที่เป็นเป้าหมายในการแนะนำกับเปรียบเทียบกับบุคคลอื่นที่มีความชอบคล้ายคลึงกัน โดยจะนำไปพิจารณากับไอเทมที่ผู้ใช้ยังไม่ให้ความนิยมไว้แต่มีความคล้ายกับไอเทมที่ผู้ใช้เคยให้คะแนนไว้แล้ว

วิธีการเก็บข้อมูลเพื่อนำมาคำนวณความชอบที่มีต่อไอเทมสามารถทำได้สองวิธีคือ หนึ่ง) การเก็บข้อมูลจากผู้ใช้โดยตรง (Explicit feedback) เช่น การตอบแบบสอบถาม คะแนนความนิยม (Rating) ข้อดีคือได้ข้อมูลจากผู้ใช้โดยตรง ข้อมูลมีความถูกต้องน่าเชื่อถือกว่าวิธี Implicit feedback ข้อเสียคือผู้ใช้ต้องเสียเวลาในการให้ข้อมูลแก่ระบบ และสอง) การเก็บข้อมูลจากโดยอ้อมหรือโดยนัย (Implicit feedback) วิธีนี้อาศัยการสังเกตพฤติกรรมหรือดึงข้อมูลการใช้งานระบบในอดีตของผู้ใช้มาทำการวิเคราะห์ เช่น ประวัติในการซื้อไอเทม หรือ ประวัติในการฟังเพลง ข้อดีคือผู้ใช้ไม่ต้องให้ข้อมูลแก่ระบบแต่ระบบเป็นผู้รวบรวมข้อมูลจากผู้ใช้เอง ข้อเสียคือการเก็บข้อมูลจากผู้ใช้เพื่อนำมาวิเคราะห์บางครั้งอาจได้ผลลัพธ์ที่ไม่ตรงใจผู้ใช้เท่ากับวิธีแบบ Explicit feedback

วิธีการหาความคล้ายของผู้ใช้เป้าหมายกับผู้ใช้คนอื่นที่มีอยู่ในฐานข้อมูล [5] เช่น วิธีการหาค่าความคล้ายคลึงแบบโคโซน์ (Cosine-based Similarity) ดังสมการที่ (1) วิธีการหาค่าความคล้ายคลึงแบบโคไซน์แบบปรับแก้ (Adjusted Cosine Similarity) ดังสมการที่ (2) และวิธีการหาค่าความคล้ายคลึงแบบค่าสัมประสิทธิ์สหสัมพันธ์ของเพียร์สัน (Pearson correlation coefficient) ดังสมการที่ (3) ข้อเสียของวิธีการกรองข้อมูลร่วม (Collaborative filtering) คือข้อมูล

---

[14] https://www.imdb.com/

[15] https://www.rottentomatoes.com/

ไอเทมที่มีอยู่เป็นจำนวนมากบางรายการไม่ถูกคะแนนทำให้เกิดปัญหาความบางเบาของข้อมูล (Sparsity problem) และ Cold start problem

$$sim(u_i, u_j) = \frac{\sum_{c=1}^{n} R_{i,c} \cdot R_{j,c}}{\sqrt{\sum_{c=1}^{n} R_{i,c}^2} \sqrt{\sum_{c=1}^{n} R_{j,c}^2}}$$ สมการที่ (1)

$$sim(u_i, u_j) = \frac{\sum_{c \in I_{ij}} \left(R_{i,c} - \overline{R_i}\right)\left(R_{j,c} - \overline{R_j}\right)}{\sqrt{\sum_{c \in I_i} \left(R_{i,c} - \overline{R_i}\right)^2} \sqrt{\sum_{c \in I_j} \left(R_{j,c} - \overline{R_j}\right)^2}}$$ สมการที่ (2)

$$sim(u_i, u_j) = \frac{\sum_{c \in I_{ij}} \left(R_{i,c} - \overline{R_i}\right)\left(R_{j,c} - \overline{R_j}\right)}{\sqrt{\sum_{c \in I_{ij}} \left(R_{i,c} - \overline{R_i}\right)^2} \sqrt{\sum_{c \in I_{ij}} \left(R_{j,c} - \overline{R_j}\right)^2}}$$ สมการที่ (3)

เทคนิคที่เป็นที่นิยมนำมาใช้ในการกรองข้อมูลร่วม (Collaborative filtering) เช่น การหาสมาชิกที่ใกล้ที่สุด (Nearest neighbor) การจัดกลุ่มข้อมูล (Clustering) ทฤษฎีกราฟ (Graph theory)โครงข่ายแบบเบย์เซียน (Bayesian networks) โครงข่ายประสาทเทียมที่มีการเลียนแบบสมองของมนุษย์ (Artificial neural networks) การถดถอยแบบเชิงเส้น (Linear regression) และแบบจำลองความน่าจะเป็น (Probabilistic model) ระบบแนะนำสินค้าหรือบริการ

ที่มีการกรองข้อมูลร่วม เช่น Amazon[16], Facebook[17], Twitter[18], LinkedIn[19], Spotify[20], Google News[21] และ Last.fm[22] วิธีการกรองข้อมูลร่วม (Collaborative filtering) แบ่งออกเป็น 2 ประเภท คือ

- การกรองข้อมูลร่วมโดยพิจารณาจากผู้ใช้ (User based Collaborative filtering)
- การกรองข้อมูลร่วมโดยพิจารณาจากรายการสินค้า (Item based Collaborative filtering)

3) วิธีการกรองแบบผสมผสาน (Hybrid filtering)

เป็นการผสมผสานวิธีการต่าง ๆ โดยนำจุดเด่นของแต่ละวิธีการมาปรับปรุงข้อที่บกพร่องของอีกวิธีการ เช่น การนำข้อดีของทั้งวิธีการกรองร่วมมาผสมกับข้อดีของวิธีการกรองเนื้อหาเพื่อสร้างเป็นระบบแนะนำ

2.1.2 การเรียนรู้เชิงลึก (Deep learning)

การเรียนรู้เชิงลึกเป็นหนึ่งในฟังก์ชันของปัญญาประดิษฐ์ (Artificial intelligence, AI) เป็นเซตย่อยของการเรียนรู้ด้วยเครื่อง (Machine learning) ในปัญญาประดิษฐ์ (AI) ที่เรียนแบบการทำงานของสมองมนุษย์ในกระบวนการประมวลผลข้อมูลและเป็นการสร้างรูปแบบ สำหรับใช้ในการตัดสินใจ ถูกสร้างมาจากการนำเอา Neural network หลายๆ ชั้นมาต่อกัน โดยชั้นแรกสุดทำหน้าที่เป็นชั้นนำเข้าข้อมูล (Input layer) ชั้นสุดท้ายจะทำหน้าที่ส่งผลลัพธ์การประมวลผลออกมา (Output layer) ส่วน ชั้นระหว่างชั้นแรกสุดและชั้นสุดท้ายเรียกว่าชั้นซ่อน (Hidden layer) แสดงภาพที่ 3

---

[16] https://www.amazon.com/

[17] http://www.facebook.com/

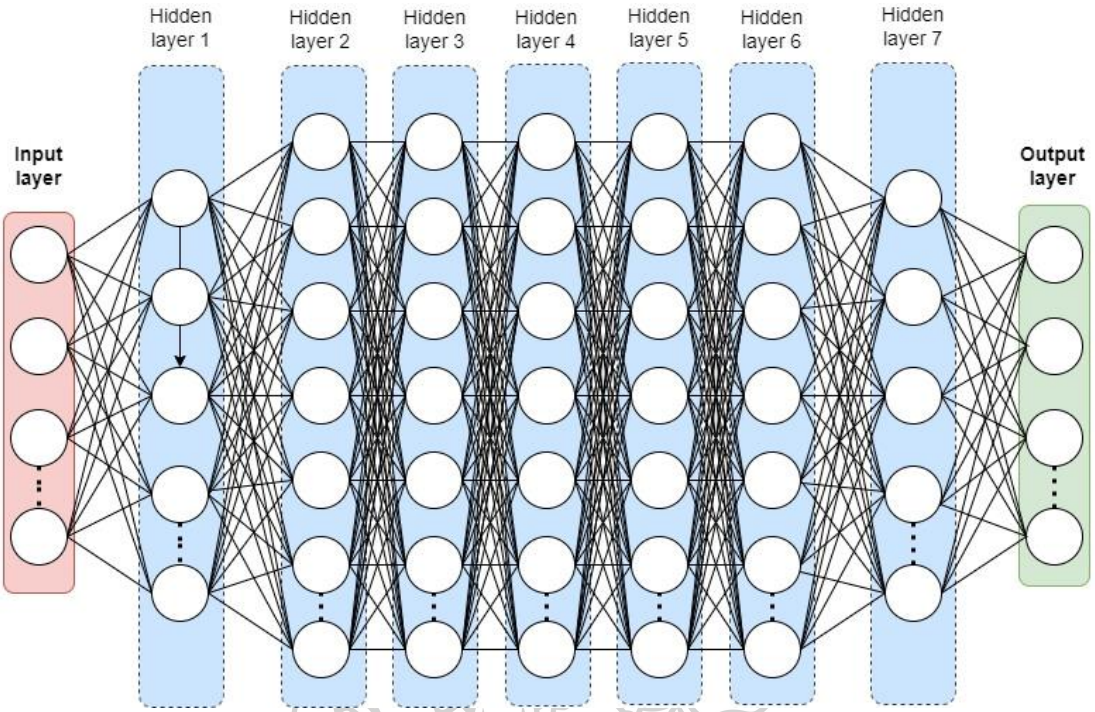
[18] http://www.twitter.com/

[19] http://www.linkedin.com/

[20] https://www.spotify.com/

[21] https://news.google.com/

[22] https://www.last.fm/



ภาพที่ 3 โครงสร้างของโครงข่ายประสาทเทียมเชิงลึกแบบ Feedforward

1) โครงข่ายประสาทเทียมเชิงลึก ประกอบด้วยเลเยอร์ต่าง ๆ ดังนี้

- Input Layer ตัวแปรข้อมูลที่ต้องการให้ Network คำนวณ เป็นจุดเริ่มต้นของขั้นตอนการทำงานสำหรับโครงข่ายประสาทเทียมจะทำหน้าที่ส่งข้อมูลไปยังแต่ละโหนด ของ Layer ต่อไป โดยจำนวนของโหนด ใน Input Layer จะขึ้นอยู่กับขนาดของข้อมูล

- Hidden Layer ประกอบไปด้วย neural node (สามารถมีได้หลายชั้น) เป็นส่วนที่ทำหน้าที่ส่งต่อข้อมูลไปยัง Output Layer โดยแต่ละครั้งที่ Training Data ผ่าน Layer นี้ แต่ละโหนดจะค่อย ๆ ปรับ Weight ให้เข้ากับ Data มากขึ้น หรือก็คือ Hidden Layer จะหาความสัมพันธ์ระหว่าง Feature ของ Data

- Output Layer ประกอบไปด้วย neural node และผลลัพธ์การคำนวณของ Network (Estimation output) เป็นส่วนที่จะแสดงผล Output ซึ่งจำนวนโหนดใน Output Layer จะขึ้นอยู่กับจำนวนคลาสในข้อมูล เช่น จะสร้างโครงข่ายประสาทเทียมเพื่อจำแนกสุนัขกับแมวมี Output โหนด 2 โหนด แต่ถ้าใช้กับ Regression จะมีแค่ 1 โหนดเท่านั้น เพราะทำนาย (Predict) เป็นตัวเลข

2) ความแตกต่างระหว่าง Machine Learning และ Deep Learning

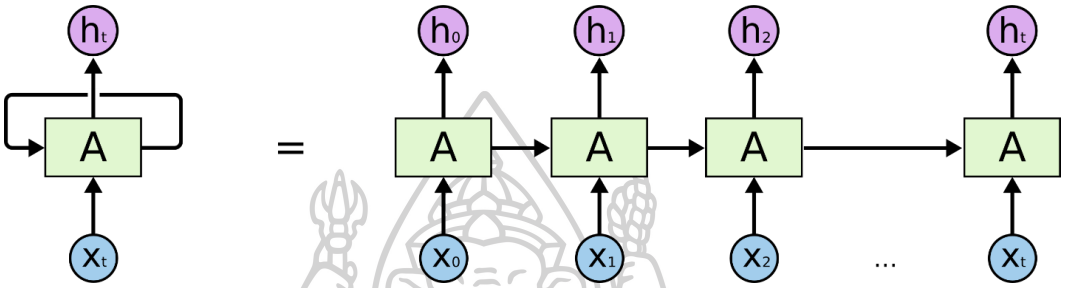
ตารางที่ 3 ความแตกต่างระหว่าง Machine Learning และ Deep Learning

	Machine Learning	Deep Learning
การอ้างอิงข้อมูล (Data Dependencies)	ประสิทธิภาพดีในชุดข้อมูลระดับเล็กถึงระดับกลาง	ประสิทธิภาพดีในชุดข้อมูลขนาดใหญ่
การอ้างอิงฮาร์ดแวร์ (Hardware Dependencies)	ดีสำหรับ machine ระดับล่าง	ต้องการ machine ที่มีศักยภาพสูง โดยเฉพาะถ้ามีการ์ดจอ (Graphic card : GPU)
วิศวกรรมคุณลักษณะ (Feature engineering)	จำเป็นต้องเข้าใจคุณลักษณะซึ่งแสดงให้เห็นถึงข้อมูล	ไม่จำเป็นต้องเข้าใจคุณลักษณะที่แสดงให้เห็นถึงข้อมูล
เวลาดำเนินการ (Execution Time)	ตั้งแต่ไม่กี่นาทีไปถึงหลายชั่วโมง	มากถึงสัปดาห์
ความสามารถในการตีความ (Interpretability)	บางอัลกอริทึมง่ายต่อการตีความ (logistic, decision tree)	ยากต่อความเข้าใจ
ชุดข้อมูลสำหรับฝึก (Training data)	ขนาดเล็ก	ขนาดใหญ่
การเลือกคุณลักษณะ (Choose features)	ใช่	ไม่ใช่
จำนวนอัลกอริทึม (Number of algorithms)	จำนวนมาก	เล็กน้อย
เวลาในการฝึกฝน (Training Time)	สั้น	ยาว

การเรียนรู้เชิงลึกได้ถูกนำไปใช้ในงานหลาย ๆ ด้าน รวมถึงระบบการแนะนำเช่นกัน เช่น ระบบการแนะนำเพลง (Music) [6], วิดีโอหรือภาพยนตร์ (Video or Movie) [7, 8] และข่าว (News) [9]

2.1.3 หน่วยความจำสั้นระยะยาว (Long short term memory, LSTM)

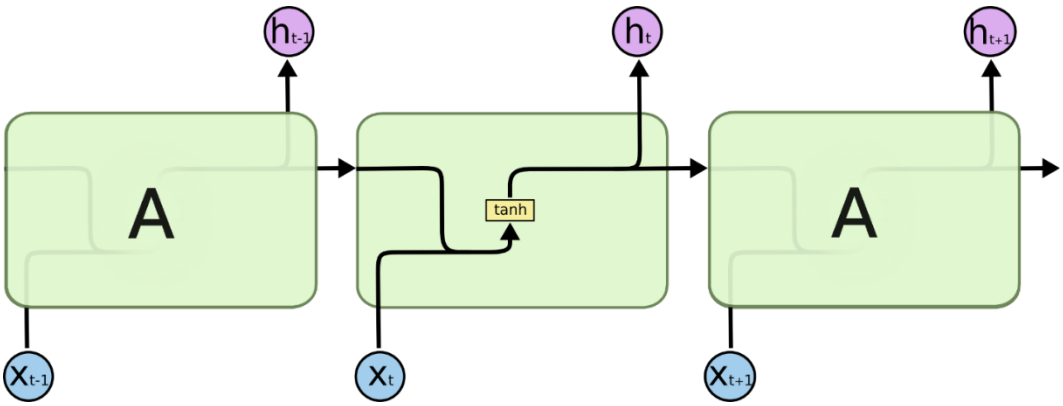
LSTM เป็นโครงข่ายประสาทเทียมแบบหนึ่งที่ถูกออกแบบมาสำหรับการประมวลผลข้อมูลที่มีความต่อเนื่องกัน หรือข้อมูลที่เหตุการณ์ก่อนหน้ามีผลต่อเหตุการณ์ที่จะเกิดขึ้นต่อไปในอนาคต LSTM ถูกพัฒนามาจาก Recurrent Neural Network (RNN) แสดงดังภาพที่ 4



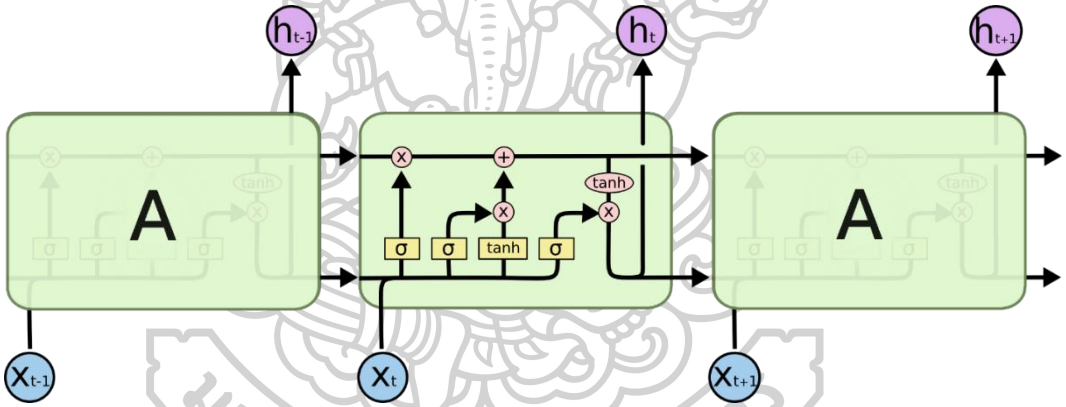
ภาพที่ 4 โครงข่ายประสาทเทียมแบบ RNN แบบคลี่ [10]

หลักการทำงานของ RNN คือ ส่งอินพุตเข้าไปในแต่ละโหนด ซึ่งในแต่ละโหนดนั้นก็จะมีการคำนวณบางอย่างอยู่ด้านใน เพื่อส่งเป็นคำตอบออกมาจากโหนดนั้น ๆ ค่าที่ได้จะถูกส่งต่อไปยังโหนดต่อไป เพื่อใช้ในการคำนวณรอบต่อไป RNN ส่งต่อข้อมูลที่ได้จากการคำนวณให้โหนดถัดไปโดยไม่สนใจจะจดจำค่านั้น ๆ เอา ทำให้เกิดปัญหาในการทำการคำนวณค่าความผิดพลาดย้อนหลังหลัง (back-propagation, BP) และทำให้เกิดปัญหา Vanishing Gradient Problem ปัญหาที่ตามมาคือ RNN จะเริ่มทำงานได้แย่ลง เมื่อระยะของข้อมูลนั้นมีความยาวมาก ๆ เพราะการจดจำของ RNN นั้นทำได้ดีแค่ในระยะสั้นเท่านั้น RNN ถูกนำไปใช้งานต่าง ๆ เช่น video (sequence of images) หรือ text (sequence of words)

ข้อดีของ RNN คือ มีการใช้ข้อมูลก่อนหน้าในการทำนายสิ่งที่อาจจะเกิดขึ้นในอนาคต (อะไรที่เคยเกิดขึ้นในอดีตย่อมส่งผลต่อเหตุการณ์ที่จะเกิดขึ้นในอนาคตด้วย) ข้อเสียของ RNN คือ สามารถดูย้อนกลับได้แค่เพียงในช่วงระยะเวลาสั้น ๆ เท่านั้น ซึ่งปัญหาหลักของ RNN เกิดมาจากค่า gradient ที่เริ่มน้อยลงในข้อมูลที่มีความยาวมากขึ้น จนแทบจะไม่สามารถเห็นความเปลี่ยนแปลงของ gradient ได้เลย ซึ่งปัญหานี้ถูกเรียกว่า Vanishing Gradient Problem ดังนั้นเพื่อแก้ปัญหาดังกล่าวจึงทำให้เกิดเทคนิค LSTM



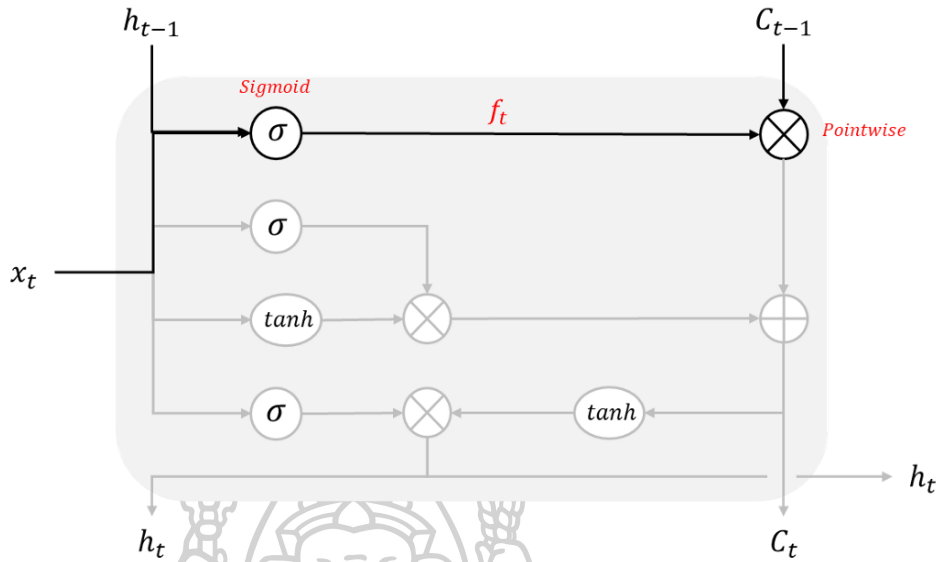
ภาพที่ 5 โมดูลที่ทำซ้ำใน RNN มาตรฐานมีเลเยอร์เดียว [10]



ภาพที่ 6 โมดูลที่ทำซ้ำใน RNN มาตรฐานมีสี่เลเยอร์ [10]

LSTM เป็นโครงข่ายประสาทเทียมประเภท RNN รูปแบบหนึ่ง โดยมีหลักการทำงานคือ สามารถเก็บ “สถานะ” หรือข้อมูลของแต่ละโหนดเอาไว้ เพื่อเวลาที่ย้อนกลับไปดูจะได้ทราบถึงที่มาของข้อมูลค่าดังกล่าวว่าของเดิมเป็นค่าอะไร และจุดเด่นของเทคนิค LSTM คือ ฟังก์ชั่นพิเศษที่มีหน้าที่เสมือน “ประตู (Gate)” ที่คอยควบคุมข้อมูลที่จะเข้ามาในแต่ละโหนด ประกอบด้วย Forget gate layer, Input gate layer และ Output gate layer [11] รายละเอียดดังภาพที่ 7 ภาพที่ 8 และภาพที่ 9 ดังนี้

1) Forget gate layer



ภาพที่ 7 โครงสร้าง Forget gate layer [12]

Forget gate คือ ประตูที่ใช้เป็นตัวกำหนดว่าข้อมูลที่เข้ามานั้นควรที่จะได้ออกไปทั้งหมดหรือไม่ ซึ่งข้อมูลที่จำเป็นต้องใช้ในการตัดสินใจว่าจะเก็บข้อมูลนี้ไว้หรือไม่นั้นก็มาจากข้อมูลขาเข้า ณ โหนด นั้น ๆ รวมกับผลลัพธ์ที่ถูกคำนวณมาจากโหนดก่อนหน้า ผ่านเข้า sigmoid function และนำไปคูณกับสถานะของโหนดก่อนหน้าโดยใช้สูตร Pointwise Multiplication ในการคำนวณ ดังสมการที่ (4)

$$f_t = \sigma(w_f \cdot [h_{t-1}, x_t] + b_f) \quad \text{สมการที่ (4)}$$

จากสมการ

f_t = Forget gate

σ = ฟังก์ชั่น sigmoid

w_f = ค่าน้ำหนักของ matrices

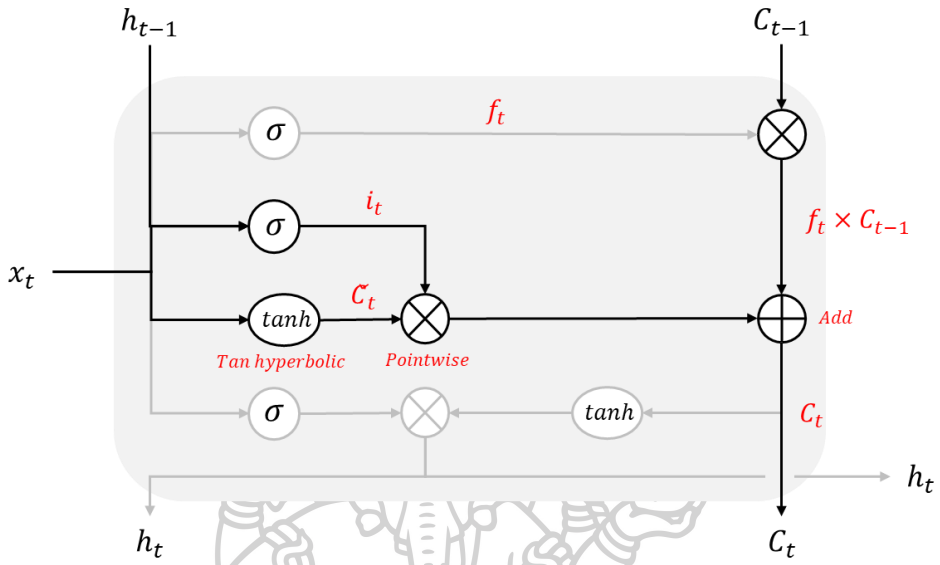
h_{t-1} = ค่า output ของ cell state ก่อนหน้า (ที่ timestamp t-1)

x_t = ค่า input ที่เข้ามาใน cell state ณ เวลา t

b_f = ค่า bias

ผลลัพธ์ที่ได้ Forget gate layer จะอยู่ระหว่างค่า 0 และ 1 ซึ่งค่า 0 หมายถึงให้ลบค่า cell state เดิมออก ในขณะที่ 1 หมายถึง ให้เก็บค่า cell state นี้ต่อไป

2) Input gate layer



ภาพที่ 8 โครงสร้าง Input gate layer [12]

Input Gate คือ ประตูที่ใช้เพื่อเปิดรับข้อมูลที่เข้ามาใหม่ เพื่อให้บันทึก หรือ ‘write’ ลงไปในแต่ละโหนด โดยมีการทำงานแบ่งออกเป็น 2 ส่วน

- ส่วนแรก คือ ถ้าต้องการ update cell state เมื่อทำการรับข้อมูล input เข้ามาแล้วฟังก์ชั่น sigmoid ที่เป็นตัวควบคุมจะเรียกใช้ input gate เพื่อเลือกว่าจะให้ update cell state หรือไม่ ดังสมการที่ (5)

$$i_t = \sigma(w_i \cdot [h_{t-1}, x_t] + b_i) \qquad \text{สมการที่ (5)}$$

จากสมการ

i_t = Input gate

σ = ฟังก์ชั่น sigmoid

w_i = ค่าน้ำหนักของ matrices

h_{t-1} = ค่า output ของ cell state ก่อนหน้า (ที่ timestamp t-1)

x_t = ค่า input ที่เข้ามาใน cell state ณ เวลา t

b_i = ค่า bias

- ส่วนที่สอง ถ้า input gate เลือกที่จะทำการ update cell state ฟังก์ชั่น tanh ก็จะทำการสร้าง candidate values (C_t) ขึ้นมาใน state ดังสมการที่ (6)

$$C_t = tanh(w_c \cdot [h_{t-1}, x_t] + b_c) \qquad \text{สมการที่ (6)}$$

จากสมการ

C_t = candidate ของ cell state ที่เวลา t

$tanh$ = ฟังก์ชั่น tanh

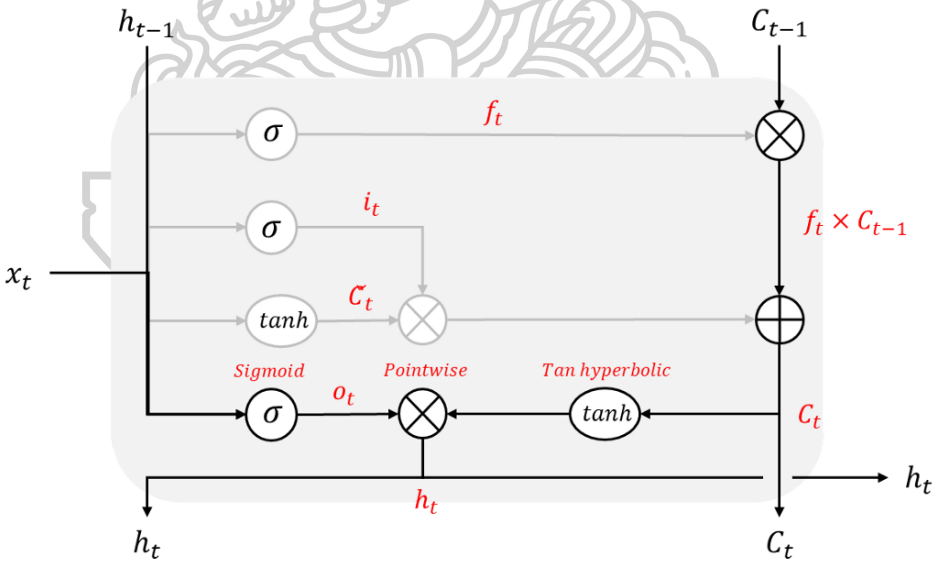
W_c = ค่าน้ำหนักของ matrices

h_{t-1} = ค่า output ของ cell state ก่อนหน้า (ที่ timestamp t-1)

x_t = ค่า input ที่เข้ามาใน cell state ณ เวลา t

b_c = ค่า bias

3) Output gate layer



ภาพที่ 9 โครงสร้าง Output gate layer [12]

Output gat เป็น Gate ที่มีหน้าที่เตรียมทำการรส่งออกข้อมูล (output data) โดยข้อมูล output นั้นจะดูจาก cell state ที่ผ่านกระบวนการคำนวณต่าง ๆ แล้ว โดยฟังก์ชั่น sigmoid จะเป็นตัวเลือกว่าข้อมูลส่วนไหนใน cell state ที่จะถูก output จากนั้นก็จะนำค่า cell state เข้า

ฟังก์ชั่น tanh (เพื่อหาว่าจะได้ค่าออกมาเป็น 1 หรือ -1) แล้วนำค่าที่ได้จากฟังก์ชั่น tanh มาทำการคำนวณกับค่า output ที่ได้จาก sigmoid gate จากนั้นก็จะได้ค่า output ที่ต้องการดังสมการที่ (7)

$$O_t = \sigma(w_o \cdot [h_{t-1}, x_t] + b_o) \qquad \text{สมการที่ (7)}$$

จากสมการ

O_t = candidate ของ cell state ที่เวลา t

σ = ฟังก์ชั่น sigmoid

w_o = ค่าน้ำหนักของ matrices

h_{t-1} = ค่า output ของ cell state ก่อนหน้า (ที่ timestamp t-1)

x_t = ค่า input ที่เข้ามาใน cell state ณ เวลา t

b_o = ค่า bias

ซึ่งค่า output ที่ได้ออกมานั้นจะถูกแบ่งออกเป็น 2 ส่วน คือ ค่า output ที่ได้จากโหนดนั้น ๆ กับค่า output ที่จะถูกส่งไปเป็นข้อมูล input ของโหนดถัดไป

2.1.4 เทคโนโลยีสำหรับข้อมูลขนาดใหญ่ (Big data)

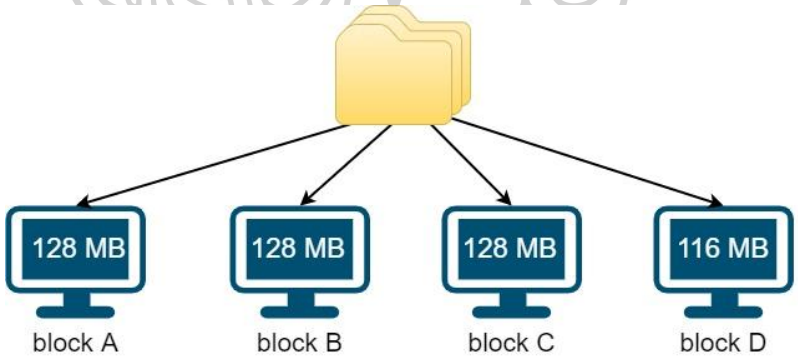
ในอดีตการจัดเก็บข้อมูลนั้นเก็บในรูปแบบของไฟล์ประเภทต่าง ๆ เช่น ไฟล์ข้อความ หรือไฟล์ประเภท Excel แต่การเก็บข้อมูลเป็นไฟล์ในลักษณะนี้จะมีความยุ่งยากหลายอย่าง เช่น การเก็บไฟล์ซ้ำซ้อนกันระหว่างผู้ใช้งาน หรือ การค้นหาข้อมูลจากหลายไฟล์ ดังนั้นเพื่อแก้ปัญหาความยุ่งยากเหล่านี้องค์กรต่าง ๆ จึงหันมาเก็บข้อมูลในฐานข้อมูล (Database) หรือฐานข้อมูลเชิงสัมพันธ์ (Relational database) การเก็บข้อมูลในฐานข้อมูลนี้จะมีระบบจัดการฐานข้อมูล (Database management system) หรือ DBMS ซึ่งเป็นซอฟต์แวร์ที่ช่วยในการจัดการและสืบค้นข้อมูล ทำให้การค้นหาข้อมูลสะดวกมากยิ่งขึ้น แต่เมื่อต้องมีการนำข้อมูลจำนวนมากซึ่งมาจากหลากหลายฐานข้อมูล มาทำการประมวลผลให้เป็นสารสนเทศเพื่อนำไปใช้ในการตัดสินใจต่อไป พบว่าค่อนข้างใช้เวลานานเนื่องจากต้องทำการปรับเปลี่ยนข้อมูลให้มีลักษณะเดียวกัน และเกิดความยุ่งยากในการดำเนินการ ในปัจจุบันเมื่อข้อมูลมีปริมาณมากขึ้นและข้อมูลมีความหลากหลายมากขึ้นจึงทำให้การเก็บข้อมูลในฐานข้อมูลหรือคลังข้อมูลแบบเดิมไม่ได้มีประสิทธิภาพเท่าที่ควร ประกอบกับปัจจุบันข้อมูลมีจำนวนเพิ่มขึ้นและเปลี่ยนแปลงอย่างรวดเร็ว ดังนั้นจำเป็นอย่างยิ่งที่ต้องอาศัยเทคโนโลยีที่สามารถจัดเก็บและประมวลผลกับข้อมูลที่มีคุณลักษณะดังกล่าวได้

1) Apache Hadoop

Apache Hadoop เป็น software framework ที่สามารถจัดการกับชุดข้อมูลขนาดใหญ่ (large dataset) และมีการประมวลผลแบบกระจาย (distributed processing) บนเครื่องคอมพิวเตอร์ที่เชื่อมต่อกันเป็นกลุ่ม (cluster) โดย Hadoop มีหน้าที่หลัก ๆ อยู่สองหน้าที่ คือ 1) บริหารจัดการหน่วยเก็บข้อมูล (storage management) และ 2) บริหารจัดการทรัพยากรสำหรับการประมวลผล (computing resource management)

เมื่อการเก็บข้อมูลขนาดใหญ่ในเครื่องเดียวนั้นแทบเป็นไปไม่ได้ การกระจายข้อมูลเข้าไปเก็บไว้เป็นบล็อก (Block) ในหลาย ๆ เครื่องและแบ่งไปตามขนาดจึงเป็นเรื่องที่สามารถทำได้ Hadoop Distributed File System (HDFS) ทำหน้าที่เป็นส่วนของระบบจัดการไฟล์ (File System) เพื่อให้ข้อมูลกระจายไปอยู่ตามเครื่องคอมพิวเตอร์หรือโหนดในระบบได้อย่างถูกต้อง และเพื่อป้องกันการเกิดปัญหาข้อผิดพลาดที่ทำให้ข้อมูลเสียหาย

HDFS จะทำการกระจายข้อมูลออกไปเก็บไว้ในโหนดต่าง ๆ เช่น ข้อมูลมีขนาด 500 MB ใน HDFS จะแบ่งข้อมูลออกเป็นก้อน ๆ เรียก บล็อก (Block) ที่มีขนาด 128 MB ซึ่งเป็นขนาดที่กำหนดไว้โดยระบบปฏิบัติการ HDFS แต่ก็สามารถเปลี่ยนแปลงค่าได้ตามต้องการ ยกตัวอย่างบล็อกที่มีขนาด 128 MB ดังนั้นข้อมูลขนาด 500 MB จะถูกแบ่งเป็นบล็อกเท่า ๆ กัน 3 บล็อก แล้วเก็บที่เหลือขนาด 116 MB ไว้ในบล็อกที่ 4 แสดงดังภาพที่ 10



ภาพที่ 10 การกระจายข้อมูลไปเก็บไว้ในแต่ละบล็อก

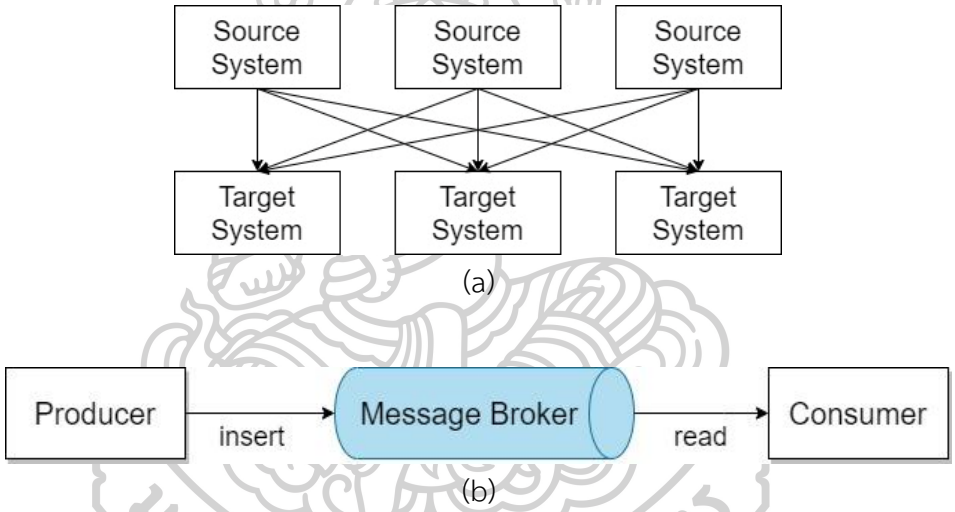
HDFS ประกอบไปด้วย 2 ส่วนย่อย คือ

- NameNode (Master Node) คือ ส่วนเก็บ Metadata หรือข้อมูลที่อธิบายรายละเอียดเกี่ยวกับข้อมูล เช่น จำนวนของบล็อก เครื่องที่จัดเก็บ ข้อมูลถูกเก็บไว้ที่ไหน

- DataNode (Slave Node) คือ ส่วนที่เก็บข้อมูลจริง ๆ ของ HDFS ทำหน้าที่อ่าน เขียนตามคำสั่ง และทำหน้าที่ทำสำเนาข้อมูล (Replica) ของแต่ละบล็อกเก็บไว้ในอีก DataNode เพื่อป้องกันการสูญหายของข้อมูล (fault-tolerant)

2) Apache Kafka

Apache Kafka เป็นแพลตฟอร์มสตรีมมิ่งแบบกระจาย (distributed streaming platform) หรือ Message Broker ที่สามารถจัดการกับข้อมูลที่ถูกสร้างขึ้นอย่างต่อเนื่องจากแหล่งข้อมูลจำนวนมาก (Streaming Data) ซึ่งจะต้องผ่านการประมวลผลตามลำดับทีละ record แสดงดังภาพที่ 11



ภาพที่ 11 Apache Kafka แพลตฟอร์มสตรีมมิ่งแบบกระจาย

Apache Kafka สามารถนำมาสร้างเป็นระบบ real-time streaming data pipeline ที่มีความน่าเชื่อถือในการรับข้อมูลระหว่างระบบ (system) ได้ และถูกนำไปประยุกต์กับระบบส่งต่อข้อความ (messaging System) เครื่องมือบันทึกกิจกรรม (activity Tracking) รวบรวมเก็บ Log (log aggregation) และการประมวลผลแบบต่อเนื่องของข้อมูล (stream processing) เช่นเดียวกับระบบแนะนำบทความข่าว (recommending news articles)

จุดเด่นของ Apache Kafka

- จัดการกับ real-time data feed ได้อย่างมีประสิทธิภาพ
- Publish และ Subscribe กับ Stream ของ records ข้อมูลได้เหมือนกับ message queue หรือ enterprise messaging system

- เก็บ Stream ของ records ข้อมูลในแนวทางที่ทนต่อความผิดพลาด (fault-tolerance durable)
- ดำเนินการกับ Stream ของ records ข้อมูลทันที่ที่เกิดขึ้น (real-time processing)
- มีการกระจาย (distributed) การเก็บข้อมูลใน clusters
- มีความยืดหยุ่น (resilient architecture) เช่น มีการทำสำเนาข้อมูลซ้ำ (replication)
- มีการทนต่อความเสียหาย (fault tolerant)
- มีความสามารถในการขยายเชิงขนาน หรือ เพิ่มเครื่อง (node) ใน cluster ได้ (horizontal scalability)
- มีประสิทธิภาพด้านความเร็ว (latency น้อยกว่า 10ms)
- ระบบใหญ่ ๆ ที่ใช้ Apache Kafka เช่น Linkedin, Netflix, AirBnB, Yahoo, Wallmart

2.2 งานวิจัยที่เกี่ยวข้อง

รายวิชาออนไลน์แบบเปิดสู่มวลชน (MOOCs) ได้รับความนิยมแพร่หลายไปทั่วโลกทำให้มีจำนวนรายวิชาหลายพันรายวิชาและมีผู้เรียนหลายล้านคนจากทั่วโลก [13] นักวิจัยเริ่มสนใจที่จะศึกษาวิจัยเกี่ยวกับการจัดการเรียนการสอนผ่านระบบรายวิชาออนไลน์แบบเปิดสู่มวลชน (MOOCs) ด้วยปริมาณของรายวิชาและผู้เรียนที่เพิ่มขึ้นอย่างรวดเร็วเป็นปริมาณมหาศาลดังภาพที่ 1 และตารางที่ 1 การวิเคราะห์ข้อมูลจำนวนมากนั้นต้องใช้เทคนิควิธีที่เหมาะสม Du และคณะ [14] นำเสนองานวิจัยเกี่ยวกับการวิเคราะห์ข้อมูลในรายวิชาออนไลน์แบบเปิดสู่มวลชน (MOOCs) โดยใช้วิธีการทางเทคนิคเหมืองข้อมูล (Data mining) ร่วมกับการจัดการข้อมูลขนาดใหญ่ (Big data) พบว่าสามารถใช้วิธีการทางเหมืองข้อมูลร่วมกับการจัดการข้อมูลขนาดใหญ่ในการวิเคราะห์เพื่อหารายวิชาที่เป็นที่นิยมโดยให้แสดงผลในรูปแบบกราฟฟิกแยกตามที่อยู่ของผู้เรียนได้

Yang และคณะ [15] สามารถคาดการณ์ผลการเรียนจากพฤติกรรมการเรียนรู้ของผู้เรียนโดยใช้โครงข่ายประสาทเทียมอนุกรมเวลา (Time series neural networks) แบบโครงข่ายประสาทแบบวนซ้ำ (Recurrent neural networks) ประมวลผลกับข้อมูลการทำแบบทดสอบ (Assessment performance) และข้อมูลพฤติกรรมการดูวิดีโอ (Video watching behavior) ของผู้เรียน โดยสร้างตัวแบบจำนวนสองตัวแบบ คือ การเรียนรู้จากการทำแบบทดสอบเพียงอย่างเดียว (Learning from quiz features only, FTSNN) และการเรียนรู้จากข้อมูลคุณลักษณะของพฤติกรรมและข้อมูลผลการ

ทำแบบทดสอบ (Behavioral features and quiz features, IFTSNN) พบว่าตัวแบบที่ได้ออกแบบทั้งสองตัวแบบนั้นให้ประสิทธิภาพที่ดีกว่าค่าเฉลี่ยของประสิทธิภาพในอดีตที่ผ่านมาโดยเฉลี่ยมากกว่า 60% และมีค่าการถดถอย (Lasso regression baseline) มากกว่า 15% ตัวแบบดังกล่าวแสดงให้เห็นถึงความเป็นไปได้ในการคาดการณ์ผลการเรียนในรายวิชาออนไลน์แบบเปิดสู่มวลชน (MOOCs) ได้ และยังพบว่าคุณลักษณะของข้อมูลเหตุการณ์บนวิดีโอทั้งแปดคุณลักษณะนั้นมีความสำคัญเท่าเทียมกันไม่ได้มีคุณลักษณะใดคุณลักษณะหนึ่งที่เด่นชัดส่งผลต่อประสิทธิภาพของตัวแบบ ดังนั้นตัวแบบของผู้วิจัยสามารถใช้เพื่อตรวจจับผู้เรียนที่มีคะแนนผลการทดสอบต่ำตั้งแต่ในช่วงต้นของการเรียนได้ ทำให้ผู้สอนเข้าเปลี่ยนแปลงการเรียนการสอนได้อย่างอัตโนมัติเพื่อกำหนดเป้าหมายของผู้เรียนเหล่านี้ใหม่ นอกจากนี้พบว่าตัวแบบ FTSNN มีแนวโน้มที่จะดีกว่าตัวแบบ IFTSNN เล็กน้อยเมื่อผู้เรียนมีการคำตอบคำถามบางส่วน

ข้อมูลการคลิกในวิดีโอของผู้เรียนสามารถสะท้อนศักยภาพของผู้เรียนและสะท้อนคุณภาพของสื่อการสอนประเภทวิดีโอได้ เช่น การวิเคราะห์จุดสูงสุดของการคลิกบนวิดีโอจากรายวิชาออนไลน์แบบเปิดสู่มวลชน (PeakVizor: Visual analytics of peaks in video clickstreams from Massive Open Online Courses) [16] PeakVizor เป็นวิชวลไลเซชัน (Visualization) ที่ช่วยให้อาจารย์ประจำรายวิชาและผู้เชี่ยวชาญด้านการศึกษาสามารถวิเคราะห์กลุ่มวิดีโอที่มีจำนวนการคลิกจำนวนมากได้นำไปสู่การค้นพบใหม่ๆ เกี่ยวกับพฤติกรรมการเรียนรู้ในระบบจัดการรายวิชา (MOOC Platform) รายวิชาออนไลน์แบบเปิดสู่มวลชน (MOOCs) ทางทีมวิจัยสามารถออกแบบและสร้าง PeakVizor ให้แก้ไขปัญหาหรืออุปสรรคต่าง ๆ ที่การแสดงข้อมูลวิดีโอสตรีมจากรายวิชาออนไลน์แบบเปิดสู่มวลชน (Visualizing Video Clickstream Data from Massive Open Online Courses, VisMOOC) ไม่สามารถทำได้ และยังเพิ่มประสิทธิภาพด้านการคัดกรองข้อมูลการคลิกของผู้เรียนตามเกณฑ์ต่าง ๆ สามารถวิเคราะห์กราฟได้หลายกราฟในเวลาเดียวกัน และใช้รูปแบบกราฟต่าง ๆ เพื่ออธิบายผลการวิเคราะห์ได้อย่างละเอียดและเข้าใจมากยิ่งขึ้น Fu และคณะ [17] นำเสนอวิชวลไลเซชันชื่อ iForum โดยใช้ SeededLDA ในการแยกแยะหัวข้อจากกระดานสนทนา (Forums) ร่วมกับเทคนิคการประมวลผลภาษาธรรมชาติ (Natural language processing techniques) พบว่าในการวิเคราะห์ของ iForum บางส่วนจำเป็นต้องปรับปรุงเพิ่มเติม เช่น การวิเคราะห์ความเชื่อมั่น การใช้กราฟในรูปแบบที่ให้ผู้ใช้งานสามารถเข้าใจได้ง่ายขึ้น และการลดความซ้ำซ้อนของเส้นที่แสดงผลใน iForum

จากที่รายวิชาออนไลน์แบบเปิดสู่มวลชน (MOOCs) ได้รับความนิยมเพิ่มขึ้นเรื่อย ๆ มีนักเรียนที่ลงทะเบียนเรียนเป็นจำนวนมากแต่มีผู้สอนที่ค่อนข้างน้อย ดังนั้นการมีส่วนร่วมของผู้สอนในขั้นตอนการเรียนการสอนจะต้องจำกัดเฉพาะงานที่สำคัญที่สุดเท่านั้น จึงส่งผลกระทบโดยตรงกับงานด้านการประเมินที่มุ่งเน้นการตรวจสอบความสามารถของผู้เรียน Capuano และคณะ [18] เสนอ

แนวความคิดการประเมินผลโดยเพื่อนร่วมชั้นเรียนประเมินกันเองโดยอาศัยการตัดสินใจแบบกลุ่มเข้ามาร่วม ในงานวิจัยฉบับนี้จะค้นหาเกี่ยวกับการประเมินตามลำดับขั้นแบบฟัซซี่ (Fuzzy Ordinal Peer Assessment, FOPA) การแบบประเมินผลโดยเพื่อนในรูปแบบใหม่ตามทฤษฎีฟัซซี่เซต (Fuzzy sets) และการใช้เทคนิคการตัดสินใจกลุ่ม (Group Decision Making, GDM)

จากความหลากหลายรายวิชาที่เกิดขึ้นในรายวิชาออนไลน์แบบเปิดสู่มวลชน (MOOCs) ทำให้อาจเป็นเรื่องยากสำหรับผู้เรียนในการเลือกรายวิชา Zhang และคณะ [19] พัฒนางานวิจัยเรื่อง “MCRS: A Course Recommendation System for MOOCs” เพื่อให้สามารถแนะนำรายวิชาที่เหมาะสมแก่ผู้เรียน โดยหาความสัมพันธ์ของข้อมูลผู้เรียนในอดีตเพื่อใช้ในการแนะนำรายวิชาให้กับผู้เรียนใหม่ต่อไปด้วยวิธี Apriori algorithm และจากการเปรียบเทียบการประมวลผลข้อมูลบน Spark พบว่า Apriori algorithm บน Spark มีประสิทธิภาพที่มากกว่า Apriori algorithm บน Hadoop งานวิจัย “MOOCRC: A Highly Accurate Resource Recommendation Model for Use in MOOC Environments” [20] ถูกนำเสนอสำหรับให้คำแนะนำรายวิชาในสภาพแวดล้อม MOOCs โดยใช้เทคนิค Deep belief networks (DBNs) จากผลการวิจัยพบว่าเมื่อนำ MOOCRC ไปเทียบกับเทคนิคห้าอันดับแรกของระบบให้คำแนะนำ คือ Content-based, KNN, Singular value decomposition (SVD), Restricted boltzmann machines (RBM) พบว่า MOOCRC มีค่า Root mean square error (RMSE) ที่น้อยกว่า 67.48% ซึ่งถือว่าดีที่สุดเมื่อเปรียบเทียบวิธีการที่กล่าวมา

การแนะนำรายวิชาที่เหมาะสมกับผู้เรียนโดยเฉพาะเป็นรายบุคคลเป็นงานที่ท้าทายและน่าสนใจมาก Wang และคณะ [5] ปรับปรุงวิธีสำหรับการแนะนำรายวิชาในรายวิชาออนไลน์แบบเปิดสู่มวลชน (MOOCs) ให้กับผู้เรียนเฉพาะรายบุคคล โดยเสนอขั้นตอนวิธีการให้น้ำหนักแบบหลายแอตทริบิวต์ (Multi-attribute weight algorithm, MAWA) ซึ่งประกอบด้วยน้ำหนักของ แอตทริบิวต์ (Weight of the attributes) และน้ำหนักประจำของแอตทริบิวต์ (Weight of attribute values) กับ Collaborative filtering จากการวิจัยพบว่า MAWA มีประสิทธิภาพสูงกว่าวิธี Collaborative filtering และขั้นตอนวิธี Uncertain neighbors ’collaborative filtering recommendation สรุปข้อมูลงานวิจัยเกี่ยวข้องกับรายวิชาออนไลน์แบบเปิดสู่มวลชน (MOOCs) ดังตารางที่ 4

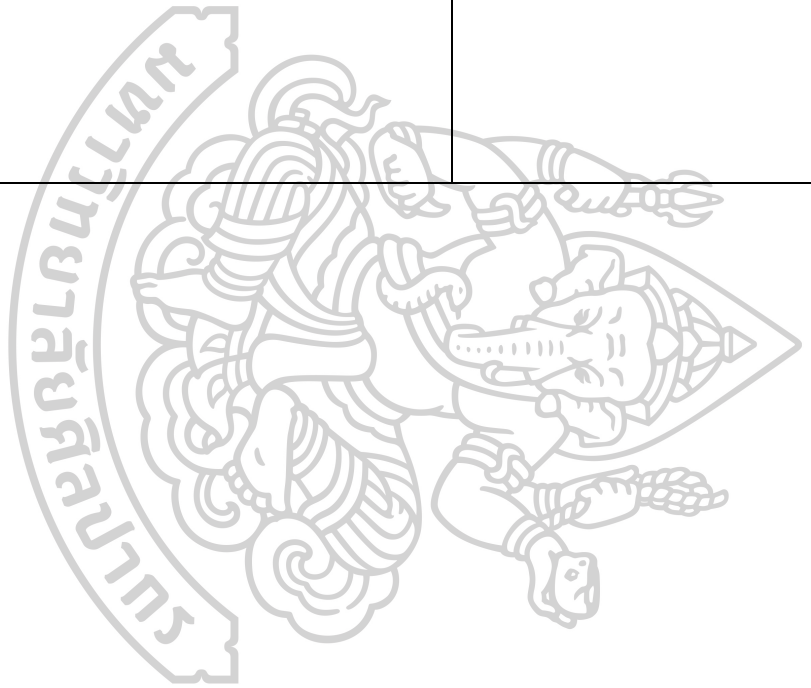
ตารางที่ 4 สรุปข้อมูลงานวิจัยเกี่ยวข้องกับรายวิชาออนไลน์แบบเปิดสู่มวลชน (MOOCs)

คณะผู้วิจัย/ ชื่องานวิจัย	วิธีการวิจัย / ตัวแบบ	ชุดข้อมูล	ผลลัพธ์ที่ได้
[14] Du และคณะ Research on the big data system of massive open online course.	- ใช้เทคนิคเหมืองข้อมูล (Data mining) เช่น Clustering, Classification ในการประมวลผลข้อมูล - ใช้ Hadoop, Spark, Storm ในการประมวลผลข้อมูลขนาดใหญ่	CNMOOC platform	- สามารถใช้เทคนิคเหมืองข้อมูล (Data mining) ประมวลผลข้อมูลใน MOOCs ได้
[15] Yang และคณะ Behavior-Based Grade Prediction for MOOCs Via Time Series Neural Networks.	- ใช้โครงข่ายประสาทแบบวนซ้ำ (Recurrent neural networks) ในการประมวลผลข้อมูล - พัฒนาตัวแบบ (Model) 2 ตัวแบบ คือ ตัวแบบการเรียนรู้จากการทำแบบทดสอบเพียงอย่างเดียว (Learning from quiz features only, FTSNN) และตัวแบบการเรียนรู้จากข้อมูลคุณลักษณะของพฤติกรรมและข้อมูลผลการทำแบบทดสอบ (Behavioral features and quiz features, IFTSNN)	Coursera platform	- ตัวแบบที่ได้ออกแบบทั้งสองตัวแบบให้ประสิทธิภาพที่ดีกว่าค่าเฉลี่ยของประสิทธิภาพในอดีตที่ผ่านมา โดยเฉลี่ยมากกว่า 60% และมีค่าการถดถอย (Lasso regression) มากกว่า 15%
[16] Chen และคณะ PeakVizor: Visual	ใช้หลักการทางสถิติ (Statistics)	edX และ Coursera platform	- PeakVizor เป็นเครื่องมือให้ผู้สอนและผู้เชี่ยวชาญด้านการศึกษาใช้ในการ

คณะผู้วิจัย/ ชื่องานวิจัย	วิธีการวิจัย / ตัวแบบ	ชุดข้อมูล	ผลลัพธ์ที่ได้
Analytics of Peaks in Video Clickstreams from Massive Open Online Courses.			วิเคราะห์พฤติกรรมการเรียนรู้ของผู้เรียนได้
[17] Fu และคณะ Visual Analysis of MOOC Forums with iForum.	ใช้ SeededLDA ในการแยกแยะหัวข้อจากกระดานสนทนา (Forums) ร่วมกับเทคนิคการประมวลผลภาษาธรรมชาติ (Natural language processing techniques)	edX platform	- ช่วยให้ค้นพบและทำความเข้าใจรูปแบบไดนามิกของฟอรัม ใน MOOC ได้อย่างมีประสิทธิภาพ - การวิเคราะห์ของ iForum บางส่วนจำเป็นต้องปรับปรุงเพิ่มเติม
[18] Capuano และคณะ A Fuzzy Group Decision Making Model for Ordinal Peer Assessment.	ใช้ Fuzzy Group Decision Making	ทดลองกับนักเรียน 100 คน	- ตัวแบบที่ได้มีประสิทธิภาพที่ดีกว่าวิธี CPA, MAL, BT

คณะผู้วิจัย/ ชื่องานวิจัย	วิธีการวิจัย / ตัวแบบ	ชุดข้อมูล	ผลลัพธ์ที่ได้
[19] Zhang และคณะ MCRS: A Course Recommendation System for MOOCs.	- ใช้เทคนิคเหมืองข้อมูล (Data mining) คือ Association Rule โดยใช้ Apriori algorithm - ใช้ Sqoop, Spark, Hadoop ในการประมวลผลข้อมูลขนาดใหญ่ (Big Data)	- ชื่อชุดข้อมูล HMXPC13 จาก Harvard and MIT - ชื่อชุดข้อมูล T10I4D100K และ T25I10D10K จาก IBM	- Apriori algorithm บน Spark มีประสิทธิภาพมากกว่าอัลกอริทึม Apriori แบบดั้งเดิมและแบบ Apriori algorithm บน Hadoop - MCRS ช่วยเพิ่มระดับการจัดเก็บข้อมูลและเพิ่มประสิทธิภาพในการคำนวณให้ได้ดียิ่งขึ้น สามารถนำไปประยุกต์ใช้กับระบบแนะนำรายวิชาใน MOOCs ได้
[5] Wang และคณะ An improved algorithm for personalized recommendation on MOOCs.	- ใช้เทคนิค Multi-attribute weight algorithm ประกอบด้วย Weight of the attributes และ Weight of attribute values	- MOOC College	- มีประสิทธิภาพสูงกว่าวิธี Collaborative Filtering และขั้นตอนวิธี Uncertain neighbors 'collaborative
[20] Zhang และคณะ MOOCRC: A Highly Accurate Resource	- ใช้เทคนิควิธี Deep belief networks (DBNs)	starC MOOC platform	- MOOCRC ให้ประสิทธิภาพที่ดีกว่าวิธีแบบเดิม คือ Content-Based, KNN, Singular Value Decomposition

คณะผู้วิจัย/ ชื่องานวิจัย	วิธีการวิจัย / ตัวแบบ	ชุดข้อมูล	ผลลัพธ์ที่ได้
Recommendation Model for Use in MOOC Environments.			(SVD), Restricted Boltzmann Machines (RBM)



บทที่ 3
วิธีดำเนินการวิจัย

การศึกษาและวิจัยในครั้งนี้มีวัตถุประสงค์เพื่อศึกษาขั้นตอนวิธีที่เหมาะสมสำหรับนำไปสร้างตัวแบบระบบแนะนำรายวิชาในรายวิชาออนไลน์แบบเปิดสู่มวลชน (MOOCs) ด้วยเทคนิคข้อมูลขนาดใหญ่ (Big Data) เพื่อนำไปปรับปรุงเทคนิคระบบแนะนำรายวิชาสำหรับข้อมูลขนาดใหญ่ (Big Data) ให้เหมาะสมและมีประสิทธิภาพ และนำไปพัฒนาเป็นตัวแบบระบบแนะนำรายวิชาในรายวิชาออนไลน์แบบเปิดสู่มวลชน (MOOCs) เพื่อแนะนำรายวิชาให้กับผู้เรียนต่อไป ก่อให้เกิดประโยชน์แก่วงการด้านการศึกษาอย่างมาก เช่น ผู้เรียนสามารถเข้าถึงรายวิชาได้ตรงกับความต้องการ ผู้เรียนได้รับคำแนะนำหรือกระตุ้นให้เกิดการเรียนรู้ ผู้สร้างรายวิชาสามารถที่จะสร้างรายวิชาซึ่งตรงกับความต้องการของผู้เรียนได้มากขึ้นตรงกับความต้องการของคนรุ่นใหม่ที่ต้องการเรียนรู้สิ่งต่าง ๆ ที่เฉพาะเจาะจง งานวิจัยฉบับนี้มีขั้นตอนการดำเนินการวิจัยดังนี้

3.1 การกำหนดปัญหา การตั้งโจทย์ สมมุติฐานงานวิจัย

3.2 เครื่องมือและอุปกรณ์ที่ใช้ในการพัฒนางานวิจัย

3.3 ข้อมูลที่ใช้ในการวิจัย (Data Overviewing)

3.4 การสำรวจข้อมูลด้วย Exploratory Data Analysis (EDA)

3.5 การกำหนดตัวแบบ (Model Creating)

3.6 การกำหนดสถาปัตยกรรมระบบแนะนำรายวิชา

3.1 การกำหนดปัญหา การตั้งโจทย์ สมมุติฐานงานวิจัย

ในโลกอินเทอร์เน็ตมีข้อมูลมากมายมหาศาล การที่จะค้นหาข้อมูลจำนวนมากนี้อาจต้องใช้เวลาจึงจะพบข้อมูลที่ต้องการ ดังนั้นหากมีระบบแนะนำ (Recommendation system, RS) ก็จะช่วยอำนวยความสะดวกให้การค้นหาแคบลงและสะดวกรวดเร็วมากขึ้น ซึ่งรายวิชาออนไลน์แบบเปิดสู่มวลชน (MOOCs) เป็นแหล่งเรียนรู้ออนไลน์ขนาดใหญ่ที่มีรายวิชาหลายพันรายวิชาและมีผู้เรียนหลายล้านคน

ความรู้ในโลกอินเทอร์เน็ตเป็นความรู้ที่มีการพัฒนาอยู่เสมอ มีการอัพเดทข้อมูลที่ทันสมัย ดังนั้นหากมีระบบแนะนำรายวิชาในรายวิชาออนไลน์แบบเปิดสู่มวลชน (MOOCs) ก็จะทำให้เป็นการสร้างโอกาสทางการเรียนรู้อย่างที่ไม่มีที่สิ้นสุด

ปัญหาที่พบ

- ระบบคำแนะนำแบบดั้งเดิม (Traditional recommendation techniques) ประสบปัญหาในการขยายขนาด (scale out) ส่งผลให้ระบบคำแนะนำมีคุณภาพไม่ดีเท่าที่ควร ไม่สามารถรองรับกับข้อมูลที่มีขนาดใหญ่ โดยเฉพาะอย่างยิ่งเมื่อมีตัวเลือกให้เลือกมากมาย กลายเป็นเรื่องยากสำหรับผู้คนที่จะค้นหาสิ่งที่ต้องการหรือสิ่งที่พวกเขาสนใจ

- ระบบแนะนำรายวิชามักจะอยู่ในสภาพแวดล้อมแบบปิดใช้งานแค่ในส่วนของตนเองเท่านั้น ทำให้ไม่สามารถนำมาประยุกต์ใช้ในวงกว้างได้

- ระบบแนะนำรายวิชาใน MOOCs ยังคงใช้เทคโนโลยีในการจัดเก็บข้อมูลและประมวลผลข้อมูลแบบเดิมทำให้ไม่สามารถใช้งานได้จริงตามสถานการณ์ที่ข้อมูลมีการเปลี่ยนแปลงและเพิ่มขึ้นอย่างรวดเร็ว

- การเรียนใน MOOCs มีอัตราการออกกลางคันของผู้เรียน ซึ่งสาเหตุส่วนหนึ่งมาจากวิชาที่เรียนไม่ตรงกับความต้องการของผู้เรียน

สมมุติฐานงานวิจัย

- การประยุกต์ใช้ Apache platform ในระบบคำแนะนำจะทำให้ระบบสามารถรองรับการปรับขนาดข้อมูลในหน่วยเก็บข้อมูล (storage) ในรูปแบบแนวนอน (scale-out) รองรับกับข้อมูลที่มีขนาดใหญ่ได้ และทำให้ระบบคำแนะนำมีคุณภาพที่ดีขึ้น

- การสร้างตัวแบบระบบคำแนะนำด้วยโครงข่ายประสาทเทียมเชิงลึก (Deep Neural Network) ร่วมกับ Long Short-Term Memory (LSTM) จะช่วยให้คุณภาพของตัวแบบมีประสิทธิภาพที่ดีขึ้น

- ระบบคำแนะนำรายวิชาจะเข้ามาช่วยพัฒนาคุณภาพการศึกษาให้ดีขึ้น เช่น อัตราการออกกลางคันของผู้เรียนในระบบออนไลน์ ส่งเสริมให้เกิดการเรียนรู้ตลอดชีวิต

3.2 เครื่องมือและอุปกรณ์ที่ใช้ในการพัฒนางานวิจัย

3.2.1 ฮาร์ดแวร์ที่ใช้ในการวิจัย ประกอบด้วย

1) เครื่องคอมพิวเตอร์แม่ข่าย
   - หน่วยประมวลผล Intel(R) Xeon(R) CPU E5-1650 0 @ 3.20GHz, 6 core
   - หน่วยความจำ 32 GB
2) การ์ดหน่วยประมวลผลด้านกราฟฟิก (GPU) GEFORCE RTX 2080 Ti
   - RTX-OPS 78T
   - Boost Clock 1635 MHz (OC)

- Frame Buffer 11GB GDDR6
- Memory Speed 14 Gbps

3.2.2 ซอฟต์แวร์ที่ใช้ในการวิจัย ประกอบด้วย

- โปรแกรมพัฒนาชุดคำสั่งคอมพิวเตอร์ใช้ Jupyter Notebook และโปรแกรม Python 3
- โปรแกรม Deep Learning Library ใช้ TensorFlow V. 1.13 และ Keras V. 2.2.4

3.3 ข้อมูลที่ใช้ในการวิจัย (Data Overviewing)

การรวบรวมข้อมูล เป็นการความเข้าใจกับข้อมูลว่าข้อมูลที่นำมาใช้มีลักษณะอย่างไร แหล่งข้อมูลอยู่ที่ใด ข้อมูลมีความถูกต้องน่าเชื่อถือหรือไม่ ข้อมูลที่ได้มีปริมาณมากพอหรือยังเป็นข้อมูลที่เหมาะสมและมีรายละเอียดเพียงพอต่อการนำไปใช้ในการวิเคราะห์หรือไม่

จากการศึกษางานวิจัยที่เกี่ยวข้องพบว่า การวิจัยเกี่ยวกับ MOOCs นั้นมีพอสมควร เช่น การวิจัยเกี่ยวกับการหยุดเรียนกลางคันของผู้เรียน การวิเคราะห์พฤติกรรมของผู้เรียนใน MOOCs ระบบแนะนำรายวิชาใน MOOCs ซึ่งจากการศึกษางานวิจัยนั้น ๆ ข้อมูลที่ผู้วิจัยนำมาใช้ ส่วนใหญ่เป็นข้อมูล MOOCs ของประเทศที่ผู้วิจัยพำนัก แต่ก็มีงานวิจัยบางส่วนที่ไม่สามารถใช้ข้อมูลภายในประเทศของตนเองได้ จำเป็นต้องอาศัยข้อมูล MOOCs ที่เปิดเผยแบบสาธารณะและอนุญาตให้นักวิจัยสามารถใช้เพื่อการวิจัยได้คือชุดข้อมูลของ MITx and HarvardX และเป็นชุดข้อมูลที่เป็นที่นิยมที่นักวิจัยได้สร้างผลงานวิจัยและสามารถตีพิมพ์เผยแพร่ในฐานข้อมูลที่มีชื่อเสียงอย่างเช่น Scopus

ดังนั้นผู้วิจัยจึงเลือกใช้ชุดข้อมูล MOOCs ของ MITx and HarvardX ชุดข้อมูลชื่อ “The HarvardX-MITx Person-Course Academic Year 2013 De-Identified dataset รุ่น 2.0” [21] สร้างขึ้นเมื่อวันที่ 14 พฤษภาคม พ.ศ.2557 ชื่อไฟล์ HMXPC13_DI_v1_5-14-14.csv ชุดข้อมูลชื่อ “The HarvardX-MITx Person-Course Academic Year 2013 De-Identified dataset รุ่น 2.0” ประกอบด้วยรายวิชาจำนวนรายวิชา 16 รายวิชา แอตทริบิวต์จำนวน 18 แอตทริบิวต์ และเรคคอร์ดจำนวน 641,138 เรคคอร์ด รายละเอียดแสดงดังตารางที่ 5

ตารางที่ 5 รายละเอียดของแอตทริบิวต์

ลำดับที่	ชื่อแอตทริบิวต์	คำอธิบาย
1.	course_id (CI)	รหัสรายวิชา (ประกอบด้วยสถาบันการศึกษา, ภาคเรียน, รหัสรายวิชา)
2.	user_id	รหัสประจำตัวผู้เรียน
3.	registered	สถานะการลงทะเบียนเรียน (ค่าข้อมูลเป็น 1 ถ้านักเรียนลงทะเบียนเรียน)
4.	Viewed (Vi)	สถานะการเข้าถึงแท็บรายวิชา (ค่าข้อมูลเป็น 1 ถ้านักเรียนเข้าถึงแท็บรายวิชา เช่น วิดีโอ แบบทดสอบ แบบฝึกหัด)
5.	Explored (EP)	สถานะการเข้าไปสำรวจในรายวิชา (อย่างน้อยครึ่งหนึ่งของรายวิชาที่เขาลงทะเบียน)
6.	Certified (CT)	สถานะผลการเรียน (ค่าข้อมูลจะเป็น 1 ถ้าผู้เรียนสำเร็จการศึกษา)
7.	Final_cc_cname (LR)	ชื่อประเทศ/ตำแหน่งที่อยู่ของผู้เรียน
8.	LoE (LE)	การศึกษาระดับสูงสุดของผู้เรียน
9.	YoB (YB)	ปีเกิดของผู้เรียน
10.	Gender (GN)	เพศของผู้เรียน
11.	Grade (Gr)	ผลการเรียนของผู้เรียน
12.	start_time (ST)	วันที่ผู้เรียนลงทะเบียนรายวิชา
13.	last_event	วันที่มีการโต้ตอบล่าสุดกับรายวิชา
14.	nevents	จำนวนครั้งของการโต้ตอบที่ผู้เรียนทำกับรายวิชา
15.	ndays_act	จำนวนวันที่ไม่ซ้ำกันของนักเรียนที่มีปฏิสัมพันธ์กับรายวิชา
16.	nplay_video	จำนวนวิดีโอที่ผู้เรียนเปิดดู
17.	Nchapters (NC)	จำนวนบทเรียนที่ผู้เรียนเข้าถึง
18.	nforum_posts	จำนวนโพสต์ในฟอรั่มสนทนาที่ผู้เรียนโพสต์

3.4 สำรวจข้อมูลด้วย Exploratory Data Analysis (EDA)

เทคนิคการเลือกข้อมูลที่ดีทำให้สามารถเลือกข้อมูลที่มีความสำคัญและสามารถใช้เป็นตัวแทนของข้อมูลส่วนใหญ่ได้ กระบวนการ Exploratory Data Analysis คือ กระบวนการตรวจสอบสำรวจข้อมูลเบื้องต้น เพื่อทำความเข้าใจข้อมูลในแต่ละคุณลักษณะ (Feature) เช่น ข้อมูลเป็นชนิดอะไร ข้อมูลเป็นแบบต่อเนื่องหรือไม่ต่อเนื่อง ช่วงของข้อมูลกว้างแค่ไหน การกระจายของข้อมูลเป็นอย่างไร มีข้อมูลขาดหายไปหรือไม่ แต่ละคุณลักษณะ (Feature) เชื่อมโยงกันอย่างไร โดยมีขั้นตอนต่าง ๆ ดังนี้

ขั้นตอนที่ 1 การเตรียมข้อมูล (Data Wrangling, Data Transformation + Data Cleaning)

- การลบ space ในข้อมูลที่เกินมา (Remove extra white spaces)
- การทำให้ค่าทั้งหมดอยู่ในมาตรฐานเดียวกัน (Normalize values)
- จัดการกับแถวที่มีข้อมูลหายไป (Handle missing values)

ขั้นตอนที่ 2 การวิเคราะห์ข้อมูล (Data Analysis)

- ความสัมพันธ์ต่าง ๆ ของตัวแปร
- การหา Patterns ที่ปรากฏ
- การคำนวณค่าสถิติต่าง ๆ

ขั้นตอนที่ 3 การแสดงผลข้อมูล (Data Visualization)

การสร้างกราฟ หรือ Chart ต่าง ๆ เพื่อให้เราเข้าใจ insights ได้ง่ายขึ้น

จากขั้นตอนการสำรวจข้อมูลด้วย Exploratory Data Analysis (EDA) ค่าความสัมพันธ์ระหว่างตัวแปรแสดงดังตารางที่ 6

ตารางที่ 6 ค่าสัมประสิทธิ์สหสัมพันธ์ระหว่างตัวแปร

	LR	YB	GN	LE	CI	IS	CT	Gr	Vi	EP	NC	ST
LR	1.000	-0.073	-0.078	-0.002	0.117	-0.159	-0.030	-0.042	-0.079	-0.044	-0.069	-0.006
YB	-0.073	1.000	-0.607	-0.419	0.035	-0.047	-0.028	-0.033	-0.033	-0.040	-0.063	-0.017
GN	-0.078	-0.607	1.000	0.379	-0.161	0.182	0.015	0.017	0.068	0.037	0.061	0.054
LE	-0.002	-0.419	0.379	1.000	-0.055	0.069	0.022	0.024	0.040	0.027	0.047	0.021
CI	0.117	0.035	-0.161	-0.055	1.000	-0.907	-0.053	-0.059	-0.106	-0.012	-0.022	-0.030
IS	-0.159	-0.047	0.182	0.069	-0.907	1.000	0.053	0.064	0.112	0.004	0.046	0.029
CT	-0.030	-0.028	0.015	0.022	-0.053	0.053	1.000	0.936	0.131	0.628	0.628	-0.006
Gr	-0.042	-0.033	0.017	0.024	-0.059	0.064	0.936	1.000	0.166	0.691	0.695	-0.007
Vi	-0.079	-0.033	0.068	0.040	-0.106	0.112	0.131	0.166	1.000	0.199	0.426	-0.087
EP	-0.044	-0.040	0.037	0.027	-0.012	0.004	0.628	0.691	0.199	1.000	0.790	-0.010
NC	-0.069	-0.063	0.061	0.047	-0.022	0.046	0.628	0.695	0.426	0.790	1.000	-0.017
ST	-0.006	-0.017	0.054	0.021	-0.030	0.029	-0.006	-0.007	-0.087	-0.010	-0.017	1.000

จากตารางที่ 6 เห็นได้ว่าแอตทริบิวต์ Final_cc_cname (LR), LoE (LE), YoB (YB) และ Gender (GN) มีค่าสหสัมพันธ์ (Correlation) โดยส่วนใหญ่มีความสัมพันธ์กันสูงในทางลบ (จากภาพปรากฏเป็นไฮไลท์สีแดง) ซึ่งสามารถนำมาใช้เป็นตัวแทนของข้อมูลในกลุ่ม User profile ได้ และแอตทริบวิต์ course_id (CI), Institution (IS), viewed (Vi), explored (EP), certified (CT), grade (Gr), start_time (ST), nchapters (NC) โดยส่วนใหญ่มีความสัมพันธ์กันสูงในทางบวก (จากภาพปรากฏเป็นไฮไลท์สีเขียว)

3.5 การกำหนดตัวแบบ (Model Creating)

จากการศึกษางานวิจัยที่เกี่ยวข้อง ผู้วิจัยพบว่ามีขั้นตอนวิธีมากมายหลากหลาย ซึ่งแต่ละวิธีมีข้อดีและข้อบกพร่องแตกต่างกันไป ดังนั้นผู้วิจัยเลือกศึกษาขั้นตอนวิธีจำนวน 3 วิธี ได้แก่ 1) กฎความสัมพันธ์ (Association rule) 2) โครงข่ายประสาทเทียมเชิงลึก (Deep neural network, DNN) และ 3) โครงข่ายประสาทเทียมเชิงลึก (Deep neural network, DNN) ร่วมกับหน่วยความจำสั้นระยะยาว Long Short-Term Memory (LSTM) โดยมีรายละเอียดในแต่ละขั้นตอนวิธีดังนี้

3.5.1 ขั้นตอนวิธีแบบกฎความสัมพันธ์

1) หลักการกฎความสัมพันธ์ (Association rule)

กฎความสัมพันธ์เป็นหนึ่งในเครื่องมือที่ใช้หาความสัมพันธ์ของข้อมูล Apriori algorithm เป็นรูปแบบของ Itemset ที่มักเกิดร่วมกัน (Frequent itemsets) คือ ถ้าเรียนวิชา A มักจะเรียนวิชา B ด้วย เป็นการนำ Frequent itemsets มาสร้างเป็นกฎความสัมพันธ์ของข้อมูล (Association Rules) หรือเรียกว่า Market Basket Analysis โดย Association Rules จะ

กำหนดเงื่อนไขว่า Frequent itemsets นั้นต้องมีค่าสนับสนุนที่ยอมรับได้ (Minimum Support) มากกว่าที่กำหนด

2) แนวคิดพื้นฐานเกี่ยวกับกฎความสัมพันธ์ (Association rule)

- กำหนดให้ D เป็นเซตของทรานแซคชั่น และแต่ละทรานแซคชั่นประกอบด้วยเซตของรายการ (items)
- กำหนดให้ I แทนเซตของรายการ โดยที่ I = $\{i_1 , i_2 , i_n\}$
- Itemsets เป็นเซตของรายวิชา
- K-Itemsets เป็น Itemsets ที่ประกอบด้วย k รายการ
- Frequent Itemsets เป็น Itemsets ที่มีรค่าสนับสนุน (support) มากกว่าหรือเท่ากับค่าสนับสนุนที่กำหนด (minimum support threshold)

3) ตัวชี้วัดที่ใช้ในการหากฎที่มีความสนใจ

กฎความสัมพันธ์ที่สนใจ คือ กฎความสัมพันธ์ที่มีค่าสนับสนุน (Support) มากกว่าหรือเท่ากับค่าสนับสนุนที่กำหนด (Minimum support threshold) คือ 50% และค่าความเชื่อมั่น (Confidence) ที่มีค่ามากกว่าหรือเท่ากับค่าความเชื่อมั่นที่กำหนด (Minimum confidence threshold) คือ 50%

ค่าสนับสนุน (Support) เป็นค่าที่บ่งบอกว่าเหตุการณ์ A กับ B มีความถี่ในการเกิดขึ้นมากน้อยแค่ไหน ดังสมการที่ (8)

$$support(A \rightarrow B) = P(A \cup B)$$

หรือ

$$support(A \rightarrow B) = \text{จำนวนทรานแซคชั่นที่ปรากฏรายการทั้ง } A \text{ และ } B$$

สมการที่ (8)

ค่าความเชื่อมั่น (Confidence) คือ ค่าที่บอกว่าเมื่อเกิดเหตุการณ์ B แล้ว มีโอกาสที่จะเกิดเหตุการณ์ A มากน้อยแค่ไหน เช่น Confidence = 100% แสดงว่าผู้เรียนที่เรียนรายวิชา B ก็จะเรียนรายวิชา A ด้วยทุกครั้ง ดังสมการที่ (9)

$$confidence(A \rightarrow B) = P(B \mid A) = P(A \cup B) \mid P(A)$$

หรือ

$$support(A \rightarrow B) = \frac{\text{จำนวนทรานแซคชั่นที่ปรากฏรายการทั้ง A และ B}}{\text{จำนวนทรานแซคชั่นที่ปรากฏรายการ A}}$$

สมการที่ (9)

4) ขั้นตอนวิธี Apriori Algorithm

- L_k: frequent k-itemset, satisfy minimum support
- C_k: candidate k-itemset, possible frequent k-itemsets

$L_1 = \{\text{frequent 1-itemsets}\};$
For ($k=2$; $L_{k-1} \neq 0$; $k++$) do begin
$C_k = \text{apriori-gen}(L_{k-1})$;
For each transactions $t \in D$ do begin
$C_t = \text{subset}(C_K, t)$
For each candidate $c \in C_t$ do
c.count++;
end
$L_k = \{ c \in C_k \mid c.\text{count} \geq \text{minsup}\}$
and
Answer = $U_k L_k$

แบ่งข้อมูลเป็นชุดฝึกสอน (Training set) 80% ได้ 512,910 รายการ และชุดทดสอบ (Test set) 20% ได้ 128,228 รายการ

3.5.2 ขั้นตอนวิธีแบบโครงข่ายประสาทเทียมเชิงลึก

โครงข่ายประสาทเทียมเชิงลึก (Deep neural network, DNN) ถูกสร้างมาจากการนำเอา Neural Network หลายๆ ชั้นมาต่อกัน โดยชั้นแรกสุดทำหน้าที่เป็นชั้นนำเข้าข้อมูล (Input Layer) ชั้นสุดท้ายจะทำหน้าที่ส่งผลลัพธ์การประมวลผลออกมา (Output Layer) ส่วน ชั้นระหว่างชั้นแรกสุดและชั้นสุดท้ายเรียกว่าชั้นซ่อน (Hidden layer) โครงข่ายประสาทเทียมเชิงลึกหรือการเรียนรู้เชิงลึก (Deep Learning) ถูกนำไปใช้ในงานหลาย ๆ ด้าน รวมถึงระบบการแนะนำ เช่น ระบบการแนะนำเพลง วิดีโอหรือภาพยนตร์และข่าว

1) การทำความเข้าใจกับข้อมูล (Data Understanding)

จากขั้นตอนในข้อที่ 3.4 สำรวจข้อมูลด้วย Exploratory Data Analysis (EDA) ผู้วิจัยเลือกใช้ข้อมูลซึ่งประกอบด้วย User profile และ User behavior ดังตารางที่ 7 และตารางที่ 8 ดังนี้

ตารางที่ 7 รายละเอียดแอตทริบิวต์ (User profile)

ลำดับที่	ชื่อแอตทริบิวต์	คำอธิบาย
1	Final_cc_cname	ชื่อประเทศ/ตำแหน่งที่อยู่ของผู้เรียน
2	LoE	การศึกษาระดับสูงสุดของผู้เรียน
3	YoB	ปีเกิดของผู้เรียน
4	gender	เพศของผู้เรียน

ตารางที่ 8 รายละเอียดแอตทริบิวต์ (User behavior)

ลำดับที่	ชื่อแอตทริบิวต์	คำอธิบาย
1	viewed	สถานะการเข้าถึงแท็บรายวิชา (ค่าข้อมูลเป็น 1 ถ้านักเรียนเข้าถึงแท็บรายวิชา เช่น วิดีโอ แบบทดสอบ แบบฝึกหัด)
2	explored	สถานะการเข้าไปสำรวจในรายวิชา (อย่างน้อยครึ่งหนึ่งของรายวิชาที่เขาลงทะเบียน)
3	certified	สถานะผลการเรียน (ค่าข้อมูลจะเป็น 1 ถ้าผู้เรียนสำเร็จการศึกษา)
4	grade	ผลการเรียนของผู้เรียน
5	nchapters	จำนวนบทเรียนที่ผู้เรียนเข้าถึง

ลำดับที่	ชื่อแอตทริบิวต์	คำอธิบาย
6	nevents	จำนวนครั้งของการโต้ตอบที่ผู้เรียนทำกับรายวิชา
7	ndays_act	จำนวนวันที่ไม่ซ้ำกันของนักเรียนที่มีปฏิสัมพันธ์กับรายวิชา
8	nplay_video	จำนวนวิดีโอที่ผู้เรียนเปิดดู
9	nforum_posts	จำนวนโพสต์ในฟอรั่มสนทนาที่ผู้เรียนโพสต์

2) การเตรียมข้อมูล (Data Preparation)

การเตรียมความพร้อมของข้อมูล มีรายละเอียดต่าง ๆ ดังนี้

- จากข้อมูลทั้งสิ้น 641,138 รายการ 13 แอตทริบิวต์ นำมาทำความสะอาดข้อมูล (Cleaning data) โดยแอตทริบิวต์ YoB, grade, nevents, ndays_act, nplay_video, nchapters และ nforum_posts ที่ไม่มีข้อมูลจะถูกแทนที่ด้วย “0” แอตทริบิวต์ gender ที่ไม่มีข้อมูลจะถูกแทนที่ด้วย “3” ซึ่งหมายถึง “ไม่ระบุเพศ” และแอตทริบิวต์ LoE ที่ไม่มีข้อมูลจะถูกแทนที่ด้วย “6” ซึ่งหมายถึง “ไม่ระบุวุฒิการศึกษา” หลังทำความสะอาดข้อมูล (Cleaning data) เสร็จสิ้นข้อมูลยังคงเหลือเท่าเดิม

- กำหนดให้แอตทริบิวต์ Course_id เป็นคลาสผลเฉลย (Class Label)

- ทำนอร์มัลไลเซชัน (Normalization) เพื่อลดความเหลื่อมล้ำของข้อมูล และทำให้ข้อมูลอยู่ในช่วงเดียวกัน โดยใช้วิธี Min-Max Normalization โดยกำหนดให้อยู่ในช่วง (0 – 1) ดังสมการที่ (10)

$$v' = \frac{v - min_A}{max_A - min_A}(new_max_A - new_min_A) + new_min_A \qquad \text{สมการที่ (10)}$$

- เมื่อ v คือ ค่าของข้อมูลเดิม v' คือ ค่าของข้อมูลใหม่ min_A คือ ค่าที่น้อยที่สุดของชุดข้อมูลเดิม max_A คือ ค่าที่มากที่สุดของชุดข้อมูลเดิม new_min_A คือ ค่าที่น้อยที่สุดช่วงชุดข้อมูลใหม่ new_max_A คือ ค่าที่มากที่สุดช่วงชุดข้อมูลใหม่

- แบ่งข้อมูลเป็นชุดฝึกสอน (Training set) 80% ได้ 512,910 รายการ และชุดทดสอบ (Testing set) 20% ได้ 128,228 รายการ ซึ่งเป็นวิธีที่เหมาะกับชุดข้อมูลขนาดใหญ่

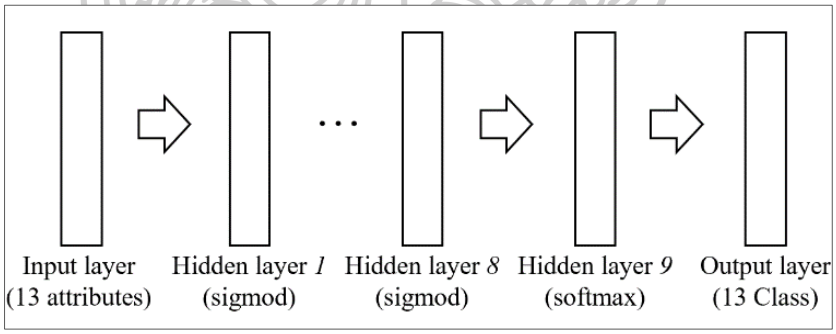
- เมื่อพิจารณาจำนวนของผู้เรียนในแต่ละรายวิชาพบว่าแต่ละรายวิชามีความถี่ของผู้เรียนที่แตกต่างกันมาก ดังนั้นจึงทำการเพิ่มปริมาณข้อมูล (Data Augmentation) โดยการเพิ่มข้อมูลในแต่ละรายวิชานั้นจะใช้วิธีสุ่ม หลังการเพิ่มข้อมูลเป็นที่เรียบร้อยแล้วทำให้แต่ละรายวิชามีความถี่ของผู้เรียนดังตารางที่ 9

ตารางที่ 9 แสดงจำนวนรายวิชาและจำนวนผู้เรียนในแต่ละรายวิชา

		จำนวนรายวิชาเรียนที่ผู้เรียนเลือกเรียนร่วมกัน (รายวิชา)					รวมผู้เรียนทั้งหมด (คน)	Training set	Test set
institution	course code	1	2	3	4	5		80	20
HarvardX	CB22x	30,002	14,491	11,725	14,378	20,912	91,508	73,206	18,302
HarvardX	CS50x	169,621	47,089	22,052	18,188	22,188	279,137	223,310	55,827
HarvardX	ER22x	57,406	22,077	14,840	15,755	21,499	131,577	105,262	26,315
HarvardX	PH207x	41,592	10,246	6,091	7,980	14,527	80,436	64,349	16,087
HarvardX	PH278x	39,602	17,260	11,832	14,054	20,709	103,457	82,765	20,691
MITx	14.73x	27,870	12,659	10,496	13,570	20,379	84,974	67,979	16,995
MITx	2.01x	5,665	3,207	3,908	7,633	14,518	34,931	27,944	6,986
MITx	3.091x	20,354	12,516	12,239	18,274	31,055	94,437	75,550	18,887
MITx	6.002x	63,046	30,214	21,825	24,900	36,976	176,960	141,568	35,392
MITx	6.00x	124,446	46,922	23,707	23,294	33,595	251,963	201,570	50,393
MITx	7.00x	21,009	12,321	11,362	13,952	19,861	78,504	62,803	15,701
MITx	8.02x	31,048	20,062	15,993	16,539	21,439	105,080	84,064	21,016
MITx	8.MReV	9,477	5,647	5,681	8,253	13,515	42,572	34,058	8,514
SUM		**641,138**	**254,711**	**171,751**	**196,770**	**291,173**	**1,555,534**	**1,244,427**	**311,107**

3) การกำหนดตัวแบบ (Model Creating)

พัฒนาตัวแบบโดยใช้โครงข่ายประสาทเทียมเชิงลึก (Deep Neural Network)



ภาพที่ 12 โครงข่ายประสาทเทียมแบบเพอร์เซปตรอนหลายชั้น (Multilayer Perceptron, MLP) โดยแต่ละชั้นซ่อน (Hidden layer) ใช้ฟังก์ชันการกระตุ้น (Activation Function) แบบ Sigmoid และใช้ Softmax ในชั้นสุดท้าย

จากภาพที่ 12 ซึ่งเป็นโครงข่ายของตัวแบบที่ผู้วิจัยออกแบบไว้ ขั้นตอนถัดไปคือการพัฒนาตัวแบบโดยใช้วิธีการแบบการเรียนรู้ของเครื่องแบบลึก (Deep Learning) ด้วยโครงสร้างแบบ Multilayer Perceptron แสดงดังภาพที่ 12 รายละเอียดดังนี้

- กำหนดให้ชั้นข้อมูลนำเข้า (Input Layer) ประกอบด้วยข้อมูลนำเข้า จำนวน 13 แอตทริบิวต์

- กำหนดจำนวนชั้นซ่อน (Hidden Layer) เป็นห้ารูปแบบ คือ 1, 3, 5, 7 และ 9 ชั้น (ตามลำดับ) เพื่อหาจำนวนชั้นซ่อนที่เหมาะสมที่สุด ภาพที่ 13 แสดงตัวแบบชั้นซ่อนจำนวน 9 ชั้น โดยแต่ละชั้นซ่อนมีจำนวนโหนดแต่ละชั้นคือ 26, 52, 104, 208, 416, 832, 1664, 3328 และ 6656 ตามลำดับ รวมพารามิเตอร์ที่ต้องประมวลผลมากกว่า 29 ล้านตัว
- กำหนดให้แอตทริบิวต์ Course_id เป็นคลาสผลเฉลย (Class Label)
- ปรับค่าน้ำหนัก (Weight) และค่าไบแอส (bias) ที่เชื่อมต่อในแต่ละโหนด เพื่อให้ผลลัพธ์ (Output) ใกล้เป้าหมายให้มากที่สุด โดยเลือกใช้อัลกอริทึมการเพิ่มประสิทธิภาพที่ชื่อ Nadam (Nadam Optimization Algorithms)
- ใช้เทคนิคการปรับปรุงประสิทธิภาพและความเสถียรของโครงข่ายประสาทเทียม ซึ่งเรียกว่า การทำแบตนอร์มัลไลเซชัน (Batch Normalization) เพิ่มเข้าไปในแต่ละชั้นซ่อน (Hidden layer) แสดงในภาพที่ 13

มหาวิทยาลัยศิลปากร

```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
input_1 (InputLayer)         (None, 13)                0
_________________________________________________________________
dense (Dense)                (None, 26)                364
_________________________________________________________________
batch_normalization_v1 (Batc (None, 26)                104
_________________________________________________________________
dense_1 (Dense)              (None, 52)                1404
_________________________________________________________________
batch_normalization_v1_1 (Ba (None, 52)                208
_________________________________________________________________
dense_2 (Dense)              (None, 104)               5512
_________________________________________________________________
batch_normalization_v1_2 (Ba (None, 104)               416
_________________________________________________________________
dense_3 (Dense)              (None, 208)               21840
_________________________________________________________________
batch_normalization_v1_3 (Ba (None, 208)               832
_________________________________________________________________
dense_4 (Dense)              (None, 416)               86944
_________________________________________________________________
batch_normalization_v1_4 (Ba (None, 416)               1664
_________________________________________________________________
dense_5 (Dense)              (None, 832)               346944
_________________________________________________________________
batch_normalization_v1_5 (Ba (None, 832)               3328
_________________________________________________________________
dense_6 (Dense)              (None, 1664)              1386112
_________________________________________________________________
batch_normalization_v1_6 (Ba (None, 1664)              6656
_________________________________________________________________
dense_7 (Dense)              (None, 3328)              5541120
_________________________________________________________________
batch_normalization_v1_7 (Ba (None, 3328)              13312
_________________________________________________________________
dense_8 (Dense)              (None, 6656)              22157824
_________________________________________________________________
dense_9 (Dense)              (None, 17)                113169
=================================================================
Total params: 29,687,753
Trainable params: 29,674,493
Non-trainable params: 13,260
_________________________________________________________________
```

ภาพที่ 13 ตัวแบบสำหรับระบบแนะนำรายวิชาจำนวนชั้นซ่อน 9 ชั้น โดยชั้นซ่อนลำดับที่ 1 มีโหนด 26 โหนด ชั้นซ่อนลำดับที่ 2 มีโหนด 52 โหนด ชั้นซ่อนลำดับที่ 9 มีโหนด 6656 โหนด รวมมีพารามิเตอร์ที่ใช้ทั้งสิ้นกว่า 29 ล้านพารามิเตอร์

- กำหนดค่าการเรียนรู้ (Learning Rate) เป็นสองรูปแบบ คือ 1e-3 และ 5e-4 สำหรับการฝึกตัวแบบ
- ใช้ฟังก์ชันการกระตุ้น (Activation Function) แบบ Sigmoid ในแต่ละชั้นชั้นซ่อน (Hidden Layer) และใช้ Softmax ในชั้นสุดท้าย แสดงในภาพที่ 12
- กำหนดจำนวนรอบที่ให้ตัวแบบได้เรียนรู้ในขั้นฝึกสอนสูงสุดจำนวน 300 Epoch

4) การประเมินผล (Evaluation)

การวัดและประเมินประสิทธิภาพของตัวแบบโดยใช้ข้อมูลชุดทดสอบ (Test set) 542,784 รายการ ด้วยการหาค่า Precision โดยประเมินผลลัพธ์การทำนายเปรียบเทียบกับผลลัพธ์จริง ดังสมการที่ (11)

$$Precision = \frac{True\ Positive}{Actual\ Results} \quad \text{สมการที่ (11)}$$

โดยที่

True Positive คือ ผลลัพธ์ที่ตัวแบบทำนายคลาสผลบวกของคลาสได้อย่างถูกต้อง

Actual Results คือ ผลลัพธ์ที่แท้จริงทั้งหมดของคลาสผลบวก

3.5.3 ขั้นตอนวิธีแบบโครงข่ายประสาทเทียมเชิงลึกร่วมกับหน่วยความจำสั้นระยะยาว

1) การทำความเข้าใจกับข้อมูล (Data Understanding)

จากขั้นตอนในข้อที่ 3.4 สำรวจข้อมูลด้วย Exploratory Data Analysis (EDA) ผู้วิจัยเลือกใช้ข้อมูลซึ่งประกอบด้วย User profile และ User behavior เป็นตัวแทนของข้อมูล ดังตารางที่ 10 และ
ตารางที่ 11

ตารางที่ 10 รายละเอียดแอตทริบิวต์ User profile

ลำดับที่	ชื่อแอตทริบิวต์	คำอธิบาย
1	Final_cc_cname	ชื่อประเทศ/ตำแหน่งที่อยู่ของผู้เรียน
2	LoE	การศึกษาระดับสูงสุดของผู้เรียน
3	YoB	ปีเกิดของผู้เรียน
4	gender	เพศของผู้เรียน

ตารางที่ 11 รายละเอียดแอตทริบิวต์ Course learning

ลำดับที่	ชื่อแอตทริบิวต์	คำอธิบาย
1	course_id	รหัสรายวิชา (ประกอบด้วยสถาบันการศึกษา, ภาคเรียน, รหัสรายวิชา)
2	Institution	สถาบันเจ้าของรายวิชา
3	viewed	สถานะการเข้าถึงแท็บรายวิชา (ค่าข้อมูลเป็น 1 ถ้านักเรียนเข้าถึงแท็บรายวิชา เช่น วิดีโอ แบบทดสอบ แบบฝึกหัด)
4	explored	สถานะการเข้าไปสำรวจในรายวิชา (อย่างน้อยครึ่งหนึ่งของรายวิชาที่เขาลงทะเบียน)
5	certified	สถานะผลการเรียน (ค่าข้อมูลจะเป็น 1 ถ้าผู้เรียนสำเร็จการศึกษา)
6	grade	ผลการเรียนของผู้เรียน
7	start_time	วันที่ผู้เรียนลงทะเบียนรายวิชา
8	nchapters	จำนวนบทเรียนที่ผู้เรียนเข้าถึง

2) การเตรียมข้อมูล (Data Preparation)

การเตรียมความพร้อมของข้อมูล มีรายละเอียดต่าง ๆ ดังนี้

- ข้อมูลเข้าประกอบด้วย User profile จำนวน 4 แอตทริบิวต์ และ Course Learning จำนวน 8 แอตทริบิวต์ รายละเอียดดังตารางที่ 10 และ
- ตารางที่ 11
- จากข้อมูลทั้งสิ้น 641,138 รายการ 13 แอตทริบิวต์ นำมาทำความสะอาดข้อมูล (Cleaning data) โดยแอตทริบิวต์ YoB, grade, nevents, ndays_act, nplay_video, nchapters และ nforum_posts ที่ไม่มีข้อมูลจะถูกแทนที่ด้วย “0” แอตทริบิวต์ gender ที่ไม่มีข้อมูลจะถูกแทนที่ด้วย “3” ซึ่งหมายถึง “ไม่ระบุเพศ” และแอตทริบิวต์ LoE ที่ไม่มีข้อมูลจะถูกแทนที่ด้วย “6” ซึ่งหมายถึง “ไม่ระบุวุฒิการศึกษา” หลังทำความสะอาดข้อมูล (Cleaning data) เสร็จสิ้นข้อมูลยังคงเหลือเท่าเดิม
- กำหนดให้แอตทริบิวต์ Course_id เป็นคลาสผลเฉลย (Class Label)

- ทำนอร์มัลไลเซชัน (Normalization) เพื่อลดความเหลื่อมล้ำของข้อมูล และทำให้ข้อมูลอยู่ในช่วงเดียวกัน โดยใช้วิธี Min-Max Normalization โดยกำหนดให้อยู่ในช่วง (0 – 1) ดังสมการที่ (12)

$$v' = \frac{v - min_A}{max_A - min_A}(new_max_A - new_min_A) + new_min_A \qquad \text{สมการที่ (12)}$$

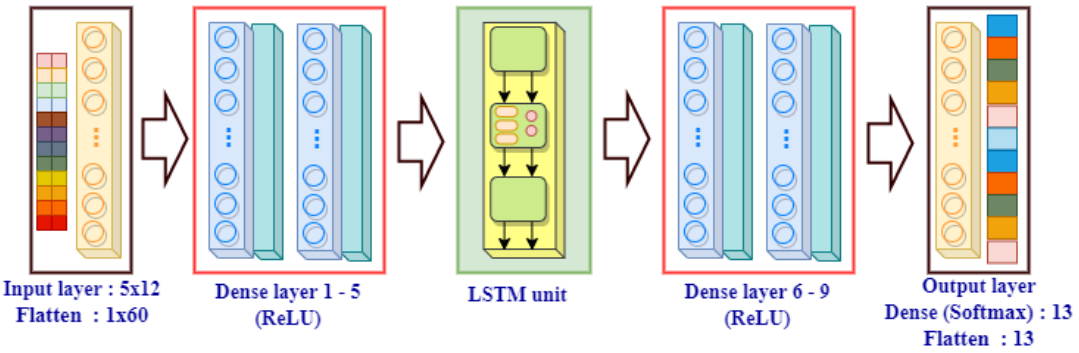
- เมื่อ v คือ ค่าของข้อมูลเดิม v' คือ ค่าของข้อมูลใหม่ min_A คือ ค่าที่น้อยที่สุดของชุดข้อมูลเดิม max_A คือ ค่าที่มากที่สุดของชุดข้อมูลเดิม new_min_A คือ ค่าที่น้อยที่สุดช่วงชุดข้อมูลใหม่ new_max_A คือ ค่าที่มากที่สุดช่วงชุดข้อมูลใหม่

- แบ่งข้อมูลเป็นชุดฝึกสอน (Training set) 80% ได้ 512,910 รายการ และชุดทดสอบ (Test set) 20% ได้ 128,228 รายการ ในแต่ละรายวิชา ซึ่งเป็นวิธีที่เหมาะกับชุดข้อมูลขนาดใหญ่

- เมื่อพิจารณาจำนวนของผู้เรียนในแต่ละรายวิชาพบว่าแต่ละรายวิชามีความถี่ของผู้เรียนที่แตกต่างกันมาก ดังนั้นจึงทำการเพิ่มปริมาณข้อมูล (Data Augmentation) โดยการเพิ่มข้อมูลในแต่ละรายวิชานั้นจะใช้วิธีสุ่ม หลังการเพิ่มข้อมูลเป็นที่เรียบร้อยแล้วทำให้แต่ละรายวิชามีความถี่ของผู้เรียนดังตารางที่ 9

3) การกำหนดตัวแบบ (Model Creating)

พัฒนาตัวแบบโดยใช้โครงข่ายประสาทเทียมเชิงลึก (Deep Neural Network) ร่วมกับ Long Short-Term Memory (LSTM)



ภาพที่ 14 โครงข่ายประสาทเทียมเชิงลึก (Deep neural network, DNN) ร่วมกับหน่วยความจำสั้นระยะยาว Long Short-Term Memory (LSTM)

จากภาพที่ 14 ตัวแบบระบบแนะนำรายวิชาโดยใช้โครงข่ายประสาทเทียมเชิงลึก (Deep Neural Network) ร่วมกับ Long Short-Term Memory (LSTM) ส่วนของข้อมูลนำเข้าประกอบด้วยข้อมูลประวัติผู้เรียน (User profile) และข้อมูลการเรียนรู้ของผู้เรียน (Course learning) โดยภายในโครงสร้างของตัวแบบระบบแนะนำรายวิชามีรายละเอียดดังนี้

1) Input layer ขนาด 5 x 12
2) Flattan layer ขนาด 1 x 60
3) Embedding layer ขนาด 60 x 200
4) Dense layer ที่ 1 – 5 เข้าสู่ LSTM unit = 500
5) Dense layer ที่ 6 – 9 ส่งออกไป Output layer ขนาด 5 x 13
6) กำหนดค่า Hyper parameter ต่าง ๆ ดังนี้ Optimizer เป็น Nadam, learning rate เป็น 1e-6, epsilon เป็น 1e-17, lose function เป็น SparseCategoricalCrossentropy

คำอธิบายเพิ่มเติม

- กำหนดให้ชั้นข้อมูลนำเข้า (Input Layer) ประกอบด้วยข้อมูล User profile จำนวน 4 แอตทริบิวต์ และข้อมูล Course learning จำนวน 8 แอตทริบิวต์ รวมจำนวน 12 แอตทริบิวต์
- กำหนดจำนวนชั้นซ่อน (Dense Layer, DL) 1 - 10
- กำหนดโครงสร้างของชั้นส่งออก (Output Layer) ประกอบด้วยข้อมูลส่งออกจำนวน 13 คลาส (class)
- ปรับค่าน้ำหนัก (Weight) และค่าไบแอส (bias) ที่เชื่อมต่อในแต่ละโหนดเพื่อให้ผลลัพธ์ (Output) ใกล้เป้าหมายให้มากที่สุด โดยเลือกใช้อัลกอริทึมการเพิ่มประสิทธิภาพที่ชื่อ Nadam (Nadam Optimization Algorithms)
- ใช้เทคนิคการปรับปรุงประสิทธิภาพและความเสถียรของโครงข่ายประสาทเทียม ซึ่งเรียกว่า การทำแบตนอร์มัลไลเซชัน (Batch Normalization) เพิ่มเข้าไปในแต่ละชั้นซ่อน (Hidden layer)
- กำหนดค่า Hyper parameter ต่าง ๆ ดังนี้ Optimizer เป็น Nadam, learning rate เป็น 1e-6, epsilon เป็น 1e-17, lose function เป็น SparseCategoricalCrossentropy

- ใช้ฟังก์ชันการกระตุ้น (Activation Function) แบบ ReLU ในแต่ละชั้นชั้นซ่อน (Dense Layer) และใช้ Softmax ในชั้นสุดท้าย

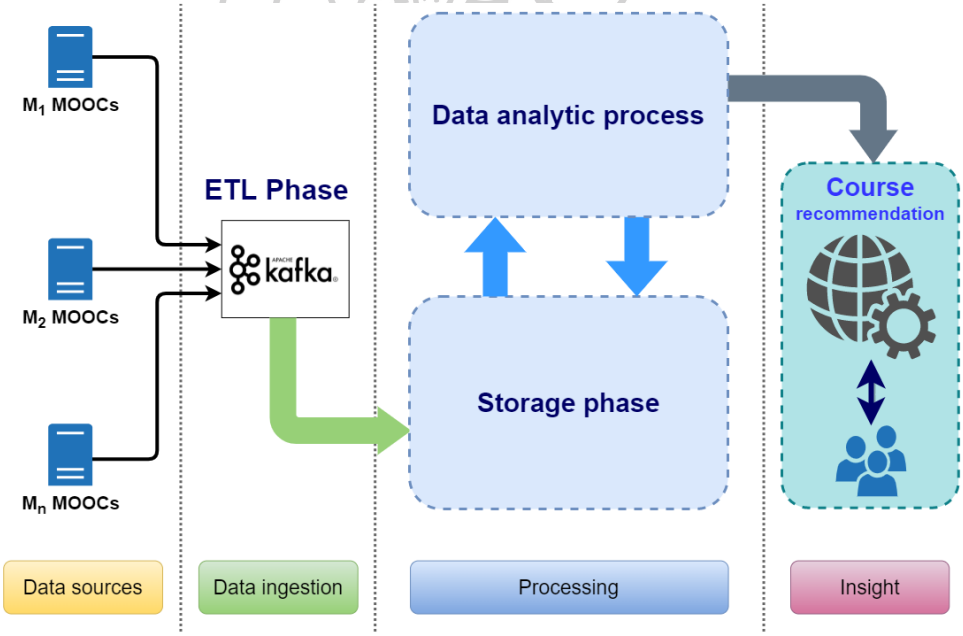
รายละเอียดใน Dense layer ที่ 1 - 10

Dense layer ที่ 1 ReLU 200
Batch normalization
Dense layer ที่ 2 ReLU 400
Batch normalization และ Max Pooling 1D size 4
Dense layer ที่ 3 ReLU 800
Batch normalization และ Max Pooling 1D size 3
Dense layer ที่ 4 ReLU 1600
Batch normalization และ Max Pooling 1D size 3
Dense layer ที่ 5 ReLU 3200
Batch normalization
Dropout 30%
LSTM unit = 500
Dropout 20%
Dense layer ที่ 6 ReLU 1000
Batch normalization
Dense layer ที่ 7 ReLU 800
Batch normalization
Dense layer ที่ 8 ReLU 400
Batch normalization
Dense layer ที่ 9 ReLU 200
Batch normalization
Dense layer ที่ 10 Softmax

3.6 การกำหนดสถาปัตยกรรมระบบแนะนำรายวิชา

ด้วยคุณลักษณะของล็อคไฟล์ข้อมูลการเรียนของผู้เรียน (event log) ใน MOOCs มีปริมาณมาก มีการเปลี่ยนแปลงอย่างรวดเร็วและมาจากหลายแหล่งข้อมูล จึงจำเป็นอย่างยิ่งที่จะต้องมีวิธีการจัดเก็บและการประมวลผลที่สามารถรองรับกับคุณลักษณะข้อมูลดังกล่าวได้ ดังนั้นผู้วิจัยจึงเลือกใช้แพลตฟอร์มอาปาเช (Apache platform) ที่ออกแบบมาสำหรับสำหรับการจัดเก็บข้อมูลขนาดใหญ่และมีระบบการจัดเก็บไฟล์ในรูปแบบ Hadoop Distributed File System (HDFS) ในการจัดเก็บข้อมูลล็อคไฟล์ (Event log) ข้อมูลการเรียนของผู้เรียนใน MOOCs แพลตฟอร์มอาปาเช (Apache platform) เป็นที่ยอมรับและนิยมนำไปใช้งานด้านต่าง ๆ มากมาย เช่น Facebook, Yahoo, Twitter, Adobe, Machine Translation, Bioinformatic, Scientific computing, Image Processing , Government [22]

จากคุณลักษณะของข้อมูลขนาดใหญ่ที่ประกอบด้วย 3V ได้แก่ Volume, Velocity และ Variaty นั้น ผู้วิจัยขอเสนอสถาปัตยกรรมระบบแนะนำรายวิชาสำหรับข้อมูลขนาดใหญ่ ประกอบด้วยส่วนต่าง ๆ ดังนี้ Data sources, Data ingestion, Processing, และ Insight แสดงดังภาพที่ 15



ภาพที่ 15 กรอบสถาปัตยกรรมระบบแนะนำรายวิชาโดยภาพรวมที่ผู้วิจัยนำเสนอ

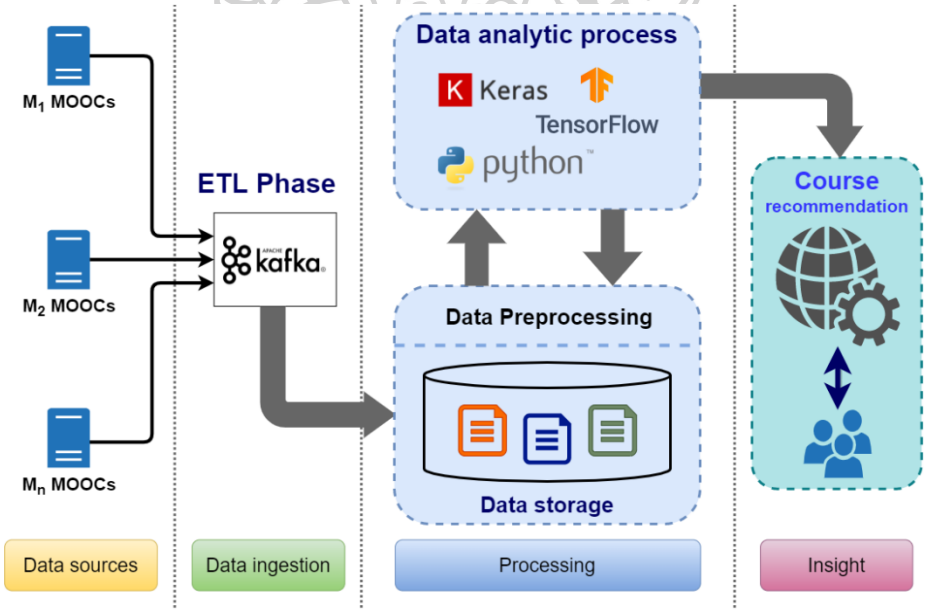
“Data sources” แหล่งข้อมูลจาก MOOCs ต่าง ๆ แทนด้วย M_1, M_2, ... M_n โดยสถาปัตยกรรมตัวแบบระบบแนะนำรายวิชาสามารถรองรับล็อคไฟล์เกี่ยวกับข้อมูลการเรียนของผู้เรียน (Event log) จากหลาย MOOCs ที่มีรูปแบบของโครงสร้างข้อมูลที่หลากหลาย เช่น CSV, JSON, HDFS ได้

“Data ingestion” เป็นการรับโอนข้อมูลจาก MOOCs ต้นทางส่งไปยัง Storage phase ทำหน้าที่ Extract, Load, Transform (ELT) เพื่อให้ข้อมูลที่มีขนาดใหญ่และมีการเกิดขึ้นตลอดเวลาสามารถส่งผ่านได้อย่างต่อเนื่องและมีคุณภาพ

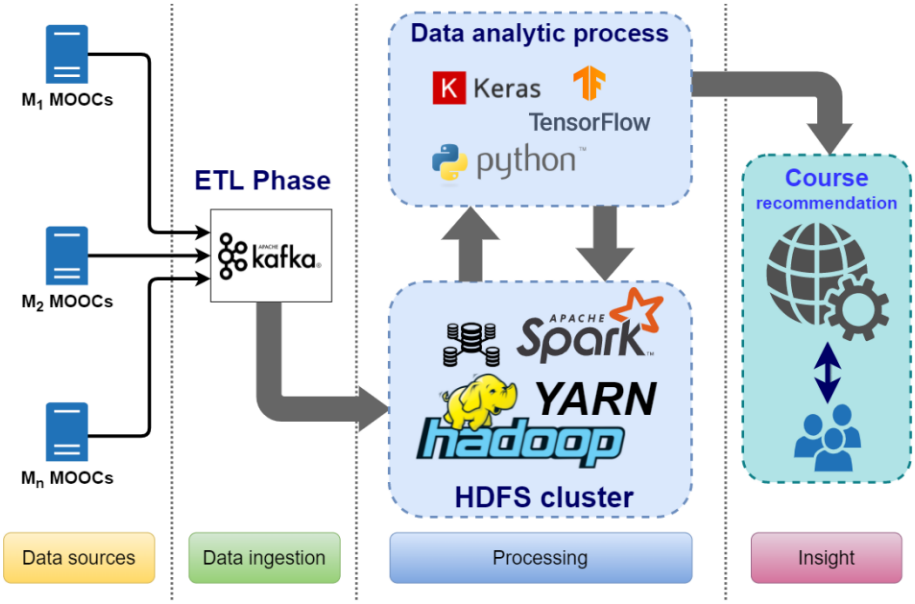
“Processing” เป็นส่วนที่ทำหน้าที่จัดเก็บข้อมูลและประมวลผลข้อมูล ประกอบด้วยการทำงาน 2 ส่วน คือส่วนของ storage phase และส่วนของ data analytic

“Insight” ตัวแบบระบบแนะนำรายวิชาเสนอรายวิชาให้แก่ผู้เรียน

จากขั้นตอน “Data ingestion” การรับโอนข้อมูลจาก MOOCs ต้นทางเพื่อส่งต่อไปขั้นตอน “Processing” นั้น ผู้วิจัยกำหนดกรอบสถาปัตยกรรมระบบแนะนำรายวิชาเป็น 2 รูปแบบเพื่อเปรียบเทียบประสิทธิภาพของระบบ แสดงดังภาพที่ 16 และภาพที่ 17 คือ หนึ่ง) การทำงานรูปแบบที่ 1 เรียกว่าแบบ Non-clustering และสอง) การทำงานรูปแบบที่ 2 เรียกว่า Clustering

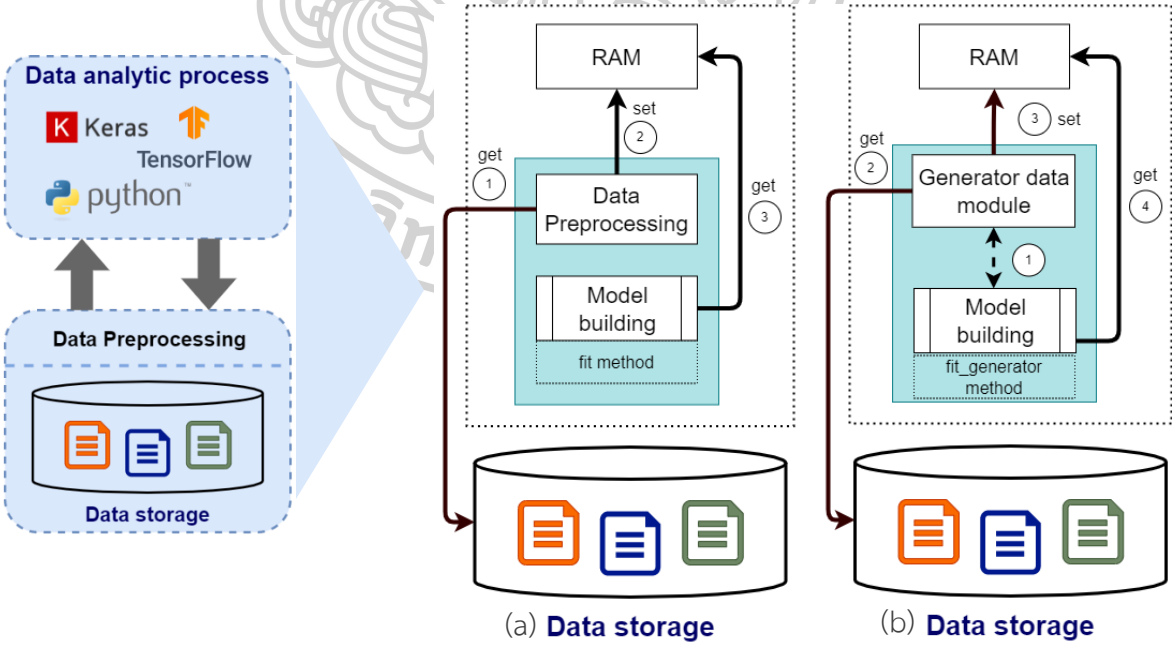


ภาพที่ 16 กรอบสถาปัตยกรรมระบบแนะนำรายวิชารูปแบบที่ 1 เรียกว่าแบบ Non-clustering



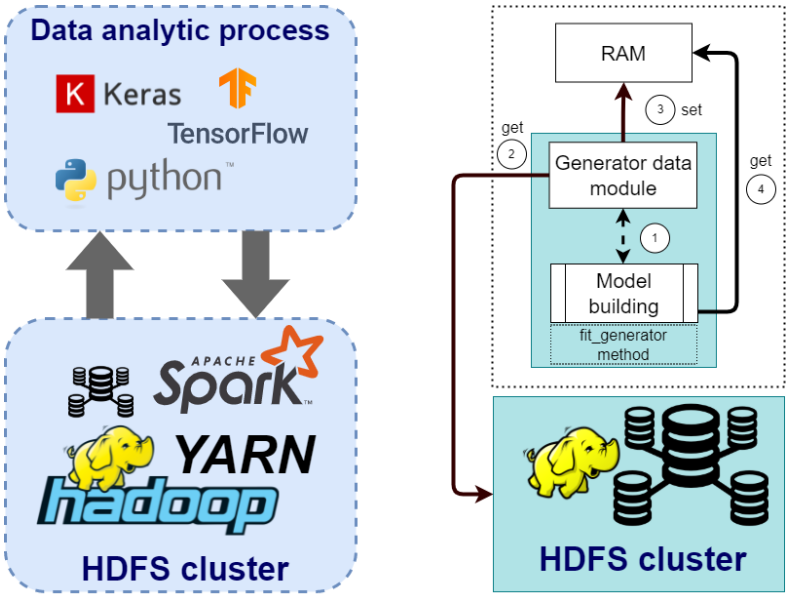
ภาพที่ 17 กรอบสถาปัตยกรรมระบบแนะนำรายวิชารูปแบบที่ 2 เรียกว่า แบบ Clustering

จากภาพที่ 16 แสดงกรอบสถาปัตยกรรมระบบแนะนำรายวิชารูปแบบที่ 1 เรียกว่าแบบ Non-clustering นั้นแยกตามวิธีในการฝึกสอนตัวแบบได้ 2 วิธีคือ หนึ่ง) ฝึกสอนด้วย fit method เรียกว่า “MCR-NF” และสอง) ฝึกสอนด้วย fit-generator method เรียกว่า “MCR-NG”



ภาพที่ 18 กรอบสถาปัตยกรรมระบบแนะนำรายวิชารูปแบบที่ 1 เรียกว่า แบบ Non-clustering (a) ฝึกสอนด้วย fit method เรียกว่า “MCR-NF” (b) ฝึกสอนด้วย fit-generator method เรียกว่า “MCR-NG”

จากภาพที่ 17 แสดงกรอบสถาปัตยกรรมระบบแนะนำรายวิชารูปแบบที่ 2 เรียกว่าแบบ Clustering นั้น ใช้วิธีในการฝึกสอนตัวแบบด้วย fit-generator method เรียกว่า “MCR-NG”



ภาพที่ 19 กรอบสถาปัตยกรรมระบบแนะนำรายวิชารูปแบบที่ 2 เรียกว่า แบบ Clustering) ฝึกสอนด้วย fit-generator method เรียกว่า “MCR-CG”

ตัวแบบที่มีแนวโน้มให้ผลค่าความแม่นยำที่ดีที่สุดที่ได้จากการทดลองคือตัวแบบ DNN + LSTM ซึ่งแสดงรายละเอียดในข้อที่ 3.5 เพื่อเป็นการเปรียบเทียบประสิทธิภาพการทำงานเชิงเวลาและความแม่นยำของตัวแบบ นำมาทดสอบโดยแยกการทดลองเป็นสามรูปแบบ คือ หนึ่ง) การทำงานแบบ ไม่ใช่คลัสเตอริ่ง (non-Clustering) ฝึกสอนด้วย fit method เรียกว่า “MCR-NF” แสดงดังภาพที่ 18 (a) สอง) การทำงานแบบ ไม่ใช่คลัสเตอริ่ง (non-Clustering) ฝึกสอนด้วย fit-generator method เรียกว่า “MCR-NG” แสดงดังภาพที่ 18 (b) และ สาม) การทำงานแบบคลัสเตอริ่ง (Clustering) ฝึกสอนด้วย fit-generator method เรียกว่า “MCR-CG” แสดงดังภาพที่ 19 เพื่อเปรียบเทียบประสิทธิภาพเชิงเวลาและค่าความแม่นยำของตัวแบบ โดยมีรายละเอียดดังนี้

1) การทำงานแบบไม่ใช่คลัสเตอริ่ง (non-Clustering) ฝึกสอนด้วย fit method เรียกว่า “MCR-NF”

จากภาพที่ 18 (a) ตัวแบบ “MCR-NF” ฝึกสอนด้วย fit method เริ่มจากส่วนของ “Data Preprocessing” จะต้องไปดึงข้อมูลจาก “Data storage” เพื่อเตรียมชุดข้อมูลสำหรับการฝึกสอนตัวแบบ เมื่อเตรียมข้อมูลเสร็จจะส่งข้อมูลไปเก็บไว้ที่ “RAM” หลังจากนั้นส่วนของ “Model building” จะเรียกใช้ “fit method” ให้ทำงาน และทำซ้ำในขั้นตอนที่ 3 ไปจนครบรอบการฝึกสอนตัวแบบ (จำนวน 500 epoch) จะเห็นได้ว่า “fit method” จะใช้ชุดข้อมูลเดิมจาก “RAM” ในทุกรอบการฝึกตัวแบบ

2) การทำงานแบบไม่ใช่คลัสเตอริ่ง (non-Clustering) ฝึกสอนด้วย fit-generator method เรียกว่า “MCR-NG”

จากภาพที่ 18 (b) ตัวแบบ “MCR-NG” ฝึกสอนด้วย fit-generator method เริ่มจากส่วนของ “Model building” จะเรียกใช้ “Generator data module” เพื่อเตรียมชุดข้อมูลสำหรับการฝึกสอนตัวแบบ โดย “Generator data module” จะไปดึงข้อมูลจาก “Data stroage” และนำข้อมูลที่ได้ไปเก็บไว้ที่ “RAM” หลังจากนั้นส่วนของ “Model building” จะเรียก “fit-generator method” ให้ทำงาน และทำซ้ำในขั้นตอนที่ 1 - 4 ไปจนครบรอบการฝึกตัวแบบ (จำนวน 500 epoch) จะเห็นได้ว่า “fit-generator method” จะสับเปลี่ยนชุดข้อมูล (shuffle data) ให้ลำดับข้อมูลไม่เหมือนกันทุกครั้งในทุกรอบการฝึกตัวแบบ

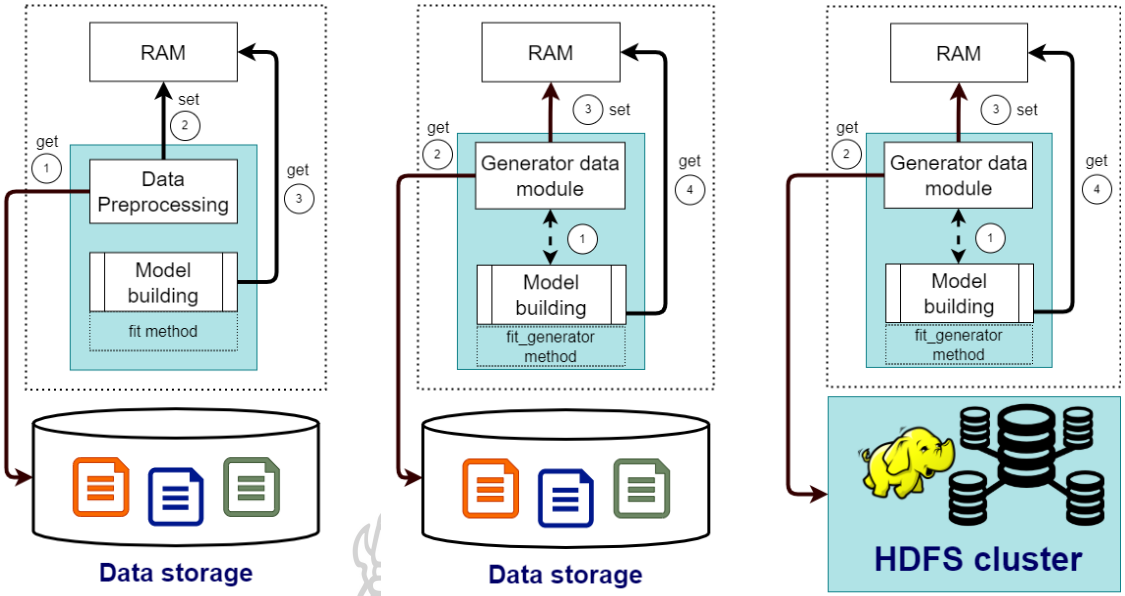
3) การทำงานแบบคลัสเตอริ่ง (Clustering) ฝึกสอนด้วย fit-generator method เรียกว่า “MCR-CG”

จากภาพที่ 19 ตัวแบบ “MCR-CG” ฝึกสอนด้วย fit-generator method เริ่มจากส่วนของ “Model building” จะเรียกใช้ “Generator data module” เพื่อเตรียมชุดข้อมูลสำหรับการฝึกสอนตัวแบบ โดย “Generator data module” จะไปดึงข้อมูลจาก “Hadoop cluster” และนำข้อมูลที่ได้ไปเก็บไว้ที่ “RAM” หลังจากนั้นส่วนของ “Model building” จะเรียก “fit-generator method” ให้ทำงาน และทำซ้ำในขั้นตอนที่ 1 - 4 ไปจนครบรอบการฝึกตัวแบบ (จำนวน 500 epoch) จะเห็นได้ว่า “fit-generator method” จะสับเปลี่ยนชุดข้อมูล (shuffle data) ให้ลำดับข้อมูลไม่เหมือนกันทุกครั้งในทุกรอบการฝึก สรุปกรอบสถาปัตยกรรมระบบแนะนำรายวิชาแสดงดังภาพที่ 20

ในแต่ละ Epoch ของการเทรน Machine Learning สอนตัวแบบ Deep Neural Network สิ่งที่สำคัญคือไม่ควรป้อนข้อมูลที่เรียงลำดับเหมือนกันทุกครั้งให้ตัวแบบ ดังนั้นผู้วิจัยจึงเลือกใช้วิธีการ

สับเปลี่ยนชุดข้อมูลหรือการสลับตำแหน่งข้อมูล (shuffle data) ก่อนป้อนข้อมูลเข้าสู่ตัวแบบ เพื่อเป็นการลดการจำข้อมูลของตัวแบบ ช่วยให้ตัวแบบมีความเป็น Generalization ได้ดีขึ้น ลด Variance ของตัวแบบ เช่น การที่แบ่งข้อมูลออกเป็น Mini-Batch ที่มีขนาด Batch Size = 32 ป้อนข้อมูลตัวอย่าง Feedforward ให้กับตัวแบบ ทีละ 32 (x, y) ตัวแบบจะใช้ข้อมูลตามลำดับที่ได้รับเข้ามา และจะเป็นแบบนี้ทุก ๆ Epoch ทำให้ตัวแบบเรียนรู้ได้ยาก การสลับตำแหน่งข้อมูล (shuffle data) จะสามารถช่วยแก้ปัญหานี้ได้ ตัวอย่างเช่น Source code ด้านล่าง มี arr เป็นตัวแปรอาเรย์ที่เก็บข้อมูล [1, 2, 3, 4, 5] ไว้ เมื่อมีการเรียกใช้ shuffle ผลลัพธ์จะปรากฏในส่วนของ Output จากการรันครั้งที่ 1 จะได้ [3 2 1 5 4] การรันครั้งที่ 2 จะได้ [5 3 4 1 2] การรันครั้งที่ 3 จะได้ [4 1 3 2 5] โดยจะทำการเปลี่ยนแปลงข้อมูลดั้งเดิม

Source code	**Output**
from numpy import random import numpy as np arr = np.array([1, 2, 3, 4, 5]) random.shuffle(arr) print(arr)	# ผลลัพธ์จากการรันครั้งที่ 1 [3 2 1 5 4] # ผลลัพธ์จากการรันครั้งที่ 2 [5 3 4 1 2] # ผลลัพธ์จากการรันครั้งที่ 3 [4 1 3 2 5]



ภาพที่ 20 สรุปกรอบสถาปัตยกรรมระบบแนะนำรายวิชาทั้ง 2 รูปแบบ (a) การทำงานแบบ Non-clustering ฝึกสอนด้วย fit method เรียกว่า “MCR-NF” (b) การทำงานแบบ Non-clustering ฝึกสอนด้วย fit-generator method เรียกว่า “MCR-NG” และ (c) การทำงานแบบ Clustering ฝึกสอนด้วย fit-generator method เรียกว่า “MCR-CG”

บทที่ 4
ผลการดำเนินการวิจัย

ตามแนวทางการดำเนินการวิจัยที่ได้กล่าวไว้ในบทที่ 3 ผู้วิจัยจากการพัฒนาระบบแนะนำรายวิชาสำหรับรายวิชาออนไลน์แบบเปิดสู่มวลชนด้วยเทคนิคข้อมูลขนาดใหญ่ ผลการดำเนินการวิจัยมีดังต่อไปนี้

1) ผลการศึกษาขั้นตอนวิธีที่เหมาะสมสำหรับนำไปสร้างตัวแบบระบบแนะนำรายวิชาในรายวิชาออนไลน์แบบเปิดสู่มวลชน (MOOCs) ด้วยเทคนิคข้อมูลขนาดใหญ่ (Big Data)

2) ผลการปรับปรุงเทคนิคระบบแนะนำรายวิชาสำหรับข้อมูลขนาดใหญ่ (Big Data) ให้เหมาะสมและมีประสิทธิภาพ เช่น ลดเวลาในการทำงาน หรือมีความแม่นยำ

3) ผลการพัฒนาตัวแบบระบบแนะนำรายวิชาในรายวิชาออนไลน์แบบเปิดสู่มวลชน (MOOCs) ให้กับผู้เรียน

4.1 ผลการศึกษาขั้นตอนวิธีที่เหมาะสมสำหรับนำไปสร้างตัวแบบระบบแนะนำรายวิชา

จากการศึกษาขั้นตอนวิธีที่เหมาะสมสำหรับนำไปสร้างตัวแบบระบบแนะนำรายวิชาแสดงรายละเอียดดังบทที่ 3 ผู้วิจัยดำเนินการด้วยขั้นตอนวิธีกฎความสัมพันธ์ (Association rule) ขั้นตอนวิธีแบบแบบโครงข่ายประสาทเทียมเชิงลึก (Deep neural network, DNN) และขั้นตอนวิธี Deep neural network (DNN) ร่วมกับ Long Short-Term Memory (LSTM) ผลการดำเนินการวิจัยเป็นดังนี้

4.1.1 ผลการใช้ขั้นตอนวิธีกฎความสัมพันธ์

1) ผลการใช้กฎความสัมพันธ์ (Association rule) แบบ Apriori ตัววัดประสิทธิภาพมี 2 ตัว คือ ค่าสนับสนุน (Support) และค่าความเชื่อมั่น (Confidence) ดังนี้

ผู้วิจัยแบ่งข้อมูลเป็นชุดฝึกสอน (Training set) 80% ได้ 512,910 รายการ และชุดทดสอบ (Testing set) 20% ได้ 128,228 รายการ วัตถุประสงค์ของการแบ่งข้อมูลออกเป็น 2 ชุดทดสอบนั้นเพื่อใช้ตรวจสอบประสิทธิภาพของขั้นตอนวิธีแบบ Apriori โดยผู้วิจัยค้นหาความสัมพันธ์ในชุดข้อมูลจากข้อมูลชุดฝึกสอน (Training set) แล้วนำกฎที่ได้ที่ได้ไปเปรียบเทียบกับข้อมูลชุดทดสอบ (Testing set) สอดคล้องกับกับงานวิจัย MCRs [19] ที่มีการแบ่งข้อมูลออกเป็น 2 ชุดทดสอบเช่นกัน โดยใช้ข้อมูลสำหรับเป็นชุดฝึกสอน (Training set) 84% จำนวน 100,000 รายการ และชุดทดสอบ (Test set) 16% จำนวน 19,024 รายการ

ผลการใช้ Apriori ทำให้ได้ค่าสนับสนุน (Support) และค่าความเชื่อมั่น (Confidence) แสดงตารางที่ 12 และตารางที่ 13 ตามลำดับ

ตารางที่ 12 ค่าสนับสนุน (Support)

No.	Course combination	ผู้วิจัย	MCRs
1	2	0.712	0.761
2	2, 4	0.027	0.032
3	3, 2, 1	0.008	0.009
4	3, 2, 1, 5	0.003	0.003

$$support(A \rightarrow B) = P(A \cup B)$$ สมการที่ (13)

จากตารางที่ 12 ที่คำนวณมาได้จากสมการที่ 8 การทดลองแสดงให้เห็นว่า

- 1-frequency item set คือ รายวิชา Introduction to Computer Science I (CS50X) ซึ่งแสดงให้เห็นว่านักเรียนเรียนรายวิชานี้มากที่สุด
- 2-frequency item set คือ รายวิชา Introduction to Computer Science I (CS50X) และ Health in Numbers: Quantitative Methods in Clinical & Public Health Research (PH207X)
- 3-frequency item set คือ รายวิชา The Ancient Greek Hero (CB22X), Introduction to Computer Science I (CS50X) และ Justice (ER22X)
- 4-frequency item set คือ รายวิชา The Ancient Greek Hero (CB22X), Introduction to Computer Science I (CS50X), Justice (ER22X) และ Human Health and Global Environmental Change (PH278X)

อภิปรายผลได้ว่า มีผู้เรียนเลือกเรียนรายวิชาเพียงแค่รายวิชาเดียว ได้แก่รายวิชา Introduction to Computer Science I (CS50X) มีความถี่เกิดขึ้นมากถึง 71.2% ผู้วิจัยสันนิษฐานว่ารายวิชาดังกล่าวเป็นรายวิชาที่มีความน่าสนใจสอดคล้องกับยุคสมัยและแนวโน้มของโลกที่กำลังจะเปลี่ยนแปลงไป ทำผู้เรียนเลือกเรียนรายวิชานี้เป็นจำนวนมาก

มีผู้เรียนเลือกเรียนรายวิชา 2 รายวิชาคู่กันเสมอได้แก่ รายวิชา Introduction to Computer Science I (CS50X) และรายวิชา Health in Numbers: Quantitative Methods in Clinical & Public Health Research (PH207X) ซึ่งเป็นเหตุการณ์ที่เกิดขึ้นด้วยกันถึงร้อยละ 2.7% (ไม่สนใจลำดับของวิชาที่เกิดขึ้น) ผู้วิจัยสันนิษฐานว่ารายวิชา Introduction to Computer Science I (CS50X) เป็นรายวิชาที่มีความน่าสนใจสอดคล้องกับยุคสมัยและแนวโน้มของโลกที่กำลังจะเปลี่ยนแปลงไป และเลือกเรียนรายวิชา Health in Numbers: Quantitative Methods in Clinical & Public Health Research ประกอบเพราะเป็นรายวิชาที่น่าสนใจในกลุ่มของสุขภาพกับงานวิจัย

มีผู้เรียนเลือกเรียนรายวิชา 3 รายวิชาคู่กันเสมอได้แก่ รายวิชา The Ancient Greek Hero (CB22X), Introduction to Computer Science I (CS50X) และ Justice (ER22X) โดยมีเหตุการณ์ที่เกิดขึ้นด้วยกัน 0.8 % (ไม่สนใจลำดับของวิชาที่เกิดขึ้น) ผู้วิจัยสันนิษฐานว่าผู้เรียนที่เลือกเรียนทั้ง 3 รายวิชาควบคู่กันเสมอน่าจะนักศึกษาที่สนใจเรื่องกฎหมาย อารยะธรรมกรีกโบราณ และสนใจทางด้านคอมพิวเตอร์พื้นฐาน ซึ่งรายวิชา Introduction to Computer Science I (CS50X) เป็นรายวิชาที่มีความน่าสนใจสอดคล้องกับยุคสมัยและแนวโน้มของโลกที่กำลังจะเปลี่ยนแปลงไป

ตารางที่ 13 ค่าความเชื่อมั่น (Confidence)

No.	Course combination	ผู้วิจัย	MCRs
1	1, 3 -> 2	0.435	0.492
2	1, 2 -> 3	0.412	0.463
3	3 -> 2	0.396	0.403
4	1 -> 3	0.382	0.398

$$confidence(A \rightarrow B) = P(B \mid A) = P(A \cup B) \mid P(A)$$ สมการที่ (9)

จากตารางที่ 13 ที่คำนวณมาได้จากสมการที่ 9 ผลลัพธ์แถวที่ 1 แสดงค่าความเชื่อมั่นที่บ่งบอกว่าเมื่อผู้เรียนเลือกเรียนรายวิชา 2 แล้ว มีโอกาสที่จะเกิดเหตุการณ์การเลือกรายวิชา 1 และ 3 มากถึง 43.5% สำหรับแถวที่ 3 แสดงให้เห็นว่าเมื่อผู้เรียนเลือกเรียนรายวิชา 2 แล้ว มีโอกาสที่จะเกิดเหตุการณ์การเลือกรายวิชา 3 มากถึง 39.6%

2) สรุปผลการใช้ขั้นตอนวิธีกฎความสัมพันธ์ (Association rule) แบบ Apriori สำหรับระบบแนะนำรายวิชาในรายวิชาออนไลน์แบบเปิดสู่มวลชน (MOOCs)

จากการทดลองด้วยขั้นตอนวิธี Apriori ผลการวิเคราะห์แสดงไว้ในตารางที่ 14 จะเห็นได้ว่าความสัมพันธ์ระหว่างรายวิชา 1, 3 และ 2 อยู่ในระดับสูง ความเชื่อมโยงระหว่างรายวิชา 2 และ 1 อยู่ในระดับต่ำ จากข้อมูลเหล่านี้เป็นที่ชัดเจนว่ารายวิชาใดมีแนวโน้มที่จะถูกเลือกเรียนมากกว่า เพื่อที่จะได้จัดการเรียนการสอนที่เหมาะสมมากขึ้น การเตรียมทรัพยากรต่าง ๆ สำหรับรองรับผู้เรียน ซึ่งเป็นผลประโยชน์ที่ฝั่งผู้สอนหรือผู้ดูแลระบบได้รับการการหาความสัมพันธ์ของข้อมูล

จากการพิจารณาประสิทธิภาพด้วยข้อมูลชุดทดสอบจำนวน 128,228 รายการ พบค่าความเชื่อมั่น (Confidence) ของผู้เรียนที่ลงทะเบียนรายวิชา 2 หลังจากรายวิชา 1 และ 3 เป็น 25.8% มีการแนะนำรายวิชา 3 ให้กับผู้ที่ลงทะเบียนรายวิชา 1 และ 2 ประสิทธิภาพเป็น 24.7% มีการแนะนำรายวิชา 2 ให้กับผู้ที่ลงทะเบียนรายวิชา 3 ประสิทธิภาพเป็น 23.8% มีการแนะนำรายวิชา 3 ให้กับผู้ที่ลงทะเบียนรายวิชา 1 ประสิทธิภาพเป็น 24.9% ดังนั้นขั้นตอนวิธีกฎความสัมพันธ์ (Association rule) แบบ Apriori สามารถสร้างรายการแนะนำให้กับผู้เรียนได้บางรายเท่านั้น

ตารางที่ 14 ประสิทธิภาพของการแนะนำรายวิชาด้วยขั้นตอนวิธีแบบ Apriori

No.	Course combination	Recommend efficiency
1	1, 3 -> 2	25.8%
2	1, 2 -> 3	24.7%
3	3 -> 1, 2	Not recommend
4	2 -> 1, 3	Not recommend
5	3 -> 2	23.8%
6	1 -> 3	24.9%
7	2 -> 5	Not recommend
8	2 -> 1	Not recommend

กล่าวโดยสรุปคือ เมื่อเปรียบเทียบกับงานวิจัย MCRs ที่เลือกใช้ Apache Sqoop สำหรับทำหน้าที่รับข้อมูลจากแหล่งต่าง ๆ (Data Acquisition Layer) โดยที่ข้อมูลมีลักษณะเป็นแบบ batch และข้อมูลเป็นประเภท structured แต่งานวิจัยฉบับนี้เลือกใช้ Apache Kafka ในการรับส่งข้อมูลแบบเรียลไทม์ ทำให้สามารถจัดการกับข้อมูลที่ถูกสร้างขึ้นอย่างต่อเนื่อง (Streaming Data) จากแหล่งข้อมูลจำนวนมากที่มาจากหลากหลาย MOOCs และที่สำคัญข้อมูลพฤติกรรมของผู้เรียนใน

ระบบ MOOCs จะมีการเกิดเปลี่ยนแปลงและเกิดขึ้นอยู่ตลอดเวลา ซึ่งก็คือข้อมูลมีลักษณะเป็นแบบ Stream จากเหตุผลที่กล่าวมาข้างต้น สรุปได้ว่า Apache Kafka เหมาะสมที่นำมาใช้มากกว่า Apache Sqoop

4.1.2 ผลการใช้ขั้นตอนวิธีแบบแบบโครงข่ายประสาทเทียมเชิงลึก

1) ผลการใช้โครงข่ายประสาทเทียมเชิงลึก (Deep neural network, DNN)

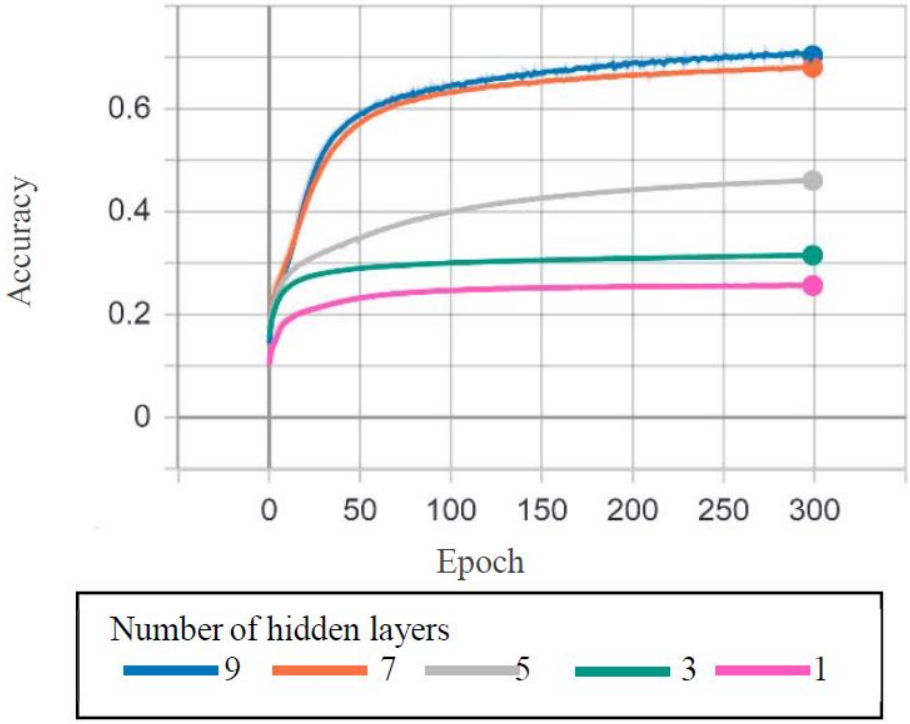
จากการออกแบบโครงข่ายประสาทเทียมเชิงลึกตามบทที่ 3 นั้น ตัวแบบสำหรับระบบแนะนำหลักสูตรจำนวนชั้นซ่อน 9 ชั้น โดยชั้นซ่อนลำดับที่ 1 มีโหนด 26 โหนด ชั้นซ่อนลำดับที่ 2 มีโหนด 52 โหนด ชั้นซ่อนลำดับที่ 9 มีโหนด 6,656 โหนด รวมมีพารามิเตอร์ที่ใช้ทั้งสิ้นกว่า 29 ล้านพารามิเตอร์ รายละเอียดดังตารางที่ 15

ตารางที่ 15 รายละเอียดโครงข่ายประสาทเทียมเชิงลึก

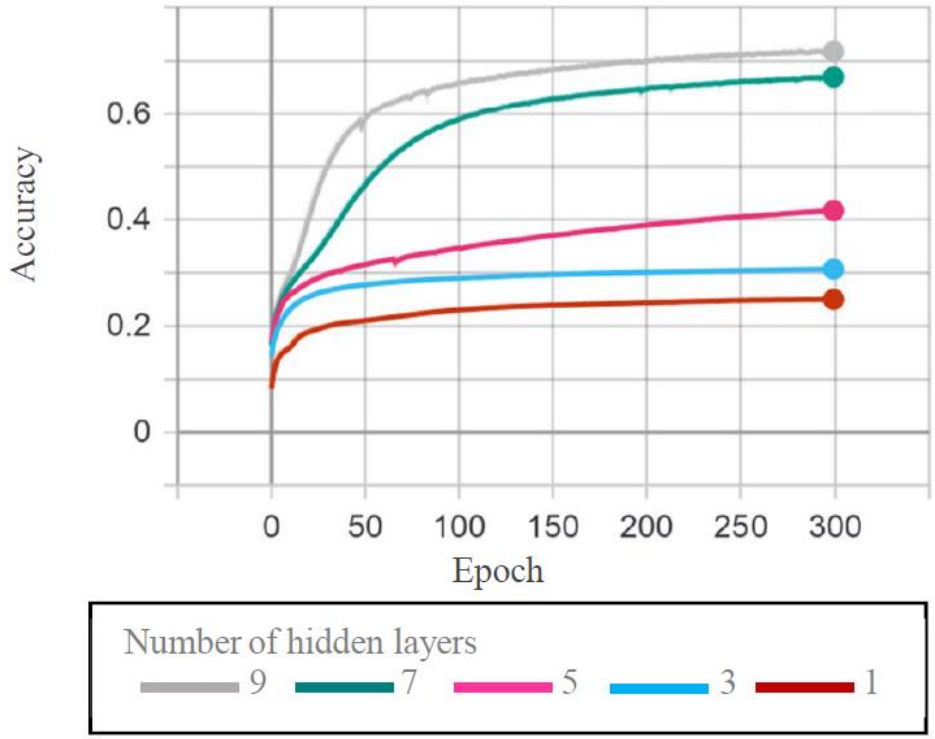
Layer	Output Shape	Parameter#
Input_1 (Dense)	(None, 13)	0
dense	(None, 26)	364
batch_normalization_v1	(None, 26)	104
dense_1	(None, 52)	1,404
batch_normalization_v1_1	(None, 52)	208
dense_2	(None, 104)	5,512
batch_normalization_v1_2	(None, 104)	416
dense_3	(None, 208)	21,840
batch_normalization_v1_3	(None, 208)	832
dense_4	(None, 416)	86,944
batch_normalization_v1_4	(None, 416)	1,664
dense_5	(None, 832)	346,944
batch_normalization_v1_5	(None, 832)	3,328
dense_6	(None, 1664)	1,386,112
batch_normalization_v1_6	(None, 1664)	3356
dense_7	(None, 3328)	5,541,120
batch_normalization_v1_7	(None, 3328)	13,312

Layer	Output Shape	Parameter#
dense_8	(None, 6656)	22,157,824
dense_9	(None, 17)	113,169

จากการฝึกตัวแบบการทำนายรายวิชาโดยใช้วิธีการเรียนรู้ของเครื่องแบบเชิงลึก (Deep Learning) เพื่อหาตัวแบบที่เหมาะสมที่สุดด้วยการกำหนดค่าการเรียนรู้เป็น 1e-3 กับจำนวนชั้นซ่อนทั้งห้ารูปแบบ แสดงดังภาพที่ 21 พบว่าตัวแบบจำนวนชั้นซ่อนแบบ 9 ชั้นให้ผลความถูกต้องในขั้นตอนการฝึกสอน (Training) ที่ดีที่สุดคือ 70% และจากการกำหนดค่าการเรียนรู้เป็น 5e-4 กับจำนวนชั้นซ่อนทั้งห้ารูปแบบ แสดงดังภาพที่ 22 พบว่าตัวแบบจำนวนชั้นซ่อนแบบ 9 ชั้นให้ผลความถูกต้องในขั้นตอนการฝึกสอน (Training) ที่ดีที่สุดคือ 72%



ภาพที่ 21 ค่าความถูกต้องในขั้นตอนการฝึกสอน (Training) ตัวแบบการทำนายรายวิชาด้วยค่าการเรียนรู้เป็น 1e-3 จำแนกตามจำนวนชั้นซ่อน โดยชั้นซ่อนที่ให้ค่าความถูกต้องมากที่สุดคือชั้นซ่อน (Hidden layer) จำนวน 9 ชั้น ให้ผลความถูกต้องเป็น 70%



ภาพที่ 22 ค่าความถูกต้องในขั้นตอนการฝึกสอน (Training) ตัวแบบการทำนายรายวิชาด้วยค่าการเรียนรู้เป็น 5e-4 จำแนกตามจำนวนชั้นซ่อน โดยชั้นซ่อนที่ให้ค่าความถูกต้องมากที่สุดคือชั้นซ่อน (Hidden layer) จำนวน 9 ชั้น ให้ผลความถูกต้องเป็น 72%

เมื่อพิจารณาประสิทธิภาพเชิงเวลา แสดงดังตารางที่ 16 ปรากฏว่าจำนวนชั้นซ่อน 7 ชั้น ค่าการเรียนรู้เป็น 1e-3 และจำนวนรอบที่ 250 Epoch ใช้เวลาในการเรียนรู้เป็น 72 นาที 7 วินาทีซึ่งเป็นค่าพารามิเตอร์ที่เหมาะสมที่สุดในการสร้างตัวแบบสำหรับระบบแนะนำรายวิชา เมื่อพิจารณาจากค่าความถูกต้องในขั้นตอนการฝึกสอน (Training) ร่วมกับเวลาที่ใช้ในการประมวลผล แม้ว่าในขั้นตอนการฝึกสอน (Training) จำนวนชั้นซ่อนแบบ 9 ชั้นจะให้ผลความถูกต้องที่สูงที่สุด แต่เมื่อพิจารณาเวลาที่ใช้ในการประมวลผลพบว่าจำนวนชั้นซ่อนแบบ 9 ชั้นใช้เวลาในการประมวลมากถึง 9 เท่าเมื่อเทียบเวลากับจำนวนชั้นซ่อนแบบ 7 ชั้น สำหรับขั้นตอนของการทดสอบ (Testing) พบว่าเมื่อนำตัวแบบที่ได้ไปทดสอบกับข้อมูลชุดทดสอบ (Test data) ตัวแบบให้ผลความถูกต้องที่ 22%

ตารางที่ 16 ผลการเปรียบเทียบเชิงเวลาในขั้นตอนการเรียนรู้ เมื่อกำหนดค่าพารามิเตอร์ดังนี้ หนึ่ง) กำหนดค่าการเรียนรู้แบบ 1e-3 และ 5e-4 สอง) กำหนดจำนวนชั้นซ่อนเป็นห้ารูปแบบคือจำนวนชั้นซ่อนแบบ 1, 3, 5, 7 และ 9 ชั้น และ สาม) กำหนดจำนวนรอบในการประมวลผลเป็น 250, 300 Epoch

Number of hidden layers	Learning rate = 1e-3				Learning rate = 5e-4			
	250 epoch		300 epoch		250 epoch		300 epoch	
	minute	second	minute	second	minute	second	minute	second
9	654	7	785	7	654	56	785	0
7	72	7	86	38	97	39	105	17
5	21	44	26	6	22	39	27	12
3	11	44	14	5	12	29	15	0
1	8	51	10	38	8	49	10	35

2) สรุปผลการพัฒนาตัวแบบโดยใช้โครงข่ายประสาทเทียมเชิงลึก (Deep neural network, DNN)

ตัวแบบสำหรับระบบแนะนำรายวิชาในสภาพแวดล้อมของ MOOCs โดยใช้ตัวแบบของการเรียนรู้ของเครื่องแบบลึก (Deep Learning) ด้วยโครงสร้างแบบ Multilayer Perceptron ซึ่งรูปแบบการเรียนรู้ของเครื่องแบบลึกนี้เหมาะกับข้อมูลที่มีปริมาณมหาศาล โดยตัวแบบที่เหมาะสมคือตัวแบบที่มีจำนวนชั้นซ่อน 7 ชั้น และค่าการเรียนรู้เป็น 1e-3 เมื่อทำการประมวลผลบน GPU-accelerated ด้วยจำนวนรอบที่เหมาะสมคือ 250 Epoch ตัวแบบที่ได้ให้ค่าความถูกต้องในขั้นตอนการฝึกสอน (Training set) เป็น 70% และเมื่อนำตัวแบบที่ได้ไปทดสอบกับข้อมูลชุดทดสอบ (Test data) พบว่าตัวแบบให้ผลความถูกต้องเป็น 22%

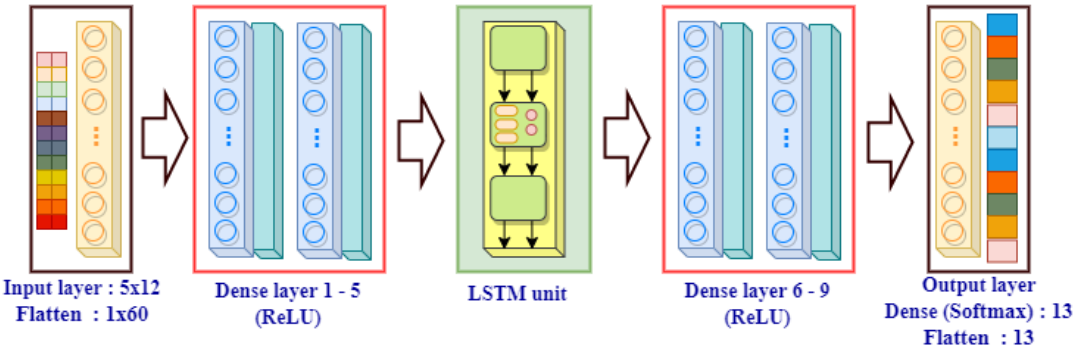
ผลของการดำเนินงานวิจัยแสดงให้เห็นถึงแนวโน้มที่ตัวแบบดังกล่าวสามารถแนะนำรายวิชาให้กับผู้ใช้รายใหม่ได้ แต่จะยังมีสิ่งที่ต้องพัฒนาตัวแบบให้ได้ประสิทธิภาพในขั้นการทดสอบ (Testing) ให้ดีขึ้นได้โดยการใช้ปรับตัวแบบให้มีชั้นของการตัดทิ้งข้อมูลการฝึก (Dropout) เพิ่มเข้าไปในตัวแบบต่อไป

4.1.3 สรุปผลการใช้ขั้นตอนวิธีโครงข่ายประสาทเทียมเชิงลึกร่วมกับหน่วยความจำสั้นระยะยาว

จากล็อคไฟล์ข้อมูล (Event log) การเรียนของผู้เรียนที่นำมาใช้ในระบบแนะนำรายวิชาบนสภาพแวดล้อมบน MOOCs พบว่ามีลักษณะเป็นแบบลำดับ (Sequence) ดังนั้นเทคนิคที่เหมาะสำหรับการประมวลผลกับข้อมูลลักษณะดังกล่าว และเหมาะกับการนำมาสร้างตัวแบบระบบแนะนำรายวิชาคือ โครงข่ายประสาทเทียมเชิงลึก (Deep Neural Network) ร่วมกับ Long Short-Term Memory (LSTM) เพื่อให้ตัวแบบสามารถเรียนรู้และจดจำได้อย่างมีประสิทธิภาพ เทคโนโลยีและขั้นตอนวิธีที่ใช้ในการสร้างตัวแบบระบบแนะนำรายวิชา แสดงดังตารางที่ 17

ตารางที่ 17 เทคโนโลยีและขั้นตอนวิธีที่ใช้ในการสร้างตัวแบบระบบแนะนำรายวิชา

วัตถุประสงค์	Technology	เหตุผลสนับสนุน
การจัดเก็บข้อมูล	Apache hadoop	Repliation, Fault Tolerance, Scalability, Data Integrity, Data Locality and High Throughput [23, 24]
เครื่องมือสำหรับพัฒนาตัวแบบระบบแนะนำ	TensorFlow, Keras, Python	- TensorFlow, Keras เป็นไลบรารี (Library) ที่มีประสิทธิภาพสูงในการนำไปสร้างตัวแบบ - Python เป็นภาษาโปรแกรมที่นิยมสำหรับทำ Machine Learning
หน่วยประมวลผล	Geforce RTX 2080 Ti Graphics Card	- ทำให้การประมวลผลเร็วขึ้น
ขั้นตอนวิธีสำหรับการประมวลผล	โครงข่ายประสาทเทียมเชิงลึก (Deep Neural Network) ร่วมกับ Long Short-Term Memory (LSTM)	- โครงข่ายประสาทเทียมเชิงลึกสามารถค้นหาสิ่งที่ซ่อนในข้อมูลมิติสูง ๆ ได้ดี - Long Short-Term Memory (LSTM) เหมาะกับการประมวลผลข้อมูลแบบลำดับ [11, 25, 26]

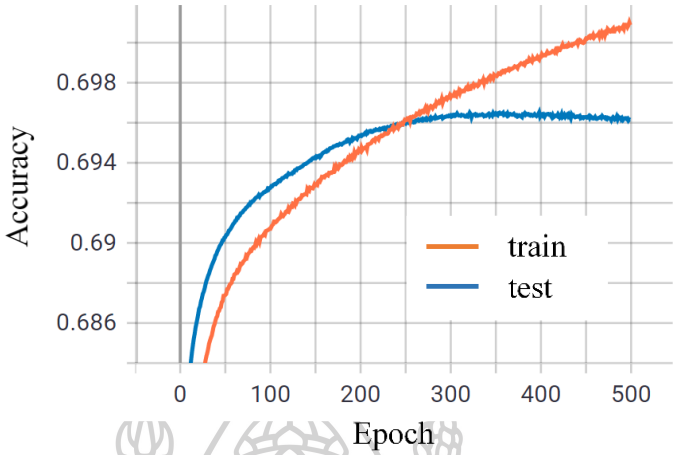


ภาพที่ 23 โครงข่ายประสาทเทียมเชิงลึก (Deep neural network, DNN) ร่วมกับหน่วยความจำสั้นระยะยาว Long Short-Term Memory (LSTM)

จากการพัฒนาตัวแบบระบบแนะนำรายวิชาโดยใช้โครงข่ายประสาทเทียมเชิงลึก (Deep Neural Network) ร่วมกับ Long Short-Term Memory (LSTM) บนการทำงานแบบคลัสเตอริ่ง (Clustering) และการทำงานแบบไม่ใช่คลัสเตอริ่ง (Non-clustering) ที่ฝึกสอนตัวแบบด้วยวิธี “fit method” และวิธี “fit-generator method” พบว่าในขั้นตอนการทดสอบตัวแบบ (Model testing) ตัวแบบ MCR-NF, MCR-NG และ MCR-CG ช่วง epoch ที่ 301 - 350 ค่าความแม่นยำของตัวแบบมีแนวโน้มการเปลี่ยนแปลงที่ไม่เพิ่มขึ้น แม้ว่าค่าความแม่นยำของตัวแบบในขั้นตอนการฝึกสอนตัวแบบ (Model training) จะยังคงมีแนวโน้มเพิ่มขึ้นก็ตาม ตามกราฟภาพที่ 24 ภาพที่ 25 และภาพที่ 26 ดังนั้นผู้วิจัยจึงเลือกช่วง epoch ที่ 301 - 350 สำหรับการเปรียบเทียบความแม่นยำและประสิทธิเชิงเวลาของตัวแบบ แสดงดังตารางที่ 18 และตารางที่ 19

จากการพัฒนาตัวแบบระบบแนะนำรายวิชาโดยใช้โครงข่ายประสาทเทียมเชิงลึก (Deep Neural Network) ร่วมกับ Long Short-Term Memory (LSTM) บนการทำงานแบบคลัสเตอริ่ง (Clustering) และการทำงานแบบไม่ใช่คลัสเตอริ่ง (Non-clustering) ที่ฝึกสอนตัวแบบด้วยวิธี “fit method” และวิธี “fit-generator method” พบว่าในขั้นตอนการทดสอบตัวแบบ (Model testing) ตัวแบบ MCR-NF, MCR-NG และ MCR-CG ช่วง epoch ที่ 301 - 350 ค่าความแม่นยำของตัวแบบมีแนวโน้มการเปลี่ยนแปลงที่ไม่เพิ่มขึ้น แม้ว่าค่าความแม่นยำของตัวแบบในขั้นตอนการฝึกสอนตัวแบบ (Model training) จะยังคงมีแนวโน้มเพิ่มขึ้นก็ตาม ตามกราฟภาพที่ 24 ภาพที่ 25 และภาพที่ 26 ดังนั้นผู้วิจัยจึงเลือกช่วง epoch ที่ 301 - 350 สำหรับการเปรียบเทียบความแม่นยำและประสิทธิเชิงเวลาของตัวแบบ แสดงดังตารางที่ 18 และตารางที่ 19

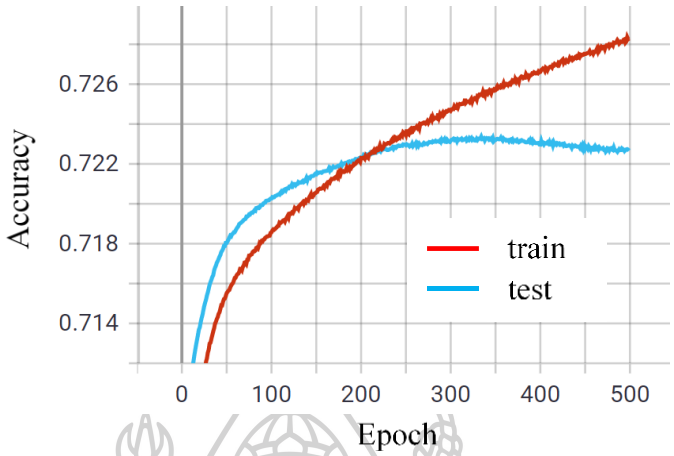
1) ประสิทธิภาพเชิงเวลาและค่าความแม่นยำของตัวแบบ MCR-NF



ภาพที่ 24 ประสิทธิภาพเชิงเวลาและค่าความแม่นยำของตัวแบบ MCR-NF

จากขั้นตอนการทำงานตัวแบบ “MCR-NF” แสดงดังกราฟภาพที่ 24 แสดงให้เห็นว่าขั้นตอน “Model building” ที่ใช้ “fit method” จะใช้ชุดข้อมูลเดิมจาก “RAM” ในทุกรอบการฝึกตัวแบบ (จำนวน 500 epoch) โดยทำซ้ำในขั้นตอนที่ 3 ไปจนครบรอบการฝึกสอนตัวแบบ ส่งผลโดยตรงต่อค่าความแม่นยำที่ไม่สูงเท่าที่ควรและยังคงค่าความแม่นยำต่อเนื่องไปเรื่อย ๆ ของตัวแบบ “MCR-NF” เพราะตัวแบบจะใช้ข้อมูลเดิมในการฝึกสอน จึงส่งผลให้มีการจำข้อมูลเกิดขึ้นสังเกตได้จากกราฟแสดงดังภาพที่ 24 เห็นได้ว่าค่าความแม่นยำในขั้นตอนการทดสอบตัวแบบ (Model testing) (เส้นสีน้ำเงิน) มีค่าความแม่นยำที่สูงกว่าในขั้นตอนการฝึกสอนตัวแบบ (Model training) (เส้นสีส้ม) ตั้งแต่ในรอบการทดสอบช่วงแรก ๆ จนถึงประมาณช่วงที่ 250 แต่ตั้งแต่ช่วงที่ 250 เป็นต้นไปค่าความแม่นยำในขั้นตอนการทดสอบตัวแบบ (Model testing) จะค่อย ๆ คงที่

2) ประสิทธิภาพเชิงเวลาและค่าความแม่นยำของตัวแบบ MCR-NG

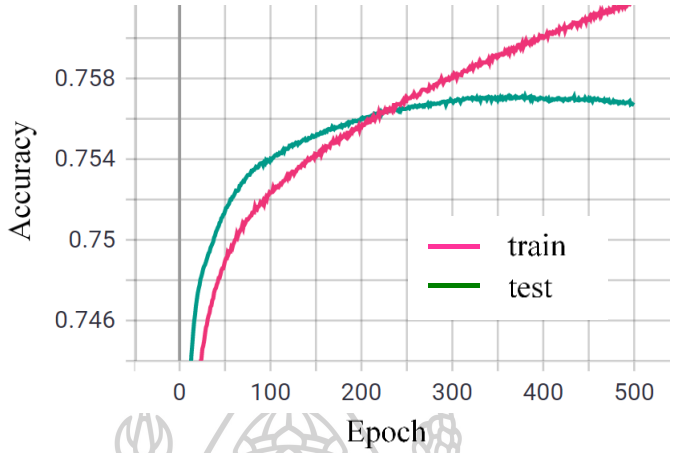


ภาพที่ 25 ประสิทธิภาพเชิงเวลาและค่าความแม่นยำของตัวแบบ MCR-NG

จากขั้นตอนการทำงานตัวแบบ “MCR-NG” แสดงดังกราฟภาพที่ 25 เริ่มจากส่วนของ “Model building” จะติดต่อกับ “Generator data module” เพื่อเตรียมชุดข้อมูลสำหรับการฝึกสอนตัวแบบ โดย “Generator data module” จะไปดึงข้อมูลจาก “Data storage” และนำข้อมูลที่ได้ไปเก็บไว้ที่ “RAM” และ “Model building” จะเรียก “fit-generator method” ให้ทำงาน โดยทำซ้ำในขั้นตอนที่ 1 - 4 ไปจนครบรอบการฝึกตัวแบบ (จำนวน 500 epoch) เห็นได้ว่าวิธี “fit-generator method” จะสับเปลี่ยนชุดข้อมูล (shuffle data) ให้ลำดับข้อมูลไม่เหมือนกันทุกครั้งในทุกรอบการฝึกสอนตัวแบบ

จากกราฟแสดงดังภาพที่ 25 เห็นได้ว่าค่าความแม่นยำในขั้นตอนการทดสอบตัวแบบ (Model testing) (เส้นสีฟ้า) มีค่าความแม่นยำที่สูงกว่าในขั้นตอนการฝึกสอนตัวแบบ (Model training) (เส้นสีแดง) ในรอบการทดสอบช่วงแรก ๆ ถึงประมาณช่วงที่ 200 และตั้งแต่ช่วงที่ 200 เป็นต้นไปค่าความแม่นยำในขั้นตอนการทดสอบตัวแบบ (Model testing) จะค่อย ๆ คงที่ แต่ก็ให้ค่าความแม่นยำของตัวแบบในขั้นตอนการฝึกสอนและขั้นตอนทดสอบตัวแบบที่สูงกว่า MCR-NF

3) ประสิทธิภาพเชิงเวลาและค่าความแม่นยำของตัวแบบ MCR-CG



ภาพที่ 26 ประสิทธิภาพเชิงเวลาและค่าความแม่นยำของตัวแบบ MCR-CG

จากขั้นตอนการทำงานตัวแบบ “MCR-CG” แสดงดังกราฟภาพที่ 26 เริ่มจากส่วนของ “Model building” จะติดต่อกับ “Generator data module” เพื่อเตรียมชุดข้อมูลสำหรับการฝึกสอนตัวแบบ โดย “Generator data module” จะไปดึงข้อมูลจาก “Hadoop cluster” และนำข้อมูลที่ได้ไปเก็บไว้ที่ “RAM” และ “Model building” จะเรียกวิธี “fit-generator method” ให้ทำงาน โดยทำซ้ำในขั้นตอนที่ 1 - 4 ไปจนครบรอบการฝึกตัวแบบ (จำนวน 500 epoch) เห็นได้ว่าวิธี “fit-generator method” จะสับเปลี่ยนชุดข้อมูล (shuffle data) ให้ลำดับข้อมูลไม่เหมือนกันทุกครั้งในทุกรอบการฝึกสอนตัวแบบ และด้วยการจัดเก็บข้อมูลไว้ใน Hadoop cluster จึงส่งผลดีในเรื่องการบริหารจัดการข้อมูลภายในคลัสเตอริ่ง (Clustering) และการขยายขนาดแบบแนวโดยที่ไม่กระทบต่อระบบ

จากกราฟแสดงดังภาพที่ 26 เห็นได้ว่าค่าความแม่นยำในขั้นตอนการทดสอบตัวแบบ (Model testing) (เส้นสีเขียว) มีค่าความแม่นยำที่สูงกว่าในขั้นตอนการฝึกสอนตัวแบบ (Model training) (เส้นสีชมพู) ในรอบการทดสอบช่วงแรก ๆ ถึงประมาณช่วงที่ 200 และตั้งแต่ช่วงที่ 200 เป็นต้นไปค่าความแม่นยำในขั้นตอนการทดสอบตัวแบบ (Model testing) จะค่อย ๆ คงที่ แต่ก็ให้ค่าความแม่นยำของตัวแบบในขั้นตอนการฝึกสอนและขั้นตอนทดสอบตัวแบบที่สูงที่สุดกว่าทั้งแบบ MCR-NF และ MCR-NG

จากตารางที่ 18 พิจารณาประสิทธิภาพด้านความแม่นยำของตัวแบบ พบว่าช่วง epoch ที่ 301 - 350 ให้ค่าความแม่นยำ ที่ 69.785 , 72.526 และ 75.861 ตามลำดับ โดยค่าความแม่นยำที่ดีที่สุดได้จากตัวแบบ MCR-CG สูงถึง 75.861 เนื่องจากตัวแบบ MCR-CG ถูกฝึกสอนด้วยวิธี “fit-

generator method” ซึ่งจะมีกระบวนการนำเข้าข้อมูลจาก “Hadoop cluster” และทำการสับเปลี่ยน (Shuffle) ชุดข้อมูลทุกครั้งในทุกรอบการฝึกสอนตัวแบบ จึงส่งผลทำให้ประสิทธิภาพของตัวแบบ MCR-CG สูงกว่า MCR-NF และ MCR-NG

จากตารางที่ 19 เมื่อพิจารณาประสิทธิภาพเชิงเวลาของตัวแบบ พบว่าช่วง epoch ที่ 301 - 350 ตัวแบบ MCR-NF, MCR-NG และ MCR-CG มีค่าเวลารวมสะสมในขั้นตอนการฝึกสอนตัวแบบ (Model training) เป็น 18:57:24 ชั่วโมง, 17:49:54 ชั่วโมง และ 11:33:12 ชั่วโมง โดยประสิทธิภาพเชิงเวลาที่ดีที่สุดคือตัวแบบ MCR-CG ซึ่งเป็นตัวแบบที่ใช้เวลาในขั้นตอนการฝึกสอนตัวแบบ (Model training) น้อยที่สุด และเมื่อเปรียบเทียบประสิทธิภาพเชิงเวลาของตัวแบบ MCR-CG กับ MCR-NF และ MCR-NG พบว่าประสิทธิภาพเชิงเวลาของตัวแบบ MCR-CG เมื่อเปรียบเทียบกับตัวแบบ MCR-NF ใช้เวลาน้อยกว่าประมาณ 444 นาที และประสิทธิภาพเชิงเวลาของตัวแบบ MCR-CG เมื่อเทียบกับตัวแบบ MCR-NG ใช้เวลาน้อยกว่าประมาณ 376 นาที

ตัวแบบ MCR-CG มีประสิทธิภาพเชิงเวลาและค่าความแม่นยำที่ดีที่สุดเมื่อเทียบกับตัวแบบ MCR-NF และ MCR-NG จากกราฟภาพที่ 24 ภาพที่ 25 และภาพที่ 26 แสดงให้เห็นถึงประสิทธิภาพของตัวแบบ ณ ตำแหน่งช่วง epoch ที่ 301 - 350 เป็นจุดที่ดีที่สุดสำหรับการเลือกตัวแบบที่ได้รับการฝึกไปใช้งานจริง ซึ่งจะเห็นได้ว่า ณ ตำแหน่ง epoch ที่ 350 ขึ้นไปแนวโน้มของการฝึกตัวแบบจะแสดงลักษณะของการเกิด Overfit ที่สูงขึ้น

สรุปประสิทธิภาพของการพัฒนาระบบแนะนำรายวิชาด้วยขั้นตอนวิธีต่าง ๆ แสดงดังตารางที่ 20 ตารางที่ 21 และตารางที่ 22

มหาวิทยาลัยศิลปากร

ตารางที่ 18 ค่าความความแม่นยำของตัวแบบในขั้นตอนการฝึกสอนและขั้นตอนการทดสอบ

Epoch	Training precision			Testing precision		
	MCR-NF	MCR-NG	MCR-CG	MCR-NF	MCR-NG	MCR-CG
001 - 050	67.969 ± 1.172	70.712 ± 1.427	73.566 ± 2.952	68.549 ± 0.654	71.236 ± 0.877	74.310 ± 1.487
051 - 100	68.935 ± 0.095	71.724 ± 0.089	75.091 ± 0.095	69.176 ± 0.070	71.937 ± 0.063	75.303 ± 0.074
101 - 150	69.190 ± 0.063	71.964 ± 0.058	75.334 ± 0.056	69.355 ± 0.043	72.093 ± 0.035	75.466 ± 0.034
151 - 200	69.381 ± 0.049	72.145 ± 0.047	75.496 ± 0.042	69.485 ± 0.032	72.191 ± 0.024	75.561 ± 0.023
201 - 250	69.539 ± 0.043	72.291 ± 0.040	75.632 ± 0.037	69.571 ± 0.018	72.269 ± 0.019	75.631 ± 0.015
251 - 300	69.672 ± 0.037	72.415 ± 0.035	75.754 ± 0.034	69.619 ± 0.011	72.305 ± 0.010	75.671 ± 0.012
301 - 350	69.785 ± 0.031	72.526 ± 0.031	75.861 ± 0.032	69.638 ± 0.005	72.324 ± 0.006	75.698 ± 0.006
351 - 400	69.885 ± 0.028	72.622 ± 0.025	75.962 ± 0.029	69.640 ± 0.005	72.315 ± 0.008	75.704 ± 0.006
401 - 450	69.976 ± 0.024	72.711 ± 0.025	76.052 ± 0.026	69.634 ± 0.006	72.295 ± 0.009	75.697 ± 0.005
451 - 500	70.056 ± 0.021	72.791 ± 0.024	76.140 ± 0.024	69.622 ± 0.006	72.274 ± 0.008	75.687 ± 0.008

ตารางที่ 19 ประสิทธิภาพเชิงเวลาของตัวแบบในขั้นตอนการฝึกสอน

Epoch	Training precision			Training time (hh:mm:ss)		
	MCR-NF	MCR-NG	MCR-CG	MCR-NF	MCR-NG	MCR-CG
001 - 050	67.969 ± 1.172	70.712 ± 1.427	73.566 ± 2.952	02:41:03	02:33:36	01:38:00
051 - 100	68.935 ± 0.095	71.724 ± 0.089	75.091 ± 0.095	05:24:48	05:06:44	03:16:20
101 - 150	69.190 ± 0.063	71.964 ± 0.058	75.334 ± 0.056	08:08:26	07:38:15	04:55:37
151 - 200	69.381 ± 0.049	72.145 ± 0.047	75.496 ± 0.042	10:50:38	10:09:48	06:34:51
201 - 250	69.539 ± 0.043	72.291 ± 0.040	75.632 ± 0.037	13:31:39	12:44:21	08:14:30
251 - 300	69.672 ± 0.037	72.415 ± 0.035	75.754 ± 0.034	16:14:19	15:17:34	09:54:13
301 - 350	69.785 ± 0.031	72.526 ± 0.031	75.861 ± 0.032	18:57:24	17:49:54	11:33:12
351 - 400	69.885 ± 0.028	72.622 ± 0.025	75.962 ± 0.029	21:40:26	20:21:31	13:11:49
401 - 450	69.976 ± 0.024	72.711 ± 0.025	76.052 ± 0.026	00:21:27	22:54:02	14:50:21
451 - 500	70.056 ± 0.021	72.791 ± 0.024	76.140 ± 0.024	02:59:24	01:24:43	16:27:30

ตารางที่ 20 สรุปผลประสิทธิภาพของการแนะนำรายวิชาด้วยขั้นตอนวิธีของกฎความสัมพันธ์ (Association rule) แบบ Apriori

ขั้นตอนวิธี	Support	Confidence
Apriori	70.000%	43.500%

ตารางที่ 21 สรุปผลประสิทธิภาพของการแนะนำรายวิชาด้วยขั้นตอนโครงข่ายประสาทเทียมเชิงลึก (Deep neural network, DNN)

ขั้นตอนวิธี	Precision		Accuracy		Recall	
	Training	Testing	Training	Testing	Training	Testing
DNN	70.04%	22.03%	-	-	-	-

ตารางที่ 22 สรุปผลประสิทธิภาพของการแนะนำรายวิชาด้วยขั้นตอนโครงข่ายประสาทเทียมเชิงลึก (Deep neural network, DNN) ร่วมกับหน่วยความจำสั้นระยะยาว Long Short-Term Memory (LSTM)

ขั้นตอนวิธี	Precision		Accuracy		Recall	
	Training	Testing	Training	Testing	Training	Testing
DNN + LSTM (MCR-CG)	75.86%	75.70%	76.00%	76.00%	81.48%	78.85%

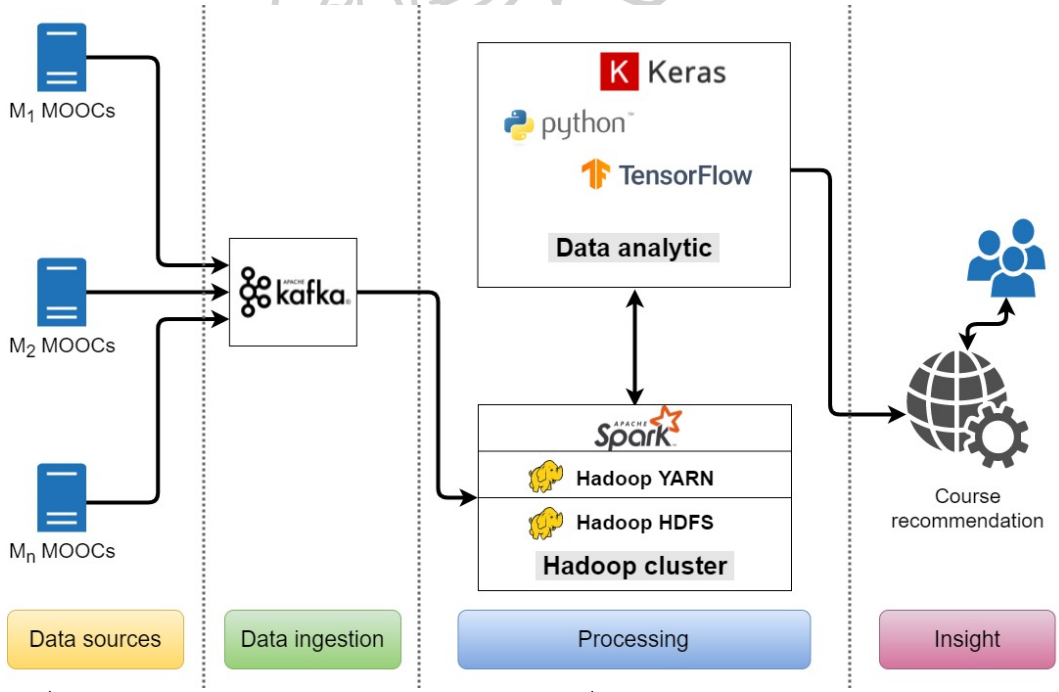
4.2 ผลการปรับปรุงเทคนิคระบบแนะนำรายวิชาสำหรับข้อมูลขนาดใหญ่

ผู้วิจัยเลือกที่จะสร้าง Hadoop cluster ภายในเครื่องเสมือนจริง (Virtual Machine) ขึ้นมาจำนวน 4 เครื่อง เพื่อใช้เป็นแพลตฟอร์มสำหรับการทดสอบระบบประกอบด้วยโหนดหลัก (Master node) จำนวน 1 เครื่อง และมีโหนดรอง (Slave node) จำนวน 3 เครื่อง มีหน่วยประมวลผลเป็น Intel(R) Xeon(R) CPU E5-2670 v2 @2.5GHz รายละเอียดคุณลักษณะของ Hadoop cluster แสดงดังตารางที่ 23

ตารางที่ 23 คุณลักษณะของ Hadoop cluster

Node	CPU	Memory	Disk
Master	Intel ® Xeon ® E5-2650 v2 2.60 GHz	16 GB	150 GB
Slave 1 – Slave 3	CPU 2 core	8 GB	100 GB

จากสถาปัตยกรรมตัวแบบระบบแนะนำรายวิชาที่ได้ออกแบบไว้ มีการใช้ Apache Kafka Hadoop cluster สำหรับเป็น storage phase และใช้ตัวแบบโครงข่ายประสาทเทียมเชิงลึก (Deep Neural Network) ร่วมกับ Long Short-Term Memory (LSTM) แสดงดังภาพที่ 27 บนการทำงานแบบคลัสเตอริ่ง (Clustering) สำหรับระบบแนะนำรายวิชามีประสิทธิภาพเชิงเวลาและค่าความแม่นยำสูงที่สุด

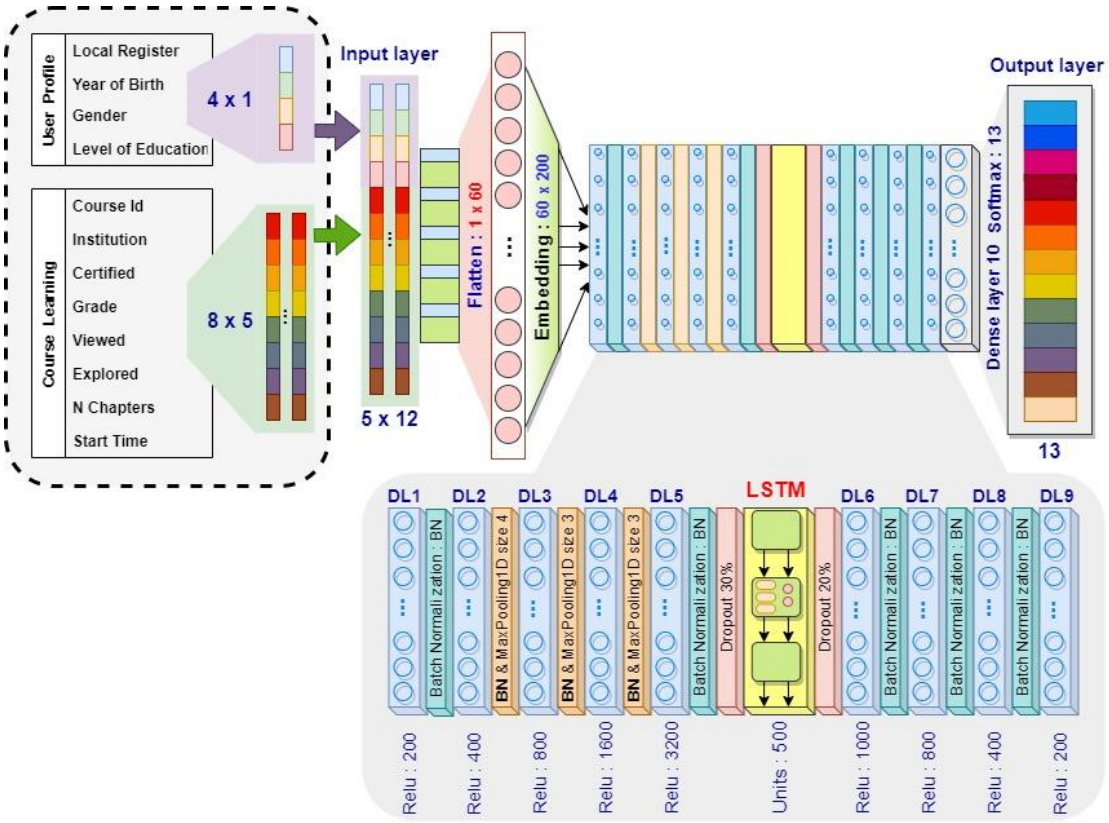


ภาพที่ 27 สถาปัตยกรรมตัวแบบระบบแนะนำรายวิชาที่ผู้วิจัยนำเสนอ

ผลการพัฒนาตัวแบบด้วยโครงข่ายประสาทเทียมเชิงลึก (Deep neural network, DNN) ร่วมกับหน่วยความจำสั้นระยะยาว Long Short-Term Memory (LSTM) กับการทำงานแบบคลัส

เตอริ่ง (Clustering) ที่ฝึกสอนด้วยวิธี fit-generator ผลการวิจัยแสดงให้เห็นว่าตัวแบบ MCR-CG มีประสิทธิภาพเชิงเวลาและค่าความถูกต้องที่ดีที่สุดเมื่อเทียบกับตัวแบบ MCR-NF และ MCR-NG โดยการทดสอบกับข้อมูลจริงของ Harvard and MIT published the edX ประสิทธิภาพเชิงด้านความถูกต้องพบว่าในขั้นตอนการฝึกสอนตัวแบบ (Model training) ให้ค่าความถูกต้องสูงถึง 75.9% และในขั้นตอนการทดสอบตัวแบบ (Model testing) ให้ค่าความถูกต้องสูงถึง 75.7% สำหรับประสิทธิภาพเชิงเวลาพบว่าในขั้นตอนการฝึกสอนตัวแบบ (Model training) MCR-CG ใช้เวลาน้อยกว่าประมาณ 444 นาทีเมื่อเปรียบเทียบกับตัวแบบ MCR-NF และตัวแบบ MCR-CG ใช้เวลาน้อยกว่าประมาณ 376 นาทีเมื่อเทียบกับตัวแบบ MCR-NG ด้วยความสามารถของ Long Short-Term Memory (LSTM) ในการคำนวณผลลัพธ์ที่ได้โดยคำนวณจากโหนดก่อนหน้าแล้วส่งต่อเป็นข้อมูลเข้าของโหนดถัดไป ทำให้ Long Short-Term Memory (LSTM) สามารถนำข้อมูลก่อนหน้า (ข้อมูลในอดีต) มาใช้ในการทำนายสิ่งที่อาจจะเกิดขึ้นในอนาคตได้ และการใช้การฝึกสอนตัวแบบด้วยวิธี “fit-generator” ที่ทำงานร่วมกับ Hadoop cluster ทำให้ข้อมูลเข้ามีการสับเปลี่ยน (Shuffle) ของชุดข้อมูลทุกครั้งในทุกรอบการฝึกสอนตัวแบบ จึงส่งผลทำให้ประสิทธิภาพของตัวแบบ MCR-CG ดีกว่าแบบอื่น ๆ และเป็นแนวทางที่ดีสำหรับการพัฒนาตัวแบบระบบคำแนะนำรายวิชา และที่สำคัญ MCR-CG ยังสามารถรองรับการปรับเพิ่มขนาดของพื้นที่การจัดเก็บข้อมูลในรูปแบบแนวนอน (Scale-out) ได้ และรองรับกับข้อมูลที่มีขนาดใหญ่ได้

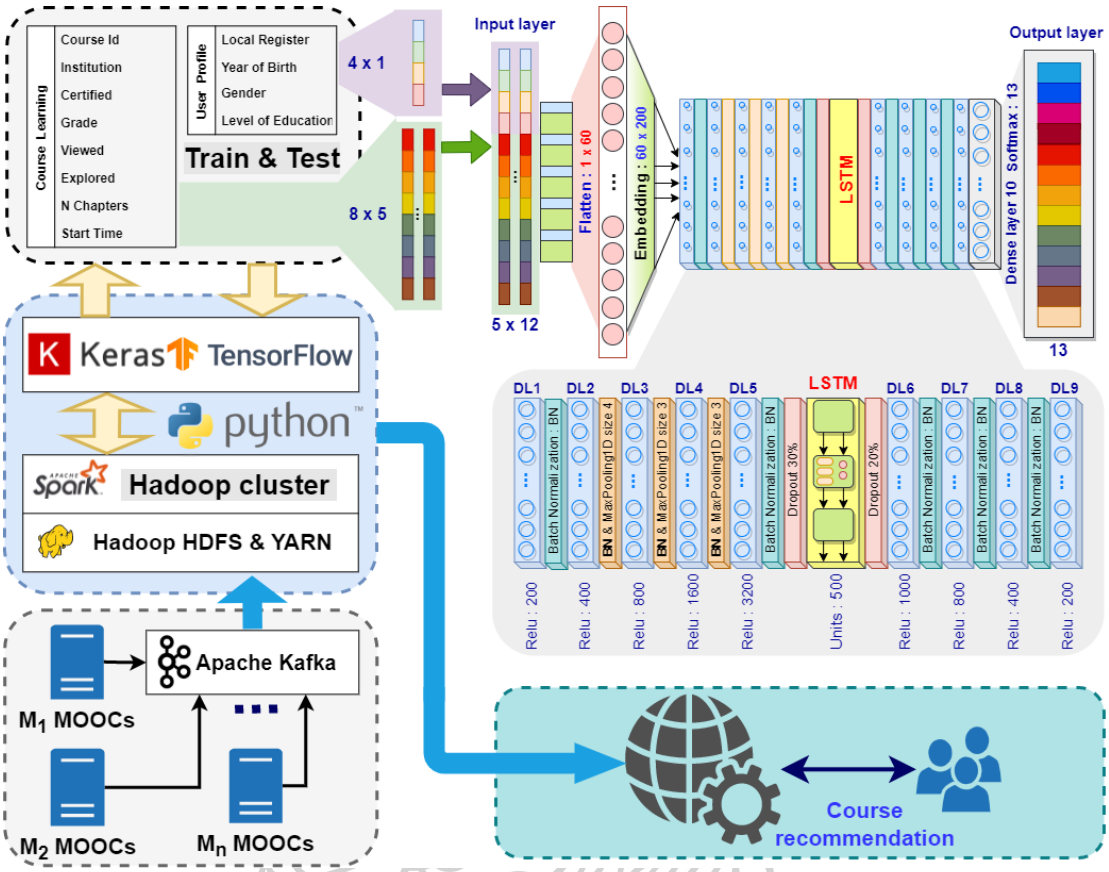
มหาวิทยาลัยศิลปากร



ภาพที่ 28 ตัวแบบระบบแนะนำรายวิชาโดยใช้โครงข่ายประสาทเทียมเชิงลึก (Deep Neural Network) ร่วมกับ Long Short-Term Memory (LSTM)

4.3 ผลการพัฒนาตัวแบบระบบแนะนำรายวิชา

จากการพัฒนาตัวแบบระบบแนะนำรายวิชาในรายวิชาออนไลน์แบบเปิดสู่มวลชน (MOOCs) ให้กับผู้เรียน พบว่าการใช้โครงข่ายประสาทเทียมเชิงลึก (Deep Neural Network) ร่วมกับ Long Short-Term Memory (LSTM) บนการทำงานแบบคลัสเตอริ่ง (Clustering) แสดงสถาปัตยกรรมและขั้นตอนการประมวลผลข้อมูลตัวแบบแสดงดัง รายละเอียดดังนี้



ภาพที่ 29 สถาปัตยกรรมและขั้นตอนการประมวลผลข้อมูลตัวแบบระบบแนะนำรายวิชาในรายวิชาออนไลน์แบบเปิดสู่มวลชน (MOOCs)

4.3.1 ขั้นตอนการนำเข้าข้อมูลจากระบบ MOCC ต่าง ๆ ผู้วิจัยเลือกใช้ Apache Kafka เป็นเครื่องมือสำหรับดำเนินการกลไกการนำเข้าสกัดและส่งต่อข้อมูล (Extract-Transform-Load : ETL) เพื่อจัดเก็บบนระบบ Hadoop HDFS ซึ่งใช้ YARN และ Apache Spark เป็นเครื่องมือสำหรับการประมวลผลข้อมูลแบบ Real-time ด้านการจัดการบันทึกข้อมูลตามโครงสร้างที่กำหนดบนระบบ Hadoop Cluster ในรูปแบบ HDFS และข้อมูลที่จัดเก็บใน HDFS จะถูกเรียกใช้งานด้วยโปรแกรมภาษา python เพื่อนำไปใช้สำหรับขั้นตอนการฝึกสอนและทดสอบตัวแบบระบบแนะนำรายวิชา

4.3.2 สำหรับขั้นตอนนี้จะนำข้อมูลจาก HDFS และดำเนินการจัดเตรียมข้อมูลแสดงดังภาพที่ 32 สำหรับส่งต่อเพื่อใช้งานขั้นตอนฝึกสอนและทดสอบตัวแบบ โดยใช้โปรแกรมภาษา Python ร่วมกับ Keras framework เป็นเครื่องมือสร้างตัวแบบแสดงตัวอย่างชุดคำสั่งของโปรแกรมดังภาพที่ 28 โดยใช้ TensorFlow เป็นเครื่องมือการบริหารจัดการการเรียนรู้ของเครื่องเพื่อฝึกสอนและทดสอบตัวแบบระบบแนะนำรายวิชา

4.3.3 จากภาพที่ 29 แสดงตัวแบบระบบแนะนำรายวิชาโดยใช้โครงข่ายประสาทเทียมเชิงลึก (Deep Neural Network) ร่วมกับ Long Short-Term Memory (LSTM) ส่วนของข้อมูลนำเข้าคือ ประวัติผู้เรียน (User profile) และการเรียนรู้ของผู้เรียน (Course learning) เข้าสู่กระบวนการเตรียมข้อมูล ส่งต่อไปยัง Input layer ขนาด 5 x 12 ส่งต่อไปยัง Flatten layer ขนาด 1 x 60 ส่งต่อไปยัง Embedding layer ขนาด 60 x 200 ส่งต่อไปยัง Dense layer ที่ 1 – 5 เข้าสู่ Long Short-Term Memory (LSTM) unit = 500 ส่งต่อไปยัง Dense layer ที่ 6 – 9 แล้วส่งออกไป Output layer ขนาด 1 x 13

User Profile		USER		u0	u1	u2	u3	u4	u5	u6	u7	u8	u9
	1	Local Register	LR	LR01	LR02	LR03	LR04	LR05	LR06	LR07	LR08	LR09	LR10
	2	Year of Birth	YB	YB01	YB02	YB03	YB04	YB05	YB06	YB07	YB08	YB09	YB10
	3	Gender	GN	GN01	GN02	GN03	GN04	GN05	GN06	GN07	GN08	GN09	GN10
	4	Level of Education	LE	LE01	LE02	LE03	LE04	LE05	LE06	LE07	LE08	LE09	LE10

(a) โครงสร้างตาราง User Profile แสดงการจัดเก็บข้อมูลผู้เรียน u0 – u9

Course Learning		USER		u0	u1	u1	u2	u3	u3	u3	u4	u5	u5	u5	u5	u6	u7	u7	u7	u7	u7	u8	u9
	5	Course Id	CI	CI01	CI02	CI03	CI04	CI05	CI06	CI07	CI08	CI09	CI10	CI11	CI12	CI13	CI14	CI15	CI16	CI17	CI18	CI19	CI20
	6	Institution	IS	IS01	IS02	IS03	IS04	IS05	IS06	IS07	IS08	IS09	IS10	IS11	IS12	IS13	IS14	IS15	IS16	IS17	IS18	IS19	IS20
	7	Certified	CT	CT01	CT02	CT03	CT04	CT05	CT06	CT07	CT08	CT09	CT10	CT11	CT12	CT13	CT14	CT15	CT16	CT17	CT18	CT19	CT20
	8	Grade	Gr	Gr01	Gr02	Gr03	Gr04	Gr05	Gr06	Gr07	Gr08	Gr09	Gr10	Gr11	Gr12	Gr13	Gr14	Gr15	Gr16	Gr17	Gr18	Gr19	Gr20
	9	Viewed	Vi	Vi01	Vi02	Vi03	Vi04	Vi05	Vi06	Vi07	Vi08	Vi09	Vi10	Vi11	Vi12	Vi13	Vi14	Vi15	Vi16	Vi17	Vi18	Vi19	Vi20
	10	Explored	EP	EP01	EP02	EP03	EP04	EP05	EP06	EP07	EP08	EP09	EP10	EP11	EP12	EP13	EP14	EP15	EP16	EP17	EP18	EP19	EP20
	11	N Chapters	NC	NC01	NC02	NC03	NC04	NC05	NC06	NC07	NC08	NC09	NC10	NC11	NC12	NC13	NC14	NC15	NC16	NC17	NC18	NC19	NC20
	12	Start Time	ST	ST01	ST02	ST03	ST04	ST05	ST06	ST07	ST08	ST09	ST10	ST11	ST12	ST13	ST14	ST15	ST16	ST17	ST18	ST19	ST20

(b) โครงสร้างตาราง Course Learning แสดงการจัดเก็บข้อมูลรายวิชาของผู้เรียน u0 – u9

มหาวิทยาลัยศิลปากร

USER		LR	YB	GN	LE	CI	IS	CT	Gr	Vi	EP	NC	ST
u0	1	LR01	YB01	GN01	LE01	CI01	IS01	CT01	Gr01	Vi01	EP01	NC01	ST01
	2	LR01	YB01	GN01	LE01	CI01	IS01	CT01	Gr01	Vi01	EP01	NC01	ST01
	3	LR01	YB01	GN01	LE01	CI01	IS01	CT01	Gr01	Vi01	EP01	NC01	ST01
	4	LR01	YB01	GN01	LE01	CI01	IS01	CT01	Gr01	Vi01	EP01	NC01	ST01
	5	LR01	YB01	GN01	LE01	CI01	IS01	CT01	Gr01	Vi01	EP01	NC01	ST01
u1	1	LR02	YB02	GN02	LE02	CI02	IS02	CT02	Gr02	Vi02	EP02	NC02	ST02
	2	LR02	YB02	GN02	LE02	CI03	IS03	CT03	Gr03	Vi03	EP03	NC03	ST03
	3	LR02	YB02	GN02	LE02	CI02	IS02	CT02	Gr02	Vi02	EP02	NC02	ST02
	4	LR02	YB02	GN02	LE02	CI03	IS03	CT03	Gr03	Vi03	EP03	NC03	ST03
	5	LR02	YB02	GN02	LE02	CI02	IS02	CT02	Gr02	Vi02	EP02	NC02	ST02
u2	1	LR03	YB03	GN03	LE03	CI04	IS04	CT04	Gr04	Vi04	EP04	NC04	ST04
	2	LR03	YB03	GN03	LE03	CI04	IS04	CT04	Gr04	Vi04	EP04	NC04	ST04
	3	LR03	YB03	GN03	LE03	CI04	IS04	CT04	Gr04	Vi04	EP04	NC04	ST04
	4	LR03	YB03	GN03	LE03	CI04	IS04	CT04	Gr04	Vi04	EP04	NC04	ST04
	5	LR03	YB03	GN03	LE03	CI04	IS04	CT04	Gr04	Vi04	EP04	NC04	ST04
u3	1	LR04	YB04	GN04	LE04	CI05	IS05	CT05	Gr05	Vi05	EP05	NC05	ST05
	2	LR04	YB04	GN04	LE04	CI06	IS06	CT06	Gr06	Vi06	EP06	NC06	ST06
	3	LR04	YB04	GN04	LE04	CI07	IS07	CT07	Gr07	Vi07	EP07	NC07	ST07
	4	LR04	YB04	GN04	LE04	CI05	IS05	CT05	Gr05	Vi05	EP05	NC05	ST05
	5	LR04	YB04	GN04	LE04	CI06	IS06	CT06	Gr06	Vi06	EP06	NC06	ST06
u4	1	LR05	YB05	GN05	LE05	CI08	IS08	CT08	Gr08	Vi08	EP08	NC08	ST08
	2	LR05	YB05	GN05	LE05	CI08	IS08	CT08	Gr08	Vi08	EP08	NC08	ST08
	3	LR05	YB05	GN05	LE05	CI08	IS08	CT08	Gr08	Vi08	EP08	NC08	ST08
	4	LR05	YB05	GN05	LE05	CI08	IS08	CT08	Gr08	Vi08	EP08	NC08	ST08
	5	LR05	YB05	GN05	LE05	CI08	IS08	CT08	Gr08	Vi08	EP08	NC08	ST08

(c) โครงสร้างตารางข้อมูลสำหรับส่งเข้าสู่ชั้น Input Layer ของตัวแบบระบบแนะนำรายวิชา

มหาวิทยาลัยศิลปากร

USER		LR	YB	GN	LE	CI	IS	CT	Gr	Vi	EP	NC	ST
u5	1	LR06	YB06	GN06	LE06	CI09	IS09	CT09	Gr09	Vi09	EP09	NC09	ST09
	2	LR06	YB06	GN06	LE06	CI10	IS10	CT10	Gr10	Vi10	EP10	NC10	ST10
	3	LR06	YB06	GN06	LE06	CI11	IS11	CT11	Gr11	Vi11	EP11	NC11	ST11
	4	LR06	YB06	GN06	LE06	CI12	IS12	CT12	Gr12	Vi12	EP12	NC12	ST12
	5	LR06	YB06	GN06	LE06	CI09	IS09	CT09	Gr09	Vi09	EP09	NC09	ST09
u6	1	LR07	YB07	GN07	LE07	CI13	IS13	CT13	Gr13	Vi13	EP13	NC13	ST13
	2	LR07	YB07	GN07	LE07	CI13	IS13	CT13	Gr13	Vi13	EP13	NC13	ST13
	3	LR07	YB07	GN07	LE07	CI13	IS13	CT13	Gr13	Vi13	EP13	NC13	ST13
	4	LR07	YB07	GN07	LE07	CI13	IS13	CT13	Gr13	Vi13	EP13	NC13	ST13
	5	LR07	YB07	GN07	LE07	CI13	IS13	CT13	Gr13	Vi13	EP13	NC13	ST13
u7	1	LR08	YB08	GN08	LE08	CI14	IS14	CT14	Gr14	Vi14	EP14	NC14	ST14
	2	LR08	YB08	GN08	LE08	CI15	IS15	CT15	Gr15	Vi15	EP15	NC15	ST15
	3	LR08	YB08	GN08	LE08	CI16	IS16	CT16	Gr16	Vi16	EP16	NC16	ST16
	4	LR08	YB08	GN08	LE08	CI17	IS17	CT17	Gr17	Vi17	EP17	NC17	ST17
	5	LR08	YB08	GN08	LE08	CI18	IS18	CT18	Gr18	Vi18	EP18	NC18	ST18
u8	1	LR09	YB09	GN09	LE09	CI19	IS19	CT19	Gr19	Vi19	EP19	NC19	ST19
	2	LR09	YB09	GN09	LE09	CI19	IS19	CT19	Gr19	Vi19	EP19	NC19	ST19
	3	LR09	YB09	GN09	LE09	CI19	IS19	CT19	Gr19	Vi19	EP19	NC19	ST19
	4	LR09	YB09	GN09	LE09	CI19	IS19	CT19	Gr19	Vi19	EP19	NC19	ST19
	5	LR09	YB09	GN09	LE09	CI19	IS19	CT19	Gr19	Vi19	EP19	NC19	ST19
u9	1	LR10	YB10	GN10	LE10	CI20	IS20	CT20	Gr20	Vi20	EP20	NC20	ST20
	2	LR10	YB10	GN10	LE10	CI20	IS20	CT20	Gr20	Vi20	EP20	NC20	ST20
	3	LR10	YB10	GN10	LE10	CI20	IS20	CT20	Gr20	Vi20	EP20	NC20	ST20
	4	LR10	YB10	GN10	LE10	CI20	IS20	CT20	Gr20	Vi20	EP20	NC20	ST20
	5	LR10	YB10	GN10	LE10	CI20	IS20	CT20	Gr20	Vi20	EP20	NC20	ST20

(d) โครงสร้างตารางข้อมูลสำหรับส่งเข้าสู่ชั้น Input Layer ของตัวแบบระบบแนะนำรายวิชา

ภาพที่ 30 โครงสร้างตารางข้อมูล User Profile และ Course Learning บน HDFS สำหรับการประมวลผลข้อมูลเพื่อส่งต่อขั้นตอนฝึกสอนและทดสอบตัวแบบระบบแนะนำรายวิชา

โดยกำหนดค่า Hyper parameter ต่าง ๆ ดังนี้ Optimizer เป็น Nadam, learning rate เป็น 1e-6, epsilon เป็น 1e-17, lose function เป็น “SparseCategoricalCrossentropy” แสดงชุดคำสั่งโปรแกรมดังภาพที่ 31 - ภาพที่ 32

```python
from tensorflow import keras
from tensorflow.keras import Model, Input, layers
import tensorflow as tf

input0 = Input(shape=(5,12))
x0 = layers.Flatten()(input0)
x0 = layers.Embedding(60,200,trainable=True)(x0)
x0 = layers.TimeDistributed(layers.Dense(200,  activation = 'relu'))(x0)
x0 = layers.BatchNormalization()(x0)
x0 = layers.TimeDistributed(layers.Dense(400,  activation = 'relu'))(x0)
x0 = layers.BatchNormalization()(x0)
x0 = layers.MaxPooling1D(pool_size=(4))(x0)
x0 = layers.TimeDistributed(layers.Dense(800,  activation = 'relu'))(x0)
x0 = layers.BatchNormalization()(x0)
x0 = layers.MaxPooling1D(pool_size=(3))(x0)
x0 = layers.TimeDistributed(layers.Dense(1600,  activation = 'relu'))(x0)
x0 = layers.BatchNormalization()(x0)
x0 = layers.MaxPooling1D(pool_size=(3))(x0)
x0 = layers.TimeDistributed(layers.Dense(3200, activation = 'relu'))(x0)
x0 = layers.BatchNormalization()(x0)

x0 = layers.Dropout(.30,)(x0)
x0 = layers.Bidirectional(keras.layers.LSTM(500, return_sequences=True))(x0)
x0 = layers.Dropout(.20,)(x0)

x0 = layers.TimeDistributed(layers.Dense(1000 , activation = 'relu'))(x0)
x0 = layers.BatchNormalization()(x0)
x0 = layers.TimeDistributed(layers.Dense(800 , activation = 'relu'))(x0)
x0 = layers.BatchNormalization()(x0)
x0 = layers.TimeDistributed(layers.Dense(400 , activation = 'relu'))(x0)
x0 = layers.BatchNormalization()(x0)
x0 = layers.TimeDistributed(layers.Dense(200 , activation = 'relu'))(x0)
x0 = lay ers.TimeDistributed(layers.Dense(13 , activation = 'softmax'))(x0)
out = layers.Flatten()(x0)

model = Model(inputs=input0, outputs=out)
model.summary()
```

ภาพที่ 31 ชุดคำสั่งโปรแกรมภาษา Python ใช้ Keras framework สร้างตัวแบบระบบแนะนำรายวิชา

```python
opt = keras.optimizers.Nadam(learning_rate=1e-6, beta_1=0.9, beta_2=0.9999, epsilon=1e-17)
path_log="logs/test_lstm/"
model.compile(
    loss=keras.losses.CategoricalCrossentropy(from_logits=True),
    optimizer=opt,
    metrics=["accuracy"],
)

log_dir = path_log+ datetime.now().strftime("%Y%m%d-%H%M%S")
tensorboard_callback = keras.callbacks.TensorBoard(log_dir=log_dir, histogram_freq=1)

batch_size = 1000
epochs = 10
model.fit(
            x_train, y_train, validation_data=(x_test, y_test),
            batch_size=batch_size, epochs=epochs,
            callbacks=[tensorboard_callback]
)
```

ภาพที่ 32 ชุดคำสั่งโปรแกรมฝึกสอนและทดสอบตัวแบบระบบแนะนำรายวิชาด้วย TensorFlow

บทที่ 5
สรุปผลการดำเนินการวิจัย

ตามที่ผู้วิจัยได้นำเสนอแนวทางการพัฒนาระบบแนะนำรายวิชาสำหรับรายวิชาออนไลน์แบบเปิดสู่มวลชน (MOOCs) ด้วยเทคนิคข้อมูลขนาดใหญ่ โดยวัตถุประสงค์ ดังนี้

1) เพื่อศึกษาขั้นตอนวิธีที่เหมาะสมสำหรับนำไปสร้างตัวแบบระบบแนะนำรายวิชาในรายวิชาออนไลน์แบบเปิดสู่มวลชน (MOOCs) ด้วยเทคนิคข้อมูลขนาดใหญ่ (Big Data)
2) เพื่อปรับปรุงเทคนิคระบบแนะนำรายวิชาสำหรับข้อมูลขนาดใหญ่ (Big Data) ให้เหมาะสมและมีประสิทธิภาพ เช่น ลดเวลาในการทำงาน หรือมีความแม่นยำ
3) เพื่อพัฒนาตัวแบบระบบแนะนำรายวิชาในรายวิชาออนไลน์แบบเปิดสู่มวลชน (MOOCs) ให้กับผู้เรียน

5.1 สรุปผลการวิจัย

รายวิชาออนไลน์แบบเปิดสู่มวลชน (Massive Open Online Courses, MOOCs) คือ นวัตกรรมการศึกษาแบบเสรี ผู้คนทั่วโลกให้ความสนใจและเข้าเรียนใน MOOCs เป็นจำนวนมาก โดยนับตั้งแต่ปี 2012 ที่ MOOCs เปิดตัวขึ้นเป็นครั้งแรกจนถึงปัจจุบันมีจำนวนผู้เรียนหลายล้านคน มีรายวิชามากกว่าหนึ่งหมื่นรายวิชา แต่เมื่อรายวิชามีจำนวนเพิ่มมากขึ้นก่อให้เกิดปัญหาการเผชิญกับข้อมูลมหาศาล ซึ่งไม่สามารถจัดการกับข้อมูลและใช้ข้อมูลเหล่านั้นให้เกิดประโยชน์ได้ ผู้ใช้งานบางส่วนไม่เข้าใจข้อมูลรายวิชา เนื่องจากประสบการณ์ในการเรียนหรือระดับความรู้ความเข้าใจที่ไม่มากพอ และปัญหาการเรียนการสอนในรายวิชาออนไลน์แบบเปิดสู่มวลชน (MOOCs) คือมีอัตราการหยุดเรียนสูงมากส่วนหนึ่งอาจเนื่องจากเนื้อหาที่นำเสนอไม่ตรงกับที่ผู้เรียนต้องการหรือปัญหาในการกำกับตนเองของผู้เรียน หรือแม้กระทั่งการเรียนการสอนในระบบชั้นเรียนปกติผู้เรียนต้องเผชิญกับข้อมูลรายวิชาที่มากเกินไปทำให้ยากในการตัดสินใจเลือกเรียน

MOOCs เป็นหนึ่งในความท้าทายบนข้อมูลขนาดใหญ่ (Big Data) เช่นเดียวกัน เนื่องจากคุณลักษณะของข้อมูลขนาดใหญ่ที่ข้อมูลมีปริมาณมาก (Volume) ข้อมูลมีรูปแบบที่หลากหลาย (Variety) และข้อมูลมีการเปลี่ยนแปลงตลอดเวลาและรวดเร็ว (Velocity) เช่น ข้อมูลการคลิกเล่น ยกเลิกหรือหยุดวิดีโอของผู้เรียน การจัดเก็บข้อมูล การประมวลผลข้อมูลแบบเดิมไม่สามารถที่จะจัดการหรือวิเคราะห์ได้อย่างมีประสิทธิภาพ จากอัตราการเพิ่มขึ้นของข้อมูลเป็นไปอย่างรวดเร็วเป็นสาเหตุทำให้การประมวลผลเป็นไปได้อย่างยากลำบากจึงต้องหาวิธีประมวลผลข้อมูลให้มีผลดีและถูกต้องเพื่อหาผลลัพธ์ที่ต้องการ โดยเฉพาะการตัดสินใจเลือกสิ่งของต่าง ๆ เป็นสิ่งที่ต้องอาศัย

ประสบการณ์ ความรู้ เวลา หรือปัจจัยอื่น ๆ เข้ามาช่วยพิจารณา ซึ่งจะเห็นได้ว่าระบบแนะนำ (RS) เข้ามามีบทบาทในชีวิตของมนุษย์มากยิ่งขึ้นช่วยอำนวยความสะดวกให้การค้นหาแคบลงและสะดวกรวดเร็ว

จากการศึกษาขั้นตอนวิธีที่เหมาะสมสำหรับนำไปสร้างตัวแบบระบบแนะนำรายวิชาในรายวิชาออนไลน์แบบเปิดสู่มวลชน (MOOCs) ด้วยเทคนิคข้อมูลขนาดใหญ่ (Big Data) นั้น ผู้วิจัยศึกษางานวิจัยที่เกี่ยวข้องพบว่ามีขั้นตอนวิธีมากมายหลากหลาย ซึ่งแต่ละวิธีมีข้อดีและข้อบกพร่องแตกต่างกันไป ดังนั้นจึงเลือกศึกษาขั้นตอนวิธีจำนวน 3 วิธี ได้แก่ 1) กฎความสัมพันธ์ (Association rule) 2) โครงข่ายประสาทเทียมเชิงลึก (Deep neural network, DNN) และ 3) โครงข่ายประสาทเทียมเชิงลึก (Deep neural network, DNN) ร่วมกับหน่วยความจำสั้นระยะยาว Long Short-Term Memory (LSTM) และที่สำคัญการออกแบบสถาปัตยกรรมที่สามารถรองรับการจัดเก็บและการประมวลผลข้อมูลขนาดใหญ่ได้นั้นต้องอาศัยเทคโนโลยีสำหรับข้อมูลขนาดใหญ่

สถาปัตยกรรมที่สามารถรองรับการจัดเก็บและการประมวลผลข้อมูลขนาดใหญ่ได้นั้นต้องอาศัยเทคโนโลยีสำหรับข้อมูลขนาดใหญ่ ดังนั้นผู้วิจัยเลือกใช้ Apache Platform ซึ่งเป็น software framework สามารถจัดการกับชุดข้อมูลขนาดใหญ่ (large dataset) และมีการประมวลผลแบบกระจาย (distributed processing) บนเครื่องคอมพิวเตอร์ที่เชื่อมต่อกันเป็นกลุ่ม (cluster) ข้อมูลที่มาจากแต่ละ MOOCs จะถูกจัดเก็บไว้ในรูปแบบบล็อกโดยมี Hadoop Distributed File System (HDFS) เป็นส่วนของระบบจัดการไฟล์ (File System) เพื่อให้ข้อมูลกระจายไปอยู่ตามเครื่องคอมพิวเตอร์หรือโหนดในระบบได้อย่างถูกต้อง และเพื่อป้องกันการเกิดปัญหาข้อผิดพลาดที่ทำให้ข้อมูลเสียหาย ตามสถาปัตยกรรมที่ผู้วิจัยได้ออกแบบไว้หากมีข้อมูลเพิ่มขึ้น ก็สามารถเพิ่มโหนดเก็บข้อมูลได้อย่างต่อเนื่องโดยไม่กระทบต่อระบบซึ่งเป็นลักษณะของการ Scale out มีการเลือกใช้ Apache Kafka ในการรับส่งข้อมูลแบบเรียลไทม์ Apache Kafka เป็นแพลตฟอร์มสตรีมมิ่งแบบกระจาย (distributed streaming platform) หรือ Message Broker ที่สามารถจัดการกับข้อมูลที่ถูกสร้างขึ้นอย่างต่อเนื่องจากแหล่งข้อมูลจำนวนมาก (Streaming Data)

ผลการพัฒนาตัวแบบด้วยโครงข่ายประสาทเทียมเชิงลึก (Deep neural network, DNN) ร่วมกับหน่วยความจำสั้นระยะยาว Long Short-Term Memory (LSTM) กับการทำงานแบบคลัสเตอริ่ง (Clustering) ที่ฝึกสอนด้วยวิธี fit-generator ผลการวิจัยแสดงให้เห็นว่าตัวแบบ MCR-CG มีประสิทธิภาพเชิงเวลาและค่าความถูกต้องที่ดีที่สุดเมื่อเทียบกับตัวแบบ MCR-NF และ MCR-NG โดยการทดสอบกับข้อมูลจริงของ Harvard and MIT published the edX ประสิทธิภาพเชิงด้านความถูกต้องพบว่าในขั้นตอนการฝึกสอนตัวแบบ (Model training) ให้ค่าความถูกต้องสูงถึง 75.9% และในขั้นตอนการทดสอบตัวแบบ (Model testing) ให้ค่าความถูกต้องสูงถึง 75.7% สำหรับประสิทธิภาพเชิงเวลาพบว่าในขั้นตอนการฝึกสอนตัวแบบ (Model training) MCR-CG ใช้เวลาน้อย

กว่าประมาณ 444 นาทีเมื่อเปรียบเทียบกับตัวแบบ MCR-NF และตัวแบบ MCR-CG ใช้เวลาน้อยกว่าประมาณ 376 นาทีเมื่อเทียบกับตัวแบบ MCR-NG ด้วยความสามารถของ Long Short-Term Memory (LSTM) ในการคำนวณผลลัพธ์ที่ได้โดยคำนวณจากโหนดก่อนหน้าแล้วส่งต่อเป็นข้อมูลเข้าของโหนดถัดไป ทำให้ Long Short-Term Memory (LSTM) สามารถนำข้อมูลก่อนหน้า (ข้อมูลในอดีต) มาใช้ในการทำนายสิ่งที่อาจจะเกิดขึ้นในอนาคตได้ และการใช้การฝึกสอนตัวแบบด้วยวิธี “fit-generator” ที่ทำงานร่วมกับ Hadoop cluster ทำให้ข้อมูลเข้ามีการสับเปลี่ยน (Shuffle) ของชุดข้อมูลทุกครั้งในทุกรอบการฝึกสอนตัวแบบ จึงส่งผลทำให้ประสิทธิภาพของตัวแบบ MCR-CG ดีกว่าแบบอื่น ๆ และเป็นแนวทางที่ดีสำหรับการพัฒนาตัวแบบระบบคำแนะนำรายวิชา และที่สำคัญ MCR-CG ยังสามารถรองรับการปรับเพิ่มขนาดของพื้นที่การจัดเก็บข้อมูลในรูปแบบแนวนอน (Scale-out) ได้ และรองรับกับข้อมูลที่มีขนาดใหญ่ได้

5.1.1 สรุปสถาปัตยกรรมระบบแนะนำรายวิชา

การใช้ Apache Platform เป็น software framework สำหรับจัดการกับชุดข้อมูลขนาดใหญ่ (large dataset) สามารถดำเนินการได้เป็นอย่างดีสำหรับข้อมูลขนาดใหญ่อย่างข้อมูล MOOCs โดยข้อมูลถูกจัดเก็บไว้ในรูปแบบบล็อก ซึ่งมี Hadoop Distributed File System (HDFS) เป็นส่วนของระบบจัดการไฟล์ (File System) มีการใช้ Apache Kafka ในการรับส่งข้อมูลแบบเรียลไทม์ Apache Kafka ทำให้สามารถจัดการกับข้อมูลที่ถูกสร้างขึ้นอย่างต่อเนื่องจากแหล่งข้อมูลจำนวนมาก (Streaming Data) ที่มาจากหลากหลาย MOOCs

5.1.2 สรุป Hyperparameter ที่ส่งผลต่อตัวแบบระบบแนะนำรายวิชา

1) Learning rate เป็น hyperparameter ที่ใช้ในการกำหนดขนาดของการไต่เขา (ใช้ gradient เพื่อกำหนดทิศทาง) จากการวิจัยพบว่า Learning rate เป็นส่วนสำคัญอันดับต้น ๆ ในการเทรนตัวแบบ โดยยิ่ง learning rate มาก ๆ ยิ่งช่วยทำให้โมเดลไต่ไปยังเป้าหมายได้เร็วขึ้น แต่ถ้า learning rate มากเกินไปทำให้การไต่เขาของตัวแบบกระโดดไปกระโดดมาในบางครั้งอาจจะทำให้หลุดไปจากเป้าหมายที่ต้องการ ถ้า learning rate น้อยเกินไปทำให้การไต่เขาของตัวแบบเป็นไปช้ามากจนอาจจะไม่ถึงเป้าหมายที่ต้องการในเวลาที่กำหนด ดังนั้นการปรับ learning rate เป็นขั้นตอนสำคัญในการเทรนตัวแบบ ทั้งนี้ learning rate ที่เหมาะสมก็ขึ้นอยู่กับทั้ง loss function , ตัวแบบและข้อมูลที่ใช้ฝึกสอน ฉะนั้นงานวิจัยนี้ใช้กำหนดค่าเริ่มต้นให้กับ learning rate เป็น 1e-6 ซึ่งเป็นค่าที่เหมาะสมสำหรับตัวแบบระบบแนะนำรายวิชา

2) Optimizer เป็นฟังก์ชันทางคณิตศาสตร์ (หรือขั้นตอนวิธี) เป็นตัวช่วยลดค่า Loss หรือ Error ทำให้ตัวแบบมีประสิทธิภาพที่ดีขึ้น ฉะนั้นงานวิจัยนี้ใช้ Optimizer แบบ Nadam (Nesterov Accelerated Adaptive Moment Estimator)

3) epochs คือ จำนวนรอบการทำการเรียนรู้ จากผลการวิจัยพบว่าช่วงของค่าที่เหมาะสมคือ 301 – 350

4) epsilon คือ ค่าคงที่ขนาดเล็กสำหรับความเสถียรเชิงตัวเลข จากผลการวิจัยพบว่ากำหนดค่าเริ่มต้นเป็น 1e-17 เป็นค่าที่เหมาะสมสำหรับตัวแบบ

5) ผู้วิจัยเลือกใช้ Sparse Categorical Cross Entropy เป็น lose function เพราะเหมาะใช้ในการแก้ปัญหาที่มีผลเฉลยหรือ Label จำนวนมาก และเหมาะกับการนำไปใช้แก้ปัญหาประเภท Classification ซึ่งโดยปกติในการสร้างตัวแบบแบบ Multi-Class Classification จะต้องคอนฟิก Loss Function เป็น Categorical Crossentropy Loss แต่สำหรับปัญหาบางปัญหาที่มีผลเฉลย (Label) จำนวนมาก ๆ การเข้ารหัสผลเฉลยแบบ One-hot Encoding เพื่อสร้างการแจกแจงความน่าจะเป็นจะทำให้สิ้นเปลืองหน่วยความจำ (Memory) Sparse Categorical Crossentropy สามารถแก้ปัญหานี้ได้โดยไม่ต้องสร้างผลเฉลยแบบ One-hot Encoding แต่ยังคงมีการคำนวนค่า Cross-entropy ได้เหมือนเดิม ซึ่งในการคอนฟิกตัวแบบให้ใช้งาน Sparse Categorical Crossentropy Loss ยังคงใช้ Activate Function เป็น Softmax ที่ Output Layer เช่นเดิม

5.2 การเตรียมความพร้อมสำหรับนำระบบไปใช้งาน

5.2.1 ความพร้อมด้านทรัพยากร

1) ระบบเครือข่าย

ระบบเครือข่ายที่สามารถแบ่งโซนการสื่อสารได้เป็นการสื่อสารผ่านอินเทอร์เน็ตและเครือข่ายท้องถิ่น (LAN) ได้ สำหรับ

- การสื่อสารผ่านอินเทอร์เน็ตที่ความเร็วรับและส่งไม่น้อยกว่า 500 Mbps เพื่อใช้สำหรับถ่ายโอนข้อมูลจากแหล่งข้อมูล MOOCs ต่าง ๆ มายังขั้นตอน Data Ingestion ดังรูปที่...

- เครือข่ายท้องถิ่นที่ความเร็วรับและส่งไม่น้อยกว่า 1 Gbps เพื่อใช้สำหรับการสื่อสารและถ่ายโอนข้อมูลของระบบบนเครื่องคอมพิวเตอร์แม่ข่ายที่ทำหน้าที่เก็บข้อมูลและประมวลผลข้อมูลผล

2) เครื่องคอมพิวเตอร์แม่ข่าย

- เครื่องคอมพิวเตอร์แม่ข่ายสำหรับเก็บข้อมูล จำนวนไม่น้อยกว่า 3 เครื่อง มีหน่วยประมวลผล Intel(R) Xeon(R) CPU E5-1650 0 @ 3.20GHz, 6 core และหน่วยความจำไม่น้อยกว่า 32 GB พร้อมติดตั้ง Apache Hadoop

- เครื่องคอมพิวเตอร์แม่ข่ายสำหรับประมวลผลข้อมูล จำนวน 1 เครื่อง มีหน่วยประมวลผล Intel(R) Xeon(R) CPU E5-1650 0 @ 3.20GHz, 6 core และหน่วยความจำไม่น้อยกว่า 64 GB มีการ์ดหน่วยประมวลผลด้านกราฟฟิก (GPU) GEFORCE RTX 2080 Ti เป็นอย่างน้อย พร้อมติดตั้ง Docker, Python, TensorFlow และ Keras

5.2.2 ความรู้ทางด้านเครือข่ายคอมพิวเตอร์และเฟรมเวิร์คสำหรับตัวแบบ

1) ความรู้ทางด้านเครือข่ายคอมพิวเตอร์

- การติดตั้งระบบปฏิบัติ Linux และ Docker container

- การตั้งค่า IP address และการแบ่งเครือข่ายย่อยบนระบบปฏิบัติการ Linux สำหรับเครื่องคอมพิวเตอร์แม่ข่ายสำหรับเก็บข้อมูล และเครื่องคอมพิวเตอร์แม่ข่ายสำหรับประมวลผลข้อมูล

2) ความรู้ทางด้านเฟรมเวิร์คสำหรับตัวแบบ

- การติดตั้งโปรแกรม Docker, Python, TensorFlow และ Keras

5.3 การนำสถาปัตยกรรมระบบและตัวแบบระบบแนะนำรายวิชาไปใช้

1) สถาปัตยกรรมระบบแนะนำรายวิชาและตัวแบบระบบแนะนำรายวิชาที่ผู้วิจัยได้นำเสนอนั้น สามารถนำไปใช้กับ Learning platform เช่น Open EdX platform, Moodle หรือ Learning platform ที่มีการจัดเก็บข้อมูลแอตทริบิวต์ตามตัวแบบที่ผู้วิจัยนำเสนอ ได้แก่ โดยมีผลลัพธ์ของระบบแนะนำรายวิชาไม่เกิน 13 รายวิชา ผู้วิจัยแนะนำสำหรับการนำตัวแบบไปใช้โดยตรงไม่มีการปรับหรือแก้ไขในส่วนของ input layer และ output layer ของตัวแบบ

2) สำหรับระบบแนะนำแบบอื่น ๆ เช่น ระบบแนะนำโรงแรม ระบบแนะนำภาพยนตร์ ระบบแนะนำสถานที่ท่องเที่ยว มีแนวโน้มที่สามารถนำสถาปัตยกรรมระบบแนะนำรายวิชาและตัวแบบระบบแนะนำรายวิชาไปประยุกต์หรือพัฒนาต่อได้ เนื่องจากตัวแบบนี้มีการแบ่งชุดข้อมูลเข้าออกเป็นสองกลุ่ม คือ User Profile และ Course Learning โดยที่ User Profile เป็นข้อมูลแบบทั่วไปที่มีในทุกระบบ แต่สำหรับข้อมูล Course Learning จะมีการปรับเปลี่ยนประเภทของแอตทริบิวต์ข้อมูลไป

ตามระบบแนะนำแบบต่าง ๆ ซึ่งสามารถที่จะปรับลดหรือเพิ่มเติมแอตทริบิวต์ได้ โดยที่ส่วนสำคัญของตัวแบบไม่มีการเปลี่ยนแปลง เช่นเดียวกับส่วนของผลลัพธ์ก็สามารถที่จะปรับลดหรือเพิ่มเติมได้ โดยการเทรนตัวแบบใหม่

5.4 ข้อเสนอแนะ

จากการวิจัยเพื่อพัฒนาระบบแนะนำรายวิชาสำหรับรายวิชาออนไลน์แบบเปิดสู่มวลชนด้วยเทคนิคข้อมูลขนาดใหญ่นั้น ผู้วิจัยใช้ข้อมูล MITx and HarvardX ซึ่งถูกเผยแพร่แบบสาธารณะและสนับสนุนให้นักวิจัยสามารถนำไปใช้เพื่อการวิจัยได้ ชุดข้อมูลชื่อ “The HarvardX-MITx Person-Course Academic Year 2013 De-Identified dataset รุ่น 2.0” [21] แต่ด้วยข้อมูลดังกล่าวเป็นข้อมูลบนบริบทของชาวต่างชาติ ดังนั้นหากนำตัวแบบระบบแนะนำรายวิชาฯ ที่ผู้วิจัยได้เสนอนั้น หากนำมาใช้จริงบนบริบทของประชาชนชาวไทย อาจต้องมีการปรับค่าพารามิเตอร์ (Parameter) ต่าง ๆ ของตัวแบบ เพื่อให้ตัวแบบมีประสิทธิภาพถูกต้องแม่นยำและเหมาะสมกับบริบทของคนไทย

5.5 ข้อจำกัดของตัวแบบระบบแนะนำรายวิชา

ตัวแบบระบบแนะนำรายวิชาที่ผู้วิจัยได้พัฒนาขึ้นนั้น ไม่รองรับเงื่อนไขต่าง ๆ ดังนี้

1) กรณีที่มีข้อกำหนดหรือเงื่อนไขของการเรียนรายวิชาต่าง ๆ เช่น ผู้เรียนต้องลงทะเบียนรายวิชาการเขียนโปรแกรมเบื้องต้น (Fundamental Programming) ก่อนจึงจะสามารถลงทะเบียนรายวิชาโครงสร้างข้อมูล (Data Structures) ได้ หรือผู้เรียนต้องลงทะเบียนรายวิชาการเขียนโปรแกรมเบื้องต้น (Fundamental Programming) และการเขียนโปรแกรมขั้นสูง (Advance Programming) ก่อนจึงจะสามารถลงทะเบียนรายวิชาขั้นตอนวิธี (Algorithms) ได้

รายวิชาที่เขียนเป็นตัวอย่างข้างต้นนั้น ผู้วิจัยเลือกเป็นวิชาที่เกี่ยวกับคอมพิวเตอร์ เนื่องจากผู้วิจัยศึกษาอยู่ในภาควิชาคอมพิวเตอร์ และน่าจะเป็นประโยชน์สำหรับนักศึกษาภาคคอมพิวเตอร์ที่สนใจงานวิจัยเรื่องนี้ และต้องการต่อยอดงานวิจัยเพิ่มเติม จะทำให้เกิดความเข้าใจได้อย่างรวดเร็ว

2) กรณีที่มีข้อกำหนดหรือเงื่อนไขของการเรียนรายวิชาต่าง ๆ เช่น ผู้เรียนต้องผ่านรายวิชาการเขียนโปรแกรมเบื้องต้น (Fundamental Programming) อย่างน้อย 80% จึงจะสามารถลงทะเบียนรายวิชาโครงสร้างข้อมูล (Data Structures) ได้

3) กรณีที่ผู้เรียนต้องการเรียนรายวิชาเดิมซ้ำ ระบบจะไม่สามารถสร้างรายการแนะนำรายวิชาให้ได้

5.6 ปัญหา และอุปสรรคที่เกิดขึ้นกับจากการวิจัย

1) ข้อมูลผู้เรียนใน MOOCs ถือเป็นข้อมูลส่วนบุคคลและเป็นข้อมูลสำคัญของประเทศนั้น ๆ จึงทำให้การเปิดเผยข้อมูลจำกัดอยู่ในเฉพาะในประเทศของตนเองเท่านั้น สำหรับประเทศไทยก็มีแพลตฟอร์ม (Platform) ชื่อ Thai MOOC[23] เป็นของตนเองเช่นกัน แต่เนื่องด้วยข้อมูลผู้เรียนและข้อมูลพฤติกรรมของผู้เรียนถือเป็นสิทธิ์เฉพาะของหน่วยงานต้นสังกัด Thai MOOC ไม่ได้มีสิทธิในข้อมูลนั้น จึงทำให้ผู้วิจัยไม่สามารถนำข้อมูลดังกล่าวมาวิจัยได้

2) ข้อมูลที่ผู้วิจัยนำมาใช้ในงานวิจัยเป็นข้อมูลของ MITx and HarvardX ซึ่งถูกเผยแพร่แบบสาธารณะและสนับสนุนให้นักวิจัยสามารถนำไปใช้เพื่อการวิจัยได้ ชุดข้อมูลชื่อ “The HarvardX-MITx Person-Course Academic Year 2013 De-Identified dataset รุ่น 2.0” [21] สร้างขึ้นเมื่อวันที่ 14 พฤษภาคม พ.ศ.2557 สังเกตได้ว่าข้อมูลดังกล่าวเป็นข้อมูลในช่วงปี ค.ศ. 2013 และนับจากนั้นเป็นต้นไป MOOC แต่ละทีไม่มีการเปิดเผยข้อมูลอีก เนื่องจากข้อมูลดังกล่าวเป็นข้อมูลส่วนบุคคลและส่วนใหญ่จะอนุญาตให้ใช้วิจัยเฉพาะภายในประเทศของตนเองเท่านั้น

3) จากปัญหาและอุปสรรคข้อที่ 2 และเพื่อให้มีข้อมูลที่เป็นปัจจุบันสำหรับงานวิจัย ผู้วิจัยได้ติดต่อผู้วิจัยที่วิจัยเกี่ยวกับ MOOCs เช่น ทีมวิจัยของ MIT เพื่อขอข้อมูลผู้เรียนและข้อมูลพฤติกรรมของผู้เรียน ผ่านช่องทางไปรษณีย์อิเล็กทรอนิกส์ (e-mail) แต่ไม่ได้รับการติดต่อกลับ อาจเนื่องมาจากสาเหตุของ “ความเป็นข้อมูลส่วนบุคคล” และความเป็นข้อมูลเฉพาะของประเทศนั้น ๆ

5.7 แนวทางการวิจัยในอนาคต

แนวทางการวิจัยในอนาคต ผู้วิจัยมุ่งมั่นที่จะปรับปรุงแนวทางการวิจัยโดยใช้เทคนิค Long Short-Term Memory ทำงานร่วมกับคุณลักษณะ (Feature) อื่น ๆ เช่น การใช้คุณลักษณะด้านอารมณ์ความรู้สึก (Sentiment) รวมทั้งการหาจุดสมดุล (Optimization) ของตัวแบบ เพื่อให้ทำงานตัวแบบระบบแนะนำรายวิชาสำหรับรายวิชาในรายวิชาออนไลน์แบบเปิดสู่มวลชนภายใต้สภาพแวดล้อมข้อมูลขนาดใหญ่ทำงานได้อย่างถูกต้องแม่นยำยิ่งขึ้น นอกจากนี้ผู้วิจัยยังตั้งเป้าหมายที่

---

[23] เป็นแพลตฟอร์ม (Platform) ที่สนับสนุนในด้านการจัดการเรียนการสอนออนไลน์แบบเปิด (Thailand Massive Open Online Course) อยู่ภายใต้การกำกับของโครงการพัฒนามหาวิทยาลัยไซเบอร์ไทย โดยมีวัตถุประสงค์เพื่อเพิ่มช่องทางการเรียนรู้ที่มีมาตรฐานให้กับประชาชนไทย โดยสามารถเรียนได้ฟรีและเข้าถึงแหล่งเรียนรู้ได้ทุกที่ทุกเวลาผ่านการจัดการเรียนการสอนออนไลน์ เพื่อส่งเสริมให้ประชาชนไทยเกิด “การเรียนรู้ตลอดชีวิต” (Lifelong Learning)

จะนำแนวทางการวิจัยนี้ไปใช้ในด้านศึกษาอย่างจริงจัง เช่น การนำงานวิจัยไปใช้กับหน่วยงานต้นสังกัดที่ได้ให้ทุนสนับสนุนการศึกษา

รายการอ้างอิง

[1] M. Binkley *et al.*, "Defining Twenty-First Century Skills," in *Assessment and Teaching of 21st Century Skills*, P. Griffin, B. McGaw, and E. Care Eds. Dordrecht: Springer Netherlands, 2012, pp. 17-66.

[2] D. Shah. "By The Numbers: MOOCs in 2021." https://www.classcentral.com/report/mooc-stats-2021/ (accessed August 6, 2021).

[3] J. D. Kelleher and B. Tierney, *Data science*. (in English), 2018.

[4] C. C. Aggarwal, "An Introduction to Recommender Systems," in *Recommender Systems: The Textbook*. Cham: Springer International Publishing, 2016, pp. 1-28.

[5] Y. Wang, B. Liang, W. Ji, S. Wang, and Y. Chen, "An improved algorithm for personalized recommendation on MOOCs," *International Journal of Crowd Science,* vol. 1, no. 3, pp. 186-196, 2017.

[6] X. Wang and Y. Wang, "Improving Content-based and Hybrid Music Recommendation using Deep Learning," in *Proceedings of the 22nd ACM international conference on Multimedia*, 2014: ACM, pp. 627-636.

[7] J. Chen, H. Zhang, X. He, L. Nie, W. Liu, and T.-S. Chua, "Attentive Collaborative Filtering: Multimedia Recommendation with Item- and Component-Level Attention," in *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2017, pp. 335-344.

[8] P. Covington, J. Adams, and E. Sargin, "Deep Neural Networks for YouTube Recommendations," in *Proceedings of the 10th ACM Conference on Recommender Systems*, 2016, pp. 191-198.

[9] S. Okura, Y. Tagami, S. Ono, and A. Tajima, "Embedding-based News Recommendation for Millions of Users," in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2017, pp. 1933-1942.

[10] C. Olah. "Understanding LSTM Networks." http://colah.github.io/posts/2015-08-Understanding-LSTMs/ (accessed April 22, 2021).

[11] J. Zhao, F. Deng, Y. Cai, and J. Chen, "Long short-term memory-Fully connected (LSTM-FC) neural network for PM2. 5 concentration prediction," *Chemosphere,* vol. 220, pp. 486-492, 2019.

[12] R. Jozefowicz, W. Zaremba, and I. Sutskever, "An Empirical Exploration of Recurrent Network Architectures," presented at the Proceedings of the 32nd International Conference on Machine Learning, 2015.

[13] S. Dhawal. "Year of MOOC-based Degrees: A Review of MOOC Stats and Trends in 2018." (accessed May, 2019).

[14] Z. Du, H. Chen, and J. Jiang, "Research on the big data system of massive open online course," in 2016 *IEEE International Conference on Big Data (Big Data)*, Dec. 2016, pp. 1931-1936.

[15] T. Yang, C. G. Brinton, C. Joe-Wong, and M. Chiang, "Behavior-Based Grade Prediction for MOOCs Via Time Series Neural Networks," *IEEE Journal of Selected Topics in Signal Processing,* vol. 11, no. 5, pp. 716-728, 2017.

[16] Q. Chen, Y. Chen, D. Liu, C. Shi, Y. Wu, and H. Qu, "PeakVizor: Visual Analytics of Peaks in Video Clickstreams from Massive Open Online Courses," *IEEE Transactions on Visualization and Computer Graphics,* vol. 22, no. 10, pp. 2315-2330, 2016.

[17] S. Fu, J. Zhao, W. Cui, and H. Qu, "Visual Analysis of MOOC Forums with iForum," *IEEE Transactions on Visualization and Computer Graphics,* vol. 23, no. 1, pp. 201-210, 2017.

[18] N. Capuano, V. Loia, and F. Orciuoli, "A Fuzzy Group Decision Making Model for Ordinal Peer Assessment," *IEEE Transactions on Learning Technologies,* vol. 10, no. 2, pp. 247-259, 2017.

[19] H. Zhang, T. Huang, Z. Lv, S. Liu, and Z. Zhou, "MCRS: A course recommendation system for MOOCs," *Multimedia Tools and Applications,* vol. 77, no. 6, pp. 7051-7069, 2018.

[20] H. Zhang, T. Huang, Z. Lv, S. Liu, and H. Yang, "MOOCRC: A Highly Accurate Resource Recommendation Model for Use in MOOC Environments," *Mobile Networks and Applications,* journal article vol. 24, no. 1, pp. 34-46, 2019.

[21] HarvardX. *HarvardX Person-Course Academic Year* 2013 *De-Identified dataset, version* 3.0, Harvard Dataverse. [Online]. Available: https://doi.org/10.7910/DVN/26147

[22] L. Duan and Y. Xiong, "Big data analytics and business analytics," *Journal of Management Analytics,* vol. 2, no. 1, pp. 1-21, 2015.

[23] K. Chen, J. Powers, S. Guo, and F. Tian, "CRESP: Towards Optimal Resource Provisioning for MapReduce Computing in Public Clouds," *IEEE Transactions on Parallel and Distributed Systems,* vol. 25, no. 6, pp. 1403-1412, 2014.

[24] F. Shabestari, A. M. Rahmani, N. J. Navimipour, and S. Jabbehdari, "A taxonomy of software-based and hardware-based approaches for energy efficiency management in the Hadoop," *Journal of Network and Computer Applications,* vol. 126, pp. 162-177, 2019.

[25] A. Sherstinsky, "Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) network," *Physica D: Nonlinear Phenomena,* vol. 404, p. 132306, 2020 2020.

[26] J. Zhao, X. Mao, and L. Chen, "Speech emotion recognition using deep 1D & 2D CNN LSTM networks," *Biomedical Signal Processing and Control,* vol. 47, pp. 312-323, 2019.

มหาวิทยาลัยศิลปากร

ภาคผนวก

ประวัติผู้เขียน

ชื่อ-สกุล	นางศิริพร ศักดิ์บุญญารัตน์
วัน เดือน ปี เกิด	22 พฤศจิกายน 2521
สถานที่เกิด	จังหวัดนครปฐม
วุฒิการศึกษา	ปริญญาโท สาขาวิชาวิทยาการคอมพิวเตอร์ ภาควิชาคอมพิวเตอร์ มหาวิทยาลัยศิลปากร วิทยาเขตพระราชวังสนามจันทร์ ปริญญาตรี สาขาวิชาวิทยาการคอมพิวเตอร์ ภาควิชาวิทยาการคอมพิวเตอร์ มหาวิทยาลัยราชภัฏนครปฐม
ที่อยู่ปัจจุบัน	บ้านเลขที่ 129/5 หมู่ 1 ตำบล นครปฐม อำเภอ เมืองนครปฐม จังหวัด นครปฐม
ผลงานตีพิมพ์	ปานใจ ธารทัศนวงศ์ และ ศิริพร ศักดิ์บุญญารัตน์, "บทเรียนออนไลน์แบบเปิดเพื่อมหาชน (MOOCs) เพื่อการศึกษาไทยหลังการแพร่ระบาดของ โรคติดเชื้อไวรัสโคโรนา 2019", คุรุสภาวิทยาจารย์, 2021, 2: 17-26. S. Sakboonyarat and P. Tantatsanawong, "Massive Open Online Courses (MOOCs) Recommendation Modeling using Deep Learning," 2019 23rd International Computer Science and Engineering Conference (ICSEC), Phuket, Thailand, 2019, pp. 275-280.