

การพยากรณ์เวลาการเดินทางจากข้อมูล GPS



นายชาญวิทย์ จักรธีรังกูร

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต

สาขาวิชาวิทยาศาสตร์คอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

ปีการศึกษา 2562

ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

Travel Time Prediction From GPS Trace Data



A Thesis Submitted in Partial Fulfillment of the Requirements  
for the Degree of Master of Science in Computer Science

Department of Computer Engineering

FACULTY OF ENGINEERING

Chulalongkorn University

Academic Year 2019

Copyright of Chulalongkorn University

หัวข้อวิทยานิพนธ์	การพยากรณ์เวลาการเดินทางจากข้อมูล GPS
โดย	นายชาญวิทย์ จักรธีรังกูร
สาขาวิชา	วิทยาศาสตร์คอมพิวเตอร์
อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก	ผู้ช่วยศาสตราจารย์ ดร.วีระ เหมเมืองสิน

---

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้บัณฑิตวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต

.....	คณบดีคณะวิศวกรรมศาสตร์
(ศาสตราจารย์ ดร.สุพจน์ เตชวรสินสกุล)	
คณะกรรมการสอบวิทยานิพนธ์	
.....	ประธานกรรมการ
(รองศาสตราจารย์ ดร.เกริก ภิรมย์โสภา)	
.....	อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก
(ผู้ช่วยศาสตราจารย์ ดร.วีระ เหมเมืองสิน)	
.....	กรรมการ
(รองศาสตราจารย์ ดร.สรวิศ นฤปิติ)	
.....	กรรมการภายนอกมหาวิทยาลัย
(ผู้ช่วยศาสตราจารย์ ดร.ธีรยุทธ โทหรานนท์)	

ชาญวิทย์ จักรธีรังกูร : การพยากรณ์เวลาการเดินทางจากข้อมูล GPS. ( Travel Time Prediction From GPS Trace Data) อ.ที่ปรึกษาหลัก : ผศ. ดร.วีระ เหมืองสิน

การพยากรณ์ระยะเวลาเดินทางเป็นส่วนสำคัญของระบบจราจรอัจฉริยะ วิธีการที่เป็นที่นิยมในการพยากรณ์ระยะเวลาเดินทางคือการนำระยะเวลาเดินทางในอดีตมาใช้เพื่อพยากรณ์ระยะเวลาเดินทางในอนาคตอันใกล้ งานวิจัยนี้เป็นการเปรียบเทียบโมเดล LSTM ที่ได้จากข้อมูลประเภท GPS trace กับโมเดลที่สร้างด้วย Neural Network แบบวนซ้ำ ประกอบด้วยโมเดล RNN, LSTM, GRU และการนำโมเดลมาประกอบกันคือ LSTM ร่วมกับ RNN, LSTM ร่วมกับ GRU และโมเดล LSTM ร่วมกับ DNN. โดยทำการศึกษาผลกระทบจากขนาดของ Sliding Window ของข้อมูลที่แตกต่างกัน การนำโมเดลที่สร้างด้วยช่วงถนนช่วงหนึ่งไปใช้พยากรณ์กับถนนช่วงอื่น การนำทิศทางการเดินทางมาเป็นส่วนในการพิจารณา การเปรียบเทียบผลลัพธ์ที่ได้จากวิธีการทำ Link level และ Path level และการพิจารณาในแต่ละช่วงเวลาชั่วโมงของวันโดยผลลัพธ์ที่ได้จากการทดลอง พบว่าโดยรวมโมเดล LSTM-DNN ให้ผลลัพธ์ที่ดีที่สุด ขณะเดียวกันระยะเวลาที่ใช้สอนและทดสอบโมเดล RNN สามารถทำได้เร็วที่สุด นอกจากนี้ยังคงพบว่าหากเป็นช่วงเวลาเย็นของวัน ผลการพยากรณ์ที่ได้มีค่าความผิดพลาดสูงกว่าช่วงเวลาอื่นของวันอีกด้วย

จุฬาลงกรณ์มหาวิทยาลัย  
CHULALONGKORN UNIVERSITY

สาขาวิชา วิทยาศาสตร์คอมพิวเตอร์  
ปีการศึกษา 2562

ลายมือชื่อนิสิต .....  
ลายมือชื่อ อ.ที่ปรึกษาหลัก .....

# # 6170918421 : MAJOR COMPUTER SCIENCE

KEYWORD: Travel Time, GPS data, Intelligent Transportation System, Long Short-Term Memory, Deep Neural Network

Charnwith Jakteerangkool : Travel Time Prediction From GPS Trace Data.

Advisor: Asst. Prof. VEERA MUANGSIN, Ph.D.

Prediction of travel time is an essential part of Intelligent Transportation Systems (ITS). A popular travel time prediction approach uses the historical travel time series to predict travel time shortly. This research compares short-term travel time prediction models from GPS trace data based on Recurrent Neural Network (RNN) including vanilla RNN, Long Short-Term Memory (LSTM), Gated Recurrent Unit (GRU), and some of their combinations including, LSTM-RNN, LSTM-GRU, and LSTM with Deep Neural Network layers (LSTM-DNN). The effects of a different sliding window of data, use the model created by one road segment to predict another road segment, the effect of trip direction, the different result of use data resolution both link-level and path level, and the period of day which make the model inaccuracy prediction. The evaluation results show that LSTM-DNN is the most accurate model, while the vanilla RNN model is the fastest. Furthermore, the model is more inaccurate when predicting the data in the evening period than in another period.

จุฬาลงกรณ์มหาวิทยาลัย  
CHULALONGKORN UNIVERSITY

Field of Study: Computer Science

Student's Signature .....

Academic Year: 2019

Advisor's Signature .....

## กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงได้ด้วยดีเพราะความอนุเคราะห์อย่างยิ่งของผู้ช่วยศาสตราจารย์ ดร. วีระ เหมืองสิน อาจารย์ที่ปรึกษาวิทยานิพนธ์ ซึ่งท่านได้สละเวลาให้ความรู้ คำปรึกษา ตรวจสอบ ให้คำแนะนำแนวทางการวิจัย และสนับสนุนจนทำให้การวิจัยครั้งนี้ออกมาด้วยดี ข้าพเจ้าจึงขอกราบระลึกถึงพระคุณของผู้ช่วยศาสตราจารย์ ดร. วีระ เหมืองสิน ไว้ ณ ที่นี้

ขอขอบพระคุณ รองศาสตราจารย์ ดร. เกริก ภิรมย์โสภา, รองศาสตราจารย์ ดร. สรวิต นฤปิติ และ ผู้ช่วยศาสตราจารย์ ดร. ชีรยุทธ โทหรานนท์ กรรมการสอบวิทยานิพนธ์ที่กรุณาเสียสละเวลาให้คำแนะนำ ตรวจสอบ และแก้ไขวิทยานิพนธ์ฉบับนี้

นอกจากนี้ขอขอบพระคุณกองทุนToyota Mobility ที่สนับสนุนงานวิจัยนี้ที่เป็นส่วนหนึ่งของโครงการพระราม 4

ท้ายที่สุดผู้เสนอวิทยานิพนธ์ขอกราบขอบพระคุณ คุณพ่อ และครอบครัวสำหรับกำลังใจที่มีค่า ยิ่ง รวมถึงขอขอบพระคุณผู้บังคับบัญชาในสายงาน เพื่อนร่วมงาน และมิตรสหาย ที่คอยติดตามให้กำลังใจ ให้การสนับสนุน และความช่วยเหลือในด้านต่างๆ และท่านอื่นๆ ที่มีได้กล่าวไว้ ณ ที่นี้ที่มีส่วนช่วยให้วิทยานิพนธ์ของข้าพเจ้าสำเร็จไปได้ด้วยดี

ชาญวิทย์ จักรธีรังกูร

จุฬาลงกรณ์มหาวิทยาลัย  
CHULALONGKORN UNIVERSITY

## สารบัญ

## หน้า

.....	ค
บทคัดย่อภาษาไทย.....	ค
.....	ง
บทคัดย่อภาษาอังกฤษ.....	ง
กิตติกรรมประกาศ.....	จ
สารบัญ.....	ฉ
สารบัญตาราง.....	ฅ
สารบัญภาพ.....	ญ
บทที่ 1 บทนำ.....	1
1.1. ความเป็นมาและความสำคัญของปัญหา.....	1
1.2. วัตถุประสงค์ของงานวิจัย.....	6
1.3. ขอบเขตการวิจัย.....	6
1.4. ขั้นตอนการดำเนินงาน.....	6
1.5. ประโยชน์ที่คาดว่าจะได้รับ.....	7
1.6. งานวิจัยที่ได้รับการตีพิมพ์.....	7
บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง.....	8
2.1. ทฤษฎีที่เกี่ยวข้อง.....	8
2.1.1. ข้อมูลที่นำมาใช้ในการหาระยะเวลาเดินทาง.....	8
2.1.2. โครงข่ายประสาทแบบวนซ้ำ (Recurrent Neural Network).....	13
2.1.3. โครงข่ายหน่วยความจำระยะสั้นระยะยาว (Long Short-Term Memory).....	14
2.2. งานวิจัยที่เกี่ยวข้อง.....	16

2.2.1. ข้อมูลที่ได้จากบันทึกสรุปเวลาเดินทาง .....	17
2.2.2. ข้อมูลที่ได้จากบันทึกเวลาที่รถแต่ละคันผ่านตำแหน่งที่กำหนด .....	21
2.2.3. ข้อมูลที่ได้จากเซ็นเซอร์วัดความเร็ว .....	25
2.2.4. ข้อมูลที่เก็บจากอุปกรณ์ GPS ตลอดเส้นทางการเดินทาง .....	26
2.2.5. ตารางสรุปงานวิจัยที่เกี่ยวข้อง .....	29
2.2.6. ข้อสรุปที่ได้จากการศึกษางานวิจัยที่เกี่ยวข้อง .....	33
บทที่ 3 แนวคิดและวิธีวิจัย .....	34
3.1. ข้อมูลที่นำมาวิจัย .....	34
3.2. ภาพรวมของวิธีการ .....	35
3.3. การเลือกถนนช่วงที่ใช้ในการวิจัย .....	37
บทที่ 4 การทดลองและผลการทดลอง .....	38
4.1. เครื่องมือที่ใช้ในการพัฒนา .....	38
4.1.1. เครื่องคอมพิวเตอร์ที่ใช้ในการวิจัย .....	38
4.1.2. เครื่องมือที่ใช้ในการวิจัย .....	38
4.2. การเตรียมข้อมูลก่อนเริ่มทำการวิจัย .....	38
4.3. ขั้นตอนการหาระยะเวลาเดินทาง .....	39
4.4. การเตรียมข้อมูลสำหรับนำมาสร้างโมเดลเครือข่ายประสาทเทียมแบบวนซ้ำ .....	41
4.5. การทดลองและวัดผล .....	43
4.6. ผลการทดลอง .....	49
4.6.1. การทดลองที่ 1 โมเดล Timeseries ที่ได้ผลการพยากรณ์ถูกต้องที่สุด .....	49
4.6.2. การทดลองที่ 2 ขนาดข้อมูลของ Sliding Window และ จำนวน Epoch เท่าไรให้ค่าความถูกต้องที่สุด .....	51
4.6.3. การทดลองที่ 3 นำโมเดลที่ได้จากการทดลองก่อนหน้ามาทดลองเพื่อพยากรณ์กับถนนช่วงอื่น .....	54



4.6.4 การทดลองที่ 4 การนำทิศทางการเดินทางมาพิจารณาจะให้ผลลัพธ์ที่ถูกต้องมากขึ้นหรือไม่.....	56
4.6.5 การทดลองที่ 5 เปรียบเทียบการพยากรณ์ระยะเวลาเดินทางระหว่างการนำข้อมูลในลักษณะ Link Level มาเปรียบเทียบกับ Path Level .....	57
4.6.5 การทดลองที่ 6 ทดสอบว่าหากพิจารณาในรายละเอียดของ RSME แต่ละช่วงย่อยว่า ช่วงไหนของการพยากรณ์ที่ให้ผลที่ผิดพลาดสูงหรือน้อย.....	58
4.6.6 การทดลองที่ 7 ขนาดของ Timestep และ ทำไรให้ค่าความถูกต้องที่สุด .....	59
บทที่ 5 สรุปผลการวิจัยและข้อเสนอแนะ .....	64
5.1 สรุปผลการวิจัย .....	64
5.2 ข้อจำกัด.....	66
5.3 แนวทางการวิจัยในอนาคต.....	67
5.4 ข้อเสนอแนะ .....	67
บรรณานุกรม.....	68
ประวัติผู้เขียน.....	72

## สารบัญตาราง

### หน้า

ตารางที่ 1 ผลการทดลองเปรียบเทียบที่ดีที่สุดของแต่ละโมเดลในแต่ละชุดข้อมูลที่แตกต่างกัน .....	20
ตารางที่ 2 ผลการทดลองด้วยโครงสร้างที่แตกต่างกันของแต่ละโมเดล .....	21
ตารางที่ 3 สรุปผลการทดลองในการปรับแต่งช่วงเวลาที่นำเข้าและช่วงเวลาที่พยากรณ์ล่วงหน้า.....	24
ตารางที่ 4 สรุปผลการทดลองค่าความผิดพลาดแต่ละโมเดล .....	26
ตารางที่ 5 สรุปผลการทดลองเปรียบเทียบระหว่างการนำข้อมูลค่า Median กับ Percentile ที่ 95 และการพยากรณ์จะแบ่งเป็น 1 ช่วงเวลาถัดไปจนถึง 4 ช่วงเวลาถัดไป.....	29
ตารางที่ 6 สรุปรายละเอียดงานวิจัยที่เกี่ยวข้อง.....	30
ตารางที่ 7 จำนวนทริบในแต่ละทิศทางการเดินทาง.....	47
ตารางที่ 8 การกำหนดค่าของแต่ละโมเดล.....	50
ตารางที่ 9 ผลการทดลองด้วยข้อมูลขนาด Timestep เท่ากับ 5 นาที.....	50
ตารางที่ 10 สรุปผลการทดลองเปรียบเทียบ RSME ต่างโมเดลและขนาดของ Sliding Window....	52
ตารางที่ 11 สรุปผลการทดลองเปรียบเทียบเวลาสอนและทดสอบต่างโมเดลและขนาดของ Sliding Window.....	53
ตารางที่ 12 สรุปผลการทดลองที่ 4 โดยแยกผลการพยากรณ์ด้วยโมเดลจากการทดลองที่ 2 .....	56
ตารางที่ 13 เปรียบเทียบแบบผลพยากรณ์แบบ path level และ link level.....	57
ตารางที่ 14 แสดงค่า RMSE ของโมเดลในแต่ละช่วงเวลาย่อย.....	58
ตารางที่ 15 แสดงค่า RSME ของโมเดลในแต่ละวันของสัปดาห์ .....	59
ตารางที่ 16 สรุปผลการทดลองเปรียบเทียบต่างโมเดลและขนาดของ Timestep .....	60

## สารบัญภาพ

	หน้า
ภาพที่ 1 ตัวอย่างการคำนวณระยะเวลาเดินทาง .....	2
ภาพที่ 2 ตัวอย่างการคำนวณระยะเวลาเดินทางทางอ้อม .....	2
ภาพที่ 3 วิธีการพยากรณ์ระยะเวลาเดินทางแบ่งตามประเภท .....	3
ภาพที่ 4 ตัวอย่างจุดหมายปลายทางที่มีผลต่อระยะเวลาเดินทาง .....	5
ภาพที่ 5 เวลาเดินทางจับกลุ่มตามวันของสัปดาห์.....	9
ภาพที่ 6 ตัวอย่างอุปกรณ์ Inductive Loop Detectors .....	10
ภาพที่ 7 การติดตั้งอุปกรณ์เพื่อเก็บข้อมูลการรถยนต์ที่วิ่งผ่าน.....	10
ภาพที่ 8 การติดตั้งอุปกรณ์บนบริเวณที่ทำการศึกษ.....	11
ภาพที่ 9 สองเส้นทางที่ได้รับความนิยมจาก Haidian Bridge ไปที่ Wudaokou.....	13
ภาพที่ 10 โครงสร้าง RNN ที่มีการส่งต่อค่าจาก Hidden Layer.....	14
ภาพที่ 11 โครงสร้างแบบจำลอง LSTM .....	15
ภาพที่ 12 ผลค่าความผิดพลาดของแต่ละโมเดล.....	18
ภาพที่ 13 ผลค่าความผิดพลาดของแต่ละโมเดล.....	18
ภาพที่ 14 เปรียบเทียบค่าพยากรณ์และค่าจริงตามระยะทาง .....	19
ภาพที่ 15 แผนผังโครงสร้างโมเดล DenseNet 2.....	21
ภาพที่ 16 โครงสร้าง LSTM-DNN และใช้ Linear Activation Function .....	22
ภาพที่ 17 ถนนที่เก็บข้อมูลของงานวิจัย .....	25
ภาพที่ 18 ผลลัพธ์จากการพยากรณ์.....	26
ภาพที่ 19 เปรียบเทียบผลลัพธ์จากการทดสอบโมเดลทั้ง 3 โครงสร้าง .....	27
ภาพที่ 20 การแจกแจงค่า MAE และ RMSE ของแต่ละช่วงเวลาล่วงหน้าของการพยากรณ์.....	29
ภาพที่ 21 ตัวอย่างเส้นทางที่เป็นไปได้ของรถยนต์ที่ขับบนถนนเส้นเดียวกัน .....	33

ภาพที่ 22	บริเวณของข้อมูลที่ได้รับเพื่อใช้ในการวิจัย .....	35
ภาพที่ 23	ช่วงถนนหลักที่ใช้ในการวิจัย .....	37
ภาพที่ 24	ตัวอย่างเส้นทางแท็กซี่ที่จะนำมาประมวล .....	40
ภาพที่ 25	การคำนวณระยะเวลาเดินทางของถนนช่วง B .....	40
ภาพที่ 26	ทริปที่ถูกนำออกก่อนการวิเคราะห์ .....	42
ภาพที่ 27	ช่วงถนนที่เลือกนำมาทดลองสำหรับการทดลองที่ 3 ช่วงที่ 1 .....	45
ภาพที่ 28	ช่วงถนนที่เลือกนำมาทดลองสำหรับการทดลองที่ 3 ช่วงที่ 2 .....	45
ภาพที่ 29	ช่วงถนนที่เลือกนำมาทดลองสำหรับการทดลองที่ 3 ช่วงที่ 3 .....	46
ภาพที่ 30	ช่วงถนนที่เลือกมาใช้เพื่อทดลองการทดลองที่ 4 .....	47
ภาพที่ 31	ช่วงถนนที่เลือกสำหรับการทดลองที่ 5 .....	48
ภาพที่ 32	ถนนช่วงที่ 1 ที่นำมาทดลองและผลการพยากรณ์ด้วยโมเดล LSTM-DNN .....	54
ภาพที่ 33	ถนนช่วงที่ 2 ที่นำมาทดลองและผลการพยากรณ์ด้วยโมเดล LSTM-DNN .....	55
ภาพที่ 34	ถนนช่วงที่ 3 ที่นำมาทดลองและผลการพยากรณ์ด้วยโมเดล LSTM-DNN .....	55
ภาพที่ 35	เปรียบเทียบผลจริงกับผลพยากรณ์ด้วย LSTM-DNN ขนาด Timestep เท่ากับ 5 นาที .	62
ภาพที่ 36	เปรียบเทียบผลจริงกับผลจากการพยากรณ์ด้วย LSTM-DNN ด้วยขนาด Timestep เท่ากับ 10 นาที .....	63

## บทที่ 1

### บทนำ

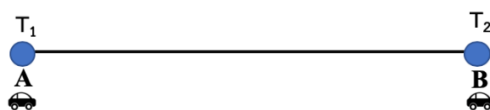
#### 1.1. ความเป็นมาและความสำคัญของปัญหา

การพยากรณ์ระยะเวลาเดินทางโดยรถยนต์จากจุดหนึ่งไปอีกจุดหนึ่งเป็นส่วนประกอบสำคัญของระบบขนส่งอัจฉริยะ (Intelligent Transportation Systems) ที่มีการศึกษาเพื่อหาวิธีที่สามารถหาระยะเวลาเดินทางที่ถูกต้องที่สุด ซึ่งแต่ละงานวิจัยมีความแตกต่างกันในแง่ต่างๆ ได้แก่ ข้อมูลที่ใช้วัดเวลาเดินทาง, วิธีการที่ใช้พยากรณ์เวลาเดินทาง, ระดับความละเอียดของการวัดเวลาเดินทาง และช่วงเวลาที่ใช้พยากรณ์

- ข้อมูลที่ใช้วัดเวลาเดินทาง สามารถแบ่งได้เป็นข้อมูล GPS trace ที่สามารถคำนวณเวลาเดินทางระหว่าง 2 ตำแหน่งใด ๆ ที่มีการบันทึกไว้, ข้อมูลเซ็นเซอร์วัดความเร็วที่สามารถคำนวณเวลาเดินทางจากความเร็วเฉลี่ยและระยะทาง, บันทึกเวลาที่รถแต่ละคันผ่านตำแหน่งที่กำหนด เช่น ด่านเก็บค่าผ่านทางโดยอุปกรณ์ RFID ที่สามารถระบุรถแต่ละคันได้ และบันทึกสรุปเวลาเดินทาง เช่น ตำแหน่งและเวลาที่รับส่งผู้โดยสารแท็กซี่
- วิธีการที่ใช้พยากรณ์เวลาเดินทาง สามารถแบ่งได้เป็นแบบ Model-Based ใช้โมเดลทางคณิตศาสตร์แสดงความสัมพันธ์ระหว่างตัวแปรต่างๆ กับเวลาเดินทาง และแบบ Data-Driven ใช้ข้อมูลในอดีตและเทคนิคด้าน Machine Learning (Bai, Lin, Ma, & Wang, 2018)
- ระดับความละเอียดของการวัดเวลาเดินทาง สามารถแบ่งได้เป็น Link level พยากรณ์เวลาเดินทางในแต่ละช่วงถนนและนำมารวมกัน, Path level พยากรณ์เวลาเดินทางตลอดเส้นทาง และ OD level พยากรณ์เวลาเดินทางระหว่าง 2 พื้นที่
- ช่วงเวลาที่พยากรณ์ สามารถแบ่งได้เป็นแบบ Short-term prediction พยากรณ์เวลาเดินทางในอนาคตไม่เกิน 1 ชั่วโมง และแบบ Long-term prediction พยากรณ์ล่วงหน้ามากกว่า 1 ชั่วโมง

การคำนวณระยะเวลาเดินทางจะแบ่งได้เป็นการคำนวณทางตรงคือการที่สามารถคำนวณระยะเวลาเดินทางได้จากชุดข้อมูลที่มีอยู่ เช่น การนำเวลาจาก GPS ณ จุดหนึ่งมาลบกับเวลาจาก GPS อีกจุดหนึ่ง และการคำนวณทางอ้อมที่จำเป็นต้องผ่านกระบวนการ หรือเครื่องมือก่อนถึงจะสามารถระยะเวลาเดินทางได้

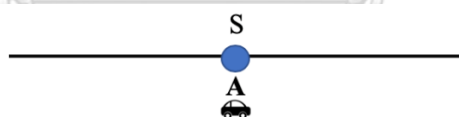
จากตัวอย่างภาพที่ 1 การคำนวณทางตรงสามารถคำนวณระยะเวลาเดินทางได้โดยนำ  $T_2 - T_1$  เช่นการที่มีอุปกรณ์ที่ติดตั้งในตำแหน่งที่กำหนด และคำนวณจากเวลาที่รถยนต์คันเดียวกันวิ่งผ่านทั้งจุด A และ B ก็สามารถคำนวณได้โดย  $T_2 - T_1$



ภาพที่ 1 ตัวอย่างการคำนวณระยะเวลาเดินทาง

การคำนวณทางอ้อมคือการที่ไม่สามารถนำข้อมูลมาคำนวณได้โดยตรงจำเป็นต้องนำข้อมูลมาแปลงก่อนการคำนวณ เช่น ชุดข้อมูลมีเพียงความเร็วจึงจำเป็นต้องนำความเร็วแปลงเป็นระยะเวลาเสียก่อนการคำนวณ, ชุดข้อมูลมีเพียงจุดที่รับและส่งผู้โดยสารจึงมีการใช้เครื่องมือ Open Source Routing Machine (OSRM) (Kankanamge et al., 2019) เพื่อช่วยในการคาดการณ์ว่ารถยนต์จะวิ่งผ่านเส้นทางใด ซึ่งมักตั้งสมมุติฐานว่ารถยนต์จะวิ่งผ่านเส้นทางที่สั้นที่สุด

ตัวอย่างการคำนวณทางอ้อมดังภาพที่ 2 จะเป็นการนำความเร็ว  $S$  ที่รถยนต์ผ่าน ณ จุด A เป็นตัวแทนของความเร็ว ณ ช่วงเวลานั้นของถนนช่วงนั้น ๆ เมื่อมาคำนวณพร้อมกับระยะทางของถนนช่วงนั้นก็จะสามารถคำนวณระยะเวลาเดินทางได้

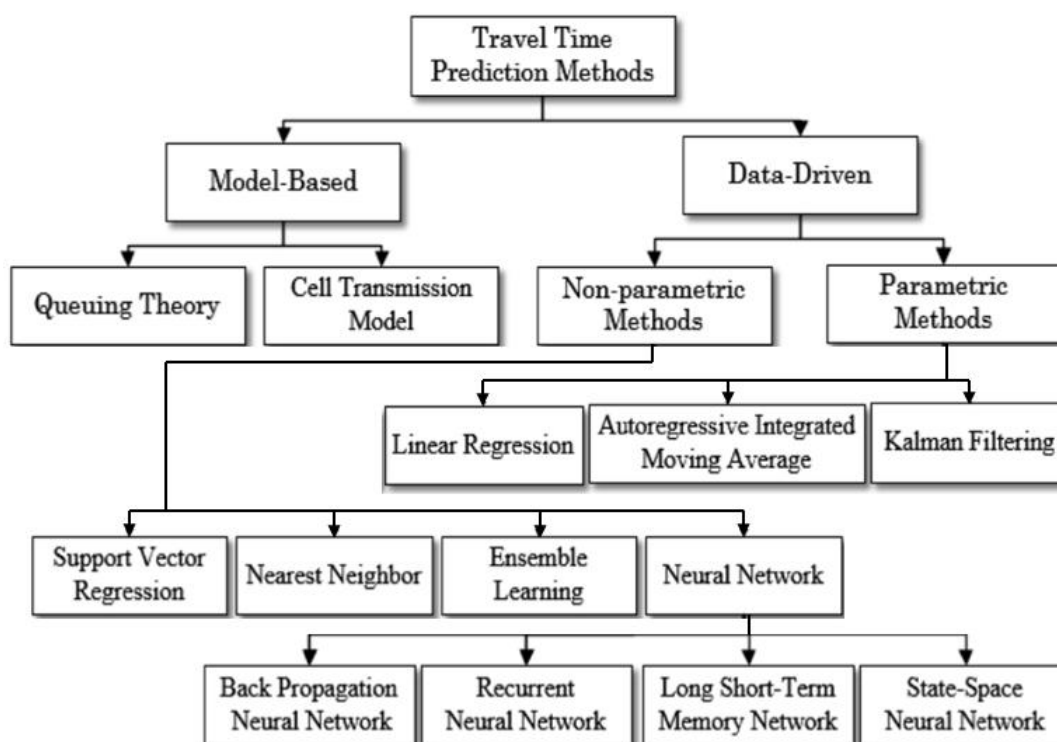


ภาพที่ 2 ตัวอย่างการคำนวณระยะเวลาเดินทางทางอ้อม

สำหรับข้อมูลสำหรับผู้วิจัยศึกษาเป็นข้อมูลที่สามารถคำนวณทางตรง โดยมาจากรถแท็กซี่ที่มีการเก็บข้อมูลตลอดเส้นทางการเดินทาง ข้อมูลประกอบด้วยพิกัด GPS, เวลา ณ ขณะนั้น, ความเร็ว, ทิศทาง และอื่น ๆ ที่เกี่ยวข้องกับการเดินทาง

ส่วนของการพยากรณ์ข้อมูลจะมีนักวิจัยพยายามหาวิธีการที่ได้คำตอบที่ถูกต้องที่สุดในการพยากรณ์ ซึ่งพบว่ามีงานวิจัยที่ทำการศึกษาเรื่องการพยากรณ์ระยะเวลาเดินทางจำนวนมาก โดย Mengting Bai และคณะได้ทำการศึกษาและรวบรวมวิธีการที่มีนักวิจัยเลือกนำมาวิจัย (Bai et al., 2018) สามารถแยกประเภทได้ดังภาพที่ 3 โดยสามารถแบ่งกลุ่มวิธีการพยากรณ์ได้เป็น 2 ประเภทหลักได้แก่วิธีการ Model-Based และวิธีการ Data-Driven

วิธีการ Model-Based คือวิธีการที่นำวิธีการทางสถิติมาสร้างโมเดลทางคณิตศาสตร์เพื่อจำลองสถานการณ์จริง โดยข้อมูลที่ได้จะนำมาปรับแต่งโมเดลเพื่อให้สามารถจำลองสถานการณ์ที่สะท้อนกับความเป็นจริงได้ถูกต้องมากยิ่งขึ้น ขณะที่วิธีการ Data-Driven จะเป็นการนำข้อมูลที่เกิดขึ้นจริงในอดีตมาสอนโมเดลที่สร้างโดยคอมพิวเตอร์ด้วยวิธีการต่าง ๆ เพื่อให้คอมพิวเตอร์ได้เรียนรู้ และพัฒนาการพยากรณ์ให้ผลลัพธ์ถูกต้องมากขึ้น



ภาพที่ 3 วิธีการพยากรณ์ระยะเวลาเดินทางแบ่งตามประเภท

ซึ่งหากพิจารณาวิธีการ Data-Driven ที่เหมาะสมกับชุดข้อมูลขนาดใหญ่จะสามารถแบ่งประเภทตามลักษณะของข้อมูลที่ใช้สอน และวิธีการที่ใช้สอนได้เป็น 2 วิธีหลักคือ วิธีการ Parametric และ วิธีการ Non-parametric ซึ่งวิธีการ Parametric เริ่มต้นจากการที่นักวิจัยตั้งสมมุติฐานว่าข้อมูลทั้งหมดมีรูปแบบการกระจายตัวอย่างใดอย่างหนึ่ง จากนั้นกำหนดจำนวนพารามิเตอร์ตามลักษณะข้อมูล และสอนโมเดลตามสมมุติฐานที่กำหนดไว้ เช่น สมมุติฐานว่าข้อมูลมีความสัมพันธ์แบบเส้นตรง จึงเลือกโมเดล Linear Regression ที่เหมาะสมกับความสัมพันธ์แบบเส้นตรง ขณะที่วิธีการ Non-parametric คือไม่มีการกำหนดจำนวนพารามิเตอร์เบื้องต้น และไม่มีการตั้งสมมุติฐานของลักษณะ

การกระจายตัวของข้อมูลเบื้องต้น โดยทำการสอนคอมพิวเตอร์ให้เรียนรู้จากข้อมูล และสร้างโมเดล ซึ่งจะเน้นที่นำข้อมูลรับเข้ามาสอน และปรับแต่งฟังก์ชันภายใน เช่น วิธีการ Neural Network

วิธีการ Neural Network เป็นวิธีการใหม่ที่ใช้ข้อมูลที่เกิดขึ้นในอดีตไปพยากรณ์ผลลัพธ์ที่จะเกิดขึ้นในอนาคต (Kosolsombat & Saraubon, 2018) วิธีการ Neural Network จะเป็นวิธีการที่มีการปรับแต่งฟังก์ชัน และ คำนวณภายในตลอดเวลาตามลักษณะของข้อมูลนำเข้า ส่งผลให้วิธีการ Neural Network เรียนรู้ตลอดเวลาและได้ค่าพยากรณ์ที่ถูกต้องมากขึ้น ซึ่ง Neural Network ที่เหมาะสมสำหรับการนำมาพยากรณ์ข้อมูลที่มีลักษณะเป็น Time Series คือแบบ Recurrent Neural Network (RNN) และ Neural Network ที่พัฒนาต่อจาก RNN เช่น Long Short Term Memory Neural Network (LSTM), Gated Recurrent Unit (GRU) เป็นต้น

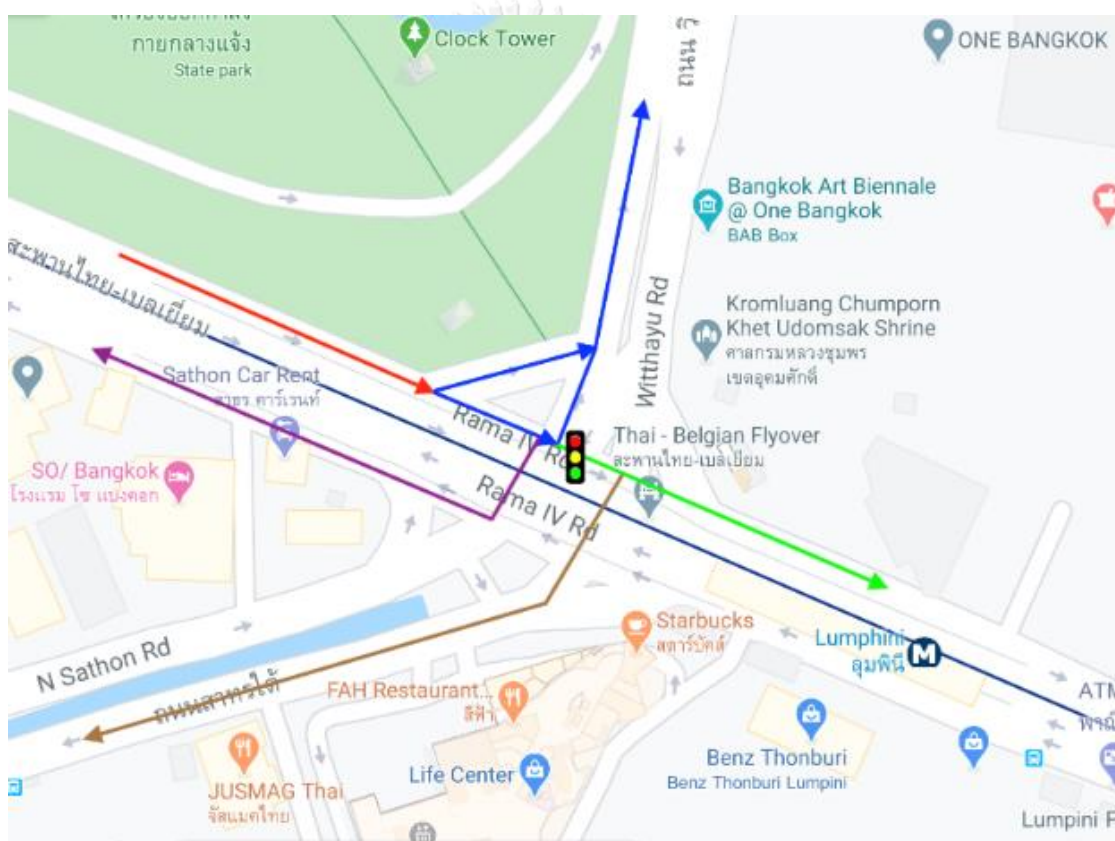
จากการศึกษาเพิ่มเติมพบงานวิจัยของ Narayanan และคณะ จะเป็นการพยากรณ์ระยะเวลาเดินทางล่วงหน้า โดย Narayanan และคณะทำการเปรียบเทียบระหว่างวิธีการ LSTM, วิธีการ Linear Regression และ วิธีการ Support Vector Machine (SVM) (Narayanan, Pranesh, Nagavarapu, Kumar, & Dauwels, 2019) พบว่าวิธีการ LSTM ให้ผลลัพธ์ที่ดีกว่าเมื่อวัดค่าความผิดพลาดด้วยวิธี Mean Absolute Percentage Error (MAPE) อยู่ระหว่าง 0.5% ถึง 6.8% ซึ่งเมื่อนำมาเปรียบเทียบกับวิธีการ Linear Regression และ Support Vector Machine พบว่าวิธีการ LSTM ให้ผลลัพธ์ที่ดีกว่า 3.5% และ 2.5% ตามลำดับ ดังนั้นวิธีการ LSTM เหมาะสมให้นำมาวิเคราะห์ข้อมูลประเภท Time Series มากกว่าวิธีการ Linear Regression และ Support Vector Machine

นอกจากนี้ยังมีงานวิจัยของ Liu และคณะ ที่นำวิธีการ LSTM มารวมกับวิธีการ Deep Neural Network (DNN) เพื่อช่วยในการพยากรณ์ให้ได้ผลลัพธ์ที่ดีขึ้น (Liu, Wang, Yang, & Zhang, 2017) ซึ่งพบว่าวิธีการดังกล่าวให้ผลลัพธ์ที่ดีกว่าวิธีการ Linear Regression, วิธีการ Ridge Regression, วิธีการ Lasso Regression, วิธีการ Deep Neural Network (DNN) และวิธีการ Auto Regressive Integrated Moving Average (ARIMA) ถึงแม้ว่าจะได้ผลลัพธ์ที่ถูกต้องมากกว่าวิธีการอื่น แต่นักวิจัยให้ความเห็นว่าวิธีการดังกล่าวยังคงมีปัญหาในการปรับแต่งโมเดล DNN เนื่องจากโมเดลมีความซับซ้อนเป็นอย่างมากทำให้การปรับแต่งโครงสร้างโมเดลทำได้ยาก ซึ่งส่งผลต่อการไปประยุกต์ใช้งานในสถานการณ์จริง

จากการศึกษาวิจัยที่เกี่ยวข้องพบว่าหลายงานวิจัยใช้วิธีคำนวณเวลาเดินทางบนถนนเป็นช่วง ๆ (link-level method) ถ้าหากต้องการพิจารณาการเดินทางจากจุด A ไปจุด B ที่เดินทางผ่าน



หลายช่วงถนนจะใช้วิธีการนำเวลาแต่ละช่วงถนนมารวมกันเพื่อให้ได้เวลาการเดินทางรวมทั้งหมด ซึ่งจากการนำข้อมูลมาพิจารณาเบื้องต้นพบว่าระยะเวลาเดินทางขึ้นกับจุดหมายปลายทาง ระยะเวลาเดินทางผ่านถนนช่วงหนึ่งมีความแตกต่างกันถึงแม้ว่าจะอยู่บนถนนช่วงเดียวกันดังภาพที่ 4 ที่จุดหมายปลายทางเป็นสิ่งกำหนดช่องทางการเดินรถยนต์ และช่องทางการเดินรถยนต์จะเป็นสิ่งกำหนดว่ารถยนต์จำเป็นต้องรอสัญญาณไฟจราจรก่อนการเดินทางต่อหรือไม่ ด้วยปัญหาในลักษณะดังกล่าวจึงเป็นที่มาของปัญหาที่สนใจจะทำการศึกษาวิจัยเพิ่มเติมว่า ลักษณะการเดินทางที่แตกต่างกันมีผลต่อการพยากรณ์ระยะเวลาเดินทางหรือไม่



ภาพที่ 4 ตัวอย่างจุดหมายปลายทางที่มีผลต่อระยะเวลาเดินทาง

ดังนั้นงานวิจัยนี้จะทำการพยากรณ์ระยะเวลาเดินทางโดยรถยนต์ด้วยโมเดล LSTM จากข้อมูล GPS ที่เก็บข้อมูลตลอดเส้นทางการเดินทางและผ่านการแบ่งข้อมูลออกตามลักษณะการเดินทางของรถยนต์ โดยจะทำการเปรียบเทียบกับโมเดล LSTM จากข้อมูลที่ไม่มีการแบ่งออกตามลักษณะการเดินทางของรถยนต์ และเปรียบเทียบกับโมเดลอื่นด้วยข้อมูลชุดเดียวกัน

## 1.2. วัตถุประสงค์ของงานวิจัย

เพื่อพัฒนาวิธีการพยากรณ์ระยะเวลาเดินทางที่มีความแม่นยำ โดยใช้ข้อมูล GPS trace และเทคนิค Recurrent Neural Network ที่ขึ้นกับเส้นทางการเดินทาง

## 1.3. ขอบเขตการวิจัย

- ข้อมูลที่ใช้ในการวิจัยจะเป็นข้อมูลจากรถแท็กซี่ที่เก็บข้อมูลการเดินทางที่เดินทางภายในบริเวณถนนพระราม 4
- ชุดข้อมูลจะมีเพียงข้อมูล GPS จากรถแท็กซี่ที่วิ่งผ่านในพื้นที่ถนนพระราม 4 ซึ่งประกอบด้วยข้อมูล GPS, เวลา ณ ขณะนั้น, ความเร็ว, จุดเริ่มต้นช่วงของถนนที่จุดนั้นปรากฏ, จุดสิ้นสุดช่วงของถนนที่จุดนั้นปรากฏ และ อื่น ๆ
- ใช้ Neural Network แบบ Recurrent Neural Network (RNN) และ Neural Network ที่พัฒนาต่อจากโมเดล RNN เช่น Long Short Term Memory Neural Network (LSTM), Gated Recurrent Unit (GRU) เป็นต้น
- เครื่องมือที่ใช้จะเป็นฐานข้อมูลโพสต์เกรสคิวแอล และส่วนขยายเพิ่มเติมที่ชื่อว่า โพสต์จิส
- จะทำการพยากรณ์เวลาเดินทางระหว่างทางแยกหลักบนถนนพระราม 4
- โมเดลทั้งหมดถูกสร้างโดยภาษาคอมพิวเตอร์ Python และประเมินผลงานวิจัยเปรียบเทียบกับวิธีการฟังก์ชันค่าความผิดพลาด Root Mean Square Error

## 1.4. ขั้นตอนการดำเนินงาน

1. ศึกษาและทำความเข้าใจทฤษฎี และงานวิจัยที่เกี่ยวข้อง
2. ศึกษาและเตรียมข้อมูล ทำความสะอาด เพื่อใช้ในการวิจัย
3. ศึกษาและเลือกเครื่องมือสำหรับสร้างโมเดลเพื่อพยากรณ์ช่วงเวลาเดินทาง
4. ออกแบบและทำการสร้างโมเดลด้วยการตั้งค่าที่แตกต่างกัน
5. ทดสอบความถูกต้องของโมเดลด้วยวิธีการฟังก์ชันค่าความผิดพลาด Root Mean Square Error
6. วิเคราะห์และประเมินผลการวิจัย
7. ทำการสร้างโมเดลด้วยวิธีการอื่นเพื่อนำผลที่ได้มาเปรียบเทียบ
8. สรุปผลการวิจัยและข้อเสนอแนะ

9. ตีพิมพ์ผลงานทางวิชาการ
10. จัดทำวิทยานิพนธ์

#### 1.5. ประโยชน์ที่คาดว่าจะได้รับ

1. ได้แนวทางการสร้างโมเดลสำหรับการพยากรณ์ระยะเวลาเดินทางล่วงหน้าระยะสั้น
2. ได้แนวทางการเตรียมข้อมูลสำหรับการสร้างโมเดลจากข้อมูลรถแท็กซี่
3. ได้โมเดลที่มีค่าความผิดพลาดน้อยที่สุดเพื่อไปประยุกต์ใช้ในงานจริง
4. ได้แนวทางในการปรับการตั้งค่าโมเดลให้เหมาะสมกับข้อมูลจริงที่แตกต่างกันเพื่อปรับใช้ในอนาคต

#### 1.6. งานวิจัยที่ได้รับการตีพิมพ์

ส่วนหนึ่งของวิทยานิพนธ์นี้ได้นำเสนอในการประชุมวิชาการดังนี้

Charnwith Jakteerangkool and Veera Muangsin, “Short-Term Travel Time Prediction from GPS Trace Data using Recurrent Neural Networks”, 2020 Asia Conference on Computers and Communications (ACCC 2020), Singapore, December 4-6, 2020



## บทที่ 2

### ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

#### 2.1. ทฤษฎีที่เกี่ยวข้อง

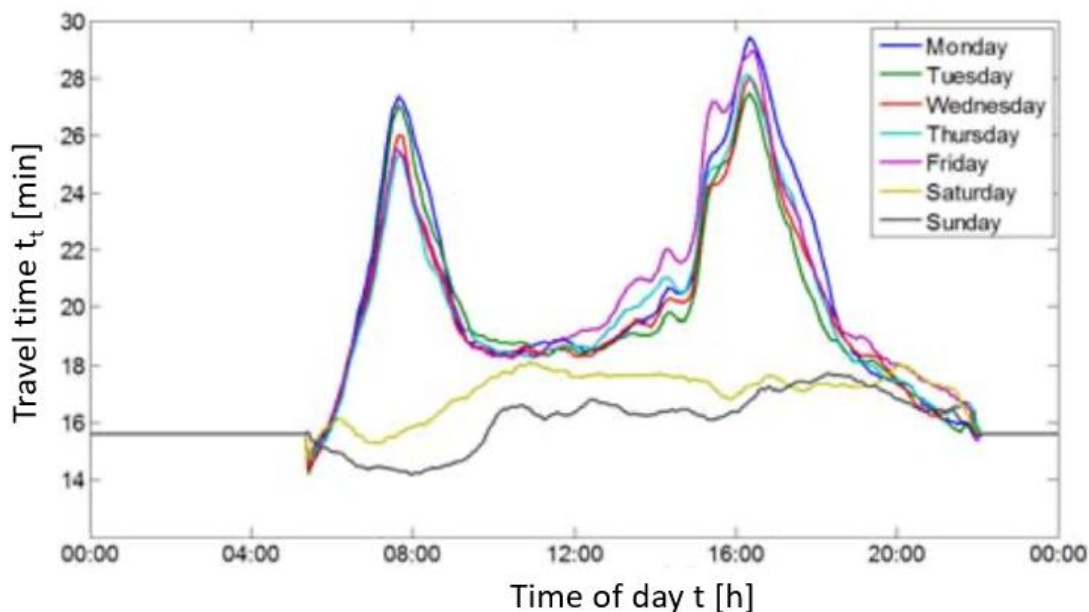
##### 2.1.1. ข้อมูลที่นำมาใช้ในการหาระยะเวลาเดินทาง

จากการศึกษางานวิจัยที่เกี่ยวข้องพบว่าสามารถแบ่งข้อมูลที่นำมาใช้ในการหาระยะเวลาที่ใช้ในการเดินทางได้เป็น 4 ลักษณะตามแหล่งที่มาของข้อมูลโดย

- ข้อมูลที่ได้จากเซ็นเซอร์วัดความเร็ว

แหล่งข้อมูลที่ได้จากเซ็นเซอร์วัดความเร็วจะเป็นแหล่งข้อมูลความเร็วเฉลี่ยของรถยนต์ที่วิ่งบนถนนเส้นที่สนใจเพื่อคำนวณเวลาเดินทางจากความเร็วเฉลี่ยและระยะทาง เช่น งานวิจัยของ (Erdelić, Ravlić, & Carić, 2016) มีการใช้ข้อมูลประวัติความเร็วในอดีตซึ่งเป็นข้อมูลของ SORDITO project (RC.2.2.08 – 0022) ประเทศโครเอเชีย เมือง Zagreb, Split, Osijek และ Rijeka โดยเก็บข้อมูลจากแต่ละส่วนของถนนบริเวณจากสี่แยกหนึ่งไปอีกสี่แยกหนึ่ง

โดยนักวิจัยได้แยกข้อมูลออกเป็น 2 กลุ่มคือกลุ่มกลางวันคือระหว่างเวลา 05:30 น. ถึง 22:00 น. และกลุ่มกลางคืนคือระหว่างเวลา 22:00 น. ถึง 05:30 น. นอกจากนี้ยังจับกลุ่มตามวันของสัปดาห์ และตามฤดูกาลเพราะนักวิจัยคาดว่าช่วงหน้าร้อนที่เป็นช่วงนักท่องเที่ยวมาท่องเที่ยวจำนวนมากจะส่งผลต่อความเร็วเช่นกันดังภาพที่ 5 ที่แสดงเวลาเดินทางเฉลี่ยที่คำนวณจากความเร็วโดยจับกลุ่มตามวันของสัปดาห์

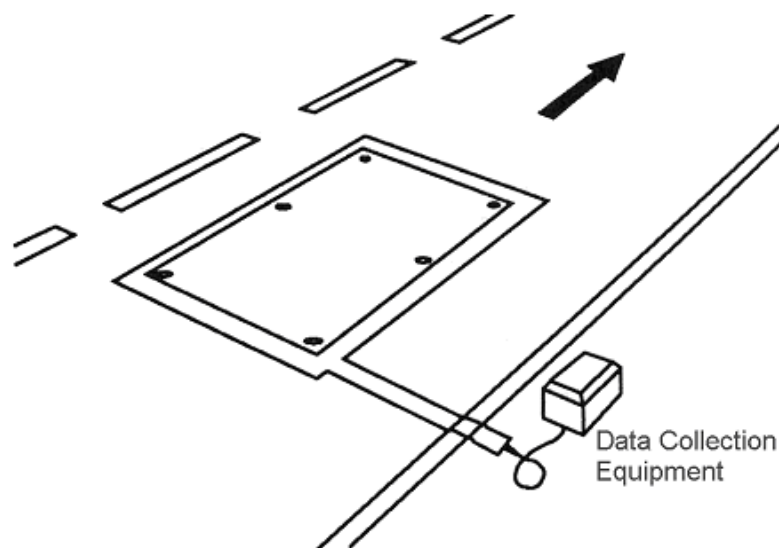


ภาพที่ 5 เวลาเดินทางจับกลุ่มตามวันของสัปดาห์

เมื่อมีข้อมูลความเร็วที่วิจัยจึงสามารถคำนวณเวลาเดินทางได้ หาเส้นทางการเดินทางที่สั้นที่สุดด้วยอัลกอริทึม Dijkstra แล้วจึงพยากรณ์เวลาเดินทางในแต่ละช่วงเวลาจากเวลาเดินทางที่ได้จากความเร็ว ซึ่งเมื่อนักวิจัยเปรียบเทียบกับอุปกรณ์นำทางที่มีขายอยู่ด้วยวิธีการ relative error ก็พบว่าวิธีการดังกล่าวให้ความถูกต้องที่มากกว่า โดยอุปกรณ์ที่มีขายอยู่ให้ค่าความผิดพลาดเท่ากับ 22.07% ขณะที่วิธีที่นักวิจัยนำเสนอให้ค่าความผิดพลาดระหว่าง -1.86% ถึง -12.35%

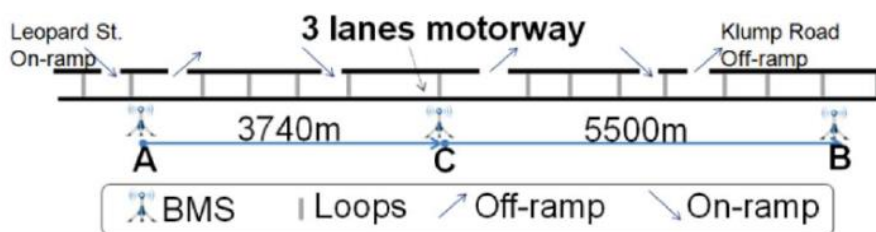
- ข้อมูลที่ได้บันทึกเวลาที่รถแต่ละคันผ่านตำแหน่งที่กำหนด

แหล่งข้อมูลที่ได้จากการติดตั้งอุปกรณ์ตรวจจับ ณ ตำแหน่งที่กำหนดเพื่อนำข้อมูลที่ได้มาคำนวณเพื่อหาระยะเวลาเดินทาง ตัวอย่างอุปกรณ์ที่นิยมนำมาใช้ในการเก็บข้อมูล คือ อุปกรณ์ Bluetooth, อุปกรณ์ Inductive Loop Detectors, กล้องตรวจจับ เป็นต้น อุปกรณ์ที่แตกต่างกันนำมาซึ่งวิธีการคำนวณที่แตกต่างกัน เช่น การติดตั้งอุปกรณ์ Bluetooth 2 จุดจะได้ข้อมูลเวลาที่รถยนต์คันเดียวกันวิ่งผ่าน 2 จุดทำให้สามารถนำข้อมูลที่ได้มาคำนวณทางตรงเพื่อหาระยะเวลาเดินทาง, การติดตั้งอุปกรณ์ Inductive Loop Detectors ดังภาพที่ 6 จำนวน 2 จุดที่ใกล้กันจะสามารถนำเวลาสองจุดมาคำนวณความเร็ว และนำระยะทางในช่วงที่สนใจ กับความเร็วที่ได้มาคำนวณทางอ้อมเพื่อหาระยะเวลาในการเดินทาง แต่ข้อเสียของอุปกรณ์ดังกล่าวคือไม่สามารถตรวจจับได้ว่าข้อมูลสองจุดเป็นรถยนต์คันเดียวกันหรือไม่ จึงจำเป็นต้องวางอุปกรณ์ไว้ใกล้กันเพื่อให้สามารถมั่นใจได้ว่าข้อมูล 2 จุดเกิดจากรถยนต์คันเดียวกัน



ภาพที่ 6 ตัวอย่างอุปกรณ์ Inductive Loop Detectors

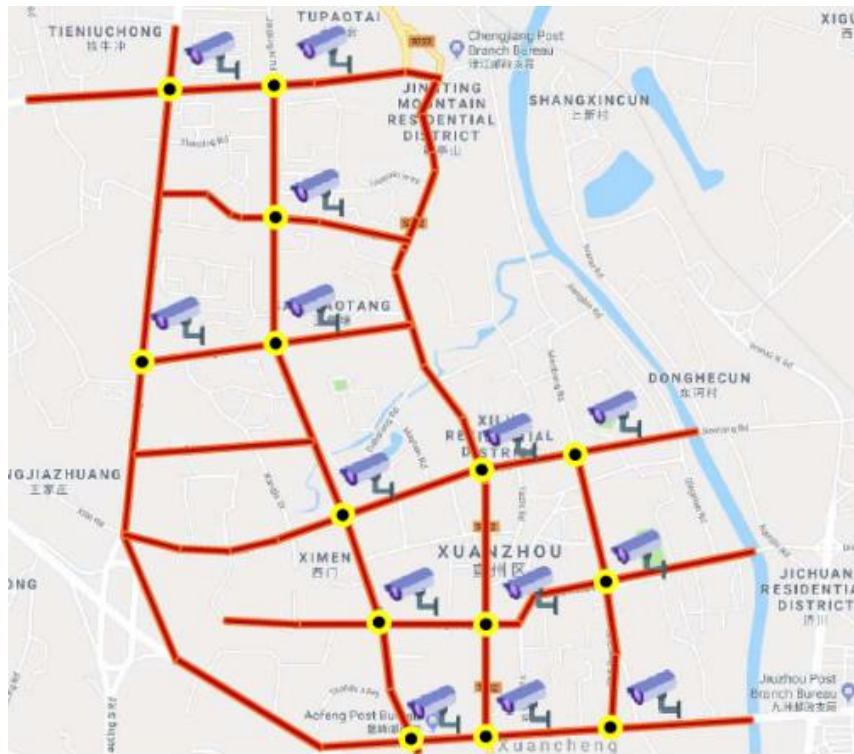
จากการศึกษาพบว่างานวิจัยของ Ashish Bhaskar และคณะ (Bhaskar, Qu, & Chung, 2015) ได้ศึกษาเรื่อง Bluetooth Vehicle Trajectory by Fusing Bluetooth and Loops: Motorway Travel Time Statistics โดยทำการติดตั้งอุปกรณ์ Bluetooth 3 จุดเพื่อเก็บข้อมูลบนถนน Pacific Motorway เมือง Brisbane ประเทศ Australia ดังภาพที่ 7 ที่แสดงการติดตั้งอุปกรณ์ 3 จุดเพื่อเก็บข้อมูล โดยใช้ข้อมูลร่วมกับอุปกรณ์ Inductive Loop Detectors ที่ติดตั้งบริเวณพื้นถนน โดยนำข้อมูลที่ได้มาทำการสร้างแบบจำลองเพื่อสร้างโมเดลทางสถิติ



ภาพที่ 7 การติดตั้งอุปกรณ์เพื่อเก็บข้อมูลการรถยนต์ที่วิ่งผ่าน

นอกจากนี้ยังมีงานวิจัยของ Yiting Zhu และคณะ (Zhu, He, & Sun, 2019) ได้ทำการศึกษาเรื่อง Network-Wide Link Travel Time Inference Using Trip-Based Data From Automatic Vehicle Identification Detectors โดยใช้ข้อมูลจาก automatic vehicle identification (AVI) ซึ่งข้อมูลประกอบด้วยทะเบียนรถยนต์ที่วิ่งผ่านจุดที่ติดตั้งอุปกรณ์ดังภาพที่ 8 และอื่น ๆ นักวิจัยได้ทำการศึกษาระยะเวลาเดินทางจากอุปกรณ์กล้องตรวจจับที่ติดตั้ง 2 จุด และ

นำเข้าโมเดล probabilistic trip travel time allocation (PTTA) เพื่อหาเส้นทางที่เป็นไปได้ที่รถยนต์คันดังกล่าวจะวิ่งผ่าน เพื่อใช้ในการคำนวณระยะเวลาเดินทาง



ภาพที่ 8 การติดตั้งอุปกรณ์บนบริเวณที่ทำการศึกษ

- ข้อมูลที่ได้บันทึกสรุปเวลาเดินทาง

แหล่งข้อมูลในลักษณะดังกล่าวจะเป็นข้อมูลในลักษณะสรุปการเดินทางของรถแท็กซี่ในแต่ละครั้งที่ให้บริการผู้โดยสาร โดยข้อมูลลักษณะดังกล่าวมักจะประกอบด้วยจุดเริ่มต้นการเดินทาง, จุดสิ้นสุดการเดินทาง, ระยะเวลาเดินทาง, ระยะทางที่เดินทาง และข้อมูลอื่นที่เกี่ยวข้องกับการให้บริการ ข้อมูลในลักษณะดังกล่าวจะไม่สามารถนำมาคำนวณระยะเวลาเดินทางได้โดยตรงเนื่องจากไม่ทราบแน่ชัดว่ารถแท็กซี่วิ่งผ่านเส้นทางใดบ้างในการเดินทาง การนำข้อมูลในลักษณะดังกล่าวมาใช้ในการวิจัยมักจะเป็นต้องหาเครื่องมือเพิ่มเติมเพื่อหาเส้นทางเดินทางที่เป็นไปได้ เช่น เครื่องมือ OSRM ที่ช่วยหาเส้นทางที่เป็นไปได้จากจุดเริ่มต้นและจุดสิ้นสุด งานวิจัยที่ใช้ข้อมูลในลักษณะดังกล่าวมักจะตั้งสมมุติฐานว่ารถแท็กซี่จะวิ่งด้วยเส้นทางที่สั้นที่สุดที่เป็นไปได้ ซึ่งความเป็นจริงรถแท็กซี่อาจจะไม่ได้วิ่งเส้นทางดังกล่าว

ด้วยลักษณะข้อมูลดังกล่าวมีข้อเสียที่ไม่สามารถระบุได้ชัดเจนว่ารถแท็กซี่วิ่งผ่านถนนเส้นใด และวิ่งผ่านด้วยระยะเวลาเท่าใด จึงไม่สามารถสรุปได้ว่าระยะเวลาเดินทางบนถนนช่วงที่สนใจจะใช้ เวลาตั้งที่ได้จากการคำนวณเส้นทางผ่านเครื่องมือ OSRM

งานวิจัยที่ใช้ข้อมูลในลักษณะนี้จะต้องทำการผ่านโมเดลเพื่อคำนวณหาเส้นทางที่เป็นไปได้ที่ รถแท็กซี่จะวิ่งผ่านคล้ายกับแบบแรก เช่น งานวิจัยของ Raghv Ganti และคณะ (Ganti, Srivatsa, & Abdelzahr, 2014) ได้ทำการศึกษารื่อง On Limits of Travel Time Predictions: Insights from a New York City Case Study ได้ใช้ข้อมูลสรุปเส้นทางการเดินทางของรถแท็กซี่ทุกคันใน เมือง New York ทั้งหมด 2 ปี

ข้อมูลที่ได้ประกอบด้วย ระยะเวลาเดินทาง, ระยะทางที่ใช้ในการเดินทาง, จุดเริ่มต้น, จุดสิ้นสุด, เวลาที่เริ่มต้น, เวลาที่สิ้นสุด และค่าโดยสาร โดยนักวิจัยได้ทำการศึกษาพบว่าโมเดลการ พยากรณ์ระยะเวลาเดินทางจะมีความถูกต้องถ้าใช้ Parameter เป็นระยะทาง และเวลาของวัน แต่ กระจนั้นนักวิจัยก็ระบุว่า Parameter ดังกล่าวเหมาะกับข้อมูลรถแท็กซี่ภายในเมือง New York เท่านั้นและไม่อาจสรุปได้ว่าจะเหมาะกับข้อมูลรถแท็กซี่ในเมืองอื่น ซึ่งเป็นสิ่งที่นักวิจัยตั้งใจจะ ทำการศึกษาเพิ่มเติม

- **ข้อมูลที่เกิดจากอุปกรณ์ GPS ตลอดเส้นทางการเดินทาง**

ข้อมูลที่เกิดจากอุปกรณ์ GPS ตลอดเส้นทางการเดินทาง (GPS Trace Data) เป็น แหล่งข้อมูลในลักษณะดังกล่าวจะมีการเก็บข้อมูล GPS จากอุปกรณ์ที่ติดตั้งภายในรถยนต์รวมถึง โทรศัพท์เคลื่อนที่ภายในรถแท็กซี่ ข้อมูลมักจะประกอบด้วยพิกัด GPS, เวลา ณ ขณะนั้น, ความเร็ว ซึ่งความถี่ของการเก็บข้อมูลก็จะแตกต่างกันตามแหล่งของข้อมูล เช่น ข้อมูลจาก Intelligent Traffic Information Center (ITIC) ที่จะเก็บข้อมูลทุก 1 นาที, ข้อมูลจาก Grab ที่ใช้ในการวิจัยจะเก็บ ข้อมูลทุก 4 วินาที

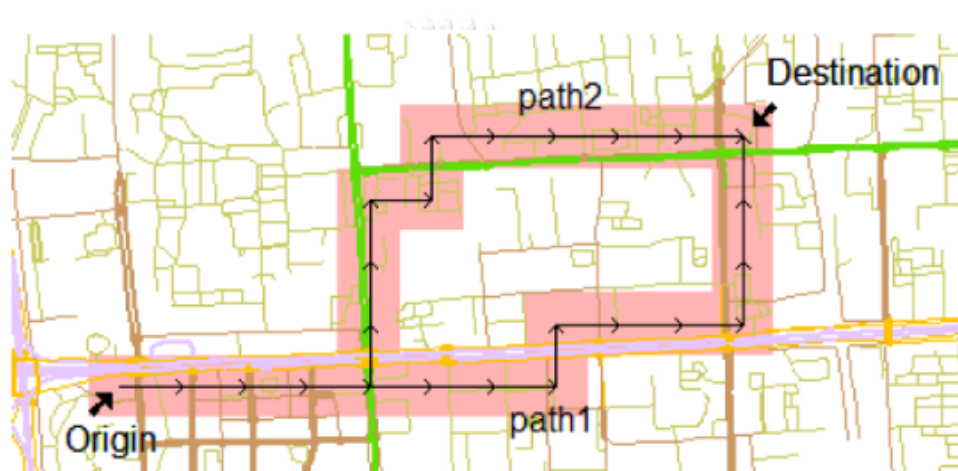
ข้อมูลในลักษณะนี้มีข้อดีที่สามารถระบุได้ชัดเจนว่ารถแท็กซี่ผ่านเส้นทางไหนบ้างเมื่อเวลาใด ส่งผลให้เมื่อทำการหาระยะเวลาเดินทางบนช่วงของถนนที่สนใจจะทำให้ผลลัพธ์ที่ถูกต้องมากกว่า ข้อมูลในลักษณะอื่น เพราะเป็นข้อมูลที่เกิดขึ้นจริงบนถนน

จากการศึกษาพบว่าข้อมูลดังกล่าวมีการถูกนำมาใช้งานวิจัยของ Mengdan Gao และคณะ (Gao, Zhu, Wan, & Wang, 2013) ได้ทำการวิจัยเรื่องของการ Analysis of Travel Time Patterns in Urban Using Taxi GPS Data โดยใช้ข้อมูลที่เกิดจากอุปกรณ์ GPS ที่มีลักษณะดังกล่าวเพื่อเส้นทาง



ที่มีประสิทธิภาพที่สุดในการเดินทางจากจุดตั้งต้นไปยังจุดหมายปลายทาง นักวิจัยมีข้อมูลจำนวน 32.6 ล้านเส้นทางการเดินทางภายในเมือง Beijing จากการเก็บข้อมูลจำนวน 3 เดือน

โดยนักวิจัยได้ทำการนำข้อมูลเข้าโมเดล Weighted Moving Average (WMA) เพื่อหา รูปแบบระยะเวลาเดินทางจากข้อมูลเส้นทางการเดินทางทั้งหมด รูปแบบเส้นทางที่ได้คือเส้นทางที่ มักจะถูกใช้ในการเดินทางจากจุดหนึ่งไปยังอีกจุดหนึ่ง ณ ช่วงเวลาของวัน ซึ่งผลลัพธ์ที่ได้สามารถ แสดงตัวอย่างดังภาพที่ 9 ที่จะคำนวณเส้นทางการเดินทางที่ได้รับความนิยมสำหรับการเดินทางจาก Origin ไป Destination

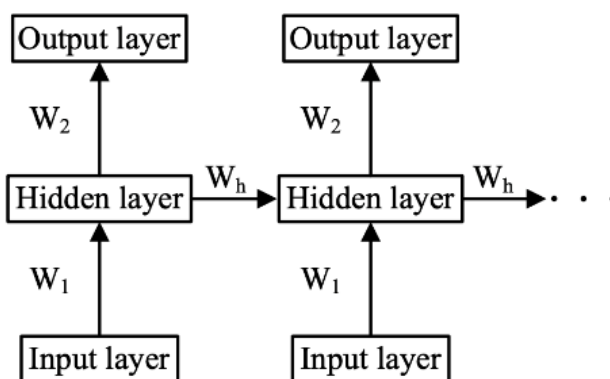


ภาพที่ 9 สองเส้นทางที่ได้รับความนิยมจาก Haidian Bridge ไปที่ Wudaokou

ซึ่งผลลัพธ์ที่ได้จากโมเดลจะถูกนำมาใช้เพื่อแนะนำเส้นทางที่เหมาะสมสำหรับการเดินทาง จากจุด Origin ไป Destination ณ เวลา  $T$  ว่าควรจะเดินทางด้วยรถยนต์ไปเส้นทาง path1 หรือ path2 เพื่อให้ได้ระยะเวลาเดินทางที่สั้นที่สุด

### 2.1.2. โครงข่ายประสาทแบบวนซ้ำ (Recurrent Neural Network)

Recurrent Neural Network หรือ RNN เป็น Neural Network ที่มีการส่งต่อค่าที่ได้จาก Hidden Layer ก่อนหน้าไปยัง Hidden Layer ในรอบถัดไปตามลำดับ (Duan, Lv, & Wang, 2016) การเชื่อมต่อในลักษณะนี้จะส่งผลให้ผลลัพธ์ที่ได้จากการคำนวณรอบก่อนหน้ามีผลสืบเนื่องมาจากการคำนวณในรอบปัจจุบัน เช่น การพยากรณ์ตัวอักษรที่จะพบว่าตัวอักษรของรอบก่อนหน้ามีผลต่อตัวอักษรถัดไปว่าจะเป็นตัวอักษรอะไร โครงสร้าง Recurrent Neural Network สามารถแสดงได้ดังภาพที่ 10



ภาพที่ 10 โครงสร้าง RNN ที่มีการส่งต่อค่าจาก Hidden Layer

จากภาพที่ 10 แสดงให้เห็นถึงการส่งต่อค่าน้ำหนักจาก Hidden Layer ก่อนหน้าไปยัง Hidden Layer ลำดับถัดไปเพื่อใช้ในการประมวลผลผลลัพธ์ของรอบนั้น ซึ่งสามารถอธิบายได้ดังสมการ (1) และ (2)

$$h_t = f(h_{t-1}, x_t) \quad (1)$$

$$y_t = W_{hy} h_t \quad (2)$$

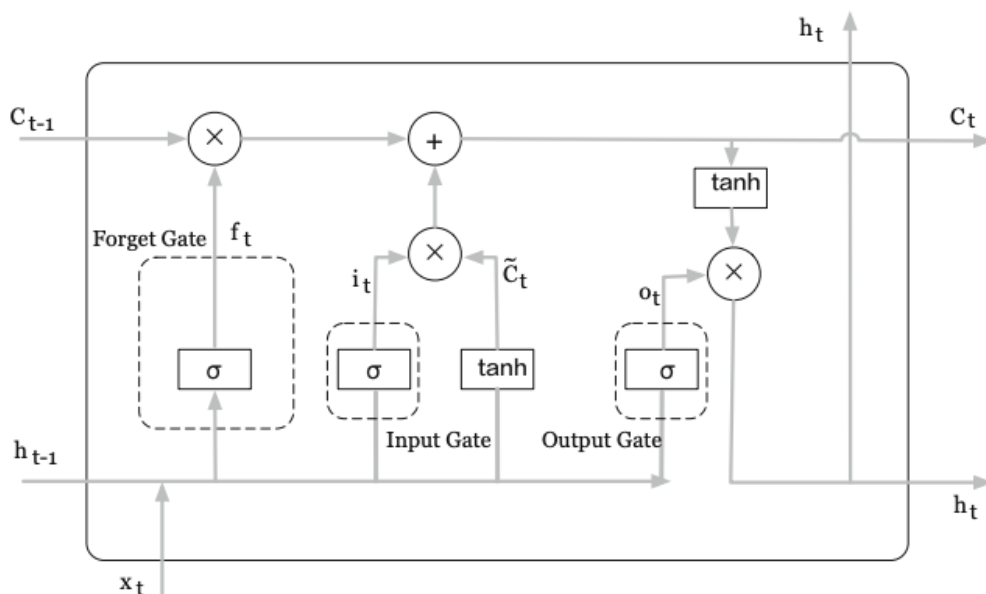
เมื่อ  $h_t$  คือค่าของ Hidden Layer ที่ผ่านการคำนวณด้วยฟังก์ชัน  $f$  เช่น tanh โดยนำค่าของ Hidden Layer ก่อนหน้าไปทำการคำนวณพร้อมกับค่า input ผลลัพธ์ที่ได้คือ Hidden Layer ของรอบนั้น จากนั้นจึงทำการคำนวณร่วมกับค่าน้ำหนักเพื่อให้ได้ค่า Output

### 2.1.3. โครงข่ายหน่วยความจำระยะสั้นระยะยาว (Long Short-Term Memory)

Long Short-Term Memory หรือ LSTM คือ Neural Network อีกประเภทที่ได้รับการพัฒนามาจากวิธีการ Recurrent Neural Network โดย LSTM มีการเพิ่มความสามารถในการตัดสินใจเลือกค่าที่ได้จากการคำนวณรอบก่อนหน้าว่าจะนำมาคำนวณร่วมกับรอบปัจจุบันหรือไม่ โดย LSTM มีการเพิ่ม Forget Gate เพื่อช่วยในการตัดสินใจ (Wei, Jia, Liu, & Yu, 2018) ซึ่งการเพิ่ม Forget Gate ทำให้ LSTM มีลักษณะเหมือนกับลักษณะการตัดสินใจของมนุษย์มากขึ้น

วิธีการ LSTM ถูกนำเสนอครั้งแรกโดย S. Hochreiter (Li & Jiang, 2017) วิธีการ LSTM ประสบความสำเร็จอย่างมากในการปรับใช้กับข้อมูลที่มีลักษณะ Time Series เช่น การประมวลผลภาษาธรรมชาติ (Natural Language Processing), การรู้จำเสียง (Speech Recognition) รวมไปถึงการประยุกต์ใช้ร่วมกับข้อมูลประเภทการจราจรที่จะใช้ในงานวิจัย

วิธีการ LSTM มีโครงสร้างที่สามารถอธิบายได้ดังภาพที่ 11



ภาพที่ 11 โครงสร้างแบบจำลอง LSTM

เมื่อกำหนดให้ ณ เวลา  $t$ , มีข้อมูล Input  $x_t$ , ผลลัพธ์ของชั้น Hidden Layer ของรอบปัจจุบันคือ  $h_t$  และผลลัพธ์ของ Hidden Layer ของช่วงเวลาก่อนหน้า  $h_{t-1}$ , สภาวะ Cell Input ของรอบปัจจุบัน  $\tilde{C}_t$  และสภาวะ Cell Input ของช่วงเวลาก่อนหน้า  $\tilde{C}_{t-1}$  และสภาวะของทั้ง 3 Gate คือ Input Gate  $i_t$  Forget Gate  $f_t$  และ Output Gate  $o_t$  จะได้สมการเพื่ออธิบายทั้ง 3 สภาวะของ Gate และ สภาวะ Cell Input ดังนี้

Input Gate

$$i_t = \sigma(W_1^i \cdot x_t + W_h^i \cdot h_{t-1} + b_i) \quad (3)$$

Forget Gate

$$f_t = \sigma(W_1^f \cdot x_t + W_h^f \cdot h_{t-1} + b_f) \quad (4)$$

Output Gate

$$o_t = \sigma(W_1^o \cdot x_t + W_h^o \cdot h_{t-1} + b_o) \quad (5)$$

Cell Input

$$\tilde{C}_t = \tanh(W_1^C \cdot x_t + W_h^C \cdot h_{t-1} + b_C) \quad (6)$$

จากสมการ (3) (4) (5) และ (6) จะพบว่ามีการเติมค่า  $b_i$ ,  $b_f$ ,  $b_o$  และ  $b_c$  ในส่วนของค่า Bias ของค่าน้ำหนัก  $W_1^i$ ,  $W_1^f$ ,  $W_1^o$  และ  $W_1^c$  และ ค่าส่วนของน้ำหนักจาก Hidden Layer ที่ได้ในแต่ละตัวแปร  $W_h^i$ ,  $W_h^f$ ,  $W_h^o$  และ  $W_h^c$  โดยนำค่าน้ำหนักที่ได้มาคำนวณรวมกับผลลัพธ์ของ Hidden Layer จากช่วงเวลาก่อนหน้า  $h_{t-1}$

เมื่อนำสมการข้างต้นมาประกอบเป็นโมเดลจะได้สมการดังนี้ (Liu et al., 2017)

ผลลัพธ์จากการพยากรณ์

$$C_t = f_t \cdot C_{t-1} + i_t \cdot \tilde{c}_t \quad (7)$$

ผลลัพธ์ของ Hidden Layer

$$h_t = o_t \cdot \tanh(C_t) \quad (8)$$

Sigmoid Activation Functions

$$\sigma(x) = \frac{1}{1+e^{-x}} \quad (9)$$

Tanh Activation Functions

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (10)$$

## 2.2. งานวิจัยที่เกี่ยวข้อง

งานวิจัยที่เกี่ยวข้องกับการพยากรณ์ระยะเวลาเดินทางโดยรถยนต์พบว่ามีนักวิจัยหลายท่านพยายามศึกษาเพื่อหาวิธีการพยากรณ์ที่ถูกต้องมากที่สุด โดย Mengting Bai และคณะได้ทำการรวบรวมวิธีการที่นักวิจัยเลือกใช้ (Bai et al., 2018) โดยสามารถแยกประเภทตามลักษณะของโมเดลได้ดังแสดงในภาพที่ 3 ในยุคแรกการพยากรณ์จะใช้วิธีการ Model-Based เช่น วิธีการ Queue Theory จะเป็นการนำค่าทางสถิติมาทำการสร้างโมเดลจำลองเพื่อจำลองสภาพจราจรที่มีรถยนต์อยู่ภายในแถวที่มีความยาวเท่าไร เพื่อนำความยาวที่ได้มาคำนวณหาระยะเวลาเดินทาง หรือวิธีการ Cell Transmission Model ที่เป็นการพิจารณาว่าระยะเวลาช่วงหนึ่งรถยนต์ผ่านทั้งหมดที่ Cell เพื่อจำลองสภาพการจราจร และหาสมการทางสถิติที่ใกล้เคียงกับความเป็นจริงมากที่สุด

งานวิจัยในยุคถัดมาจะเป็นงานวิจัยที่เน้นวิธีการ Data-Driven ซึ่งเป็นการนำข้อมูลในอดีตมาสร้างการเรียนรู้ให้กับคอมพิวเตอร์ด้วยวิธีต่าง ๆ เพื่อให้คอมพิวเตอร์เกิดการเรียนรู้และให้ผลลัพธ์

ใกล้เคียงกับข้อมูลในอดีตที่สอนมากที่สุด และสามารถนำผลการเรียนรู้ไปพยากรณ์ผลลัพธ์ที่จะเกิดขึ้นในอนาคต

เนื่องด้วยลักษณะของข้อมูล GPS trace มีลักษณะเป็น Time Series จึงมีนักวิจัยหลายท่านที่เลือกนำวิธีการ Recurrent Neural Network (RNN) และ Long Short Term Memory Neural Network (LSTM) มาทำการวิจัยเพราะเป็นวิธีการที่เหมาะสมกับข้อมูลประเภทดังกล่าว

สำหรับวิธีการ LSTM จะมีโครงสร้างภายในที่ต่างจากวิธีการ RNN เพราะวิธีการ LSTM ออกแบบเพื่อแก้ไขปัญหา Gradient Vanishing ที่เกิดขึ้นกับวิธีการ RNN ถ้าเกิดการวนซ้ำรอบจำนวนมากจะมีการปรับเปลี่ยนค่าน้ำหนักที่น้อยจนแทบไม่เปลี่ยนแปลง

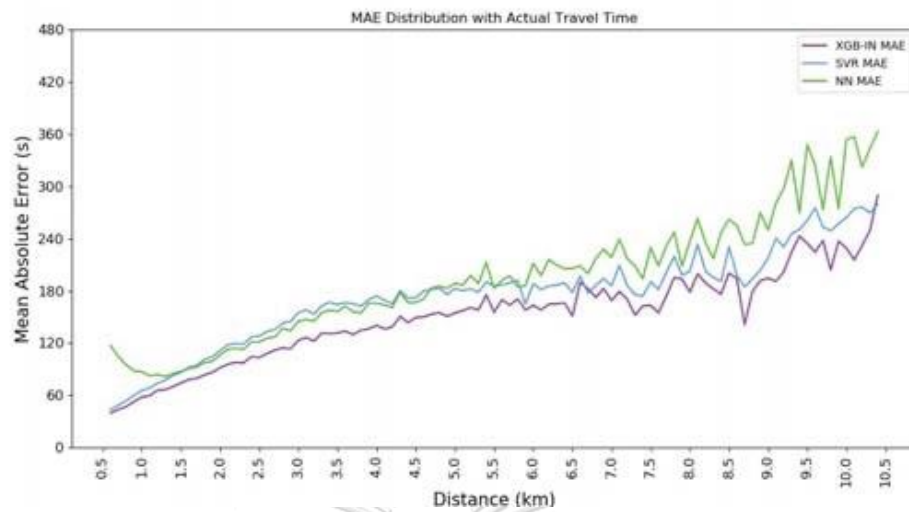
จากการศึกษาสามารถแบ่งกลุ่มงานวิจัยตามลักษณะแหล่งข้อมูล GPS ได้ 4 ลักษณะคือ

### 2.2.1. ข้อมูลที่ได้จากบันทึกสรุปเวลาเดินทาง

งานวิจัยที่ใช้ข้อมูลลักษณะดังกล่าวมีจำนวนมากเช่นงานวิจัยของ Kusal D. Kankanamge และคณะได้ทำการศึกษาศึกษาการพยากรณ์ระยะเวลาเดินทางด้วยวิธี XGBoost Regression ในหัวข้อ Taxi Trip Travel Time Prediction with Isolated XGBoost Regression (Kankanamge et al., 2019) งานวิจัยนี้นำข้อมูลการเดินทางของรถแท็กซี่ที่ได้จากหน่วยงาน New York City Taxi and Limousine Commission ข้อมูลดังกล่าวประกอบด้วยจุดรับและส่งผู้โดยสาร, เวลา ณ ขณะที่ผู้โดยสารขึ้น และลงรถแท็กซี่, รหัสคนขับรถแท็กซี่, ระยะเวลาเดินทาง, ระยะทาง, ค่าโดยสาร, ภาษี และอื่น ๆ ที่เกี่ยวข้องกับข้อมูลการเดินทางด้วยรถแท็กซี่

ข้อมูลดังกล่าวไม่สามารถระบุได้ว่าเดินทางผ่านถนนเส้นใดบ้าง ผู้วิจัยจึงตั้งสมมุติฐานว่าคนขับรถแท็กซี่จะขับรถไปตามเส้นทางที่สั้นที่สุดโดยเครื่องมือ Open Source Routing Machine (OSRM) เพื่อนำเอาพารามิเตอร์เกี่ยวกับเส้นทาง เช่น ระยะทาง จำนวนทางแยก ฯลฯ มาใช้ประกอบการสร้างโมเดล

จากนั้นนักวิจัยทำการสร้างโมเดลโดยวิธีการ Neural Network, วิธีการ Support Vector Regression, วิธีการใช้ข้อมูลจากผู้ให้บริการแผนที่ออนไลน์ Bing ด้วยเครื่องมือ Distance Matrix API และวิธีการ XGB-IN ซึ่งเป็นการสร้างโมเดลด้วย Decision Tree หลายต้นมาต่อกัน โดยที่แต่ละต้นไม้จะเรียนรู้จากข้อผิดพลาดของต้นไม้ก่อนหน้าและวนทำซ้ำจนกระทั่งถึงความลึกของต้นไม้ที่กำหนด หรือจนกว่าจะไม่มีค่าความผิดพลาดให้เรียนรู้ ซึ่งผลการทดลองที่ได้ดังภาพที่ 12 และ ภาพที่



ภาพที่ 12 ผลค่าความผิดพลาดของแต่ละโมเดล

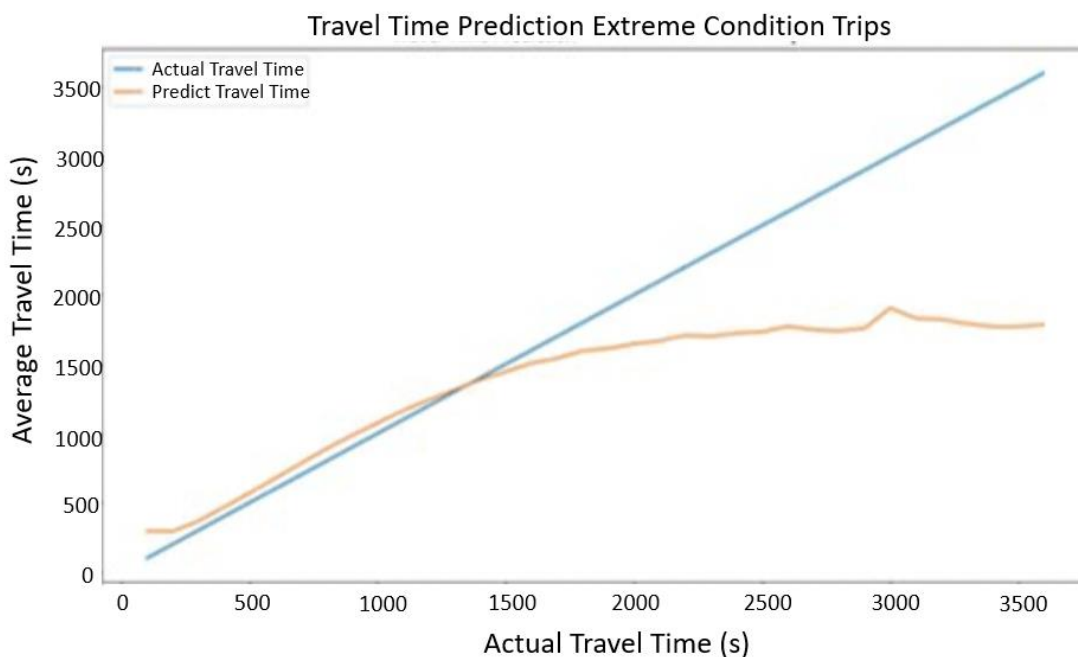
Model	MAE(s)	RMSE(s)	MAPE	Spearman Correlation
XGB-IN	94.764	130.627	17.021	0.932283
SVR	112.802	155.246	20.116	0.913727
NN	120.585	164.052	24.761	0.912958

Model	MAE(s)	RMSE(s)	MAPE	Spearman Correlation
XGB-IN	90.369	181.194	17.241	0.928567
TEMP-rel	124.609	192.794	21.954	0.833775
Online Map Services	203.933	24.543	24.761	0.811948

Model	MAE(s)	RMSE(s)	MAPE	Spearman Correlation
XGB-Extreme	190.439	287.292	24.082	0.892672

ภาพที่ 13 ผลค่าความผิดพลาดของแต่ละโมเดล

จากภาพที่ 12 และภาพที่ 13 พบว่าโมเดล Neural Network ได้ผลการทดลองที่มีค่าผิดพลาดสูงสุด และโมเดล XGB-IN ให้ค่าที่ถูกต้องที่สุด ถึงแม้ว่าวิธี XGB-Extreme จะได้ค่าผิดพลาดเท่ากับวิธีการอื่น แต่วิธีการดังกล่าวเหมาะสมกับพยากรณ์ระยะเวลาเดินทางสั้น แต่นำมาใช้กับระยะเวลาเดินทางที่นานก็จะให้ค่าผิดพลาดสูงดังภาพที่ 14



**ภาพที่ 14** เปรียบเทียบค่าพยากรณ์และค่าจริงตามระยะทาง

ถึงแม้ว่านักวิจัยกลุ่มดังกล่าวได้ทำการทดลองจนสรุปว่าโมเดล XGBoost Regression ได้ให้ความถูกต้องมากกว่าวิธีการ Neural Network แต่ตรงข้ามกับงานวิจัยของ Arthur Cruz de Araujo และคณะที่เลือกวิธีการ Deep Neural Network มาทำการพยากรณ์ระยะเวลาเดินทางในหัวข้อ Deep Neural Networks for Predicting Vehicle Travel Times (Araujo & Etemad, 2019)

นักวิจัยนำข้อมูล 2016 Yellow Taxi Trip Data ของหน่วยงาน NYC Open Data Initiative โดยข้อมูลประกอบด้วยจุดที่รับและส่งผู้โดยสาร, เวลาขณะที่รับและส่งผู้โดยสาร, ระยะทางที่เดินทางในแต่ละรอบการเดินทาง และข้อมูลอื่น ๆ ที่เกี่ยวข้อง ซึ่งเป็นข้อมูลที่มีลักษณะคล้ายกับข้อมูลงานวิจัยของ Kusal D. Kankanamge และคณะ นักวิจัยได้ทำการนำข้อมูลแท็กซี่เตรียมด้วยวิธีการที่แตกต่างกัน 5 วิธีจนได้ข้อมูลสำหรับการวิจัยจำนวน 5 ชุด โดย

**FS0:** คือชุดข้อมูลที่ประกอบด้วยจุดที่รับและส่งของผู้โดยสารเท่านั้น

**FS1:** คือชุดข้อมูล FS0 ที่เพิ่มข้อมูลระยะทางที่เดินทางที่ประมาณด้วยวิธี Naive Linear Regressor คือการตั้งสมมติฐานว่าสัดส่วนระหว่างระยะทางที่เดินทางจริง กับระยะทางที่กีดทางภูมิศาสตร์มีความสัมพันธ์แบบเส้นตรง โดยนำค่าอัตราส่วนมัธยฐาน (Median) ของสัดส่วนดังกล่าวมาใช้ในการปรับค่าข้อมูลระยะทางที่เดินทาง ซึ่งเมื่อวัดค่าความผิดพลาดด้วยวิธี Mean Absolute Percentage Error ได้ค่าเท่ากับ 15.34 %

**FS2:** คือชุดข้อมูล FS0 ที่เพิ่มข้อมูลระยะทางที่เดินทางที่ประมาณด้วยวิธี Map-based Locally-Linear Regressor คือการแบ่งแผนที่ในเมือง New York ออกเป็น 300 x 300 ตาราง และในแต่ละตารางและนำวิธีการ Linear Regression มาสอนการเดินทางในแต่ละตารางและนำอัตราส่วนของแต่ละตารางมาคำนวณพบว่าค่า Mean Absolute Percentage Error เท่ากับ 14.24 %

**FS3:** คือชุดข้อมูล FS0 ที่มีระยะทางที่เดินทางจริงที่เก็บได้จากชุดข้อมูลตั้งต้น

**FS4:** คือชุดข้อมูล FS2 ที่มีการเพิ่มระยะทางทางภูมิศาสตร์และระยะเวลา รวมไปถึงคุณสมบัติอื่นเพิ่มเติม

เมื่อได้ชุดข้อมูลทั้ง 5 ชุดนักวิจัยได้ทำการศึกษาดูด้วยวิธีการหลายวิธี และนำผลลัพธ์ที่ได้มาเปรียบเทียบ โดยทำการวิจัยด้วยวิธีการ Support Vector Machine (SVM), วิธีการ Random Forest (RF), วิธีการ XGBoost และวิธีการ Multilayer Perceptron (MLP) ซึ่งได้ผลการพยากรณ์ตารางที่ 1

**ตารางที่ 1** ผลการทดลองเปรียบเทียบที่ดีที่สุดของแต่ละโมเดลในแต่ละชุดข้อมูลที่แตกต่างกัน

Model	FS0			FS 1			FS2		
	EV	RMSLE	MAPE	EV	RMSLE	MAPE	EV	RMSLE	MAPE
SVM	0.6952	0.4060	33.79	0.7009	0.3826	30.95	0.7035	0.3817	30.85
RF	0.7011	0.4002	36.54	0.7011	0.3933	35.27	0.7013	0.3926	35.17
XGBoost	0.6784	0.4111	37.46	0.6906	0.3987	35.48	0.6891	0.3982	35.55
MLP	0.7163	0.3791	33.42	0.7034	0.3960	33.13	0.7138	0.3797	30.49

Model	FS3			FS4		
	EV	RMSLE	MAPE	EV	RMSLE	MAPE
SVM	0.7339	0.3358	26.99	0.7905	<b>0.3385</b>	<b>26.49</b>
RF	0.7356	0.3456	30.06	0.7817	0.3622	31.99
XGBoost	0.7222	0.3495	29.94	<b>0.7938</b>	0.3506	29.50
MLP	0.7403	0.3600	28.90	0.7885	0.3502	29.92

ตารางที่ 1 นำเสนอประสิทธิภาพของโมเดลที่สอนด้วยข้อมูลทั้ง 5 ชุด โดยแต่ละโมเดลมีความแตกต่างกันทั้งพารามิเตอร์ และโครงสร้างที่ใช้ในการทดลองกับข้อมูลแต่ละชุด ซึ่งผลที่ได้เป็นผลการทดลองที่ดีที่สุดของแต่ละโมเดลของข้อมูลแต่ละชุด การวัดผลวัดด้วยวิธีการที่แตกต่างกันทั้ง 3 วิธี ได้แก่วิธี Explained Variance (EV) , วิธี Root Mean Squared Logarithmic Error (RMSLE) คือวิธีการวัดผลแบบ Root Mean Square Error (RMSE) ที่มีการเพิ่มฟังก์ชัน Logarithmic ในสมการเพื่อให้ค่าที่ได้มีความเรียบ และเหมาะสมมากขึ้นสำหรับการเป็นตัววัดผล และวิธีการ Mean Absolute Percentage Error (MAPE)

ผลลัพธ์ที่ได้สามารถสรุปว่าชุดข้อมูล FS4 เมื่อเปรียบเทียบกับ FS0, FS1, FS2 และ FS3 ให้ผลลัพธ์ที่ดีที่สุดเมื่อนำมาพยากรณ์ เมื่อได้ผลดังนั้นก็จึงทำการนำชุดข้อมูล FS4 มาใช้ในการปรับแต่ง

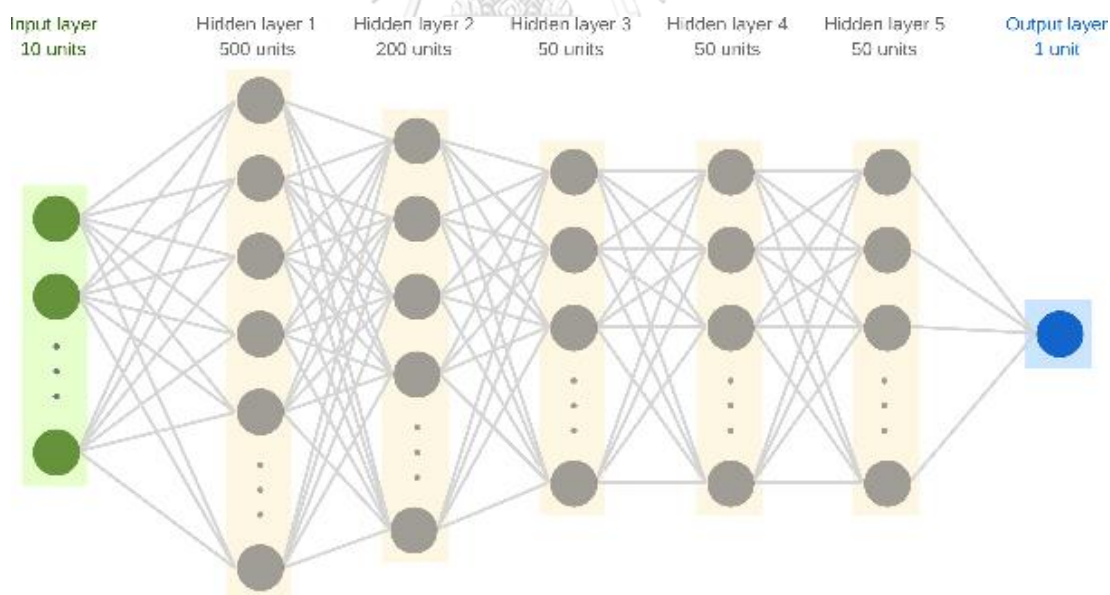


โมเดล Deep Neural Network ต่อไป โดยเลือกทำการสร้างโมเดล Deep Neural Network ที่มีลักษณะโครงสร้างที่แตกต่างกัน 4 แบบเพื่อเปรียบเทียบผลที่ถูกต้องมากที่สุด ดังแสดงในตารางที่ 2

ตารางที่ 2 ผลการทดลองด้วยโครงสร้างที่แตกต่างกันของแต่ละโมเดล

Model	Architecture	Epochs	FS4		
			EV	RMSLE	MAPE
DenseNet1	[500, 200, 50]	15	0.8102	0.3229	25.92
DenseNet2	[500, 200, 3{50}]	15	<b>0.8116</b>	<b>0.3198</b>	<b>25.70</b>
DenseNet3	[50, 100, 200, 500, 200, 100, 50]	20	0.8081	0.3346	28.62
DenseNet4	[500, 200, 2{100}, 3{50}, 10]	15	0.8072	0.3260	26.57
		30	0.8090	0.3250	26.71

จากตารางที่ 2 สรุปได้ว่าโครงสร้างตามโมเดล DenseNet2 ให้ผลลัพธ์ที่ดีที่สุดเมื่อเทียบกับโครงสร้างอื่น ซึ่งโครงสร้างของโมเดลเป็นดังภาพที่ 15 และเมื่อนำผลการทดลองไปเปรียบเทียบกับผลการทดลองก่อนหน้านี้ทำให้สรุปได้ว่าโมเดล DenseNet2 ให้ผลลัพธ์ที่ดีที่สุดเมื่อเทียบกับวิธีการ SVM, RF, XGBoost และ MLP



ภาพที่ 15 แผนผังโครงสร้างโมเดล DenseNet 2

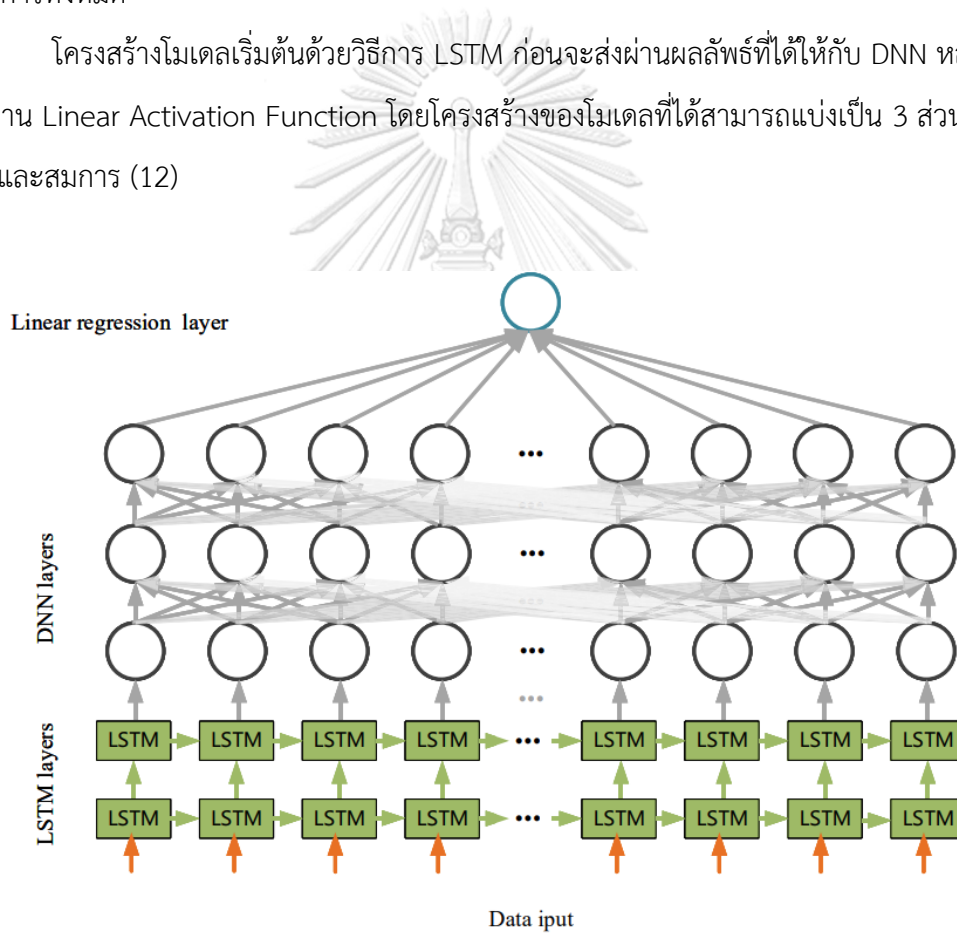
### 2.2.2. ข้อมูลที่ได้จากบันทึกเวลาที่รถแต่ละคันผ่านตำแหน่งที่กำหนด

จากการศึกษาผลงานวิจัยของ Liu และคณะหัวข้อ Short-term travel time prediction by deep learning: A comparison of different LSTM-DNN models (Liu et al., 2017) มีการ

เปรียบเทียบความถูกต้องของโมเดล LSTM ที่ต่อกับโมเดล Deep Neural Network และใช้ Linear Activation Function โดยเปรียบเทียบกับวิธีการดั้งเดิมอื่น ซึ่งวัดผลในแต่ละรอบโดยวิธี Mean Square Error

งานวิจัยใช้ข้อมูลจาก Canltrans Performance Measure System ซึ่งเป็นระบบที่เก็บข้อมูลการจราจรในรัฐแคลิฟอร์เนีย ประเทศสหรัฐอเมริกา นักวิจัยเลือกบางช่วงของถนน Corridor13 ซึ่งใช้ข้อมูล 90 วัน ตั้งแต่วันที่ 18 กันยายน ค.ศ. 2016 จนกระทั่ง 17 ธันวาคม ค.ศ. 2016 โดยเก็บช่วงเวลาที่ใช้ในการเดินทางทุก 5 นาที จะได้ว่าได้ข้อมูลทั้งหมด 288 รายการต่อวัน หรือ 25,920 รายการทั้งหมด

โครงสร้างโมเดลเริ่มต้นด้วยวิธีการ LSTM ก่อนจะส่งผ่านผลลัพธ์ที่ได้ให้กับ DNN หลังจากนั้นจึงผ่าน Linear Activation Function โดยโครงสร้างของโมเดลที่ได้สามารถแบ่งเป็น 3 ส่วนดังภาพที่ 16 และสมการ (12)



ภาพที่ 16 โครงสร้าง LSTM-DNN และใช้ Linear Activation Function

Mean Square Error

$$MSE = \frac{1}{n} \sum_{i=1}^n (x_i - \hat{x}_i)^2 \quad (12)$$

การวิจัยได้ทำการแบ่งข้อมูลที่ได้เป็น ข้อมูลสำหรับสอน 70% ข้อมูลสำหรับตรวจสอบ 10% และข้อมูลสำหรับทดสอบ 20% หลังจากเสร็จสิ้นจะทำการวัดผลในแต่ละโมเดลโดย Mean Absolute Percentage Error ดังสมการ (13) และ Root Mean Square Error ดังสมการ (14)

Mean Absolute Percentage Error

$$MAPE = \frac{1}{n} \sum_{i=1}^n \frac{|x_i - \hat{x}_i|}{x_i} \quad (13)$$

Root Mean Square Error

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (x_i - \hat{x}_i)^2} \quad (14)$$

นักวิจัยได้พยายามปรับแต่ง Long Short Term Memory Neural Network โดยกำหนดจำนวนชั้นที่แตกต่างกันเพื่อวัดผลว่าจำนวนชั้นของ Network เท่าไหร่ถึงจะได้ผลลัพธ์ถูกต้องมากที่สุด โดยปรับแต่งค่า Learning Rate เท่ากับ 0.001 จำนวนของ Epoch เท่ากับ 80 และขนาดของ Batch เท่ากับ 180 โดยได้ผลการทดสอบดังตารางที่ 3

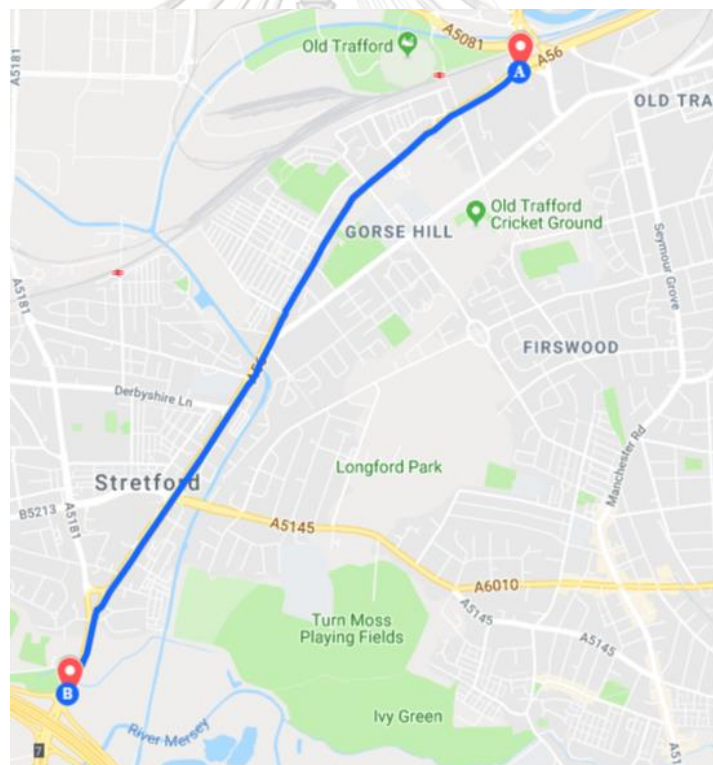
ตารางที่ 3 สรุปผลการทดลองในการปรับแต่งช่วงเวลาที่นำเข้าและช่วงเวลาที่พยากรณ์ล่วงหน้า

MAPE(%), Sliding window = 60 mins					
Prediction horizon(mins)	5	10	20	30	60
Linear Regression	1.010187	1.781746	3.157644	4.513085	8.637864
Ridge Regression	1.010173	1.781706	3.157573	4.513007	8.637823
Lasso Regression	1.048166	1.793461	3.165147	4.504833	8.671028
DNN	1.493290	1.861104	4.151781	4.876978	7.820807
ARIMA	1.208255	2.484198	5.562116	9.407818	23.736445
LSTM-DNN	<b>0.967767</b>	<b>1.621363</b>	<b>2.616827</b>	<b>4.057407</b>	<b>7.264099</b>
MAPE(%), Sliding window = 30 mins					
Prediction horizon(mins)	5	10	20	30	60
Linear Regression	1.006194	1.773854	3.122103	4.457734	8.528927
Ridge Regression	1.006172	<b>1.773804</b>	3.122005	4.457624	8.528827
Lasso Regression	1.028043	1.769604	3.107643	4.407132	8.498129
DNN	1.065725	1.778982	3.084992	4.358483	7.877626
ARIMA	1.130722	2.236957	4.790908	7.990758	20.229357
LSTM-DNN	<b>0.961468</b>	1.997938	<b>2.920018</b>	<b>3.655459</b>	<b>6.600965</b>
RMSE, Sliding window = 60 mins					
Prediction horizon(mins)	5	10	20	30	60
Linear Regression	0.367289	0.677546	1.298362	1.954813	3.853114
Ridge Regression	0.367294	0.677550	1.298375	1.954831	3.853134
Lasso Regression	0.404177	0.711253	1.340855	1.996254	3.887231
DNN	0.459849	0.731222	1.427575	2.161596	4.173776
ARIMA	0.402316	0.792926	1.687567	2.735245	4.004502
LSTM-DNN	<b>0.362902</b>	<b>0.655964</b>	<b>1.211798</b>	<b>1.771794</b>	<b>3.064785</b>
RMSE, Sliding window = 30 mins					
Prediction horizon(mins)	5	10	20	30	60
Linear Regression	0.366843	<b>0.677064</b>	1.296032	1.952448	3.858874
Ridge Regression	0.366848	0.677065	1.296040	1.952459	3.858887
Lasso Regression	0.397264	0.703925	1.329985	1.984265	3.884927
DNN	0.377937	0.687101	1.319625	1.968736	4.011477
ARIMA	0.388855	0.750618	1.554574	2.497897	5.806814
LSTM-DNN	<b>0.365456</b>	0.688558	<b>1.234101</b>	<b>1.822048</b>	<b>3.364138</b>

จากตารางที่ 3 แสดงให้เห็นว่านักวิจัยมีการพยายามเปรียบเทียบโมเดล LSTM-DNN กับโมเดลประเภทอื่น ซึ่งพบว่าโมเดล LSTM-DNN โดยรวมให้ผลลัพธ์ที่มีความถูกต้องมากที่สุด

### 2.2.3. ข้อมูลที่ได้จากเซ็นเซอร์วัดความเร็ว

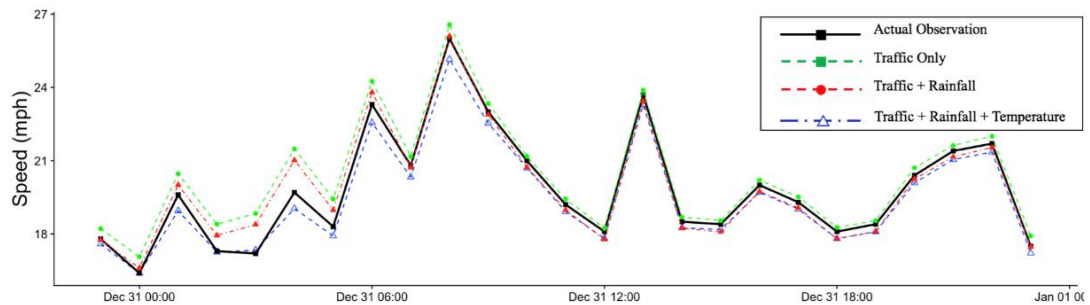
อุปกรณ์เซ็นเซอร์วัดความเร็วมีหลายรูปแบบหนึ่งในรูปแบบที่นิยมใช้กันคือการติดตั้งอุปกรณ์กล้องตรวจจับความเร็ว เช่นงานวิจัยของ Aniekan Essien และคณะได้ทำการศึกษาเรื่อง Improving Urban Traffic Speed Prediction Using Data Source Fusion and Deep Learning (Essien, Petrounias, Sampaio, & Sampaio, 2019) ใช้ข้อมูลจาก Transport for Greater Manchester (TfGM) ระหว่างช่วงเวลา 1 มกราคม ค.ศ. 2017 จนถึง 31 ธันวาคม ค.ศ. 2017 เป็นข้อมูลลักษณะการจราจร เช่น ความเร็วเฉลี่ย, การไหลของการจราจร (flow), ความหนาแน่นของการจราจร (density) ที่เก็บด้วยอุปกรณ์ Inductive Loop Detectors ที่ติดตั้งจำนวน 10 ตัวห่างกันประมาณ 0.3 ไมล์ (ประมาณ 482 เมตร) บนถนน Chester Road - A56 ดังภาพที่ 17 และใช้ข้อมูลสภาพอากาศร่วมด้วย



ภาพที่ 17 ถนนที่เก็บข้อมูลของงานวิจัย

นักวิจัยได้นำข้อมูลที่ได้เข้าโมเดล LSTM 5 ชั้น เพื่อพยากรณ์เป็นรายชั่วโมงโดยสามารถพยากรณ์ล่วงหน้าจำนวน 24 ชั่วโมง การวิจัยได้ทำการเปรียบเทียบกับโมเดล ARIMA กับ LSTM ด้วยข้อมูลหลายชุด โมเดลแรกสร้างด้วย ARIMA, โมเดลที่สองสร้างด้วย LSTM และใช้ข้อมูลการจราจร

อย่างเดียว, โมเดลที่ 3 สร้างด้วย LSTM และใช้ข้อมูลการจราจร และข้อมูลปริมาณน้ำฝน และโมเดลสุดท้ายสร้างด้วย LSTM และใช้ข้อมูลการจราจร, ข้อมูลปริมาณน้ำฝน และข้อมูลอุณหภูมิ ผลลัพธ์ที่ได้เป็นไปตามภาพที่ 18 และตารางที่ 4



ภาพที่ 18 ผลลัพธ์จากการพยากรณ์

ตารางที่ 4 สรุปผลการทดลองค่าความผิดพลาดแต่ละโมเดล

Model ID	MAE	RMSE	MSE	MAE (mph)
ARIMA	2.4640	3.1419	0.9871	~11.1
LSTM rainfall temperature	0.049	0.0892	0.008	~0.2
LSTM rainfall	0.071	0.1095	0.012	~0.3
LSTM traffic	0.1338	0.3658	0.039	~0.6

ผลลัพธ์ที่ได้จากงานวิจัยจากตารางที่ 4 พบว่าโมเดล LSTM ที่ใช้ข้อมูลการจราจร, ข้อมูลปริมาณน้ำฝน และข้อมูลอุณหภูมิให้ผลลัพธ์ที่ดีที่สุดเมื่อเทียบกับโมเดลอื่น และจากภาพที่ 18 แสดงให้เห็นว่าโมเดล LSTM สามารถใช้พยากรณ์ระยะเวลาเดินทางล่วงหน้าได้ทั้ง Short-term prediction และ Long-term prediction

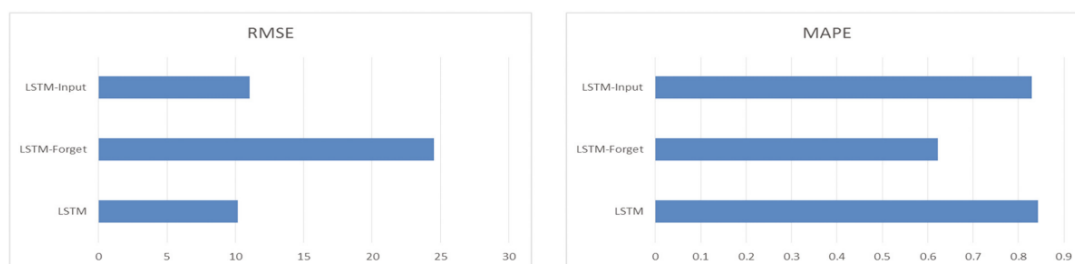
#### 2.2.4. ข้อมูลที่เก็บจากอุปกรณ์ GPS ตลอดเส้นทางการเดินทาง

มีนักวิจัยบางกลุ่มที่ใช้ข้อมูลที่มีลักษณะ GPS trace มาใช้ในการศึกษา โดยพบว่ามีนักวิจัย Li และ Jiang ทำการวิจัยหัวข้อ Predicting the Travel Time in Using Recurrent Neural Networks: A Case Study of Fuzhou (Li & Jiang, 2017) โดยข้อมูลมาจากจีพีเอสที่ติดตั้งภายในรถแท็กซี่ในเมืองฝูโจวที่เก็บข้อมูลทุก 30 วินาที

ข้อมูลที่ใช้ในการวิจัยเป็นข้อมูลระหว่างวันที่ 1 มิถุนายน ค.ศ. 2015 จนถึง 31 มิถุนายน ค.ศ. 2015 ในช่วงเวลา 7.30 จนถึง 21:30 ครอบคลุมพื้นที่ประมาณ 1,400 ตารางกิโลเมตร นักวิจัยได้แบ่งข้อมูลออกเป็นช่วงเวลาช่วงละ 10 นาที ซึ่งจะมีบางช่วงของข้อมูลที่ขาดหายเนื่องจากในช่วงเวลาดังกล่าวไม่มีรถแท็กซี่ขับผ่าน นักวิจัยได้ใช้วิธีนำข้อมูลเฉลี่ยในอดีตมาเติมในช่วงเวลาที่หายไป งานวิจัยนี้จะทำการวิจัยเพื่อวัดผลประสิทธิภาพของแต่ละ Gate ของโมเดล LSTM เพื่อใช้ในการพยากรณ์ระยะเวลาเดินทาง โดยการวิจัยจะทำการปรับเปลี่ยนรูปแบบโครงสร้างของโมเดล LSTM โดยเปรียบเทียบโมเดล 3 โครงสร้าง ในแต่ละโครงสร้างแตกต่างกันที่ โครงสร้างแรกเป็นการนำ Input Gate ออกจากโมเดล, โครงสร้างที่สองเป็นการนำ Forget Gate ออกจากโมเดล และโครงสร้างสุดท้ายเป็นโมเดล LSTM ที่ทุก Gate ไม่มีการนำออกจากโมเดล ในงานวิจัยได้เลือกใช้ Soft Sign Activation Function เพราะคำนวณได้เร็วกว่า Tanh Activation Function

นักวิจัยใช้ข้อมูลเวลาของวัน, เวลาเดินทางของช่วงถนนที่สนใจ, เวลาของช่วงถนนก่อนหน้านี้ และเวลาของช่วงถนนถัดไปเป็นคุณสมบัติที่ใช้ในการพยากรณ์เวลาที่ใช้ในการเดินทางถัดไป โดยนักวิจัยได้เลือกช่วงถนนที่สนใจจำนวน 20 ช่วงถนนมาใช้ในการทดลอง

การวัดผลโมเดลด้วยฟังก์ชัน Mean Absolute Percentage Error ดังสมการ (13) และฟังก์ชัน Root Mean Square Error ดังสมการ (14) ผลการทดลองเปรียบเทียบโมเดลโดยโครงสร้างโมเดล LSTM ปกติเปรียบเทียบกับโครงสร้างที่นำ Input Gate ออกจากโมเดล LSTM และโครงสร้างที่นำ Forget Gate ออกจากโมเดล LSTM



ภาพที่ 19 เปรียบเทียบผลลัพธ์จากการทดสอบโมเดลทั้ง 3 โครงสร้าง

จากภาพที่ 19 แสดงให้เห็นว่าผลการทดสอบโดยเปรียบเทียบโครงสร้างแรกเป็นการนำ Input Gate ออกจากโมเดล LSTM, โครงสร้างที่สองเป็นการนำ Forget Gate ออกจากโมเดล LSTM และโครงสร้างสุดท้ายเป็นโมเดล LSTM ที่ทุก Gate ไม่มีการนำออกจากโมเดล โดยผลที่ได้พบว่าถ้าหากโมเดล LSTM ถูกถอด Forget Gate ออกจะทำให้ผลการพยากรณ์มีข้อผิดพลาดสูงขึ้นกว่าโมเดลที่ครบสมบูรณ์ ขณะที่โมเดล LSTM ที่ถูกถอด Input Gate ออกส่งผลต่อความผิดพลาดเล็กน้อย

จากค่าผิดพลาดที่แสดงดังภาพที่ 19 ทำให้นักวิจัยสรุปได้ว่าโมเดล LSTM สามารถนำมาประยุกต์ใช้กับการพยากรณ์เวลาที่ใช้ในการเดินทางได้จริงสำหรับข้อมูลในเมืองฝูโจว และ Forget Gate มีผลต่อค่าความผิดพลาดของการพยากรณ์เป็นอย่างมาก

ไม่เพียงเท่านั้นนักวิจัย Duan และคณะสนใจที่จะศึกษาการพยากรณ์ระยะเวลาในการเดินทางด้วยโมเดล LSTM ในหัวข้อ Travel Time Prediction with LSTM Neural Network (Duan et al., 2016) นักวิจัยได้ข้อมูลจากกรมทางหลวงประเทศสหราชอาณาจักร (Highways England) ชุดข้อมูลประกอบด้วยระยะเวลาเดินทางจะเป็นการคำนวณจากกล้องที่ใช้ส่องทะเปียนรถยนต์ และข้อมูล GPS ภายในรถยนต์ โดยข้อมูลระยะเวลาที่ใช้ในการวิจัยจะเป็นข้อมูลระยะเวลาเดินทางเฉลี่ยทุก 15 นาที เป็นข้อมูลปี ค.ศ. 2015 จำนวนทั้งสิ้น 66 ช่วงถนนหลวง M25

นักวิจัยได้ทำการนำข้อมูลเข้าโมเดล LSTM โดยใช้ข้อมูลสอนเท่ากับ 80% ข้อมูลตรวจสอบ 10% และข้อมูลทดสอบ 10% และวัดผลของโมเดลที่ได้ด้วยฟังก์ชัน Mean Absolute Error ดังสมการ (15) ฟังก์ชัน Root Mean Square Error ดังสมการ (14) และ ฟังก์ชัน Mean Relative Error ดังสมการ (16)

Mean Absolute Error

$$MAE = \frac{1}{n} \sum_{i=1}^n |x_i - \hat{x}_i| \quad (15)$$

Mean Relative Error

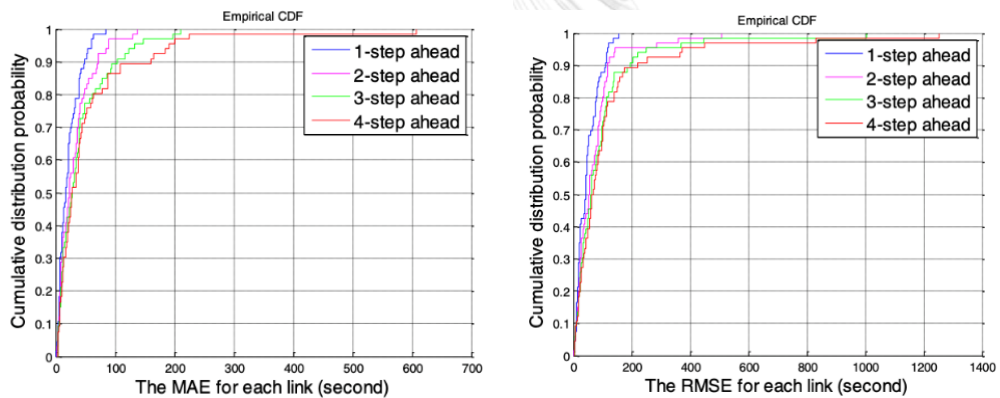
$$MRE = \frac{1}{n} \sum_{i=1}^n \frac{|x_i - \hat{x}_i|}{x_i} \quad (16)$$

การทดสอบจะเป็นการเปรียบเทียบระหว่างการนำข้อมูลค่า Median กับ Percentile ที่ 95 มาใช้ในการสอนโมเดลและพยากรณ์ การพยากรณ์จะแบ่งเป็น 1 ช่วงเวลาถัดไปจนถึง 4 ช่วงเวลาถัดไป ผลลัพธ์ที่ได้ดังตารางที่ 5 และจากภาพที่ 20 จะพบว่าการใช้ค่า Median ได้ผลลัพธ์ที่ถูกต้องมากกว่าค่า Percentile ที่ 95



ตารางที่ 5 สรุปผลการทดลองเปรียบเทียบระหว่างการนำข้อมูลค่า Median กับ Percentile ที่ 95 และการพยากรณ์จะแบ่งเป็น 1 ช่วงเวลาถัดไปจนถึง 4 ช่วงเวลาถัดไป

Step ahead	Median			95%th		
	MAE	RMSE	MRE	MAE	RMSE	MRE
1	16.9s	43.0s	0.070	60.4s	118s	0.173
2	23.1s	54.9s	0.092	89.1s	171s	0.298
3	27.8s	61.1s	0.113	136s	273s	0.446
4	27.8s	66.2s	0.108	193s	388s	0.770



ภาพที่ 20 การแจกแจงค่า MAE และ RMSE ของแต่ละช่วงเวลาล่วงหน้าของการพยากรณ์

### 2.2.5. ตารางสรุปงานวิจัยที่เกี่ยวข้อง

สามารถสรุปงานวิจัยที่เกี่ยวข้องได้ดังตารางที่ 6

ตารางที่ 6 สรุปรายละเอียดงานวิจัยที่เกี่ยวข้อง

Topic	Researcher	Year	Data	Type of Data	Resolution	Model
Taxi Trip Travel Time Prediction with Isolated XGBoost Regression	Kusal D. Kankanamge, Yasiru R. Witharanage, Chanaka S. Withanage, Malsha Hansini, Damindu Lakmal, Uthayasanker Thayasivam	2019	มาจากหน่วยงาน New York City Taxi and Limousine Commission ข้อมูลดังกล่าวประกอบด้วยจุดรับและส่งผู้โดยสาร, เวลา ณ ขณะที่ใช้โดยสารขึ้นรถแท็กซี่ และเวลาที่ผู้โดยสารลงรถแท็กซี่, รหัสคนขับรถแท็กซี่, ระยะเวลาเดินทาง, ระยะทางที่เดินทาง, ค่าโดยสาร, ภาษี และอื่น ๆ	ข้อมูลที่ได้จากบันทึกสรุปเวลาเดินทาง	path-level	โมเดล isolated XGBoost regression
Deep Neural Networks for Predicting Vehicle Travel Times	Arthur Cruz de Araujo, Ali Etemad	2019	ข้อมูล 2016 Yellow Taxi Trip Data ของหน่วยงาน NYC Open Data Initiative โดยข้อมูลประกอบด้วยจุดที่รับและส่งผู้โดยสาร, เวลาขณะที่รับและส่งผู้โดยสาร, ระยะทางที่เดินทางในแต่ละรอบการเดินทาง และข้อมูลอื่น ๆ	ข้อมูลที่ได้จากบันทึกสรุปเวลาเดินทาง	OD-level	โมเดล Deep Neural Network

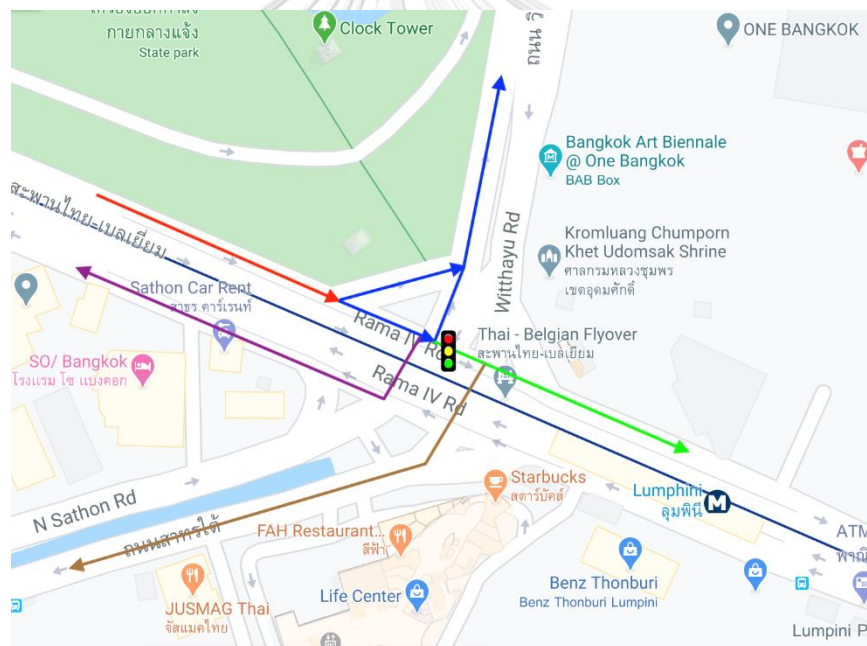
Short-term travel time prediction by deep learning: A comparison of different LSTM-DNN models	Yangdong , Yizhe , Xiaoguang , Linan Zhang	Liu Wang Yang	2017	ช่วงถนน Corridor13 ซึ่งใช้ข้อมูล 90 วัน โดยเก็บช่วงเวลาที่ใช้ในการเดินทางทุก 5 นาที จะได้ว่าได้ข้อมูลทั้งหมด 288 รายการต่อวัน หรือ 25,920 รายการทั้งหมด	ข้อมูลที่ได้จาก บันทึกเวลาที่รถ แต่ละคันผ่าน ตำแหน่งที่กำหนด	link-level	โมเดล Long Short Term Memory ร่วมกับ Deep Neural Network
Improving Urban Traffic Speed Prediction Using Data Source Fusion and Deep Learning	Aniekan Essien , Ilias Petrounias , Pedro Sampaio , Sandra Sampaio		2019	ข้อมูลจาก Transport for Greater Manchester (TfGM) ระหว่างช่วงเวลา 1 มกราคม ค.ศ. 2017 จนถึง 31 ธันวาคม ค.ศ. 2017 ซึ่งเป็นข้อมูลลักษณะของการจราจร เช่น ความเร็วเฉลี่ย, การไหลของการจราจร (flow), ความหนาแน่นของการจราจร (density) ที่เก็บด้วยอุปกรณ์ Inductive Loop Detectors บนถนน Chester Road - A56 และใช้ข้อมูลสภาพอากาศ	ข้อมูลที่ได้จาก เซ็นเซอร์วัดความเร็ว	link-level	โมเดล Long Short Term Memory

Predicting the Travel Time in Using Recurrent Neural Networks: A Case Study of Fuzhou	Luming , Xinhua Jiang	Li	2017	ข้อมูลมาจากจีพีเอสที่ติดตั้งภายในรถแท็กซี่ในเมืองฝูโจว โดยจีพีเอสจะเก็บข้อมูลทุก 30 วินาที ในช่วงเวลา 7:30 จนถึง 21:30 ครอบคลุมพื้นที่ประมาณ 1,400 ตารางกิโลเมตร นักวิจัยได้แบ่งข้อมูลออกเป็นช่วงเวลา ช่วงละ 10 นาที	ข้อมูลที่ได้จาก GPS trace	link-level	โมเดล Long Short Term Memory
Travel Time Prediction with LSTM Neural Network	Yanjie , Yisheng , Fei-Yue Wang	Duan Lv	2016	ข้อมูลมาจากการจราจรทางหลวงประเทศสหราชอาณาจักร (Highways England) ชุดข้อมูลระยะเวลาเดินทางจะเป็นการคำนวณจากกล้องที่ใช้ส่องทะเบียบรถยนต์ และข้อมูล GPS ภายในรถยนต์ จำนวนทั้งสิ้น 66 ช่วงถนนหลวง M25	ข้อมูลที่ได้จาก บันทึกเวลาที่รถแต่ละคันผ่านตำแหน่งที่กำหนด และข้อมูลที่ได้จาก GPS trace	link-level	โมเดล Long Short Term Memory

### 2.2.6. ข้อสรุปที่ได้จากการศึกษางานวิจัยที่เกี่ยวข้อง

จากการศึกษางานวิจัยก่อนหน้านี้ที่เป็นแบบ data-driven พบว่าไม่มีการพิจารณาแยกข้อมูลเส้นทางการเดินทางที่มีลักษณะการเดินทางที่แตกต่างกันออกจากกัน เช่น รถที่ตรงไป รถที่เลี้ยวซ้าย หรือเลี้ยวขวา ทั้งที่จริงแล้วเส้นทางการเดินทางที่แตกต่างกันย่อมส่งผลต่อระยะเวลาเดินทางที่แตกต่างกัน

ตัวอย่างเช่น บริเวณสี่แยกถนนวิสุทธิดงชัยกับถนนพระราม 4 ฝั่งสวนลุมพินี ประกอบด้วยทั้งช่องทางเดินรถเลี้ยวซ้ายผ่านตลอด, ช่องทางเดินรถเลี้ยวขวา, ช่องทางเดินรถกลับรถ และช่องทางเดินรถที่ตรงไป ดังภาพที่ 21 ซึ่งหากนำระยะเวลาเดินทางในทุกเส้นทางที่ผ่านถนนช่วงนี้มาพยากรณ์ระยะเวลาเดินทางสำหรับเส้นทางที่เลี้ยวซ้ายจะเกิดความคลาดเคลื่อนของผลลัพธ์ที่ได้จากการพยากรณ์



ภาพที่ 21 ตัวอย่างเส้นทางที่เป็นไปได้ของรถยนต์ที่ขับบนถนนเส้นเดียวกัน

## บทที่ 3

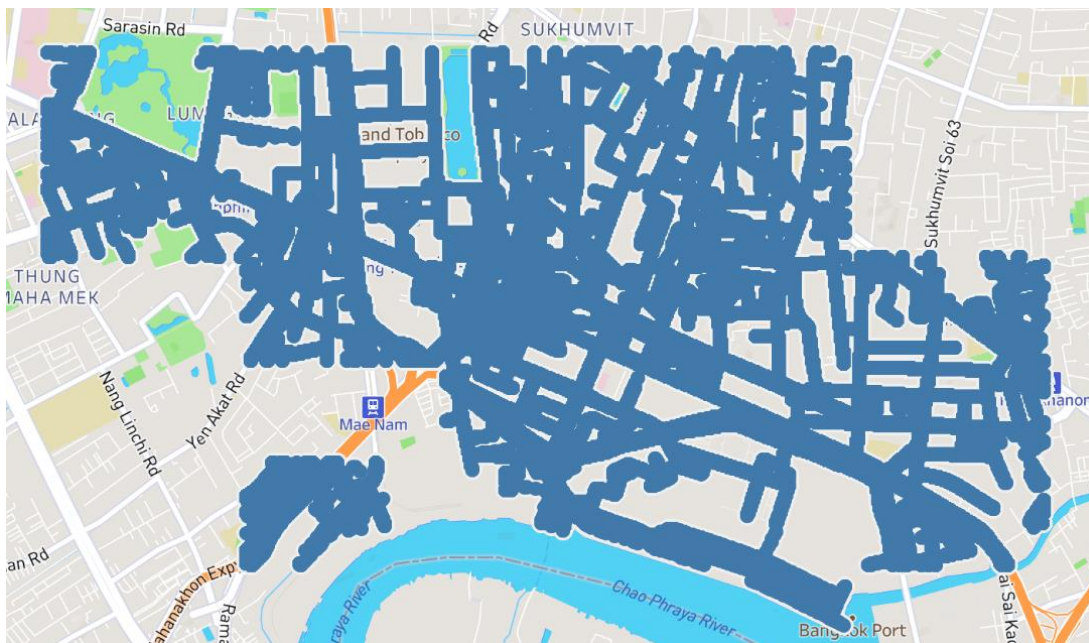
### แนวคิดและวิธีวิจัย

งานวิจัยนี้มุ่งเน้นที่การสร้างโมเดลสำหรับการพยากรณ์เวลาที่ใช้ในการเดินทางจากจุดหนึ่งไปยังอีกจุดหนึ่งโดยรถยนต์ด้วยวิธีการ Neural Network ประเภทวนซ้ำซึ่งเหมาะสมกับข้อมูลที่ลักษณะเป็นข้อมูล Timeseries โดยโมเดลที่เลือกนำมาวิจัยจะเลือกโมเดลที่ได้จากงานวิจัยที่เกี่ยวข้องอื่นและพบว่าประสิทธิภาพในการพยากรณ์ข้อมูลประกอบด้วย LSTM, RNN, GRU และ โมเดลที่เกิดจากการรวม 2 Model เข้าด้วยกันได้แก่ LSTM รวมกับ RNN, LSTM รวมกับ GRU และ LSTM รวมกับ DNN เพื่อมองหาโมเดลที่มีความสามารถในการพยากรณ์แม่นยำที่สุด และโมเดลที่ใช้เวลาในการสอนและทดสอบเร็วที่สุดแต่ความแม่นยำรองลงมา ทั้งนี้เพื่อเป็นตัวเลือกในการนำไปประยุกต์ใช้งานจริง

#### 3.1. ข้อมูลที่นำมาวิจัย

ลักษณะของข้อมูลที่ใช้ในการวิจัยโดยข้อมูลที่ใช้จะเป็นข้อมูลที่เกิดขึ้นจริงจาก GPS บนรถแท็กซี่ที่ให้บริการบนพื้นที่บริเวณถนนพระราม 4 ดังภาพที่ 22 ข้อมูลดังกล่าวจะเป็นข้อมูลที่ผ่านกระบวนการปรับตำแหน่งพิกัด GPS ให้เข้ากับแผนที่จริง (Map Matching) ซึ่งประกอบด้วยข้อมูลที่ระบุคนขับรถ, ระบุทริปการเดินทาง, ระบุพิกัด GPS และเวลา ทั้งก่อนและหลังทำ Map Matching, ระบุว่า GPS นั้นอยู่ Segment ID และ Way ID ไหน, ความเร็ว ณ ขณะนั้น และข้อมูลอื่นๆที่เกี่ยวข้องกับการเดินทาง

ข้อมูล GPS ที่ได้จะมีการเก็บข้อมูลขณะรถแท็กซี่ให้บริการผู้โดยสาร และมีการเคลื่อนที่ทุก 4 วินาที ถ้าหากรถแท็กซี่มีการหยุดนิ่งจะไม่มีเก็บข้อมูล GPS และเวลา ณ จุดหยุดนิ่ง



ภาพที่ 22 บริเวณของข้อมูลที่ได้รับเพื่อใช้ในการวิจัย

### 3.2. ภาพรวมของวิธีการ

งานวิจัยจะเริ่มต้นจากการนำข้อมูล GPS จากรถแท็กซี่เข้าฐานข้อมูลโพสตรสคิวแอล (PostgreSQL) เป็นระบบการจัดการฐานข้อมูลเชิงวัตถุ-สัมพันธ์ (Object-Relational) ประกอบกับฐานข้อมูลโพสตรสคิวแอลมีส่วนขยายเพิ่มเติมที่ชื่อว่าโพสตรจิส (PostGIS) ซึ่งโพสตรจิสเป็นเครื่องมือที่ช่วยในการจัดการข้อมูลที่มีลักษณะสัมพันธ์เชิงพื้นที่ (Spatial) เช่น ช่วยในการหาระยะทางระหว่างจุด 2 จุดของข้อมูล GPS เป็นต้น

เมื่อนำข้อมูลเข้าสู่ฐานข้อมูลจะทำการศึกษาถึงลักษณะของข้อมูล และวิธีการคำนวณระยะเวลาเดินทางจากข้อมูล (Data Processing) และวิธีการแบ่งระยะเวลาเดินทางในแต่ละ Timestep ถ้าหากหลังจากแบ่งข้อมูลออกเป็น Timestep แล้วพบว่าข้อมูลทริปการเดินทางของรถแท็กซี่ต่อ Timestep มากกว่า 1 ทริปจะใช้วิธีการหาค่ามัธยฐาน (Median) เพื่อให้ได้ค่าระยะเวลาเดินทางที่สามารถเป็นตัวแทนของ Timestep นั้นได้ โดยการคำนวณจะเลือกเพียงบางช่วงถนนมาใช้ในการพิจารณาเพื่อวิจัย

นอกจากระยะเวลาเดินทางแล้ว ยังมีการคำนวณวันที่มีการเดินทาง, ลำดับเวลาของวัน และลำดับวันของสัปดาห์เพิ่มเติม

หลังจากนั้นนำข้อมูลระยะเวลาเดินทางที่ได้ในแต่ละ Timestep มาวาดกราฟเพื่อมองหาข้อมูลที่ผิดปกติ (Outlier) เช่น มีลักษณะการเดินทางของรถแท็กซี่ที่ผิดปกติในลักษณะของการ

เดินทางในช่องทางที่ผิดดังนั้นจึงถูกบังคับให้เลี้ยวซ้ายก่อนจะมีการกลับรถแท็กซี่เข้าสู่เส้นทางหลักเดิม ซึ่งการเดินทางลักษณะดังกล่าวจะส่งผลต่อค่ามัธยฐานของระยะเวลาเดินทางเป็นอย่างมาก หลังจากนั้นก็จะทำการคำนวณระยะเวลาเดินทางใหม่ และพิจารณาว่ามีข้อมูลที่ผิดปกติหลงเหลือหรือไม่

เมื่อทำการนำข้อมูลผิดปกติออกพบว่ามีบาง Timestep ที่ไม่พบข้อมูลระยะเวลาเดินทางที่เกิดขึ้นจริงซึ่งจะส่งผลต่อการนำมาใช้เพื่อสอนโมเดล ดังนั้นจึงมีการเติมข้อมูลที่ขาดหายของบางช่วง Timestep โดยมีลำดับขั้นตอนดังนี้

1. นำข้อมูลระยะเวลาเดินทางที่มีลำดับเวลาของวัน และลำดับวันของสัปดาห์เดียวกันมาหาค่ามัธยฐานก่อนจะเติมในช่วงเวลาที่ขาด
2. ถ้าหากข้อมูลยังคงขาดอยู่จะนำระยะเวลาเดินทางที่มีลำดับเวลาของวันเดียวกันโดยไม่สนใจลำดับวันของสัปดาห์มาหาค่ามัธยฐานก่อนจะเติมในช่วงเวลาที่ขาด
3. ถ้าหากข้อมูลยังคงขาดอยู่จะนำระยะเวลาเดินทางเฉลี่ย 4 ช่วงติดกัน (นำลำดับเวลาของวันมาแบ่งเป็นกลุ่ม กลุ่มละ 4 ช่วง) มาหาค่ามัธยฐานก่อนจะเติมในช่วงเวลาที่ขาด

หลังจากขั้นตอนนี้จะได้ข้อมูลที่ครบทุกช่วง Timestep ในช่วงถนนที่เลือกมาใช้ในการวิจัย จากนั้นจึงนำข้อมูลที่ได้มาปรับรูปแบบให้อยู่ในลักษณะของการมองข้อมูลย้อนหลัง (Look Back) 1 Timestep เพื่อจะพยากรณ์ Timestep ปัจจุบัน

การสอนและทดสอบโมเดลจะเริ่มจากการนำข้อมูลที่มีการเตรียมพร้อมมาปรับค่าโดยฟังก์ชัน Min-Max เพื่อปรับค่าในแต่ละ feature และค่าที่จะพยากรณ์ให้ค่าที่ได้อยู่ระหว่าง 0 ถึง 1 จากนั้นจะแบ่งข้อมูลออกเป็นข้อมูลสำหรับสอน 80% และทดสอบ 20% และข้อมูลสำหรับสอนจะถูกแบ่งเป็นข้อมูลสำหรับสอน 80% และสำหรับตรวจสอบ 20% และนำข้อมูลเหล่านั้นเข้าสู่โมเดล Neural Network ประเภทวนซ้ำที่มีการเลือกนำมาวิจัย

การวิจัยจะทำการปรับแต่งโมเดลด้วยการปรับค่าพารามิเตอร์ (Parameter) ของแต่ละโมเดลเพื่อหาค่าที่ส่งผลให้โมเดลได้ค่าความแม่นยำที่สุดในแต่ละชนิดของโมเดล โดยวัดผลความแม่นยำด้วยฟังก์ชัน RSME และระยะเวลาที่ใช้ในการสอนและทดสอบโมเดล

เมื่อได้ค่าพารามิเตอร์ที่เหมาะสมจะนำมาใช้เพื่อพยากรณ์ช่วงถนนอื่นเพื่อทดสอบว่าหากนำไปใช้งานจริงจะสามารถใช้โมเดลที่สอนและทดสอบด้วยถนนช่วงหนึ่งไปใช้กับถนนช่วงอื่นได้หรือไม่ นอกจากนี้การวิจัยยังครอบคลุมถึงขนาดของ Timestep ว่าหากมีการปรับเปลี่ยนขนาดของ Timestep เป็น Timestep ละ 5 นาที, 10 นาที, 30 นาที และ 60 นาที ผลลัพธ์ที่ได้จะเป็นอย่างไร

การพิจารณาเปรียบเทียบผลระหว่างการพยากรณ์ด้วยข้อมูลที่มีลักษณะ Link Level และ Path Level ลักษณะไหนให้ผลลัพธ์ที่ดีกว่ากัน หรือเหมาะสมแก่การนำไปใช้งานมากกว่ากัน รวมไปถึงการแบ่ง

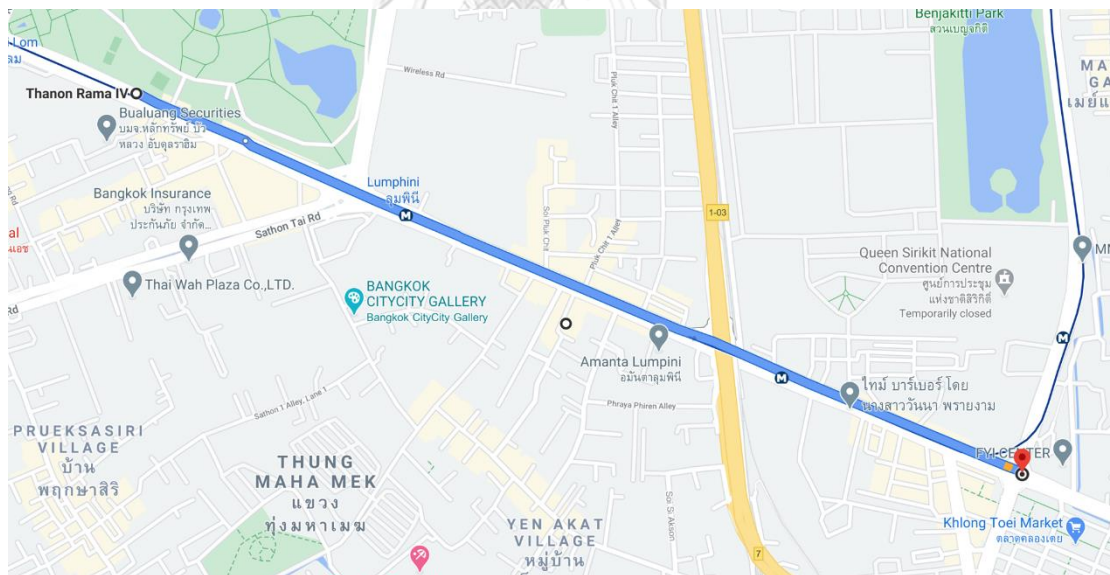


ข้อมูลเป็นชุดตามทิศทางของการเดินทางก่อนจะทำการสร้างโมเดล หรือการไม่แบ่งข้อมูลก่อนอันไหนให้ความถูกต้องที่มากกว่ากัน

สุดท้ายจะเป็นนำผลที่ได้มาสรุปและวิเคราะห์ผลการวิจัยว่าการปรับแต่งโมเดลด้วยพารามิเตอร์แบบใดเหมาะสมกับชุดข้อมูล และขนาดของชุดข้อมูลเท่าไรถึงจะเหมาะสมแก่การนำมาวิจัย และขอเสนอแนะสำหรับผู้ที่จะกระบวนการดังกล่าวไปศึกษาเพิ่มเติม

### 3.3. การเลือกถนนช่วงที่ใช้ในการวิจัย

การเลือกช่วงถนนที่ใช้ในการวิจัยหลักจะเลือกถนนช่วงบริเวณหน้าสวนลุมพินีจนถึงสี่แยกตลาดคลองเตยดังภาพที่ 23 เนื่องด้วยบริเวณหน้าสวนลุมพินีคือจุดที่ขยับ 1 ช่วงถนนของขอบเขตของข้อมูลที่ได้รับ และเป็นจุดที่น่าสนใจ (point of interest) บนถนนพระราม 4 จึงเป็นจุดเริ่มต้นถนนช่วงที่สนใจจะศึกษา ขณะเดียวกันสี่แยกตลาดคลองเตยก็เป็นสี่แยกใหญ่บนถนนพระราม 4 และเป็นสี่แยกสุดท้ายก่อนจะสุดขอบเขตของข้อมูล จึงตั้งสมมุติฐานว่าถ้าหากใช้ข้อมูลระยะเวลาเดินทางในช่วงดังกล่าวมาศึกษาจะส่งผลให้โมเดลสามารถใช้ในการพยากรณ์ระยะเวลาเดินทางของถนนช่วงอื่น ๆ บนถนนพระราม 4 ได้



ภาพที่ 23 ช่วงถนนหลักที่ใช้ในการวิจัย

## บทที่ 4

### การทดลองและผลการทดลอง

#### 4.1. เครื่องมือที่ใช้ในการพัฒนา

การพัฒนาโมเดลสำหรับการพยากรณ์ระยะเวลาเดินทางโดยรถยนต์จากจุดหนึ่งไปอีกจุดหนึ่ง โดยใช้ข้อมูลจาก GPS trace จากรถแท็กซี่จึงจำเป็นต้องมีเครื่องมือที่ใช้สำหรับการประมวลผลข้อมูล และเครื่องมือสำหรับสร้างโมเดลเพื่อพยากรณ์จากข้อมูลที่ผ่านการประมวลผล

การประมวลผลข้อมูลจำเป็นต้องทำการโหลดข้อมูล GPS trace เข้าสู่ฐานข้อมูลก่อนจึงสามารถคำนวณระยะเวลาเดินทางในแต่ละช่วงเวลาของช่วงถนนที่พิจารณาได้ โดยข้อมูลที่นำเข้ามีลักษณะเป็นข้อมูล GPS จึงจำเป็นต้องมีฐานข้อมูลที่มีเครื่องมือที่ช่วยในการจัดการข้อมูลเชิงพื้นที่ได้

การสร้างโมเดลจะทำการสร้างด้วยภาษา Python โดยใช้เครื่องมือ Keras

##### 4.1.1. เครื่องคอมพิวเตอร์ที่ใช้ในการวิจัย

- ระบบปฏิบัติการ MacOS Catalina
- Processor คือ Intel Core i9 2.4 GHz 8-Core
- Memory คือ 32 GB 2667 MHz DDR4
- Graphic Card คือ AMD Radeon Pro 5500 M 4GB
- Harddisk คือ 500 GB

##### 4.1.2. เครื่องมือที่ใช้ในการวิจัย

- ฐานข้อมูล Postgres ติดตั้งส่วนเสริม Postgis
- ภาษาคอมพิวเตอร์ Python บนแอปพลิเคชัน Jupyter Notebook
- เครื่องมือ keras, sklearn, numpy และ pandas

#### 4.2. การเตรียมข้อมูลก่อนเริ่มทำการวิจัย

ข้อมูลที่ใช้ในการวิจัยจะเป็นข้อมูลประเภท GPS Trace ของรถแท็กซี่ในบริเวณถนนพระราม 4 โดยข้อมูลผ่านการทำ Map Matching จากทางเจ้าของข้อมูลในรูปแบบของไฟล์ข้อมูลหลายไฟล์ ตามวันของข้อมูล

ข้อมูลเป็นข้อมูลการบันทึกตำแหน่ง GPS และเวลา ณ ขณะนั้นถ้าหากแท็กซีคันนั้นมีการรับผู้โดยสาร และรถยนต์มีการเปลี่ยนตำแหน่งจากเดิมซึ่งจะมีการบันทึกข้อมูลทุก ๆ 4 วินาที นอกเหนือจากนี้จะไม่มีการบันทึกข้อมูล โดยข้อมูลที่อาจจะไม่ครบทั้งทริปการเดินทางเนื่องจากมีข้อมูลเพียงบริเวณพระราม 4 เท่านั้น และ 1 ทริปการเดินทางจะประกอบด้วยหลายข้อมูลตามจำนวนที่มีการเก็บได้

ขั้นตอนแรกจะทำการรวมไฟล์ข้อมูลเป็นไฟล์เดียวและนำเข้าข้อมูลสู่ฐานข้อมูล Postgres ที่ติดตั้งส่วนเสริม PostGIS โดยข้อมูลที่จะนำมาใช้ในการวิเคราะห์จะประกอบด้วย bookingcode, pingtimestamp, projectedlat, projectedlng และ wayids สำหรับข้อมูลอื่น ๆ ที่ไม่เกี่ยวข้องจะไม่นำมาใช้ในการวิจัย

คำอธิบายข้อมูล

- bookingcode คือ ID เพื่อระบุข้อมูลว่าเป็นของแท็กซีทริปไหน
- pingtimestamp คือเวลา ณ ที่มีการบันทึกตำแหน่งนั้น โดยข้อมูลเป็นเลข Unix Timestamp
- projectedlat, projectedlng คือพิกัด Latitude และ Longitude ที่ผ่านการทำ Map Matching แล้ว
- wayids คือ ID เพื่อระบุว่าตำแหน่งที่มีการบันทึกนั้นอยู่บนถนนช่วงไหนหลังจากกระบวนการ Map Matching แล้ว

ขั้นตอนถัดไปคือการเปลี่ยนค่าข้อมูล pingtimestamp โดยจากค่า Unix Timestamp เปลี่ยนเป็น Timestamp ที่สามารถอ่านได้ในรูปแบบ ปี เดือน วัน (yyyy-mm-dd) และเปลี่ยนค่า Time zone เป็นประเทศไทย (GMT+7)

นอกจากนี้ยังมีการเตรียมข้อมูลในแต่ละ bookingcode ว่า booking นั้นผ่าน wayids อะไรบ้างเพื่อช่วยให้ง่ายต่อการเลือก booking มาใช้คำนวณระยะเวลาเดินทางของ booking ที่ผ่านช่วงถนนที่สนใจ

#### 4.3. ขั้นตอนการหาระยะเวลาเดินทาง

ข้อมูลที่ได้ครอบคลุมพื้นที่ถนนพระราม 4 ดังภาพที่ 22 ซึ่งประกอบด้วยหลากหลายเส้นทางที่เป็นไปได้ ดังนั้นขั้นตอนแรกคือการเลือกช่วงของถนนที่สนใจจะศึกษาตามการออกแบบการทดลองที่แตกต่างกัน

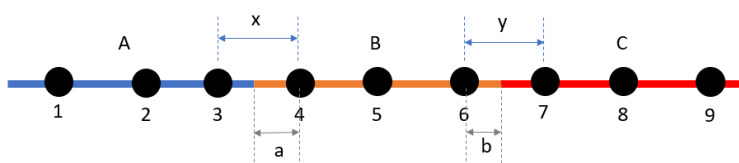
การคำนวณระยะเวลาเดินทางมีขั้นตอนดังนี้

1. เลือกช่วงของถนนที่สนใจจะทำการคำนวณระยะเวลาเดินทาง
2. หาตำแหน่ง GPS ที่เป็นจุดเริ่มต้น wayids และจุดสุดท้ายของ wayids ที่ต้องการคำนวณระยะเวลาเดินทาง
3. เลือก bookingcode จากข้อมูลที่มีการเก็บก่อนหน้าว่า ทริปไหนผ่าน wayids อะไรบ้าง โดยการพิจารณาจะต้องมองหาทริปที่มีการเดินทางจากช่วงก่อนหน้าจนถึงช่วงถัดไปของช่วงถนนที่ต้องการจะคำนวณระยะเวลาเดินทางเพื่อให้สามารถคำนวณระยะเวลาเดินทางได้ตลอดช่วงถนน ตัวอย่างเช่นภาพที่ 24 ที่ต้องการจะคำนวณระยะเวลาเดินทางของถนนช่วง wayids = B ดังนั้นจึงต้องเลือก booking ที่มีการเดินทางผ่านเส้นทางถนนช่วง A, B และ C



ภาพที่ 24 ตัวอย่างเส้นทางแท็กซี่ที่จะนำมาประมวล

4. นำทริปที่ได้จากข้อ 3 มาเลือกส่วนหนึ่งของข้อมูลของทริป โดยข้อมูลส่วนหนึ่งของทริปครอบคลุม 1 จุดก่อนเข้าสู่ช่วงถนนที่ต้องการคำนวณ และ 1 จุดหลังจากออกจากช่วงถนนที่ต้องการคำนวณ
5. คำนวณระยะเวลาเดินทางของแท็กซี่แต่ละทริปที่เดินทางตลอดช่วงถนน B ตัวอย่างเช่นภาพที่ 25 สามารถคำนวณระยะเวลาเดินทางของถนนช่วง B สามารถคำนวณได้จากสมการดังนี้ จุฬาลงกรณ์มหาวิทยาลัย



ภาพที่ 25 การคำนวณระยะเวลาเดินทางของถนนช่วง B

กำหนดให้ *Time* ใด ๆ สามารถหาได้จากความต่างของเวลา ณ จุดสองจุดที่ลำดับติดกัน และ *Distance* สามารถคำนวณได้จาก GPS 2 จุดที่ติดกันโดยใช้เครื่องมือ PostGis ที่สามารถคำนวณระยะห่างระหว่าง 2 จุดได้ จะได้ว่า

เมื่อ

$$Time_a = (Time_x * \frac{Distance_a}{Distance_x})$$

$$Time_b = (Time_y * \frac{Distance_b}{Distance_y})$$

ดังนั้น

$$Time_B = Time_a + Time_b + Time_{4-6}$$

โดย

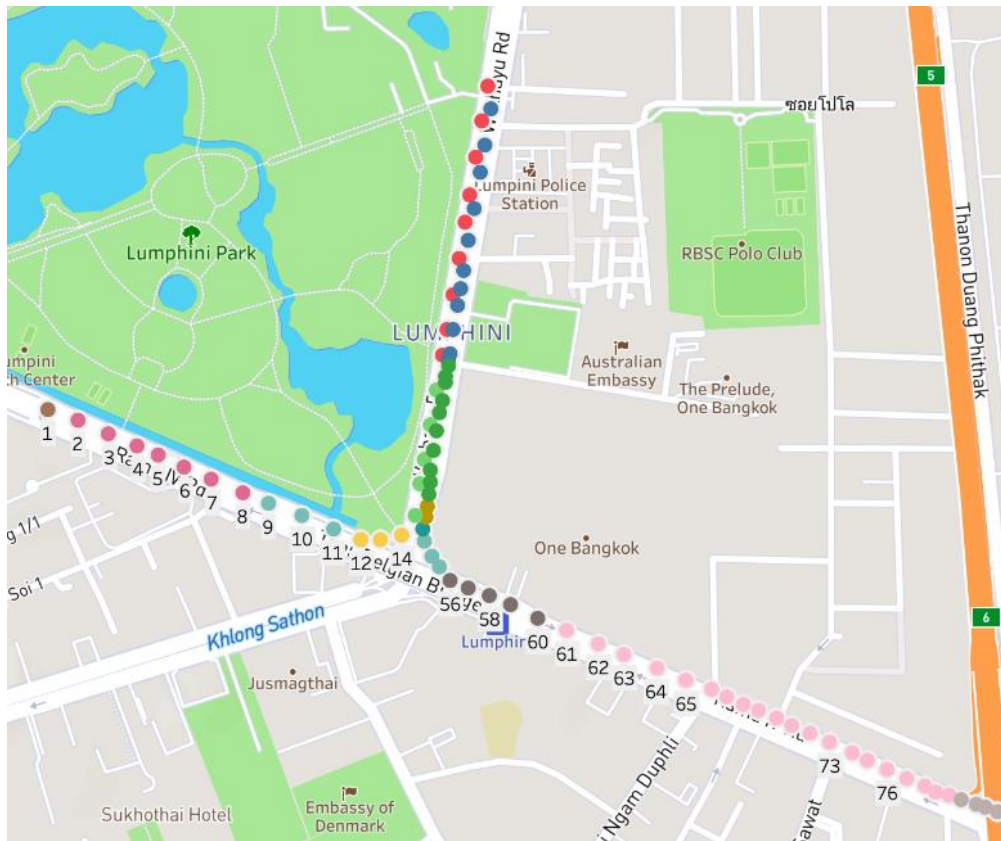
- $Time_a$ ,  $Time_b$ ,  $Time_x$  และ  $Time_y$  คือระยะเวลาเดินทางในช่วงถนน a, b, x และ y ตามลำดับ
- $Distance_a$ ,  $Distance_b$ ,  $Distance_x$  และ  $Distance_y$  คือระยะทางในช่วงถนน a, b, x และ y ตามลำดับ
- $Time_{4-6}$  คือระยะเวลาเดินทางระหว่างจุดที่ 4 จนถึง จุดที่ 6
- $Time_B$  คือระยะเวลาเดินทางตลอดช่วงถนน B

หลังจากจบขั้นตอนข้างต้นจะได้ข้อมูลที่อยู่ในรูปแบบของ 1 ข้อมูลต่อทริป

#### 4.4. การเตรียมข้อมูลสำหรับนำมาสร้างโมเดลเครือข่ายประสาทเทียมแบบวนซ้ำ

การเตรียมข้อมูลจะเริ่มต้นจากการนำข้อมูล GPS trace มาคำนวณระยะเวลาเดินทางของแต่ละทริปที่ผ่านถนนช่วงที่ต้องการจะทดลองเตรียมไว้ก่อนจะเริ่มกระบวนการดังนี้

1. นำข้อมูลระยะเวลาเดินทางที่ได้จากการคำนวณในแต่ละทริปมาพิจารณาหาทริปที่มีลักษณะผิดปกติที่จะส่งผลต่อระยะเวลาเดินทาง ตัวอย่างเช่นทริปที่มีการขับรถในเส้นทางผิดปกติ, ข้อมูลหลังจาก Map Matching แล้วพบว่ามีพิกัด GPS ที่ถูกปรับไปอยู่บนถนนเส้นอื่นที่ไม่ถูกต้อง ตัวอย่างทริปที่มีความผิดปกติดังภาพที่ 26 ที่รถแท็กซี่มีการเลี้ยวซ้ายบริเวณแยกสวนลุมพินีก่อนจะกลับรถ และกลับมาเดินทางตรงในเส้นทางหลักเหมือนเดิม ซึ่งหากไม่กรองทริปเหล่านี้ออกจะส่งผลต่อระยะเวลาเดินทางโดยรวมภายหลัง



ภาพที่ 26 ทริปที่ถูกนำออกก่อนการวิเคราะห์

2. ทำการคำนวณข้อมูลเพิ่มเติมจากวันและเวลาที่แท็กซ์ทริปนั้นเริ่มเข้าสู่ช่วงถนนที่ต้องการคำนวณ โดยเพิ่มข้อมูลสำหรับวันที่เท่าไรของสัปดาห์ (day of week) ซึ่งมีค่าอยู่ระหว่าง 0 ถึง 6 และคำนวณว่าเวลานั้นอยู่ลำดับที่เท่าไรของวัน (sequence of day - Timestep) ซึ่งแต่ละ Timestep มีขนาด 5 นาทีแต่ต่างกันว่าขนาดของ sliding window โดยขึ้นอยู่กับลักษณะการทดลอง
3. คำนวณข้อมูลระยะเวลาเดินทางที่เป็นตัวแทนของแต่ละ Timestep ของแต่ละวัน จากข้อมูลที่ได้จากข้อ 2 ที่มีค่าวันที่ และ Timestep เดียวกัน ในกรณีที่อยู่ใน Timestep นั้นประกอบด้วยข้อมูลแท็กซ์ทริปมากกว่า 1 ทริปจะคำนวณระยะเวลาเดินทางด้วยวิธีมัธยฐาน (Median) แต่ถ้าหากมีทริปเดียวก็ใช้ข้อมูลนั้นเป็นตัวแทน
4. ถ้าหากมีข้อมูลไม่ครบทุกช่วง Timestep จะทำการเติมข้อมูลที่ว่างด้วยวิธี Median จากข้อมูลระยะเวลาเดินทางจากข้อ 3 โดย
  - a. นำข้อมูลระยะเวลาเดินทางของช่วงที่มีค่า Day of week และ Timestep เดียวกันมาคำนวณด้วยวิธี Median และนำมาเติมข้อมูลของช่วง Timestep ที่มีค่า Day of week และ Timestep เดียวกัน

- b. ถ้าหากยังมีข้อมูลที่ขาดหายจะทำการเติมข้อมูลด้วยวิธีเดียวกับข้อ a แต่จะพิจารณาเพียง Timestep เดียวกันโดยไม่คำนึงว่าจะเป็น Day of week เดียวกันหรือไม่
  - c. แต่ถ้าหากยังพบว่ามีข้อมูลที่ขาดหายจะทำการเติมข้อมูลด้วยค่า Median ของข้อมูลที่มีค่า Timestep ติดกัน 4 Timestep การจับกลุ่ม 4 Timestep ที่ติดกันทำการจับกลุ่มด้วยวิธีการไล่ลำดับ โดย Timestep ที่ 1 – 4 จะถูกจับอยู่กลุ่มเดียวกัน และไล่ลำดับไปจนกระทั่งถึง Timestep สุดท้าย
5. นำข้อมูลที่ได้ด้วยวิธีข้างต้นมาจัดรูปแบบให้อยู่ในลักษณะของการนำข้อมูลในอดีตมาพิจารณาข้อมูลใน Timestep ถัดไป 1 Timestep โดยประกอบด้วยข้อมูลดังนี้
- a. วันที่ของทริป
  - b. วันของสัปดาห์ของทริป
  - c. ลำดับที่ของ Timestep
  - d. ระยะเวลาเดินทาง
  - e. ระยะเวลาเดินทางของ Timestep ถัดไป
6. เรียงลำดับข้อมูลตามลำดับของเวลา และนำข้อมูลแปลงให้อยู่ในรูปแบบของ Data Frame จากนั้นแปลงค่าในทุก column ด้วยฟังก์ชัน Min-Max ซึ่งจะได้ค่าอยู่ระหว่าง 0 ถึง 1 เพื่อเตรียมพร้อมก่อนจะนำเข้าโมเดล

#### 4.5. การทดลองและวัดผล

การทดลองจะมีการแบ่งข้อมูลด้วยสัดส่วน 80:20 สำหรับข้อมูลสอน และข้อมูลทดสอบตามลำดับ จากนั้นนำข้อมูลสอนมาแบ่งด้วยสัดส่วน 80:20 สำหรับข้อมูลสอน และข้อมูลตรวจสอบก่อนจะนำข้อมูลดังกล่าวเข้าทดลองในทุกๆโมเดลด้วยข้อมูลชุดเดียวกัน

จุดประสงค์ของการทดลองเพื่อพิจารณาว่าโมเดลใดให้ค่าความถูกต้อง และเหมาะสมมากที่สุดในแต่ละสถานการณ์ โดยการทดลองประกอบด้วย

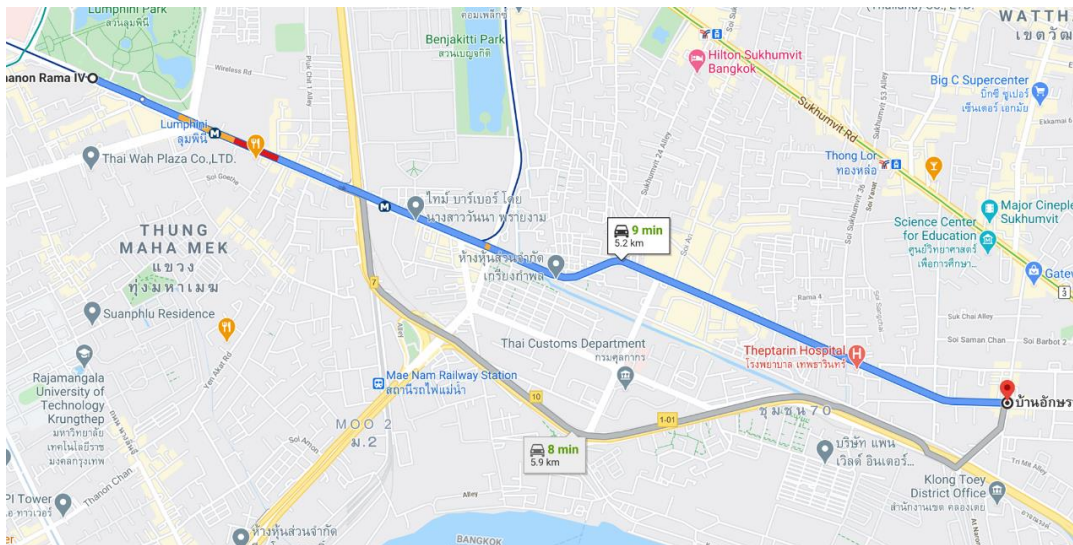
- **การทดลองที่ 1** จุดประสงค์ของการทดลองเพื่อมองหาโมเดล Timeseries ที่สามารถนำมาใช้พยากรณ์ระยะเวลาเดินทางได้ถูกต้องมากที่สุด การทดลองจะใช้ข้อมูลชุดเดียวกัน, จำนวนรอบในการสอนและทดสอบเท่ากัน, อัตราการเรียนรู้เท่ากัน และจำนวน Batch size เท่ากัน
  - การทดลองจะทำการทดลองด้วยโมเดล LSTM, RNN, GRU, LSTM+RNN, LSTM+GRU และ LSTM+DNN

- การวัดผลการทดลองจะวัดด้วยวิธีการ RSME และระยะเวลาสอนและทดสอบโมเดล
  - ข้อมูลที่ใช้ในการทดลองจะเป็นช่วงถนนดังภาพที่ 23 เนื่องจากคาดว่าจะจะเป็นช่วงถนนที่สามารถนำมาสร้างโมเดลและใช้โมเดลที่ได้พยากรณ์ช่วงถนนอื่นๆได้โดยไม่ต้องทำการสอนโมเดลอื่น
  - ผลที่ได้จากการทดลองจะสามารถสรุปได้ว่าโมเดลไหนให้ค่าถูกต้องมากที่สุด และโมเดลไหนใช้เวลาในการสอนและทดสอบเร็วที่สุด
- **การทดลองที่ 2** จุดประสงค์เพื่อต้องการทราบว่าขนาดของ Sliding Window และจำนวนรอบ (Epoch) ที่ใช้สอนที่แตกต่างกันจะส่งผลกระทบต่อความถูกต้อง ต่างกันหรือเหมือนกันหรือไม่ และโมเดลใดให้ค่าความถูกต้องมากที่สุดสำหรับข้อมูลในแต่ละขนาดของ Sliding Window และจำนวนรอบ
    - การทดลองเปรียบเทียบแต่ละโมเดลด้วยข้อมูลมีขนาดของ Sliding Window ที่แตกต่างกันประกอบด้วย 5 นาที, 10 นาที, 30 นาที และ 60 นาที , Timestep size เท่ากับ 5 นาที และจำนวนรอบสอนโมเดลประกอบด้วย 50 รอบ, 100 รอบ และ 200 รอบ
    - การวัดผลการทดลองจะวัดด้วยวิธีการ RSME และระยะเวลาสอนและทดสอบโมเดล
    - ข้อมูลที่ใช้ในการทดลองจะเป็นช่วงถนนดังภาพที่ 23 เนื่องจากคาดว่าจะจะเป็นช่วงถนนที่สามารถนำมาสร้างโมเดลและใช้โมเดลที่ได้พยากรณ์ช่วงถนนอื่นๆได้โดยไม่ต้องทำการสอนโมเดลอื่น
    - ผลที่ได้จากการทดลองจะสามารถสรุปได้ว่าโมเดลไหนเหมาะสมกับข้อมูลที่มี Sliding Window และต้องสอนและทดสอบจำนวนกี่รอบเพื่อให้ได้ผลที่ดีที่สุดในแต่ละขนาดของ Sliding Window และ Epoch
  - **การทดลองที่ 3** จุดประสงค์เพื่อต้องการทราบว่าหากนำผลที่ได้จากการทดลองที่ 1 มาสร้างโมเดลจะสามารถนำโมเดลดังกล่าวไปใช้เพื่อพยากรณ์ระยะเวลาเดินทางกับถนนช่วงอื่นได้หรือไม่
    - การทดลองจะทำการสร้างโมเดลจากการตั้งค่าโมเดลที่ได้จากผลการทดลองที่ 1 ไปพยากรณ์ระยะเวลาเดินทางถนนช่วงอื่นโดยไม่สร้างโมเดลใหม่
    - การวัดผลการทดลองจะวัดด้วยวิธีการ RSME และนำมาเปรียบเทียบกับกราฟระหว่างระยะเวลาเดินทางจริง และระยะเวลาเดินทางพยากรณ์



○ ข้อมูลที่ใช้ในการทดสอบจะเป็นช่วงถนนที่ประกอบด้วยภาพที่ 27, 28 และ 29 เนื่องจาก

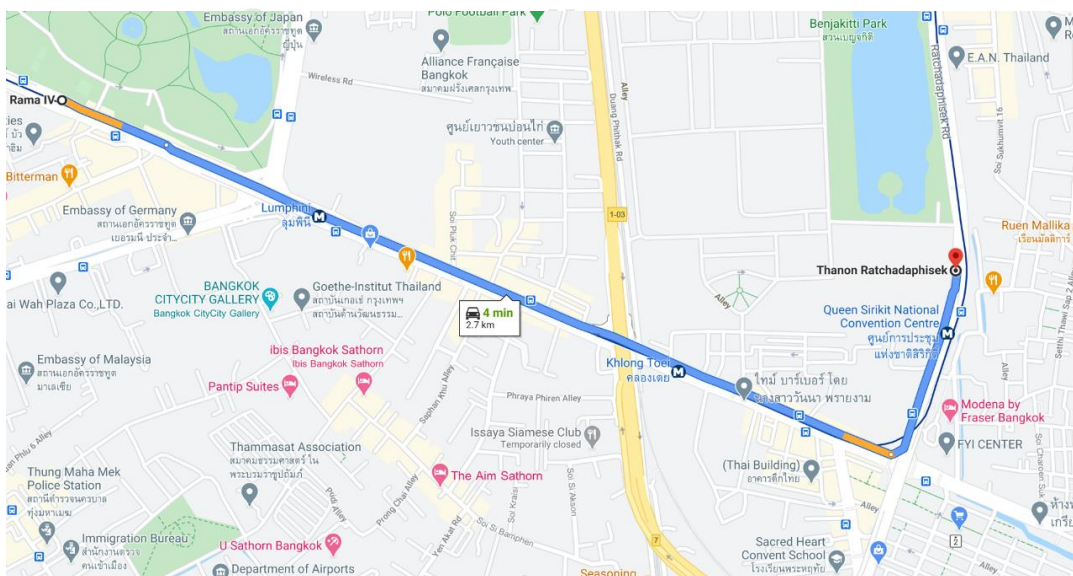
- ช่วงถนนที่ 1 คือช่วงถนนที่ยาวกว่าช่วงถนนที่ใช้ในการสร้างโมเดลเพื่อทดสอบว่าหากเป็นช่วงถนนที่ยาวกว่าช่วงถนนที่ใช้สอนจะยังคงสามารถพยากรณ์ได้หรือไม่



ภาพที่ 27 ช่วงถนนที่เลือกนำมาทดลองสำหรับการทดลองที่ 3 ช่วงที่ 1

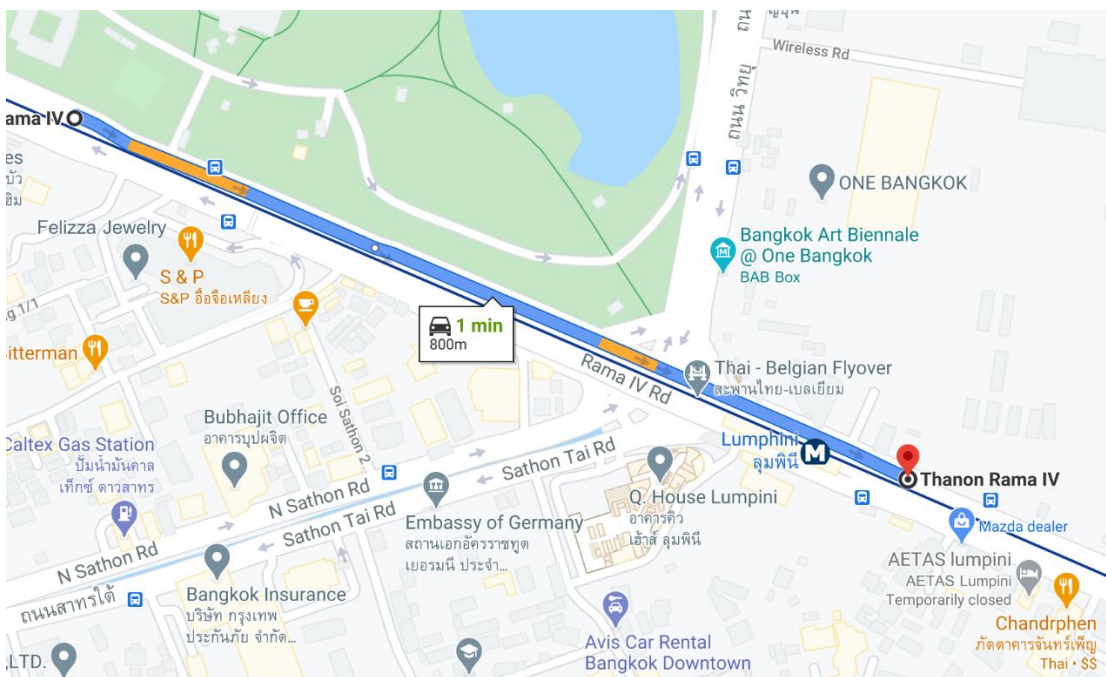
- ช่วงถนนที่ 2 คือช่วงถนนที่มีการเลี้ยวซ้ายเพื่อทดสอบว่าหากช่วงถนนไม่ใช่เป็นช่วงตรงเหมือนช่วงถนนที่ใช้สอนจะยังคงสามารถพยากรณ์ได้หรือไม่

จุดาลงกรณ์มหาวิทยาลัย



ภาพที่ 28 ช่วงถนนที่เลือกนำมาทดลองสำหรับการทดลองที่ 3 ช่วงที่ 2

- ช่วงถนนที่ 3 คือช่วงถนนที่สั้นกว่าช่วงถนนที่ใช้สอนเพื่อทดสอบว่าหากช่วงถนนสั้นกว่าช่วงถนนที่ใช้สอนจะยังคงสามารถพยากรณ์ได้หรือไม่



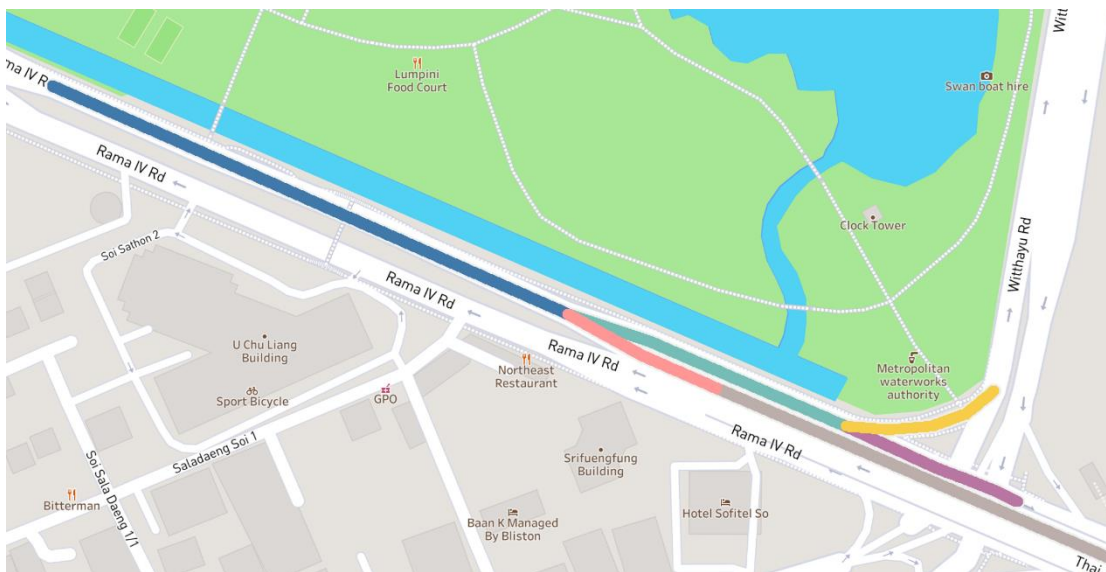
ภาพที่ 29 ช่วงถนนที่เลือกนำมาทดลองสำหรับการทดลองที่ 3 ช่วงที่ 3

- ผลที่ได้จากการทดลองจะสามารถสรุปได้ถ้าหากไม่ต้องการสร้างโมเดลใหม่สำหรับทุกช่วงถนน จะยังคงสามารถใช้โมเดลที่สร้างจากช่วงถนนที่เลือกเพียงโมเดลเดียวได้หรือไม่

จุฬาลงกรณ์มหาวิทยาลัย  
CHULALONGKORN UNIVERSITY

- การทดลองที่ 4 จุดประสงค์เพื่อต้องการทราบว่า การนำทิศทางการเดินทาง (เลี้ยวซ้าย, ตรงไป หรือเลี้ยวขวา) มาเป็นส่วนหนึ่งในการกรองแท็กซีทริปที่จะนำมาใช้เพื่อคำนวณระยะเวลาเดินทาง กับการไม่นำมาเป็นเงื่อนไขในการกรองแท็กซีทริปอันไหนให้ผลที่ให้ค่าถูกต้องมากกว่ากัน
  - การทดลองจะทำการเลือกช่วงถนนบริเวณก่อนถึงแยกมาพิจารณาโดยเตรียมข้อมูล 2 ชุด มาทำการสอนและทดสอบด้วยโมเดลที่มีการตั้งค่าเหมือนกันทุกอย่าง
  - การวัดผลการทดลองจะวัดด้วยวิธีการ RSME
  - ข้อมูลที่ใช้ในการทดสอบประกอบด้วยข้อมูล 2 ชุด โดยช่วงถนนที่เลือกเป็นช่วงถนนสีน้ำเงินดังภาพที่ 30 โดยมีสัดส่วนตามตาราง

- ข้อมูลชุดที่ 1 จะเป็นข้อมูลระยะเวลาเดินทางที่คำนวณจากแท็กซีทริปทั้งหมดโดยไม่มีการกรองข้อมูลตามทิศทางการเดินทาง
- ข้อมูลชุดที่ 2 จะเป็นข้อมูลระยะเวลาเดินทางที่คำนวณจากแท็กซีทริปที่เลือกเฉพาะทริปที่มีการตรงไปเท่านั้น



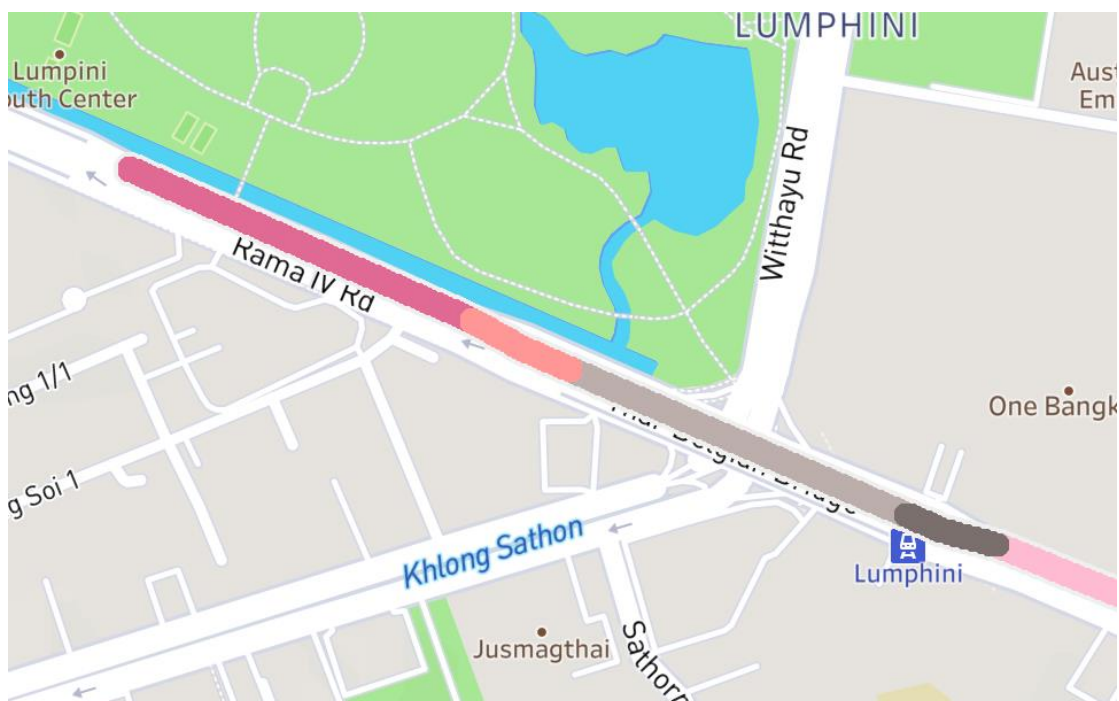
ภาพที่ 30 ช่วงถนนที่เลือกมาใช้ในการทดลองการทดลองที่ 4

ตารางที่ 7 จำนวนทริปในแต่ละทิศทางการเดินทาง

Direction	Go Straight	Go Left	Go Right	Go U-Turn	Total
Number	35,048	6,405	6,474	2,863	50,790
Percentage	69.01%	12.61%	12.75%	5.64%	100%

- ผลที่ได้จากการทดลองจะสามารถสรุปได้ว่าหากมีการเลือกสร้างโมเดลจากช่วงถนนที่สั้นกว่าจากแยกถนนหนึ่งไปอีกแยกถนนหนึ่งควรจะนำทิศทางการเดินทางมาพิจารณาหรือไม่
- การทดลองที่ 5 จุดประสงค์เพื่อต้องการทดสอบว่าการพยากรณ์ระยะเวลาเดินทางโดยข้อมูลระยะเวลาเดินทางที่คำนวณด้วยความละเอียดในลักษณะ Path Level และความละเอียดในลักษณะ Link Level หลาย Link มาต่อกัน แบบไหนให้ค่าความถูกต้องมากกว่ากัน

- การทดลองจะทำการสร้างโมเดลโดยนำข้อมูลของรถแท็กซี่ที่ผ่านตลอดเส้นทางสี่มุมพุ่มเข้มนจนถึงสี่น้ำตาลเข้มน เปรียบเทียบกับแบ่งออกเป็น 4 ช่วงถนนตามสี่แล้วสร้างโมเดลโดยพยากรณ์แต่ละช่วงถนนแล้วนำผลรวมมารวมกันดังภาพที่ 31 โดยสุ่มเส้นทางการเดินทางของรถแท็กซี่จำนวน 10 เส้นทางมาใช้ในการทดลอง
- ข้อมูลที่ใช้ในการทดสอบสุ่มเส้นทางการเดินทางของรถแท็กซี่จำนวน 10 ทริปมาใช้ในการทดลอง



ภาพที่ 31 ช่วงถนนที่เลือกสำหรับการทดลองที่ 5

- การวัดผลจะวัดด้วยการเลือกแท็กซี่ทริปจำนวน 10 ทริปโดยจะคำนวณค่าระยะเวลาเดินทางที่เกิดขึ้นจริง กับค่าที่ได้จากการพยากรณ์จากโมเดลทั้ง 2 ประเภทว่าแบบไหนค่าความผิดพลาดมากกว่ากัน
  - ผลที่ได้จากการทดลองจะสามารถสรุปได้การพยากรณ์ด้วยข้อมูลในลักษณะไหนได้ค่าที่ถูกต้องมากกว่ากัน
- **การทดลองที่ 6** จุดประสงค์เพื่อต้องการมองหาว่าช่วงเวลาชั่วโมงใดในการพยากรณ์ให้ค่าความถูกต้องมากที่สุด และช่วงเวลาชั่วโมงใดในการพยากรณ์ให้ค่าความถูกต้องน้อยที่สุด ซึ่งจะทำให้ทราบได้เวลานำโมเดลไปใช้ในสถานการณ์จริงว่าควรจะต้องระวังผลที่ได้จากการพยากรณ์ในช่วงเวลาใดของวัน

- การทดลองจะทำการสร้างโมเดลด้วยโมเดล LSTM-DNN ด้วยข้อมูล Sliding Window เท่ากับ 10 นาที, ขนาดของ Timestep เท่ากับ 5 นาที โดย Batch Size เท่ากับ 36 และ Epoch เท่ากับ 100
  - ข้อมูลที่ใช้จะเป็นช่วงถนนดังภาพที่ 23 เหมือนการทดลองที่ 1
  - การวัดผลการทดลองจะวัดด้วยวิธีการ RSME
  - ผลที่ได้จากการทดลองจะสามารถสรุปได้ว่าช่วงเวลาใดที่โมเดลสามารถพยากรณ์ได้ถูกต้องมากที่สุด และช่วงเวลาใดของวันที่ต้องระวังเวลานำโมเดลไปใช้ในสถานการณ์จริง
- **การทดลองที่ 7** จุดประสงค์เพื่อต้องการทราบว่าขนาดของ Timestep และจำนวนรอบ (Epoch) ที่ใช้สอนที่แตกต่างกันจะส่งผลกระทบต่อความถูกต้อง ต่างกันหรือเหมือนกันหรือไม่ และโมเดลใดให้ค่าความถูกต้องมากที่สุดสำหรับข้อมูลในแต่ละขนาดของ Timestep และจำนวนรอบ
    - การทดลองเปรียบเทียบแต่ละโมเดลด้วยข้อมูลมีขนาดของ Timestep ที่แตกต่างกัน ประกอบด้วย 5 นาที, 10 นาที, 30 นาที และ 60 นาที และจำนวนรอบสอนโมเดล ประกอบด้วย 50 รอบ, 100 รอบ และ 200 รอบ
    - การวัดผลการทดลองจะวัดด้วยวิธีการ RSME
    - ข้อมูลที่ใช้ในการทดลองจะเป็นช่วงถนนดังภาพที่ 23 เนื่องจากคาดว่าจะในช่วงถนนที่สามารถนำมาสร้างโมเดลและใช้โมเดลที่ได้พยากรณ์ช่วงถนนอื่นๆได้โดยไม่ต้องทำการสอนโมเดลอื่น
    - ผลที่ได้จากการทดลองจะสามารถสรุปได้ว่าโมเดลไหนเหมาะสมกับขนาดของ Timestep และจำนวนรอบสอนโมเดลเท่าไร

#### 4.6. ผลการทดลอง

##### 4.6.1. การทดลองที่ 1 โมเดล Timeseries ที่ได้ผลการพยากรณ์ถูกต้องที่สุด

การทดลองจะใช้ข้อมูลชุดเดียวกัน, จำนวนรอบในการสอนและทดสอบเท่ากัน, อัตราการเรียนรู้เท่ากัน และจำนวน Batch size เท่ากัน รวมไปถึงขนาดของ Timestep เท่ากันคือ 5 นาที และ Sliding Window เท่ากับ 5 นาทีเช่นกัน

โดยมีการกำหนดค่าการสร้างโมเดลดังตารางที่ 8 และการทดลองได้ผลลัพธ์ดังตารางที่ 9

ตารางที่ 8 การกำหนดค่าของแต่ละโมเดล

โมเดล	ลำดับชั้น	การตั้งค่าอื่น
LSTM	LSTM[288,288,288,144]+Dense[1 <sup>b</sup> ]	Learning Rate = 0.001 Epoch = 50 Batch Size = 36 Validation Split = 0.2 Optimizers = adam Loss Function = MSE
RNN	RNN[288,144,144]+Dense[1 <sup>b</sup> ]	
GRU	GRU[288,288,144]+Dense[1 <sup>b</sup> ]	
LSTM+RNN	LSTM[288]+RNN[288]+LSTM[288]+Dense[1 <sup>b</sup> ]	
LSTM+GRU	LSTM[288]+GRU[288]+LSTM[288]+Dense[1 <sup>b</sup> ]	
LSTM+DNN	LSTM[288,288,144]+Dense[36 <sup>a</sup> ,1 <sup>b</sup> ]	

a = Activation Function คือ TanH

b = Activation Function คือ Sigmoid

ตารางที่ 9 ผลการทดลองด้วยข้อมูลขนาด Timestep เท่ากับ 5 นาที

โมเดล	RSME	เวลาที่ใช้สอนโมเดล
RNN	212.58424	<b>17.59998 วินาที</b>
LSTM	222.18518	131.432652 วินาที
GRU	209.34152	78.994024 วินาที
LSTM-RNN	220.73284	75.388884 วินาที
LSTM-GRU	215.3416	95.268718 วินาที
LSTM-DNN	<b>208.40408</b>	71.940301 วินาที

จากผลการทดลองที่ 1 ตามตารางที่ 9 ได้ผลว่าโมเดล LSTM-DNN ได้ผลการพยากรณ์ที่ถูกต้องมากที่สุด แต่ก็ใช้เวลาในการสอนโมเดลนานด้วยเช่นกัน ขณะที่โมเดล RNN ได้ค่าพยากรณ์ที่ถูกต้องน้อยกว่าเป็นอันดับที่ 3 แต่เวลาที่ใช้ในการสอนก็สั้นกว่าโมเดล LSTM-DNN ประมาณ 54 วินาที หรือคิดเป็น 4 เท่า

ที่เป็นเช่นนี้คาดว่าเป็นผลจากการเพิ่ม DNN layer ให้กับโมเดล LSTM ที่เป็นโมเดลประเภท Timeseries ที่อาจจะเกิดปัญหาโมเดลจะเรียนรู้ลดลงเมื่อมีการสอนซ้ำ ๆ มากขึ้น แต่เมื่อเติมโมเดล DNN ต่อท้ายจะทำให้โมเดล LSTM-DNN เกิดการเรียนรู้เพิ่มขึ้นมากกว่าโมเดล Timeseries อื่น แต่

ขณะเดียวกันเมื่อมีจำนวน Layer ที่เพิ่มขึ้นย่อมส่งผลต่อระยะเวลาที่ใช้ในการสอนโมเดลมากขึ้นเช่นกัน

จึงสามารถสรุปจากผลการทดลองได้ว่าถ้าหากต้องการโมเดลที่สามารถพยากรณ์ได้ถูกต้องมากที่สุดควรเลือกโมเดล LSTM-DNN แต่ในกรณีที่ข้อมูลมีการเพิ่มขึ้นถี่และต้องการจะสอนโมเดลทุกครั้งที่มีการเพิ่มของข้อมูลเพื่อให้โมเดลมีการเรียนรู้ข้อมูลใหม่ โมเดล RNN ก็เหมาะสมแก่การนำไปประยุกต์ใช้งานจริง ดังนั้นการเลือกโมเดลที่จะนำไปใช้งานขึ้นอยู่กับลักษณะของงานที่จะนำไปประยุกต์ใช้

#### 4.6.2. การทดลองที่ 2 ขนาดข้อมูลของ Sliding Window และ จำนวน Epoch เท่าไรให้ค่าความถูกต้องที่สุด

การทดลองขนาดของ Sliding Window และจำนวนรอบ (Epoch) ที่ใช้สอนที่แตกต่างกันจะส่งผลต่อความถูกต้อง ต่างกันหรือเหมือนกัน หรือไม่ และโมเดลใดให้ค่าความถูกต้องมากที่สุดสำหรับข้อมูลในแต่ละขนาดของ Sliding Window และจำนวนรอบ

การทดลองจะใช้ค่าที่กำหนดตามตารางที่ 7 โดยจะทดลองทุกโมเดลสำหรับ Sliding Window ที่มีขนาด 5 นาที, 10 นาที, 30 นาที และ 60 นาที, ขนาดของ Timestep เท่ากับ 5 นาที และการทดลองแต่ละครั้งจะทดสอบ Epoch เท่ากับ 50, 100 และ 200 โดยผลที่ได้เป็นไปตามตารางที่ 10 และตารางที่ 11

จากตารางที่ 10 แสดงให้เห็นว่าโมเดล LSTM-DNN ให้ผลลัพธ์การพยากรณ์ที่ถูกต้องมากที่สุด ในหลายขนาด Sliding Window เมื่อเทียบกับโมเดลอื่น โดยพบว่ายิ่ง Sliding Window มากขึ้นเท่าไรผลลัพธ์ที่ได้ก็จะยิ่งถูกต้องมากขึ้น คาดว่าที่เป็นเช่นนี้เนื่องจากยิ่ง Sliding Window มากขึ้นระยะเวลาเดินทางในแต่ละ Timestep ก็จะมีควมราบเรียบมากขึ้นส่งผลให้โมเดลสามารถพยากรณ์ได้ใกล้เคียงมากขึ้น แต่จะสูญเสียรายละเอียดบางส่วนของแต่ละ Timestep

ตารางที่ 11 แสดงระยะเวลาที่ใช้ในการสอนโมเดลในแต่ละ Epoch ของแต่ละโมเดล และ Sliding Window ที่แตกต่างกัน ผลลัพธ์ที่ได้สอดคล้องกับการทดลองที่ 1 ที่สรุปว่าโมเดล RNN สามารถสอนและทดสอบได้เร็วที่สุดกว่าโมเดลอื่น และการเพิ่ม Epoch ถึงแม้ว่าจะได้ผลลัพธ์ที่ถูกต้องมากขึ้น แต่ระยะเวลาที่ใช้ในการสอนและทดสอบในทุกโมเดลและทุก Sliding Window ก็เพิ่มขึ้นเป็นจำนวนเท่าในอัตราส่วนแบบเส้นตรงขณะที่ความถูกต้องที่เพิ่มขึ้นไม่ได้เป็นเพิ่มขึ้นเป็นเชิงเส้น ดังนั้น

การกำหนด Epoch ต้องคำนึงถึงความคุ้มค่าระหว่างค่าความถูกต้องที่เพิ่มขึ้น และระยะเวลาสอน และทดสอบที่เพิ่มขึ้นเช่นกัน

จากข้อมูลตารางที่ 10 และตารางที่ 11 เมื่อพิจารณาในการนำไปใช้งานจริงพบว่าการใช้ Sliding Window เท่ากับ 30 นาทีและ 60 นาทีจะส่งผลให้ข้อมูลขาดความละเอียดมากเกินไปส่งผลให้ข้อมูลที่ได้จากการพยากรณ์ไม่สะท้อนความเป็นจริง และเมื่อพิจารณาระหว่าง Sliding Window ที่ 5 นาทีและ 10 นาที พบว่าโมเดล LSTM-DNN ให้ผลลัพธ์ที่ดีที่สุดที่ Sliding Window เท่ากับ 10 นาที โดยจำนวน Epoch ที่เหมาะสมขึ้นอยู่กับการนำไปประยุกต์ใช้งาน ซึ่งถ้าจำนวน Epoch ที่เพิ่มขึ้นก็จะได้ผลลัพธ์ที่ดีขึ้น แต่สำหรับการทดลองต่อไปขอเลือกใช้ Epoch ที่เท่ากับ 100

ตารางที่ 10 สรุปผลการทดลองเปรียบเทียบ RSME ต่างโมเดลและขนาดของ Sliding Window

Epoch	Model Type	5-Mins	10-Mins	30-Mins	60-Mins
50	RNN	212.58424	211.12302	158.3595	121.30106
	LSTM	222.18518	209.62986	148.34308	120.66138
	GRU	209.34152	199.58476	147.66386	116.9212
	LSTM-RNN	220.73284	202.59446	148.40382	117.6687
	LSTM-GRU	215.3416	203.6178	147.69426	122.60066
	LSTM-DNN	<b>208.40408</b>	<b>198.09074</b>	<b>131.89772</b>	<b>104.4476</b>
100	RNN	213.90082	199.12676	136.74706	<b>103.60374</b>
	LSTM	209.91882	196.64266	145.33058	104.53106
	GRU	203.78246	196.32964	143.87222	112.27664
	LSTM-RNN	211.07366	196.85322	149.50296	115.45524
	LSTM-GRU	206.88296	195.7284	147.07188	115.6341
	LSTM-DNN	<b>202.7011</b>	<b>194.0332</b>	<b>132.03476</b>	105.8706
200	RNN	215.55078	189.8799	134.29268	124.54598
	LSTM	203.26364	191.48218	131.76256	102.06312
	GRU	198.71146	188.65542	135.42756	<b>97.6311</b>



	LSTM-RNN	208.2508	192.11584	132.79822	109.06682
	LSTM-GRU	201.84642	193.82372	135.79446	103.39908
	LSTM-DNN	<u>197.35946</u>	<u>187.9567</u>	<u>130.80238</u>	100.27698

ตารางที่ 11 สรุปผลการทดลองเปรียบเทียบเวลาสอนและทดสอบต่างโมเดลและขนาดของ Sliding Window

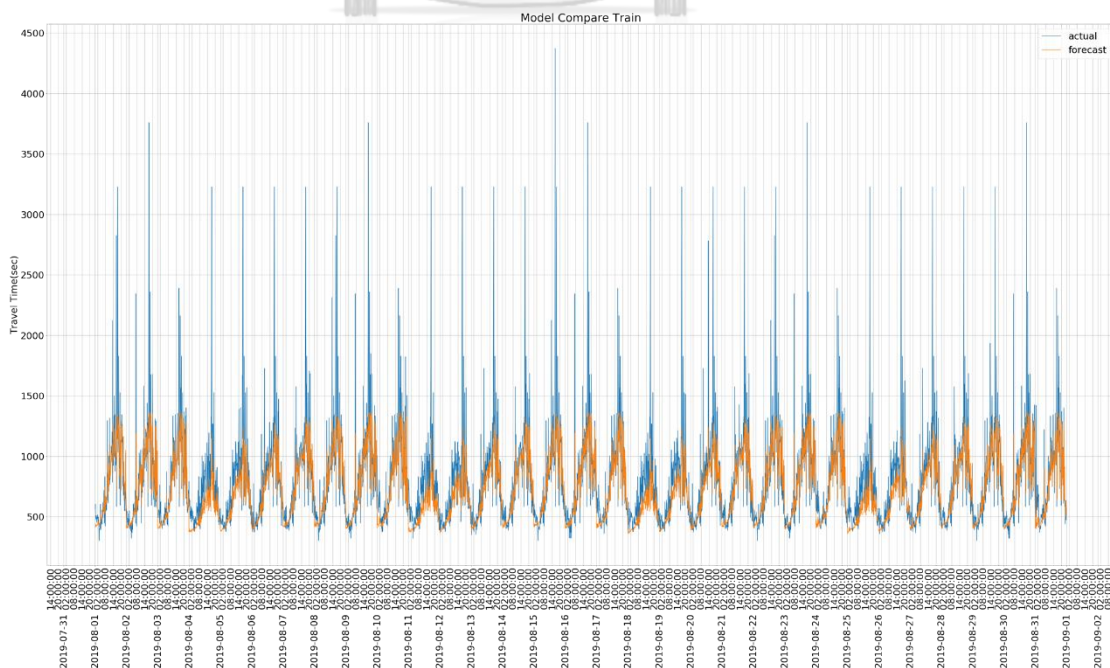
Epoch	Model Type	5-Mins	10-Mins	30-Mins	60-Mins
50	RNN	<u>17.59998</u>	<u>17.875142</u>	<u>18.314473</u>	<u>18.411348</u>
	LSTM	131.432652	119.797057	119.974053	116.020474
	GRU	78.994024	61.372592	62.828849	62.57655
	LSTM-RNN	75.388884	74.630308	71.566083	74.564912
	LSTM-GRU	95.268718	96.435227	94.79497	99.365037
	LSTM-DNN	71.940301	74.256064	72.768248	75.562167
100	RNN	<u>33.642775</u>	<u>34.441885</u>	<u>34.603348</u>	<u>34.96214</u>
	LSTM	249.86696	234.141166	237.663929	238.085531
	GRU	146.871533	120.700935	125.345985	121.583157
	LSTM-RNN	145.980427	147.882302	144.793989	147.83631
	LSTM-GRU	194.473087	186.876701	184.090759	200.435899
	LSTM-DNN	143.305472	149.901252	142.750609	145.325562
200	RNN	<u>66.772273</u>	<u>67.170929</u>	<u>68.15587</u>	<u>68.140144</u>
	LSTM	479.476445	475.182073	477.21047	474.09395
	GRU	261.452158	239.837545	245.531982	240.503284
	LSTM-RNN	317.645274	282.421082	293.730724	288.158464
	LSTM-GRU	364.738502	203.4872855	361.076772	401.638556
	LSTM-DNN	286.775731	314.253654	285.27541	287.633002

#### 4.6.3 การทดลองที่ 3 นำโมเดลที่ได้จากการทดลองก่อนหน้ามาทดลองเพื่อพยากรณ์กับถนนช่วงอื่น

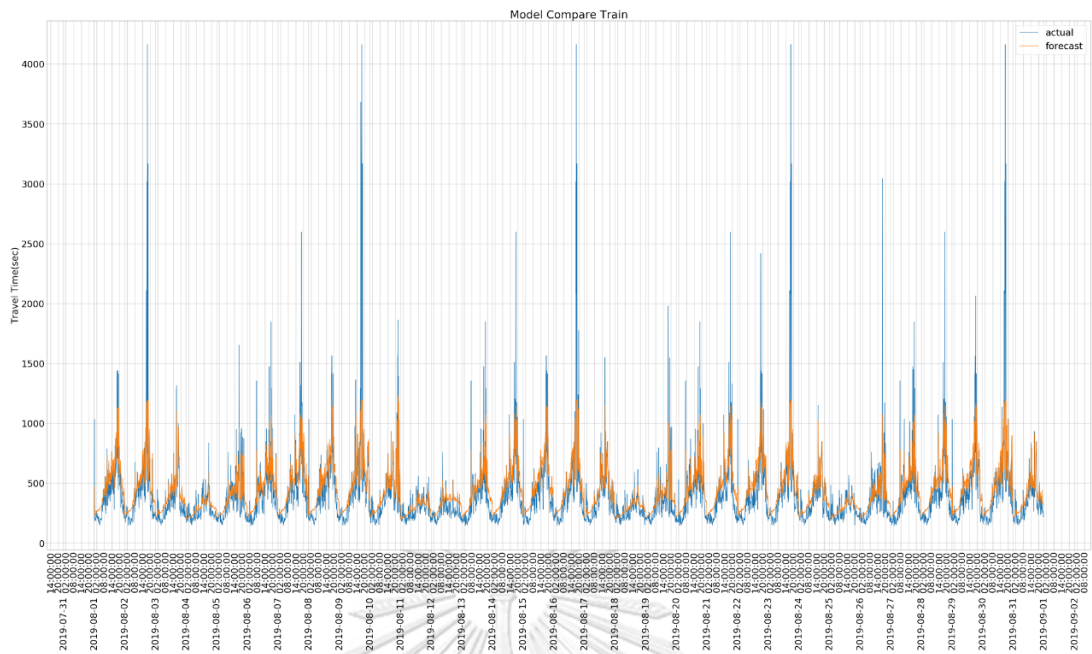
การทดลองโดยนำโมเดลที่ได้จากการทดลองที่ 2 ไปทดสอบกับช่วงถนนอื่นโดยเลือกเป็นถนนจำนวน 3 ช่วงดังภาพที่ 27 , 28 และ 29 โดยเมื่อผลการทดลองที่ได้มาวาดกราฟเพื่อเปรียบเทียบระหว่างค่าระยะเวลาเดินทางจริง และระยะเวลาเดินทางที่ได้จากค่าพยากรณ์ได้ผลการเปรียบเทียบดังภาพที่ 32, 33 และ 34

จากผลการทดลองให้ผลลัพธ์ว่าหากเรานำโมเดลที่สร้างด้วยข้อมูลจากถนนช่วงหนึ่งจากการทดลองที่ 1 ไปพยากรณ์ระยะเวลาเดินทางของถนนช่วงอื่นก็ยังคงสามารถนำมาเพื่อใช้พยากรณ์ได้ดังจะเห็นได้จากเส้นสีส้มที่ยังคงให้ค่าที่ใกล้เคียงในบางช่วงเวลาสำหรับช่วงถนนที่มีการเลี้ยว หรือทางตรงที่ระยะทางยาวขึ้น แต่ขณะเดียวกันสำหรับถนนช่วงที่ระยะทางสั้นกว่าดังภาพที่ 29 กลับพบว่าบางส่วนให้ค่าพยากรณ์ที่สูงกว่าค่าจริงจำนวนมากดังภาพที่ 34

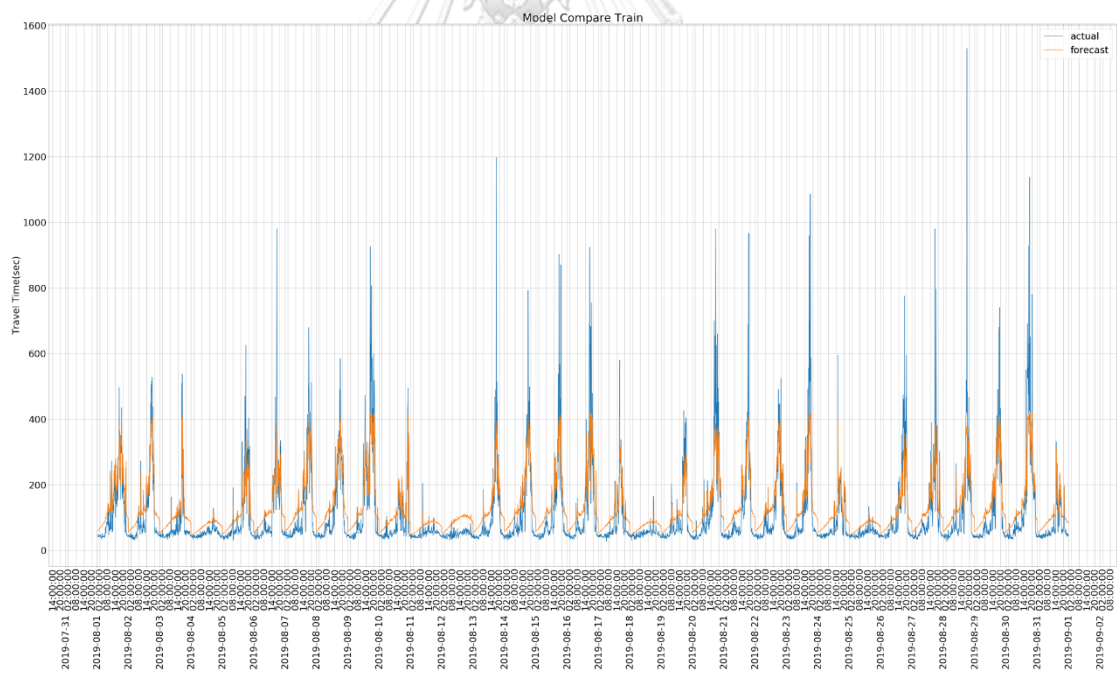
ถึงแม้ว่าผลการทดลองจะแสดงให้เห็นว่าสามารถนำมาใช้พยากรณ์ได้ก็ตามแต่ผลลัพธ์ที่ได้เมื่อเทียบกับการสร้างโมเดลสำหรับถนนช่วงนั้น จะพบว่าการสร้างโมเดลสำหรับถนนช่วงนั้นจะให้ผลลัพธ์การพยากรณ์ที่ถูกต้องมากกว่า ทั้งนี้เป็นผลมาจากความหลากหลายของระยะเวลาเดินทางในแต่ละช่วงถนนมีความแตกต่างกันในหลายช่วงเวลาของวันทำให้โมเดลที่เรียนรู้ด้วยข้อมูลรูปแบบหนึ่งเมื่อนำไปพยากรณ์ข้อมูลที่มีรูปแบบแตกต่างกันก็จะได้ผลลัพธ์ตามรูปแบบที่เรียนรู้เดิม



ภาพที่ 32 ถนนช่วงที่ 1 ที่นำมาทดลองและผลการพยากรณ์ด้วยโมเดล LSTM-DNN



ภาพที่ 33 ถนนช่วงที่ 2 ที่นำมาทดลองและผลการพยากรณ์ด้วยโมเดล LSTM-DNN



ภาพที่ 34 ถนนช่วงที่ 3 ที่นำมาทดลองและผลการพยากรณ์ด้วยโมเดล LSTM-DNN

#### 4.6.4 การทดลองที่ 4 การนำทิศทางการเดินทางมาพิจารณาจะให้ผลลัพธ์ที่ถูกต้องมากขึ้นหรือไม่

การทดลองจะเป็นการเลือก 1 ช่วงถนนจากนั้นนำข้อมูลการเดินทางโดยไม่พิจารณาทิศทางการเดินทางเทียบกับการนำข้อมูลเดินทางที่มีการเลือกเฉพาะทริปของรถแท็กซี่ที่มีทิศทางการเดินทางตรงข้ามสี่แยกดังภาพที่ 30 ก่อนนำข้อมูลทั้งสองแบบเข้าสู่โมเดล

ตารางที่ 12 สรุปผลการทดลองที่ 4 โดยแยกผลการพยากรณ์ด้วยโมเดลจากการทดลองที่ 2

Model Type	Epoch-50		Epoch-100	
	ไม่คิดทิศทาง	พิจารณาทิศทาง	ไม่คิดทิศทาง	พิจารณาทิศทาง
RNN	<b><u>132.147416</u></b>	143.59084	115.2345	<b><u>112.74254</u></b>
LSTM	116.686156	<b><u>116.4793</u></b>	117.546964	<b><u>115.98662</u></b>
GRU	117.419856	<b><u>117.08114</u></b>	121.023984	<b><u>113.34148</u></b>
LSTM-RNN	117.048164	<b><u>116.39262</u></b>	115.208524	<b><u>112.2196</u></b>
LSTM-GRU	116.771136	<b><u>116.40018</u></b>	116.026076	<b><u>112.44012</u></b>
LSTM-DNN	<b><u>114.4545</u></b>	115.386548	114.538896	<b><u>111.856064</u></b>

จากตารางที่ 12 แสดงให้เห็นว่าการแยกข้อมูลตามทิศทางการเดินทางก่อนการคำนวณระยะเวลาเดินทางและนำเข้าโมเดลให้ผลที่มีความถูกต้องมากกว่าการไม่แยกตามทิศทางการเดินทางทั้งหมดยกเว้นโมเดล LSTM-DNN กับ RNN ที่ Epoch เท่ากับ 50 เท่านั้นที่ได้ค่าที่ถูกต้องมากกว่า

การนำทิศทางการเดินทางเข้ามาช่วยในการพิจารณาเสมือนเป็นการตัดข้อมูลที่แปลกปลอมที่จะส่งผลกระทบต่อระยะเวลาเดินทางของ Timestep นั้นออกจากข้อมูลทั้งหมดก่อนจะเริ่มนำเข้าสู่โมเดล ซึ่งย่อมนำมาสู่ผลลัพธ์ที่ดีขึ้น

ดังนั้นการนำทิศทางการเดินทางเข้ามาร่วมในการพิจารณาด้วยย่อมถือเป็นสิ่งจำเป็นในขั้นตอนการเตรียมข้อมูลก่อนเริ่มทำการนำเข้าสู่โมเดล

#### 4.6.5 การทดลองที่ 5 เปรียบเทียบการพยากรณ์ระยะเวลาเดินทางระหว่างการนำข้อมูล ในลักษณะ Link Level มาเปรียบเทียบกับ Path Level

ผลการทดลองจะเห็นได้จากตารางที่ 13 ที่พบว่าแบบ Link Level ที่มีการแบ่งถนนออกเป็นช่วงย่อย 4 ช่วงให้ผลลัพธ์ที่มีความถูกต้องมากกว่าแบบใช้ข้อมูลยาวตลอดเส้นทาง แต่กระนั้นความซับซ้อนในการเตรียมข้อมูลและพยากรณ์ก็ซับซ้อนกว่าเช่นกันเนื่องด้วยต้องเตรียมข้อมูล 4 ชุด และเตรียมโมเดลสำหรับทดสอบ 4 โมเดล

สิ่งที่ทำให้แบบ Link Level ให้ผลลัพธ์ที่ดีกว่าแบบ Path Level คาดว่าเป็นผลมาจากระยะเวลาเดินทางในบางช่วงของ Link Level ที่สั้นมากจนแทบทำให้ในแต่ละช่วงระยะเวลาเดินทางมีความแตกต่างกันน้อยมากผลที่ได้ทำให้บางช่วง Link Level จะพยากรณ์ได้ดีกว่า ส่งผลให้เมื่อนำผลที่ได้จากทั้ง 4 Link มารวมกันก็เลยได้ผลลัพธ์ที่ได้จาก Path Level

แต่หากจะนำไปใช้งานจริง Path Level มีความเหมาะสมมากกว่าทั้งเรื่องความซับซ้อนของโมเดล และชุดข้อมูลที่น้อยกว่า และระยะเวลาที่ใช้ในการสอนที่น้อยกว่า และผลลัพธ์ในบางช่วงเวลาก็ต่างกันเพียงเล็กน้อยเท่านั้น

ตารางที่ 13 เปรียบเทียบแบบผลพยากรณ์แบบ path level และ link level

Trip	Start	Actual	Path Level	Link Level	Error Path Level	Error Link Level
Trip 1	2019-08-01 21:38:28	56	141.34738	94.658675	152.41%	<b><u>69.03%</u></b>
Trip 2	2019-08-03 03:58:00	36	59.25226	51.706296	64.59%	<b><u>43.63%</u></b>
Trip 3	2019-08-08 14:31:11	275	229.48903	141.236136	<b><u>16.55%</u></b>	48.64%
Trip 4	2019-08-13 23:00:26	41	75.09713	84.295247	<b><u>83.16%</u></b>	105.60%
Trip 5	2019-08-17 17:57:48	92	191.28688	138.211057	107.92%	<b><u>50.23%</u></b>
Trip 6	2019-08-18 11:17:41	43	81.388405	58.339508	89.28%	<b><u>35.67%</u></b>

Trip 7	2019-08-21 21:13:10	64	105.56396	94.455951	64.94%	<b><u>47.59%</u></b>
Trip 8	2019-08-23 10:50:23	50	107.23107	68.755957	114.46%	<b><u>37.51%</u></b>
Trip 9	2019-08-26 11:38:30	64	102.28214	66.49876	59.82%	<b><u>3.90%</u></b>
Trip 10	2019-08-28 00:46:50	42	53.34435	55.2838785	<b><u>27.01%</u></b>	31.63%

#### 4.6.5 การทดลองที่ 6 ทดสอบว่าหากพิจารณาในรายละเอียดของ RSME แต่ละช่วงย่อยว่าช่วงไหนของการพยากรณ์ที่ให้ผลที่ผิดพลาดสูงหรือน้อย

จากการคำนวณ RSME เป็นช่วงย่อยๆ ช่วงละ 1 ชั่วโมงเพื่อมองหาลักษณะของค่าที่ได้จากการพยากรณ์ว่าช่วงเวลาไหนมีค่าผิดพลาดที่สูงและช่วงเวลาไหนมีค่าผิดพลาดน้อย เมื่อพิจารณาด้วยวิธีการหาค่าเฉลี่ยและค่ามัธยฐานพบว่าช่วงเวลาเย็น (4 โมงเย็น ถึง 1 ทุ่ม) ให้ค่าผิดพลาดสูงกว่าช่วงเวลาอื่นซึ่งมากกว่า 200 วินาที ที่เป็นเช่นนี้เนื่องจากช่วงเวลาเย็นค่าที่จริงมีค่าที่สูงกว่าช่วงเวลาอื่นอย่างมาก และค่าที่สูงนั้นยังคงมีค่าไม่เหมือนกันในแต่ละวันซึ่งมีความแปรปรวนสูง ผลที่ได้เป็นไปตามตารางที่ 14 ที่แสดงค่า RMSE ทุก ๆ ชั่วโมงทั้งวิธีค่าเฉลี่ย และค่ามัธยฐาน

#### จุฬาลงกรณ์มหาวิทยาลัย

ตารางที่ 14 แสดงค่า RMSE ของโมเดลในแต่ละช่วงเวลาย่อย

ชั่วโมงที่	ค่าเฉลี่ย	ค่ามัธยฐาน	ชั่วโมงที่	ค่าเฉลี่ย	ค่ามัธยฐาน
0	47.9428687	37.5999972	12	112.192055	104.698754
1	40.5849608	36.311557	13	124.193573	119.098809
2	48.0307488	45.863921	14	127.660956	126.693948
3	58.0912393	59.3231838	15	144.911385	134.460852
4	72.9148416	69.2551443	16	<b><u>237.435337</u></b>	150.335088
5	59.0387277	50.390436	17	<b><u>454.149599</u></b>	<b><u>365.975003</u></b>
6	173.970389	76.1019054	18	<b><u>327.530629</u></b>	<b><u>335.249685</u></b>
7	108.187788	112.847177	19	<b><u>205.919723</u></b>	<b><u>217.522628</u></b>

8	105.702567	119.112211	20	152.331252	144.46004
9	106.827929	94.8560587	21	105.231457	102.74961
10	102.43849	98.3053031	22	109.218064	94.1862091
11	101.777283	92.5393339	23	87.6168289	95.3002502

เมื่อพิจารณาเพิ่มเติมถึงวันของสัปดาห์ว่ามีผลต่อค่า RSME หรือไม่ ผลลัพธ์พบว่าวันเสาร์ และวันอาทิตย์ให้ผลพยากรณ์ถูกต้องมากกว่าวันอื่น ที่เป็นเช่นนี้เนื่องจากในวันหยุดระยะเวลาเดินทาง ในทุกช่วงเวลาตลอดวันมีค่าความแปรปรวนน้อยกว่าช่วงวันธรรมดา ผลการเปรียบเทียบแสดงได้จากตารางที่ 15

ตารางที่ 15 แสดงค่า RSME ของโมเดลในแต่ละวันของสัปดาห์

วัน	ค่าเฉลี่ย	ค่ามัธยฐาน	วัน	ค่าเฉลี่ย	ค่ามัธยฐาน
วันอาทิตย์	<u>70.7720342</u>	<u>69.68703696</u>	วันพฤหัสบดี	161.3431927	167.555034
วันจันทร์	137.3648925	140.8929687	วันศุกร์	171.2500549	171.250055
วันอังคาร	125.8575662	123.7600457	วันเสาร์	<u>91.7429867</u>	<u>108.675184</u>
วันพุธ	140.7936454	137.0864315			

เมื่อพิจารณาจากตารางที่ 14 และตารางที่ 15 สามารถสรุปได้ว่าช่วงเย็นของวันให้ค่าความผิดพลาดที่สูงกว่าช่วงอื่นของวัน และวันเสาร์ และอาทิตย์สามารถพยากรณ์ได้ถูกต้องมากกว่าวันธรรมดา

#### 4.6.6 การทดลองที่ 7 ขนาดของ Timestep และ เเท่าไรให้ค่าความถูกต้องที่สุด

การทดลองขนาดของ Timestep และจำนวนรอบ (Epoch) ที่ใช้สอนที่แตกต่างกันจะส่งผลกระทบต่อความถูกต้อง ต่างกันหรือเหมือนกัน หรือไม่ และโมเดลใดให้ค่าความถูกต้องมากที่สุดสำหรับข้อมูลในแต่ละขนาดของ Timestep และจำนวนรอบ

การทดลองจะใช้ค่าที่กำหนดตามตารางที่ 7 โดยจะทดลองทุกโมเดลสำหรับ Timestep ที่มีขนาด 5 นาที, 10 นาที, 30 นาที และ 60 นาที การทดลองแต่ละครั้งจะทดสอบ Epoch เท่ากับ 50, 100 และ 200 โดยผลที่ได้เป็นไปตามตารางที่ 16

ตารางที่ 16 สรุปผลการทดลองเปรียบเทียบต่างโมเดลและขนาดของ Timestep

Epoch	Model Type	5-Mins	10-Mins	30-Mins	60-Mins
50	LSTM	222.18518	252.16786	263.79636	252.52358
	RNN	212.58424	251.9164	265.29714	251.70186
	GRU	209.34152	252.29068	263.97334	<b><u>251.67404</u></b>
	LSTM-RNN	220.73284	251.72908	263.8394	251.82348
	LSTM-GRU	215.3416	251.50934	<b><u>263.74466</u></b>	251.88552
	LSTM-DNN	<b><u>208.40408</u></b>	<b><u>250.97144</u></b>	265.52054	258.5792
100	LSTM	209.91882	251.37342	263.14138	249.33444
	RNN	213.90082	252.62592	<b><u>258.31362</u></b>	249.24136
	GRU	203.78246	253.86146	264.05186	248.88554
	LSTM-RNN	211.07366	250.87456	263.1689	249.01308
	LSTM-GRU	206.88296	250.61522	263.29874	248.89198
	LSTM-DNN	<b><u>202.7011</u></b>	<b><u>247.08206</u></b>	261.06536	<b><u>247.30428</u></b>
200	LSTM	203.26364	244.92464	259.93778	248.24088
	RNN	215.55078	254.43002	<b><u>245.52676</u></b>	<b><u>231.5616</u></b>
	GRU	198.71146	243.497616	262.45044	247.5682
	LSTM-RNN	208.2508	243.80438	260.14568	248.14604
	LSTM-GRU	201.84642	243.83956	259.42362	247.9064
	LSTM-DNN	<b><u>197.35946</u></b>	<b><u>243.137268</u></b>	255.15746	245.4443

ผลจากการทดลองพบว่าหากเปรียบเทียบที่ขนาดของ Timestep เท่ากัน ผลการทดลองพบว่าที่ Timestep เท่ากับ 5 นาที และ 10 นาที โมเดล LSTM-DNN ได้ค่าความแม่นยำที่ดีที่สุด ขณะที่ Timestep เท่ากับ 30 นาที และ Epoch เท่ากับ 50 โมเดล LSTM-GRU ให้ผลลัพธ์ที่ดีกว่า และที่ Epoch เท่ากับ 100 และ 200 โมเดล RNN ให้ผลลัพธ์ที่ดีกว่า ในขณะที่ Timestep เท่ากับ



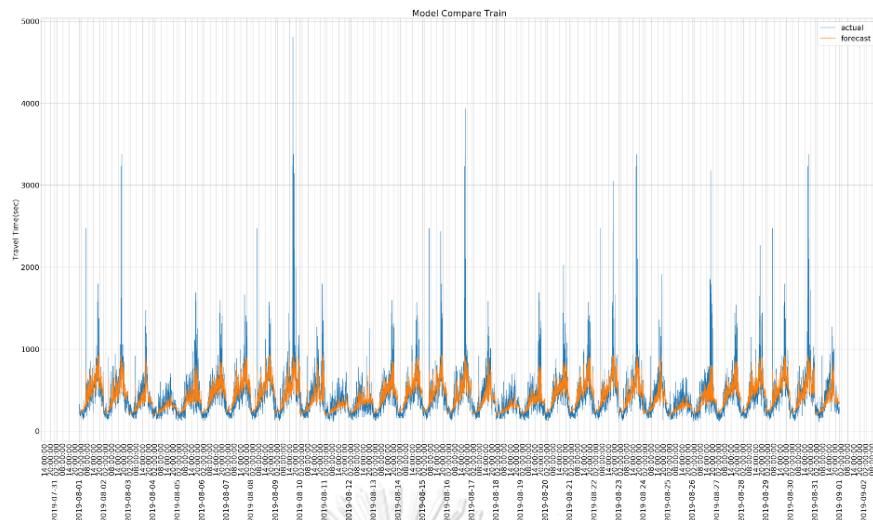
60 นาที และ Epoch เท่ากับ 50 โมเดล GRU ให้ผลลัพธ์ที่ดีที่สุด Epoch เท่ากับ 100 โมเดล LSTM-DNN ให้ผลลัพธ์ที่ดีที่สุด และที่ Epoch เท่ากับ 200 โมเดล RNN ให้ผลลัพธ์ที่ดีที่สุด

เมื่อพิจารณาถึงการจะนำโมเดลไปประยุกต์ใช้งานจริงขนาดของ Timestep เท่ากับ 30 นาที และ 60 นาทีเป็นช่วงเวลาที่นานเกินไปจึงไม่เหมาะสมแก่การนำมาใช้เพื่อพยากรณ์ระยะเวลาเดินทางใน Timestep ถัดไป

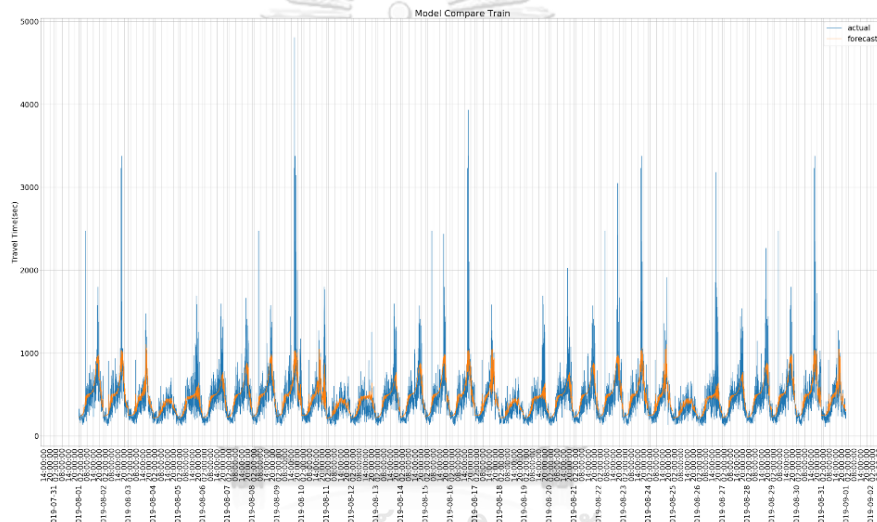
เมื่อนำผลการพยากรณ์ไปวาดกราฟจะเป็นไปดังภาพที่ 35 และภาพที่ 36 ที่แสดงการเปรียบเทียบระหว่างระยะเวลาเดินทางจริงกับระยะเวลาที่ได้จากการพยากรณ์ด้วยข้อมูลขนาด Timestep เท่ากับ 5 นาที และ 10 นาที ด้วย Epoch เท่ากับ 50, 100 และ 200 ตามลำดับ

เมื่อพิจารณาในรายละเอียดพบว่าโมเดลที่ใช้ขนาดของ Timestep ที่ 10 นาทีให้ผลลัพธ์ที่ดีกว่า 5 นาที และ Epoch เท่ากับ 200 ให้ผลลัพธ์ที่ดีที่สุด แต่ด้วย Epoch ที่ 100 กับ 200 ให้ผลลัพธ์ต่างกันไม่มาก แต่ระยะเวลาที่ใช้ในการสอนต่างกันเท่าตัว ดังนั้นจึงเลือกโมเดล LSTM-DNN ที่ขนาด Timestep เท่ากับ 10 นาทีและ Epoch เท่ากับ 100 มาใช้ในการทดลองต่อไป

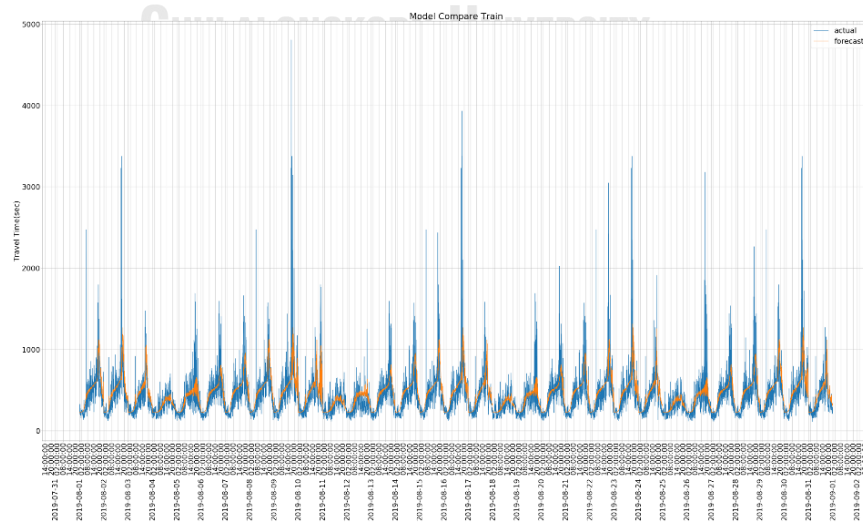
คาดว่าที่ผลการทดลองเป็นเช่นนี้เนื่องจากที่ขนาดของ Timestep เท่ากับ 5 นาทีจะพบว่าระยะเวลาเดินทางมีความเหวี่ยงของข้อมูลมากกว่าแบบ 10 นาทีที่มีรูปแบบของข้อมูลที่ราบรื่นและมีความเป็นรูปแบบมากกว่า



a) Epoch = 50

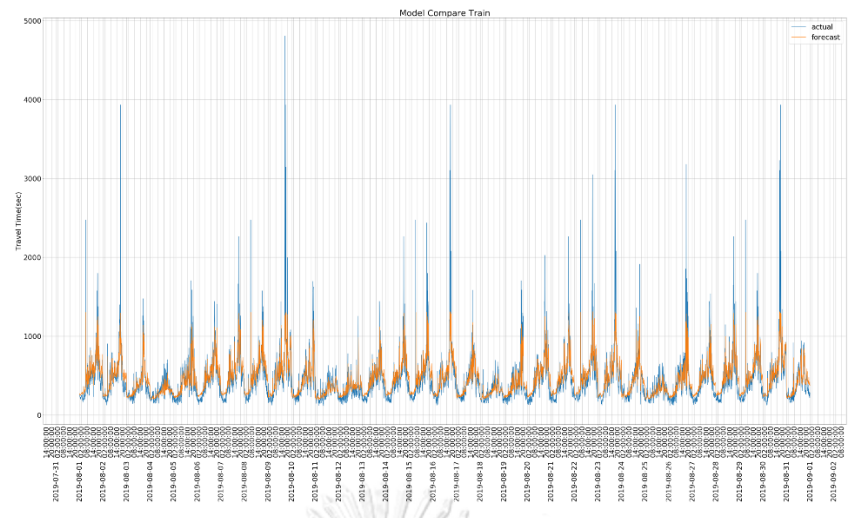


b) Epoch = 100

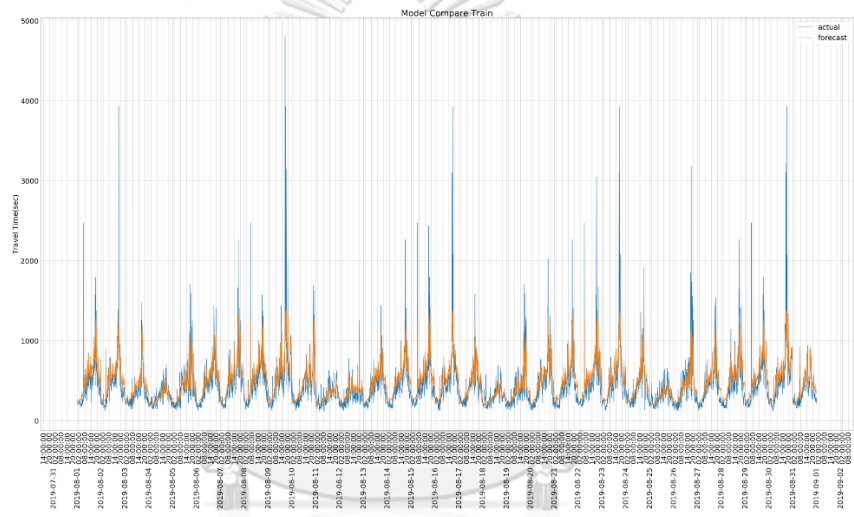


c) Epoch = 200

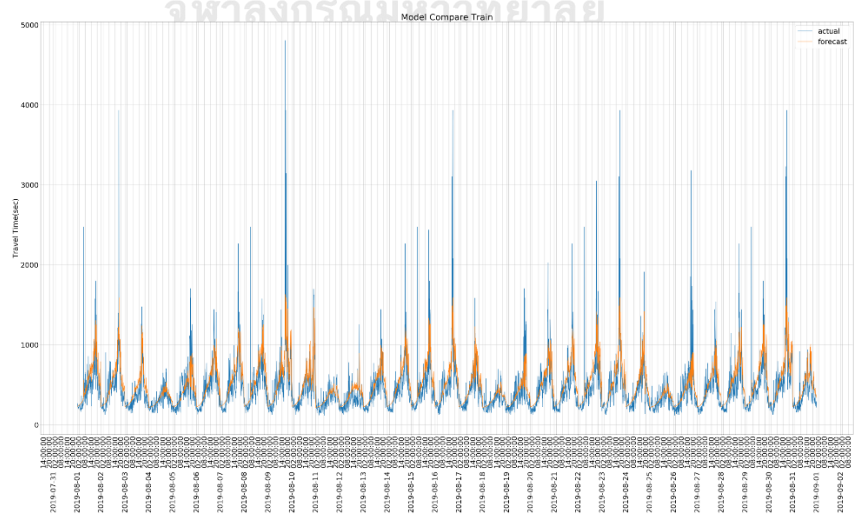
ภาพที่ 35 เปรียบเทียบผลจริงกับผลพยากรณ์ด้วย LSTM-DNN ขนาด Timestep เท่ากับ 5 นาที



a) Epoch = 50



b) Epoch = 100



c) Epoch = 200

ภาพที่ 36 เปรียบเทียบผลจริงกับผลจากการพยากรณ์ด้วย LSTM-DNN ด้วยขนาด Timestep เท่ากับ 10 นาที

## บทที่ 5

### สรุปผลการวิจัยและข้อเสนอแนะ

#### 5.1 สรุปผลการวิจัย

งานวิจัยนี้เป็นการศึกษาวิธีการพยากรณ์ระยะเวลาเดินทางด้วยโมเดลทางอนุกรมเวลาที่นำผลจากช่วงเวลาปัจจุบันมาใช้ในการพยากรณ์ระยะเวลาเดินทางในอนาคต Timestep ถัดไป โดยเปรียบเทียบผลจากโมเดลต่างๆ , ขนาดของ Timestep ที่ต่างกัน, การกำหนดค่าโมเดลที่แตกต่างกัน รวมไปถึงการนำโมเดลไปใช้ในสถานการณ์จริงต่าง ๆ เช่น การนำโมเดลที่สร้างด้วยข้อมูลจากถนนช่วงหนึ่งไปประยุกต์ใช้กับเส้นทางอื่น

ผลการวิจัยพบว่าโมเดล LSTM-DNN ให้ค่าที่ถูกต้องมากที่สุดเมื่อเทียบกับโมเดล RNN, GRU, LSTM-RNN และ LSTM-GRU แต่โมเดล RNN สามารถสอนและสร้างโมเดลได้เร็วที่สุดขณะที่ความถูกต้องเป็นอันดับ 3 เมื่อเทียบกับโมเดลอื่น ดังนั้นหากต้องการโมเดลที่เทรนได้เร็วที่สุดแต่อาจจะไม่ต้องถูกต้องมากที่สุด โมเดล RNN ก็ยังสามารถนำมาใช้งานได้

จากการทดลองโดยหาโมเดลและขนาดของ Sliding Window ที่เหมาะสมสำหรับชุดข้อมูลที่จะใช้เพื่อพยากรณ์พบว่าโมเดล LSTM-DNN ที่ใช้ข้อมูล Sliding Window เท่ากับ 10 นาทีที่เหมาะสมที่สุดเนื่องจากการใช้ Sliding Window ที่ 30 นาทีและ 60 นาทีข้อมูลมีความละเอียดลดลงส่งผลให้ผลที่ได้จากการพยากรณ์มีค่าละเอียดลดลงเช่นกัน และเมื่อพิจารณาด้วยค่า RMSE พบว่าการใช้ Sliding Window ที่ 10 นาทีเหมาะสมที่สุด

นอกจากนี้การทดลองยังพิจารณาถึงระยะเวลาที่ใช้ในการสอนและทดสอบสำหรับโมเดลต่างๆ ที่ Sliding Window ที่แตกต่างกัน พบว่าระยะเวลาที่ใช้ในการสอนและทดสอบของข้อมูลในแต่ละขนาดของ Sliding Window ที่แตกต่างกันใช้ระยะเวลาแทบไม่แตกต่างกัน แต่สำหรับโมเดลเดียวกันแต่จำนวน Epoch ที่แตกต่างกันผลลัพธ์ที่ได้พบจำนวน Epoch กับ ระยะเวลาในการสอนและทดสอบเป็นอัตราส่วนเชิงเส้น ขณะเดียวกันค่าความถูกต้องที่ได้ไม่ได้สัมพันธ์แบบเชิงเส้น ดังนั้นการเลือกใช้งาน Epoch ขึ้นอยู่กับลักษณะของการนำไปประยุกต์ใช้ว่าระยะเวลาสอนและทดสอบเท่าไรถึงเหมาะสมที่สุด

ขณะเดียวกันในการทดลองที่ 7 ทดลองเช่นเดียวกับการทดลองที่ 2 แต่เปลี่ยนจาก Sliding Window เป็น Timestep Size โดยพบว่าขนาดของ Timestep ถ้าพิจารณาจาก RSME พบว่า 5 นาทีให้ค่าที่ผิดพลาดน้อยที่สุด ที่เป็นเช่นนี้เนื่องจากจำนวนของข้อมูลที่มีมากกว่าชุดข้อมูลอื่นทำให้

เปรียบเทียบแล้วได้ค่าน้อยกว่า แต่เมื่อพิจารณาในรายละเอียดพบว่าขนาดที่เหมาะสมต่อการนำมาสร้างโมเดลเพื่อพยากรณ์ระยะเวลาเดินทางคือ 10 นาทีเมื่อเทียบกับ 5 นาทีเพราะข้อมูล 5 นาทีมีความกว้างมากกว่า 10 นาทีและมีบางช่วงที่ข้อมูล 5 นาทีพยากรณ์ได้สูงกว่าความเป็นจริงมาก และขนาดของ Timestep ที่เท่ากับ 30 นาทีและ 60 นาทีเป็นช่วงเวลาที่ห่างกันเกินไป และผลที่ได้จากการพยากรณ์ไม่สามารถนำมาใช้งานได้จริง ข้อสังเกตการกำหนดขนาดของ Timestep ส่งผลต่อจำนวนข้อมูลด้วยเช่นกัน การกำหนด Timestep ที่กว้างจะทำให้มีข้อมูลเพื่อใช้ในการสอนและทดสอบที่น้อยลงตาม

การทดลองนำโมเดลที่ได้จากข้อมูลของช่วงถนนหนึ่งไปใช้พยากรณ์ข้อมูลของช่วงถนนอื่นๆ พบว่าโมเดลที่สร้างสามารถใช้เพื่อพยากรณ์ช่วงถนนอื่นที่มีรูปแบบข้อมูลที่คล้ายคลึงกันได้ เช่นช่วงถนนที่มีลักษณะตรงยาวเช่นกัน แต่สำหรับถนนที่มีข้อมูลที่ต่างกันมาก เช่นช่วงถนนที่มีระยะสั้นกว่าช่วงถนนที่นำมาใช้สอนโมเดลมากจะพบว่ามีความผิดพลาดที่ค่าพยากรณ์สูงกว่าค่าจริง ทั้งนี้เพราะช่วงเวลาดังกล่าวของข้อมูลที่ใช้สอนมีแนวโน้มสูงกว่าช่วงถนนที่พยากรณ์ ดังนั้นหากต้องการจะนำโมเดลที่สอนด้วยข้อมูลถนนช่วงหนึ่งไปพยากรณ์ถนนช่วงอื่นก็สามารถทำได้ แต่ว่าหากต้องการได้ผลที่ดีที่สุดคือการสร้างโมเดลสำหรับช่วงถนนนั้นๆ

การแบ่งข้อมูลก็มีผลต่อการพยากรณ์เช่นกัน โดยพบว่าการพยากรณ์ด้วยข้อมูลที่มีการพิจารณาถึงทิศทางการเดินทางให้ผลลัพธ์ที่ดีกว่าเมื่อเทียบกับการไม่พิจารณาการเดินทาง เนื่องจากทิศทางการเดินทาง (เลี้ยวซ้าย, เลี้ยวขวา, กลับรถ และขับตรง) ส่งผลโดยตรงต่อระยะเวลาเดินทาง และการพิจารณาทิศทางการเดินทางจะช่วยลดข้อมูลที่อาจจะทำให้เกิดความผิดพลาดจากทริปที่มีการขับที่ผิดปกติก่อนนำข้อมูลสอนโมเดล เสมือนการกรองข้อมูลที่ไม่เกี่ยวข้อง หรือผิดปกติดออกจากข้อมูลทั้งหมด

นอกจากนี้ การแบ่งข้อมูลออกเป็นช่วงเล็กๆ แล้วนำผลพยากรณ์มารวมกัน ได้ผลลัพธ์ที่ดีกว่าการพยากรณ์ช่วงยาวๆ ช่วงเดียว เนื่องจากการแบ่งข้อมูลช่วงเล็กๆ บางช่วงสั้นมากส่งผลให้ระยะเวลาเดินทางในช่วงเหล่านี้น้อยมากทุกช่วงเวลา เมื่อรวมกันหลายช่วงถนนสั้นๆ จึงส่งผลให้โดยรวมค่าความผิดพลาดของการพยากรณ์จึงน้อยกว่าเมื่อนำข้อมูลช่วงถนนที่ยาวมาพยากรณ์ แต่การพยากรณ์ด้วยวิธีการแบ่งข้อมูลออกเป็นช่วงเล็กๆ มีความซับซ้อนทั้งการเตรียมข้อมูลสำหรับโมเดล และการเตรียมโมเดลสำหรับช่วงถนนสั้นๆ ดังนั้นถึงแม้ว่าจะมีความถูกต้องมากกว่าแต่ความซับซ้อนของกระบวนการก็มากขึ้นเช่นกัน

ท้ายที่สุดเมื่อพิจารณาเป็นรายชั่วโมงพบว่าช่วงเย็นของวัน (4 โมงเย็น ถึง 1 ทุ่ม) ให้พยากรณ์ ได้ผิดพลาดสูงกว่าช่วงอื่นของวันเนื่องจากผลของความแปรปรวนของระยะเวลาเดินทางที่สูงกว่า ช่วงเวลาอื่นส่งผลให้โมเดลไม่สามารถพยากรณ์ได้ดีเท่ากับช่วงเวลาอื่น นอกจากนี้ยังพบว่าวันเสาร์ และอาทิตย์โมเดลสามารถพยากรณ์ได้ถูกต้องมากกว่าวันธรรมดา เนื่องจากวันเสาร์อาทิตย์ระยะเวลา เดินทางตลอดทั้งวันมีความแปรปรวนน้อยกว่าวันธรรมดา

ผลลัพธ์ที่ได้จะทำให้ผู้ที่สนใจการพยากรณ์ระยะเวลาเดินทางนำไปใช้เพื่อพยากรณ์และ ปรับแต่งให้เหมาะสมกับการนำไปปรับใช้กับสถานการณ์จริง และเป็นอีกทางเลือกที่สามารถใช้ พยากรณ์ระยะเวลาเดินทางได้ แต่กระนั้นผลการพยากรณ์ที่ได้จากโมเดลยังคงพบว่าไม่สามารถ พยากรณ์บางช่วงเวลาที่มีความต่างมากๆ ได้ดังจะเห็นได้จากภาพที่ 33 a ที่ผลการพยากรณ์ไม่เกิน 2000 วินาที ที่เป็นเช่นคาดว่าเป็นเพราะข้อมูลในลักษณะดังกล่าวมีเพียงเล็กน้อยเมื่อเทียบกับจำนวน ข้อมูลทั้งหมดทำให้โมเดลที่เทรนไม่สามารถพยากรณ์ค่ามากกว่านั้นได้

ข้อจำกัดเรื่องจำนวนข้อมูลเป็นผลทำให้โมเดลสามารถพยากรณ์ได้ถูกต้องหรือไม่ ถ้าหากมี ข้อมูลที่เพิ่มมากขึ้นย่อมส่งผลให้ค่าที่ได้จากการพยากรณ์มีความถูกต้องมากขึ้นเช่นกัน รวมไปถึงมี จำนวนทรูปมากพอที่จะสามารถใช้เพื่อเป็นตัวแทนถนนช่วงนั้นได้

การพัฒนาต่อไปในอนาคตคือการสามารถนำโมเดลที่ได้พัฒนาเป็นแอปพลิเคชันจริงใน สถานการณ์จริง และการนำกระบวนการพัฒนาโมเดลไปพัฒนาโมเดลสำหรับช่วงถนนอื่นๆ ต่อไป ในอนาคต

## 5.2 ข้อจำกัด

โมเดลนี้เป็นโมเดลที่พัฒนาภายใต้ข้อมูลที่เป็นลักษณะ GPS Trace ของรถแท็กซี่ใน บริเวณถนนพระราม 4 เท่านั้น ดังนั้นการตั้งค่าดังที่แสดงในงานวิจัยนี้อาจจะไม่ได้ผลลัพธ์แบบ เดียวกันเมื่อนำไปใช้กับข้อมูลในลักษณะอื่น ซึ่งหากนำไปปรับใช้จำเป็นต้องมีการปรับเปลี่ยนการตั้งค่า โมเดลก่อนนำไปพัฒนาต่อไป

โมเดลนี้ใช้ระยะเวลาเดินทางปัจจุบันมาเพื่อพยากรณ์ระยะเวลาเดินทางถัดไป ส่งผลให้หาก ต้องการจะพยากรณ์ระยะเวลาเดินทางล่วงหน้าหลายช่วงเวลาจะส่งผลให้เกินค่าความผิดพลาดได้สูง อันเนื่องมาจากเมื่อนำระยะเวลาเดินทางที่ได้จากการพยากรณ์มาเป็น feature สำหรับโมเดลถัดไปจะ ส่งผลให้ความผิดพลาดสะสมจากการนำ feature ที่มีความคลาดเคลื่อนมาพยากรณ์

โมเดลนี้ไม่สามารถพยากรณ์ค่าระยะเวลาเดินทางที่มีความต่างของค่ามากๆ และเกิดขึ้นจำนวนไม่มาก ดังนั้นจะปรากฏบางช่วงเวลาของวันที่โมเดลจะพยากรณ์ระยะเวลาเดินทางได้น้อยกว่าค่าความเป็นจริง

### 5.3 แนวทางการวิจัยในอนาคต

ปรับปรุงวิธีการพยากรณ์ให้สามารถพยากรณ์ให้หลาย Timestep มากขึ้นเนื่องจากเมื่อนำไปประยุกต์ใช้จริงไม่สามารถปรับปรุงข้อมูลต้นทางทุก Timestep ได้ทำให้หากไม่สามารถพยากรณ์หลาย Timestep ล่วงหน้าได้จะส่งผลให้สามารถไปประยุกต์ใช้งานจริงได้ยาก

นอกจากนี้ ควรจะมีการนำข้อมูลจากหลายแหล่งที่ส่งผลกับระยะเวลาเดินทางเข้ามาประกอบในโมเดลด้วยเช่นกันเพื่อให้เกิดความรอบด้านของข้อมูล และโมเดลมีความถูกต้องมากยิ่งขึ้น เช่น ข้อมูลพยากรณ์อากาศ ข้อมูลปริมาณรถยนต์ เป็นต้น

### 5.4 ข้อเสนอแนะ

งานวิจัยนี้มีข้อจำกัดด้านข้อมูลที่มีเพียง 1 เดือนเท่านั้นและข้อมูลก็มีน้อยทำให้มีข้อมูลการเดินทางไม่ครบทุกช่วง Timestep จึงส่งผลให้บาง Timestep ได้ค่าระยะเวลาเดินทางที่เป็นการประมาณการณ์จากในอดีต หรือจากระยะเวลาที่ติดกัน ทำให้เกิดความคลาดเคลื่อน นอกจากนี้พฤติกรรมคนขับรถบางครั้งก็เลือกจะเลี้ยวซ้ายผ่านตลอดและกลับรถแทนการรอสัญญาณไฟแล้วขับตรงต่อไปทำให้ระยะเวลาที่ได้จากการพยากรณ์มีความคลาดเคลื่อนและไม่สัมพันธ์กับระยะเวลาใกล้เคียงกัน ซึ่งส่งผลต่อความถูกต้องของโมเดล ดังนั้นหากต้องการให้โมเดลมีความถูกต้องมากขึ้นจำเป็นต้องปรับปรุงข้อมูลให้มีความสมบูรณ์ขึ้นเช่นกัน

## บรรณานุกรม

- Araujo, A. C. d., & Etemad, A. (2019). *Deep Neural Networks for Predicting Vehicle Travel Times*. Paper presented at the 2019 IEEE SENSORS, Montreal, QC, Canada, Canada. <https://ieeexplore.ieee.org/document/8956878>
- Bai, M., Lin, Y., Ma, M., & Wang, P. (2018). *Travel-Time Prediction Methods: A Review*. Paper presented at the Smart Computing and Communication. SmartCom 2018, Tokyo, Japan. [https://link.springer.com/chapter/10.1007/978-3-030-05755-8\\_7](https://link.springer.com/chapter/10.1007/978-3-030-05755-8_7)
- Bhaskar, A., Qu, M., & Chung, E. (2015). *Bluetooth Vehicle Trajectory by Fusing Bluetooth and Loops: Motorway Travel Time Statistics*. Paper presented at the IEEE Transactions on Intelligent Transportation Systems. <https://ieeexplore.ieee.org/document/6841057>
- Duan, Y., Lv, Y., & Wang, F.-Y. (2016). *Travel time prediction with LSTM neural network*. Paper presented at the 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC), Rio de Janeiro, Brazil. <https://ieeexplore.ieee.org/document/7795686/>
- Erdelić, T., Ravlić, M., & Carić, T. (2016). *Travel time prediction using speed profiles for road network of Croatia*. Paper presented at the 58th International Symposium ELMAR-2016, Zadar, Croatia.
- Essien, A., Petrounias, I., Sampaio, P., & Sampaio, S. (2019). *Improving Urban Traffic Speed Prediction Using Data Source Fusion and Deep Learning*. Paper presented at the 2019 IEEE International Conference on Big Data and Smart Computing (BigComp), Kyoto, Japan, Japan.
- Ganti, R., Srivatsa, M., & Abdelzaher, T. (2014). *On Limits of Travel Time Predictions: Insights from a New York City Case Study*. Paper presented at the 2014 IEEE 34th International Conference on Distributed Computing Systems, Madrid, Spain. <https://ieeexplore.ieee.org/document/6888893/authors#authors>
- Gao, M., Zhu, T., Wan, X., & Wang, Q. (2013). *Analysis of Travel Time Patterns in Urban Using Taxi GPS Data*. Paper presented at the 2013 IEEE International Conference on Green Computing and Communications and IEEE Internet of Things and IEEE



Cyber, Physical and Social Computing, Beijing.

<https://ieeexplore.ieee.org/document/6682115/>

Kankanamge, K. D., Witharanage, Y. R., Withanage, C. S., Hansini, M., Lakmal, D., & Thayasivam, U. (2019). *Taxi Trip Travel Time Prediction with Isolated XGBoost Regression*. Paper presented at the 2019 Moratuwa Engineering Research Conference (MERCon), Moratuwa, Sri Lanka, Sri Lanka.

<https://ieeexplore.ieee.org/document/8818915>

Kosolsombat, S., & Saraubon, K. (2018, 27 December 2018). *A Review of the Prediction Method for Intelligent Transport System*. Paper presented at the 2018 18th International Symposium on Communications and Information Technologies (ISCIT), Bangkok, Thailand.

Li, L., & Jiang, X. (2017). *Predicting the Travel Time in Using Recurrent Neural Networks: A Case Study of Fuzhou*. Paper presented at the Advances in Smart Vehicular Technology, Transportation, Communication and Applications. VTCA 2017, Kaohsiung, Taiwan. [https://link.springer.com/chapter/10.1007/978-3-319-70730-3\\_4](https://link.springer.com/chapter/10.1007/978-3-319-70730-3_4)

Liu, Y., Wang, Y., Yang, X., & Zhang, L. (2017). *Short-term travel time prediction by deep learning: A comparison of different LSTM-DNN models*. Paper presented at the 2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC), Yokohama, Japan. <https://ieeexplore.ieee.org/document/8317886>

Narayanan, A. K., Pranesh, C., Nagavarapu, S. C., Kumar, B. A., & Dauwels, J. (2019). *Data-driven Models for Short-term Travel Time Prediction*. Paper presented at the 2019 IEEE Intelligent Transportation Systems Conference (ITSC), Auckland, NZ.

Wei, W., Jia, X., Liu, Y., & Yu, X. (2018). *Travel Time Forecasting with Combination of Spatial-Temporal and Time Shifting Correlation in CNN-LSTM Neural Network*. Paper presented at the Web and Big Data. APWeb-WAIM 2018, Macau, China.

[https://link.springer.com/chapter/10.1007/978-3-319-96890-2\\_25](https://link.springer.com/chapter/10.1007/978-3-319-96890-2_25)

Zhu, Y., He, Z., & Sun, W. (2019). *Network-Wide Link Travel Time Inference Using Trip-Based Data From Automatic Vehicle Identification Detectors*. Paper presented at the IEEE Transactions on Intelligent Transportation Systems.

<https://ieeexplore.ieee.org/document/8734127>



จุฬาลงกรณ์มหาวิทยาลัย  
**CHULALONGKORN UNIVERSITY**



จุฬาลงกรณ์มหาวิทยาลัย  
**CHULALONGKORN UNIVERSITY**

ชื่อ-สกุล	ชาญวิทย์ จักรธีรังกูร
วัน เดือน ปี เกิด	9 พฤษภาคม 2532
สถานที่เกิด	กรุงเทพมหานคร
วุฒิการศึกษา	บริหารธุรกิจมหาบัณฑิต
ที่อยู่ปัจจุบัน	506/416 คอนโด KnightsBridge Collage รามคำแหง ถนนรามคำแหง แขวงหัวหมาก เขตบางกะปิ กรุงเทพฯ 10240



จุฬาลงกรณ์มหาวิทยาลัย  
CHULALONGKORN UNIVERSITY