การทำนายราคาที่พักบน Airbnb โดยการแปลงข้อมูลเชิงกลุ่มให้เป็นข้อมูลเชิงปริมาณ

นิสิทรา บุญเรือง[1*], นภา แซ่เบ๊[2]

บทคัดย่อ

งานวิจัยนี้มีวัตถุประสงค์เพื่อสร้างแบบจำลองการทำนายราคาที่พักโดยใช้ชุดข้อมูลที่พัก Airbnb ในชุดข้อมูลของกรุงเทพมหานคร จำนวนข้อมูล 20,823 แถว 18 คอลัมน์ จากเว็บไซต์ http://insideairbnb.com// โดยศึกษาเปรียบเทียบการแปลงข้อมูลเชิงกลุ่มให้เป็นข้อมูลเชิงปริมาณโดยใช้การเข้ารหัสแบบ Entity Embedding และ One-hot Encoding สำหรับตัวแปรเชิงกลุ่มที่มีความหลากหลายสูงผ่านแบบจำลอง 4 รูปแบบ ได้แก่ Neural Network, Random Forest, K-Nearest Neighbors และ XGBoost ผลการทดลองแสดงให้เห็นว่าแบบจำลอง Random Forest ให้ผลการดำเนินการที่ดีที่สุดสำหรับการใช้ Entity Embedding โดยมีค่า RMSE 832.56, MAE 587.56 และ R-squared 0.25 ในขณะที่แบบจำลอง XGBoost ให้ผลลัพธ์ที่ดีที่สุดสำหรับการใช้ One-hot Encoding โดยมีค่า RMSE 787.22, MAE 544.42 และ R-squared 0.37 แม้ว่า One-hot Encoding จะให้ผลลัพธ์การทำนายที่ดีกว่าแต่ก็ยังมีค่าความคลาดเคลื่อนสูง อาจเป็นผลจากข้อมูลในชุดข้อมูลยังไม่เพียงพอที่จะสร้างแบบจำลองการทำนายราคาที่พักได้อย่างมีประสิทธิภาพ ดังนั้นการพิจารณาปัจจัยและตัวแปรอื่น เช่น สิ่งอำนวยความสะดวกในที่พัก หรือการออกแบบภายใน อาจช่วยเพิ่มประสิทธิภาพของแบบจำลองการทำนายราคา การวิจัยเพิ่มเติมในประเด็นเหล่านี้อาจนำไปสู่แบบจำลองการทำนายราคาที่พักที่แม่นยำและน่าเชื่อถือมากขึ้น นอกจากนี้ผลการทดลองโดยการนำข้อมูลที่ผ่านการเข้ารหัสโดยใช้เทคนิค Entity Embedding มาทำการแสดงผลยังให้แสดงให้เห็นถึงความสัมพันธ์ระหว่างกลุ่มต่างๆ ซึ่งเป็นอีกทางเลือกในการสำรวจและการวิเคราะห์ข้อมูลเพิ่มเติมได้

คำสำคัญ : Airbnb, Entity Embedding, One-hot Encoding

---

[1] หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาวิทยาการข้อมูล คณะวิทยาศาสตร์ มหาวิทยาลัยศรีนครินทรวิโรฒ กรุงเทพฯ 10110

[2] คณะวิทยาศาสตร์ มหาวิทยาลัยศรีนครินทรวิโรฒ กรุงเทพฯ 10110

* Corresponding author: Tel.: 095-4527067 E-mail address: Nisitra.sye@g.swu.ac.th

PREDICTING ACCOMMODATION PRICES ON AIRBNB USING ENTITY EMBEDDING

Nisitra Bunruang[1*], Napa Sae-bae[2]

Abstract

This research aims to develop a predictive model for Airbnb accommodation prices using a dataset of 20,823 rows and 18 columns from the Bangkok metropolitan area, obtained from the website http://insideairbnb.com//. The study compares the transformation of categorical data into numerical data using Entity Embedding and One-hot Encoding for high-diversity categorical variables across four models: Neural Network, Random Forest, K-Nearest Neighbors, and XGBoost. The experimental results demonstrate that Random Forest with Entity Embedding achieved the most favorable performance metrics. It achieved RMSE of 832.56, MAE of 587.63, and R-squared of 0.25. Conversely, XGBoost demonstrated superior performance when utilizing One-hot Encoding. This model yielded RMSE of 787.22, MAE of 544.42, and R-squared of 0.37. Even though One-hot Encoding slightly better predictions, it exhibited higher error rates associated with this technique. This could be attributed to the insufficient data in the dataset to effectively build a predictive model for accommodation prices. Therefore, considering additional factors and variables such as accommodation amenities or interior design, could potentially enhance the performance of the price prediction model. Further research on these aspects promises to the accommodation price prediction models with higher accuracy and reliability. Moreover, the application of Entity Embedding visualization techniques reveals the relationship among various groups, opening up new avenues for data exploration and analysis.

Keywords : Airbnb, Entity Embedding, One-hot Encoding

[1] Data Science, Faculty of Science, Srinakharinwirot University, Bangkok, 10110, Thailand

[2] Faculty of Science, Srinakharinwirot University, Bangkok, 10110, Thailand

* Corresponding author: Tel.: 095-4527067 E-mail address: Nisitra.sye@g.swu.ac.th

บทนำ

ปัจจุบันการหาที่พักแรมมีความสะดวกและรวดเร็วโดยใช้เว็บไซต์หรือแอพพลิเคชันการจองที่พักออนไลน์เพื่อค้นหาที่พักที่เป็นจุดหมายปลายทาง นอกจากนี้ยังสามารถกรองผลการค้นหาตามความต้องการเฉพาะ เช่น ราคา สิ่งอำนวยความสะดวก หรือคะแนนรีวิวจากผู้เข้าพักก่อนหน้า ทั้งยังกำหนดได้ว่าต้องการที่พักแบบไหน เช่น โรงแรม วิลล่า บ้านพักของผู้คนในท้องถิ่น หรือคอนโดมิเนียม ซึ่งสามารถเลือกให้ตรงกับความต้องการและงบประมาณของผู้เข้าพัก แต่ในช่วงไม่กี่ปีที่ผ่านมา Airbnb เป็นจุดเปลี่ยนครั้งสำคัญ เพราะเป็นจุดเริ่มต้นที่ให้เจ้าของบ้านประกาศปล่อยเช่าอะพาร์ตเมนต์ทั้งหลังหรือห้องว่างภายในบ้านภายในแพลตฟอร์ม ซึ่งต่างจากห้องพักในโรงแรมมาตรฐานในเรื่องรูปแบบของที่พัก ที่ตั้ง สิ่งอำนวยความสะดวก ดังนั้นงานวิจัยนี้จึงศึกษาวิธีการทำนายราคาที่พักบน Airbnb โดยใช้วิธีการแปลงข้อมูลเชิงกลุ่มให้เป็นข้อมูลเชิงปริมาณ เพื่อนำผลลัพธ์ที่ได้มาทำนายราคาโดยอ้างอิงจากราคาจริง

ในงานวิจัยนี้มีแนวทางการแปลงข้อมูลโดยการใช้วิธี Entity Embedding และ One-hot Encoding ร่วมกับการทำ StandardScaler และ Logarithm เพื่อตรวจสอบว่าในแบบจำลองประเภทใดจะให้ประสิทธิภาพของการทำนายราคาได้ดีที่สุด โดยการทำ Embedding Feature นั้นจะเป็นการแปลงข้อมูลที่มีลักษณะเป็นข้อมูลหมวดหมู่ (categorical data) ให้เป็นรูปแบบของเวกเตอร์ (vector format) ซึ่งเป็นการแทนที่ข้อมูลที่มีลักษณะเป็นข้อความหรือตัวเลขอื่นๆ ให้อยู่ในรูปของตัวเลขที่สามารถนำเข้าแบบจำลองได้โดยง่าย ส่วน One-hot Encoding นั้นก็เป็นเทคนิคที่ใช้ในการแปลงข้อมูลหมวดหมู่ให้เป็นรูปแบบของเวกเตอร์เช่นกัน แตกต่างกันที่แต่ละค่าของข้อมูลหมวดหมู่จะถูกแทนที่ด้วยเวกเตอร์ที่มีค่า 0 และ 1 ซึ่งช่วยให้แบบจำลองสามารถเข้าใจและเรียนรู้จากข้อมูลหมวดหมู่ได้อย่างมีประสิทธิภาพ

วิธีดำเนินการ

ขั้นตอนที่ 1 : ชุดข้อมูลที่ใช้ในการศึกษา

ผู้วิจัยนำข้อมูลเกี่ยวกับราคาที่พักบน Airbnb ซึ่งเป็นชุดข้อมูลสาธารณะจากเว็บไซต์ http://insideairbnb.com/get-the-data/ ที่เก็บรวบรวมข้อมูลต่างๆ เกี่ยวกับที่พักบน Airbnb โดยเลือกใช้ชุดข้อมูลของกรุงเทพมหานคร ชุดข้อมูลเป็นข้อมูลรายไตรมาสในช่วง 12 เดือน (สถานะสิ้นสุด 25 ธันวาคม 2565) จำนวนข้อมูลทั้งหมด 20,823 แถว 18 คอลัมน์ ดังตาราง 1

ตาราง 1 แสดงตัวแปรของข้อมูล Airbnb

Variable	Description
id	หมายเลยเลขประจำตัวที่ไม่ซ้ำในรายการ
name	ชื่อที่พัก
host_id	หมายเลขเลขประจำตัวที่ไม่ซ้ำสำหรับเจ้าของที่พักหรือผู้ใช้
host_name	ชื่อของเจ้าของที่พัก
neighbourhood_group	กลุ่มย่านที่ตั้งของที่พัก
neighborhood	ย่านที่พัก
latitude	ละติจูดตามพิกัดทางภูมิศาสตร์

ตาราง 1 (ต่อ)

Variable	Description
longitude	ลองติจูดตามพิกัดทางภูมิศาสตร์
room_type	ประเภทของที่พัก
price	ราคาของที่พัก
minimum_nights	จำนวนคืนขั้นต่ำสำหรับการเข้าพัก
number_of_reviews	จำนวนรีวิวทั้งหมด
last_review	วันที่ล่าสุดที่ผู้เข้าพักรีวิวให้กับที่พัก
reviews_per_month	จำนวนรีวิวเฉลี่ยต่อเดือน
calculated_host_listing_count	จำนวนรายการที่เจ้าของที่พักมีทั้งหมด
availability_365	ความพร้อมในการจองสำหรับ 365 วันถัดไป
number_of_reviews_ltm	จำนวนรีวิวใน 12 เดือนที่ผ่านมา
license	หมายเลขใบอนุญาต

ขั้นตอนที่ 2 : การจัดการข้อมูล

การจัดการข้อมูลเริ่มต้นด้วยการนำเข้าโมดูลสำคัญสำหรับการสร้างแบบจำลอง ต่อมานำเข้าข้อมูลและข้อมูลที่ใช้สำหรับสร้างแบบจำลอง ถัดมามีการตัดคอลัมน์ที่ไม่มีความจำเป็นสำหรับการทำนายราคาในแบบจำลอง จึงได้มีการเลือกเพียง 8 คอลัมน์ ได้แก่ neighbourhood, room_type, price, minimum_nights, number_of_reviews, reviews_per_month, calculated_host_listings_count และ availability_365 คงเหลือจำนวนข้อมูลทั้งหมด 20,823 แถว 8 คอลัมน์

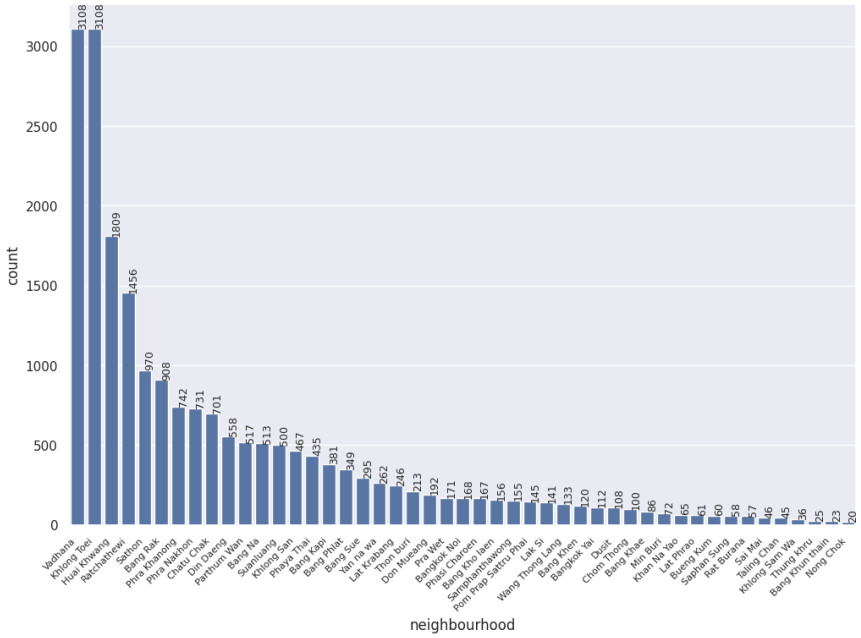
จากการสำรวจข้อมูลพบข้อมูลที่หายไปในคอลัมน์ reviews_per_month ผู้วิจัยได้ตั้งสมมติฐานว่าอาจจะเกิดขึ้นเนื่องจากไม่มีรีวิวสำหรับรายการนั้นๆ และเพื่อยืนยันสมมติฐานจึงนำมาตรวจสอบกับคอลัมน์ number_of_reviews พบว่ามีค่าเป็น 0 นั่นแสดงว่ารายการทั้งหมดที่มีข้อมูลที่หายไปไม่มีการรีวิวจากผู้เข้าพัก ดังนั้นจึงต้องเติมข้อมูลที่หายไปให้มีค่าเท่ากับ 0 ส่วนข้อมูลที่มีการรายการซ้ำได้ทำการลบข้อมูลที่มีรายการซ้ำทั้งหมดออกไป

การจัดการกับข้อมูลส่วนเกิน (outliers) จากการสำรวจข้อมูลในคอลัมน์ price พบว่าช่วงราคาของที่พักอยู่ระหว่าง 40-1,000,000 บาท ดังตาราง 2

ตาราง 2 การกระจายตัวของข้อมูลในคอลัมน์ price

Price Range	Frequency
40 - 299	23
300 - 5,000	19,450
5,001 - 10,000	933
10,001 - 100,000	395
100,001 - 500,000	17
500,001 - 1,000,000	5

จากตาราง 2 พบว่าข้อมูลมีการกระจายตัวอยู่ในช่วงกว้างมากและหากมีการเก็บค่าทุกค่าในคอลัมน์ price ไว้จะทำให้แบบจำลองที่ได้มีจำนวนของ outlier อยู่เป็นจำนวนมาก จึงได้มีการตัดข้อมูลบางส่วนที่ต่ำและสูงเกินความเป็นจริงออกและเลือกเก็บไว้เฉพาะช่วงราคาของที่พักที่อยู่ระหว่าง 300-5,000 เท่านั้น ส่วนข้อมูลในคอลัมน์ neighbourhood พบว่าค่าที่ไม่ซ้ำกันในคอลัมน์ (unique values) จำนวน 50 ย่าน และพบว่ามีบางย่านที่มีค่าที่น้อยเกินไปที่อาจจะทำให้เกิดเป็น outlier ได้ จึงลบย่านที่มีค่าที่นับได้น้อยว่า 20 ออกไป และพบว่าค่าที่ไม่ซ้ำกันในคอลัมน์มีจำนวนคงเหลือ 47 ย่าน ทำให้มีจำนวนคงเหลือของรายการทั้งหมด 18,532 รายการ 8 คอลัมน์



ภาพประกอบ 1 กราฟแสดงจำนวนข้อมูลในคอลัมน์ neighbourhood

การจัดการกับตัวแปรประเภทหมวดหมู่ (categorical variables) โดยในชุดข้อมูลที่นำมาใช้ในงานวิจัยนี้มีตัวแปรที่มีค่าความหลากหลายสูง (high cardinality) คือตัวแปร neighbourhood จึงมีการนำเทคนิคการเข้ารหัสแบบ Entity Embedding และ One-hot Encoding มาใช้ในการจัดการกับตัวแปรนี้

การจัดการกับตัวแปรที่มีค่าความหลากหลายสูงทำโดยการเข้ารหัสแบบ Entity Embedding ใช้ไลบรารี่ Fastai โดยที่ไลบรารี่นี้จะมีการกำหนดขนาด (dimension) ของ Embedding Feature อัตโนมัติ โดยใช้ค่าของตัวแปรที่ไม่ซ้ำกันในตัวแปรนั้นๆ ส่วนการจัดการกับตัวแปรที่มีค่าความหลากหลายสูงโดยการเข้ารหัสแบบ One-hot Encoding จะมีการแทนค่าแต่ละค่าด้วยตัวแปรใหม่ที่เป็น Binary โดยมีค่า 0 ถึง 1

ขั้นตอนที่ 3 : การสร้างแบบจำลอง

ก่อนการสร้างแบบจำลองมีการแบ่งข้อมูลออกเป็นชุดข้อมูลฝึก (train data) 80% และชุดข้อมูลทดสอบ (test data) 20% โดยในตัวแปรตาม (dependent variable) มีการปรับการกระจายตัวของข้อมูลด้วยวิธี StandardScaler กับคอลัมน์ที่เป็นตัวแปรเชิงปริมาณทั้งหมด และในตัวแปรอิสระ (independent variable) มีการปรับการกระจายตัวของข้อมูล 2 วิธี คือ StandardScaler และ Logarithm ในคอลัมน์ price และเพื่อวัดประสิทธิภาพของแบบจำลองมีการทำการย้อนกลับของข้อมูล (inverse) ในตัวแปรตัวแปรอิสระเพื่อให้ข้อมูลกลับมาสู่รูปแบบเดิมก่อนที่จะนำไปใช้งานต่อ

ในงานวิจัยนี้เป็นการจัดการปัญหาเชิงถดถอย (regression) ซึ่งเป็นการวิเคราะห์ความสัมพันธ์ระหว่างตัวแปรอิสระกับตัวแปรตาม มีวัตถุประสงค์ในการทำนายค่าของตัวแปรตามจากค่าของตัวแปรอิสระ คือ price และเพื่อทำนายความสัมพันธ์ระหว่างตัวแปรทั้งสองจึงมีการสร้างแบบจำลอง 4 ชนิด ได้แก่ Neural Network, Random Forest, K-Nearest Neighbors (KNN) และ XGBoost เพื่อนำมาเปรียบเทียบประสิทธิการทำงานของแบบจำลอง ดังนี้

1. Neural Network

ตาราง 3 เปรียบเทียบประสิทธิภาพของแบบจำลอง Neural Network เมื่อมีการปรับจำนวนโหนดในชั้นต่างๆ

Transform		Entity Embedding			One-hot Encoding		
Data	Layer	RMSE	MAE	R-squared	RMSE	MAE	R-squared
StandardScaler	(8, 2)	898.02	658.13	0.12	1,770.70	1,488.87	-2.41
	(16, 8)	892.77	655.54	0.13	879.16	638.34	0.16
	(32, 16)	888.53	644.74	0.14	880.58	626.38	0.16
	(84, 32)	886.81	651.80	0.14	872.16	637.09	0.17
	(128, 64)	884.15	647.37	0.15	871.82	652.93	0.17
Logarithm	(8, 2)	899.33	661.29	0.12	886.97	654.63	0.14
	(16, 8)	893.34	652.46	0.13	880.16	654.92	0.16
	(32, 16)	891.86	655.81	0.13	879.45	626.03	0.16
	(64, 32)	889.41	649.26	0.14	872.42	641.28	0.17
	(128, 64)	**885.81**	**648.58**	**0.15**	**869.72**	**618.37**	**0.18**

จากตาราง 3 เมื่อมีจำนวนชั้นที่เพิ่มขึ้นแบบจำลองสามารถเรียนรู้ได้ดีมากขึ้น เนื่องจากมีการเรียนรู้คุณลักษณะที่ซับซ้อนมากขึ้นและมีความสัมพันธ์ระหว่างตัวแปรที่มีความซับซ้อนมากขึ้นในแบบจำลองซึ่งสามารถแสดงให้เห็นถึงลักษณะของข้อมูลได้ดีขึ้น

2. Random Forest

ตาราง 4 เปรียบเทียบประสิทธิภาพของแบบจำลอง Random Forest เมื่อมีการเพิ่ม max_feature

Transform		Entity Embedding			One-hot Encoding		
Data	max_feature	RMSE	MAE	R-squared	RMSE	MAE	R-squared
StandardScaler	0.1	857.32	609.42	0.20	841.93	610.68	0.23
	0.2	851.77	603.14	0.21	830.39	594.14	0.25
	0.3	848.18	598.26	0.22	825.55	586.86	0.26
	0.4	843.76	595.50	0.23	820.31	582.60	0.27
	0.5	842.90	592.79	0.22	818.94	578.47	0.27
	0.6	841.36	591.01	0.23	817.09	575.92	0.27
	0.7	**840.33**	**590.56**	**0.24**	**816.18**	**573.80**	**0.28**
Logarithm	0.1	850.00	608.65	0.21	842.67	612.85	0.23
	0.2	844.82	603.17	0.22	829.63	594.65	0.25
	0.3	840.12	596.90	0.23	825.73	589.23	0.25
	0.4	835.99	592.90	0.24	821.28	583.21	0.26
	0.5	833.43	590.19	0.24	820.21	580.32	0.27
	0.6	833.49	589.30	0.24	817.46	577.58	0.28
	0.7	**832.56**	**587.63**	**0.25**	**817.23**	**575.17**	**0.29**

ในแบบจำลอง Random Forest มีการกำหนดพารามิเตอร์ ได้แก่ max_depth = 30, max_features = (n), min_samples_split = 5, min_samples_leaf = 5, n_estimators = 100

จากตาราง 4 จะเห็นว่าเมื่อมีการเพิ่มขึ้นของ max_feature แบบจำลองจะมีประสิทธิภาพที่ดีขึ้น นั่นคือจำนวนคุณลักษณะ (features) ที่สุ่มเลือกเข้ามาใช้ในการตัดสินใจเมื่อสร้างโหนดในต้นไม้มีเพิ่มขึ้นซึ่งสามารถช่วยลดความเชื่อมโยงระหว่างต้นไม้ ทำให้แบบจำลองมีประสิทธิภาพในการทำนายกับข้อมูลที่ไม่เคยเห็นมาก่อนได้ดีขึ้นเนื่องจากแบบจำลองไม่มีความเชื่อมโยงมากนักกับข้อมูลเดิมและมีความยืดหยุ่นมากขึ้นในการตัดสินใจ

3. K-Nearest Neighbors (KNN)

ตาราง 5 แสดงการเปรียบเทียบประสิทธิภาพของแบบจำลอง KNN เมื่อมีการเพิ่มค่า K

Transform		Entity Embedding			One-hot Encoding		
Data	K	RMSE	MAE	R-squared	RMSE	MAE	R-squared
StandardScaler	3	945.41	666.83	0.03	989.75	697.99	-0.07
	4	926.10	659.08	0.07	962.96	683.87	-0.01
	5	913.80	651.98	0.09	952.70	680.05	0.01
	6	904.98	647.93	0.11	941.35	674.27	0.04
	7	899.13	644.22	0.12	936.29	675.30	0.05
	8	897.24	643.56	0.12	935.03	674.52	0.05
	9	**894.94**	**643.47**	**0.13**	**930.76**	**674.70**	**0.06**
Logarithm	3	928.83	655.42	0.06	990.38	698.49	-0.07
	4	911.53	651.75	0.10	963.01	683.81	-0.01
	5	905.49	649.76	0.11	952.14	679.92	0.01
	6	900.69	643.04	0.12	940.69	673.68	0.04
	7	893.50	640.13	0.13	936.57	675.72	0.05
	8	889.13	637.79	0.14	935.29	674.83	0.05
	9	**885.38**	**636.56**	**0.15**	**931.12**	**675.24**	**0.06**

จากตาราง 5 จะเห็นว่าเมื่อมีการเพิ่มขึ้นของค่า K จะมีการใช้ข้อมูลจำนวนมากเพื่อตัดสินใจ และเพิ่มความสามารถในการจัดกลุ่มข้อมูลที่ถูกต้องมากขึ้นเนื่องจากมีการพิจารณาข้อมูลจำนวนมากในการตัดสินใจ จึงทำให้ประสิทธิภาพของแบบจำลองดีขึ้น

4. XGBoost

ตาราง 6 เปรียบเทียบประสิทธิภาพของแบบจำลอง XGBoost เมื่อมีการเพิ่ม learning_rate

Transform		Entity Embedding			One-hot Encoding		
Data	learning_rate	RMSE	MAE	R-squared	RMSE	MAE	R-squared
StandardScaler	0.01	882.93	620.84	0.15	839.59	607.56	0.23
	0.02	869.71	597.32	0.16	805.98	571.32	0.29
	0.03	868.70	593.12	0.16	796.85	557.62	0.31
	0.04	866.48	587.88	0.17	796.07	554.50	0.31
	0.05	865.94	583.00	0.18	795.67	553.14	0.31
	0.06	864.88	582.96	0.19	794.52	552.23	0.32
	0.07	**862.71**	**580.73**	**0.21**	**791.22**	**550.42**	**0.33**

ตาราง 6 (ต่อ)

Transform Data	learning_rate	Entity Embedding			One-hot Encoding		
		RMSE	MAE	R-squared	RMSE	MAE	R-squared
StandardScaler	0.01	893.12	625.55	0.13	838.60	606.50	0.24
	0.02	891.03	623.82	0.14	804.88	570.22	0.29
	0.03	890.08	622.19	0.14	795.55	556.60	0.32
	0.04	889.49	621.54	0.15	793.89	552.78	0.33
	0.05	887.48	617.23	0.16	792.00	550.11	0.35
	0.06	883.28	6168.17	0.17	791.52	549.23	0.37
	0.07	**881.21**	**611.53**	**0.20**	**787.22**	**544.42**	**0.37**

ในแบบจำลอง XGBoost มีการกำหนดพารามิเตอร์ ได้แก่ max_depth = 30, learning_rate = (n), reg_alpha = 0.1, reg_lambda = 0.1, subsample = 0.9, colsample_bytree = 0.9, n_estimators = 100

จากตาราง 6 จะเห็นว่าเมื่อมีการเพิ่มขึ้นของ learning_rate จะช่วยให้แบบจำลองมีประสิทธิภาพมากขึ้น เนื่องจากค่า learning rate มีบทบาทสำคัญในการกำหนดการเคลื่อนไหวของแบบจำลองในการปรับค่าพารามิเตอร์ เมื่อเพิ่ม learning rate อาจช่วยให้แบบจำลองมีการปรับค่าได้มากขึ้นในแต่ละรอบของการฝึก ทำให้มีโอกาสในการพบค่าที่ดีกว่า

ขั้นตอนที่ 4 : การประเมินผล

การประเมินประสิทธิภาพของแบบจำลองสำหรับการทำนายราคามีการคำนวณค่าความคลาดเคลื่อน (Error Metrics) เพื่อประเมินประสิทธิภาพของแบบจำลองด้วยค่า RMSE, MAE และ R-squared

1. Root Mean Squared Error (RMSE) คือ การวัดความแตกต่างระหว่างค่าจริงและค่าพยากรณ์ ถ้าหากค่าที่ได้มีค่าที่น้อยแสดงถึงค่าทำนายนั้นประมาณค่าได้ใกล้เคียงกับค่าจริง ดังสมการ 1

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(Yi - \widehat{Yi})^2} \qquad \text{สมการ (1)}$$

2. Mean Absolute Error (MAE) คือ การหาค่าเฉลี่ยของความแตกต่างสมบูรณ์ระหว่างค่าพยากรณ์และค่าจริง หากค่า MAE ยิ่งน้อยแสดงว่าค่าทำนายมีค่าใกล้เคียงกับค่าจริง ดังสมการ (2)

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|Yi - \widehat{Yi}\right| \qquad \text{สมการ (2)}$$

3. R-Squared คือ ค่าทางสถิติที่ใช้ในการอธิบายความเปลี่ยนแปลงของตัวแปรตาม (Dependent Variable) โดยเทียบกับความเปลี่ยนแปลงของตัวแปรอิสระ (Independent Variables) R-Squared หากค่า R-Squared ยิ่งเข้าใกล้ 1 หมายถึงแบบจำลองสามารถอธิบายข้อมูลได้เพียงพอทุกประการ ดังสมการ (3)

$$R^2 = 1 - \frac{SSres}{SStot} \qquad \text{สมการ (3)}$$

ผลการวิจัยและอภิปรายผลการวิจัย

ในงานวิจัยนี้ผู้วิจัยได้ทำการศึกษาการทำนายราคาที่พักบน Airbnb ในพื้นที่กรุงเทพมหานคร โดยมีการจัดการกับตัวแปรที่มีค่าความหลากหลายสูง (High Cardinality) ด้วยการเข้ารหัสแบบ Entity Embedding และ One-hot Encoding 2.ในตัวแปรตาม (dependent variable) มีการปรับปรุงข้อมูลด้วยวิธี StandardScaler กับคอลัมน์ที่เป็นตัวแปรเชิงปริมาณทั้งหมด และในตัวแปรอิสระ (independent variable) มีการปรับการกระจายตัวของข้อมูล 2 วิธี คือ StandardScaler และ Logarithm ในคอลัมน์ price ผลการวัดประสิทธิภาพของแบบจำลอง ได้ผลดังตาราง 7

ตาราง 7 แสดงผลการทดสอบประสิทธิภาพของแบบจำลอง

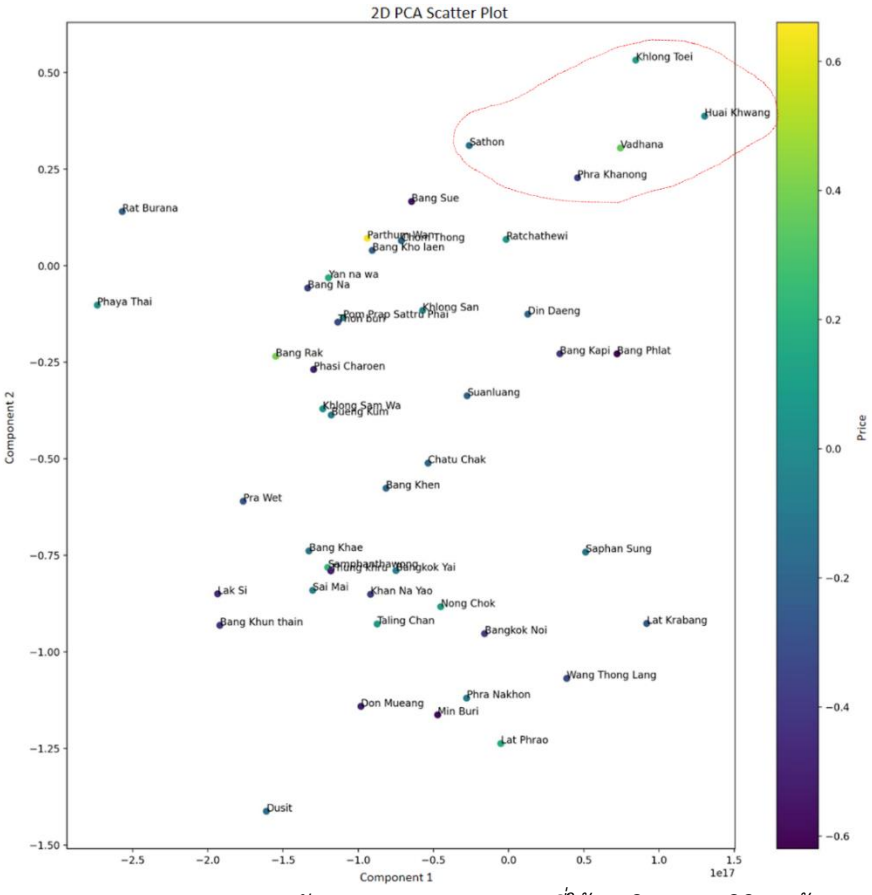
Transform		Entity Embedding			One-hot Encoding		
Data	Model	RMSE	MAE	R-squared	RMSE	MAE	R-squared
StandardScaler	NN	884.15	647.37	0.15	871.82	652.93	0.17
	RF	840.33	590.56	0.24	816.18	573.80	0.28
	KNN	894.94	643.47	0.13	930.76	674.70	0.06
	XGB	862.71	580.73	0.21	791.22	550.42	0.33
Logarithm	NN	885.81	648.58	0.15	869.72	618.37	0.18
	RF	**832.56**	**587.63**	**0.25**	817.23	575.17	0.27
	KNN	885.38	636.56	0.15	931.12	675.24	0.06
	XGB	881.21	611.53	0.20	**787.22**	**544.42**	**0.37**

จากผลการทดสอบประสิทธิภาพของแบบจำลองผ่านวิธีการเข้ารหัสแบบ Entity Embedding และ One-hot Encoding ผู้วิจัยได้เลือกผลลัพธ์ที่ดีที่สุดทั้ง 3 มิติ ได้แก่ มิติของการเข้ารหัส (encoding method) มิติของการปรับปรุงข้อมูล (transform data) มิติของการทดสอบประสิทธิภาพของแบบจำลอง (model evaluation) ในมิติของการเข้ารหัสวิธีที่ให้ผลลัพธ์ดีที่สุด คือ การเข้ารหัสแบบ One-hot Encoding ในมิติของการปรับปรุงข้อมูลวิธีที่ให้ผลลัพธ์ดีที่สุด คือ การปรับปรุงข้อมูลด้วยวิธี Logarithm และในมิติของการทดสอบประสิทธิภาพของแบบจำลองแบบจำลองที่ให้ผลลัพธ์ดีที่สุด คือ XGBoost โดยเมื่อพิจารณาค่าความคลาดเคลื่อนในทุกๆ มิติ มีค่า RMSE เท่ากับ 787.22 MAE เท่ากับ 544.42 และ R-squared เท่ากับ 0.37 ซึ่งเมื่อพิจารณาในมิติเดียวกันแต่ด้วยการเข้ารหัสแบบ Entity Embedding วิธีการปรับปรุงข้อมูลที่ให้ผลลัพธ์ดีที่สุด คือ การปรับปรุงข้อมูลด้วยวิธี Logarithm และในมิติของการทดสอบประสิทธิภาพของแบบจำลอง แบบจำลองที่ให้ผลลัพธ์ดีที่สุด คือ Random Forest โดยเมื่อพิจารณาค่าความคลาดเคลื่อนในทุกๆ มิติ มีค่า RMSE เท่ากับ 832.56 MAE เท่ากับ 587.63 และ R-squared เท่ากับ 0.25

สรุปผลการวิจัย

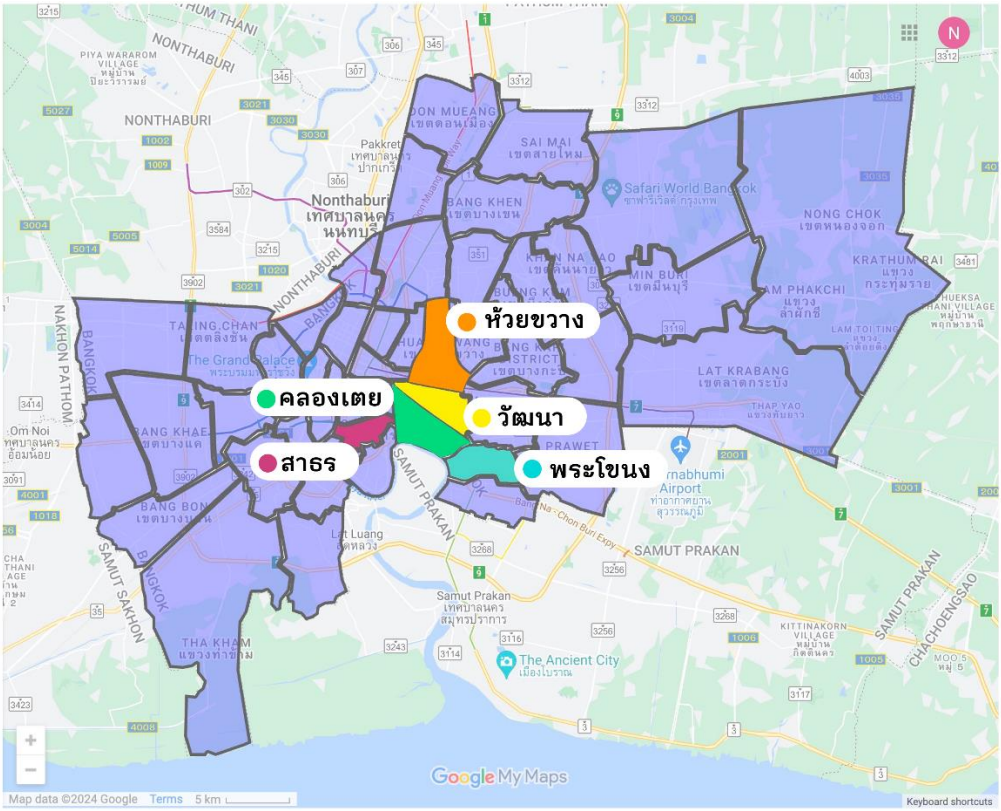
ผลการวิจัยสรุปได้ว่าการเข้ารหัสแบบ One-hot Encoding ร่วมกับการทำ Logarithm ในแบบจำลอง XGBoost ให้ผลลัพธ์ที่ดีที่สุด แต่เมื่อเปรียบเทียบกับการเข้ารหัสแบบ Entity Embedding ผลการทดสอบประสิทธิภาพของการเข้ารหัสทั้ง 2 แบบไม่ได้แตกต่างกันมากนักและไม่ได้มีประสิทธิภาพในการทำงานที่สูง นั่นเป็นเพราะข้อมูลไม่ได้มีจำนวนรายการที่เยอะและฟีเจอร์ที่มีในชุดข้อมูลไม่เพียงพอต่อการทำนายราคา จึงมีความเป็นไปได้ที่จะมีปัจจัยอื่นที่มีผล เช่น ตำแหน่งที่ตั้ง สิ่งอำนวยความสะดวกของที่พัก การตกแต่งภายใน จึงมีความเป็นไปได้ที่ปัจจัยต่างๆ เหล่านี้จะมีผลต่อการทำนายราคา แต่การเข้ารหัสแบบ Entity Embedding สามารถใช้เทคนิคการลดมิติของข้อมูล (PCA : Principal Component Analysis) เพื่อลดขนาดของ Embedded Feature ลงมาให้สามารถแสดงผลในรูปแบบการแสดงภาพ (visualization) ได้โดยที่ยังสามารถรักษาความสัมพันธ์ของข้อมูลในการแสดงผลได้ดี

แม้ว่าการเข้ารหัสแบบ Entity Embedding จะไม่ได้ให้ผลลัพธ์ที่ดีที่สุด แต่เป็นวิธีที่น่าสนใจที่สามารถนำไปประยุกต์ใช้กับข้อมูลที่มีค่าความหลากหลายสูงได้ เพราะสามารถใช้เทคนิคการลดมิติของข้อมูล (PCA : Principal Component Analysis) เพื่อลดขนาดของ Embedded Feature ลงมาให้สามารถแสดงผลในรูปแบบการแสดงภาพ (visualization) เพื่อให้เห็นความสัมพันธ์ของข้อมูลได้ง่ายขึ้น โดยที่ยังสามารถรักษาความสัมพันธ์ของข้อมูลได้ดี



ภาพประกอบ 2 การแสดงภาพของตัวแปร neighbourhood ที่ใช้เทคนิคการลดมิติของข้อมูล (PCA)

จากภาพประกอบ 2 พบว่าการลดมิติของข้อมูลด้วยวิธี PCA (Principal Component Analysis) จุดบนภาพไม่ได้มาจากองค์ประกอบ (component) ที่เกี่ยวข้องโดยตรงกับที่ตั้งทางภูมิศาสตร์แต่กลับสอดคล้องกับที่ตั้งทางภูมิศาสตร์ จึงเป็นสิ่งที่น่าสนใจและสะท้อนถึงตัวแปรอื่นๆ ในชุดข้อมูลที่นำมาใช้ในงานวิจัย เช่น ประเภทของห้องพัก ราคารายวันสำหรับการเข้าพัก จำนวนรีวิว จากการแสดงภาพจะพบว่าเขตวัฒนา เขตคลองเตย เขตห้วยขวาง เขตราชเทวี และเขตสาธร เป็นกลุ่มข้อมูลที่มีความถี่สูงและมีจำนวนรายการเพียงพอที่สามารถอยู่ในกลุ่มที่สอดคล้องกับที่ตั้งทางภูมิศาสตร์ ซึ่งทั้ง 5 เขตเป็นเขตที่มีความสำคัญทางธุรกิจ อาจเป็นเพราะเขตเหล่านี้มีพื้นที่ตั้งอยู่ใจกลางกรุงเทพมหานครซึ่งมีระบบโครงสร้างพื้นฐานที่เอื้ออำนวยต่อการทำธุรกิจ และเป็นที่ตั้งของศูนย์การค้า บริษัท และองค์กรชั้นนำ เมื่อนำมาเทียบกับแผนที่ของกรุงเทพมหานครจะพบว่าทั้ง 5 เขต เป็นเขตพื้นที่ที่ตั้งอยู่ในใกล้กัน ดังภาพประกอบ 3



ภาพประกอบ 3 แผนที่กรุงเทพมหานคร

ที่มา : https://www.google.com/maps/d/viewer?mid=19vgGq-gj8wlK47tMoXBlT7Gff4U&hl=en_US&ll=13.725127956901979%2C100.633392499999997&z=10

กิตติกรรมประกาศ

การจัดทำวิจัยฉบับนี้สำเร็จลุล่วงไปได้ด้วยดีจากการสนับสนุนและการให้ความช่วยเหลือจาก ผศ.ดร.นภา แซ่เบ๊ อาจารย์ที่ปรึกษา ที่ได้กรุณาให้ความรู้และข้อแนะนำมาโดยตลอด ขอขอบคุณคณาจารย์ทุกท่านในภาควิชาวิทยาการข้อมูล คณะวิทยาศาสตร์ มหาวิทยาลัยศรีนครินทรวิโรฒ และบัณฑิตวิทยาลัยในการสนับสนุนการนำเสนอผลงานวิจัย ผู้วิจัยจึงขอขอบคุณมา ณ ที่นี้

เอกสารอ้างอิง

[1] C. Guo and F. Berkhahn, “Entity embeddings of categorical variables,” arXiv [cs.LG], 2016.

[2] Y. He, C. Wu, and Y. Fan, “Exploring the drivers of local government budget coordination: A random forest regression analysis,” Int. Rev. Econ. Finance, 2024.

[3] C. Lee, “How can we use neural network with entity embedding for product valuations? A case study for the car industry,” International Journal of Information Management Data Insights, vol. 3, no. 2, p. 100187, 2023.

[4] Z. Allah Bukhsh, I. Stipanovic, A. Saeed, and A. G. Doree, “Maintenance intervention predictions using entity-embedding neural networks,” Autom. Constr., vol. 116, no. 103202, p. 103202, 2020.

[5] J. Lu, H. Leung, and N. Xie, “Privacy-preserving data integration and sharing in multi-party IoT environments: An entity embedding perspective,” Inf. Fusion, vol. 108, no. 102380, p. 102380, 2024.

[6] “Get the data,” Insideairbnb.com. [Online]. Available: https://insideairbnb.com/get-the-data/. [Accessed: 04-Apr-2024].

[7] S. Chaiyadecha, “One-Hot Encoding สร้างตัวแปร Dummies สำหรับ Classification model,” Medium, 23-Dec-2020. [Online]. Available: https://lengyi.medium.com/one-hot-encoding-737c66e5b1bd. [Accessed: 04-Apr-2024].