



Ứng dụng học sâu dự đoán lưu lượng giao thông trên dữ liệu thời gian thực

Nguyễn Thị Bích Ngọc¹, Đinh Công Tùng², Mai Đức Vinh^{2*}

¹Khoa Công nghệ thông tin, Trường Đại học Sao Đỏ, Hải Dương

²Khoa Công nghệ thông tin, Trường Đại học Giao thông vận tải

*Email: ducvinh1012004@gmail.com

Tóm tắt

Trong bối cảnh đô thị hóa, việc quản lý và tối ưu hóa hạ tầng giao thông nhằm giảm thiểu tình trạng ùn tắc là cần thiết. Dự đoán lưu lượng phương tiện tại từng thời điểm góp phần giải quyết bài toán ùn tắc, tạo cơ sở để xây dựng hệ thống giao thông thông minh, đô thị thông minh. Bài báo này tiến hành nghiên cứu, đánh giá các mô hình học sâu GRU, LSTM, Transformer, RNN và MLP trong phân tích chuỗi thời gian và dự đoán lưu lượng giao thông. Kết quả thử nghiệm cho thấy các mô hình học sâu đều cho độ chính xác tương đối cao, trong đó, Transformer đạt giá trị cao nhất là 93%. Những kết quả trên cho thấy tiềm năng ứng dụng của học sâu trong việc dự đoán sớm lưu lượng giao thông và quy hoạch đô thị.

Từ khóa: GRU, LSTM, Transformer, RNN, MLP, dự đoán lưu lượng giao thông.

Abstract

In the context of urbanization, managing and optimizing transportation infrastructure to reduce traffic congestion is essential. Predicting vehicle flow at different time points helps address congestion issues and provides a foundation for building intelligent transportation systems and smart cities. This paper studies and evaluates deep learning models, including GRU, LSTM, Transformer, RNN, and MLP, for time series analysis and traffic flow prediction. Experimental results show that all deep learning models achieve relatively high accuracy, with the Transformer model attaining the highest performance at 93%. These findings highlight the potential of deep learning in early traffic flow prediction and urban planning.

Keywords: GRU, LSTM, Transformer, RNN, MLP, traffic flow prediction.

1. Mở đầu

Sự phát triển nhanh chóng của đô thị đòi hỏi các chiến lược quản lý và quy hoạch hạ tầng giao thông hiệu quả nhằm giảm thiểu tình trạng ùn tắc, tối ưu hóa sử dụng không gian đô thị và hạn chế tác động tiêu cực đến môi trường. Một trong những yếu tố quan trọng trong công tác quy hoạch và xây dựng hệ thống giao thông là khả năng dự báo chính xác lưu lượng phương tiện. Điều này hỗ trợ lập kế hoạch mở rộng và nâng cấp hạ tầng, tối ưu hóa điều phối giao thông, cải thiện hiệu suất khai thác các tuyến đường. Hiện nay, mô hình học sâu đã chứng minh khả năng hiệu quả trong phân tích dữ liệu chuỗi thời gian, giúp tăng độ chính xác của dự báo lưu lượng giao thông, tạo cơ sở cho việc xây dựng, quản lý hạ tầng giao thông thông minh và đô thị.

Có nhiều giải pháp ứng dụng học sâu đã được sử dụng trong các nghiên cứu về dự đoán lưu lượng giao thông. Bài báo [1] sử dụng các mô hình học máy khác nhau trên bộ dữ liệu



Huawei Munich nhằm tối ưu hóa phân luồng phương tiện, trong đó mô hình Multilayer Perceptron Regressor đạt kết quả tốt nhất với giá trị EV đạt 0.93. Nhằm nâng cao độ chính xác của dự báo trong điều kiện có nhiều yếu tố tác động như thời tiết, tai nạn, các tác giả trong [2, 3] đề xuất mô hình kết hợp bốn thuật toán gồm DAN (Deep Autoencoder), DBN (Deep Belief Network), RF (Random Forest) và LSTM (Long Short-Term Memory), dựa trên dữ liệu từ cảm biến giao thông để cải thiện khả năng dự đoán. Khi các hệ thống giao thông thông minh dần phát triển, việc thiếu dữ liệu huấn luyện chi tiết từ trước đó trở thành vấn đề lớn. Để khắc phục vấn đề này, bài báo [4] thử nghiệm phương pháp học chuyển đổi với khung dự báo ba bước gồm lựa chọn nguồn dữ liệu, huấn luyện mô hình và chuyển đổi kết quả, mang lại hiệu quả trong dự báo ngắn hạn, nhưng còn hạn chế đối với dự báo dài hạn. Trước tính biến động cao của hệ thống giao thông, bài báo [5] giới thiệu mô hình học sâu STAWnet (Spatial-Temporal Attention Wavenet), có khả năng tự học các đặc tính giao thông khu vực thông qua cơ chế chú ý, giúp nâng cao độ chính xác dự báo theo thời gian. Các mô hình hiện tại vẫn gặp khó khăn trong việc dự báo ùn tắc giao thông trong các tình huống không cố định. Để giải quyết vấn đề này, bài báo [6] đề xuất mô hình Ising-Traffic, gồm hai bước: “Tái hiện” (Reconstruct-Ising) thu nhận dữ liệu và xử lý các yếu tố bất định, và “Dự báo” (Predict-Ising) dự đoán trạng thái ùn tắc dựa trên kết quả của bước đầu tiên. Mô hình này tối ưu hóa tài nguyên tính toán, cho kết quả nhanh hơn 98 lần, giảm 98.2% số phép tính và cải thiện độ chính xác lên 5% so với các mô hình hiện có. Hướng đến tối ưu hóa hiệu suất dự báo cho người tham gia giao thông đơn lẻ, bài báo [7] đề xuất mô hình kết hợp học máy và phương pháp Bayesian cải tiến (Improved Bayesian Combination Model with Deep Learning – IBCM-DL), áp dụng trên dữ liệu thực tế của giao thông Bắc Kinh kết hợp với lịch sử dữ liệu quá khứ, mang lại kết quả dự đoán tương đối tốt. Bài báo [8] phát triển hệ thống điều phối tín hiệu đèn giao thông phân cấp, sử dụng mạng LSTM kết hợp với học củng cố (Reinforcement Learning – RL) nhằm tối ưu hiệu quả điều tiết giao thông trên phạm vi rộng. Bài báo [9] đề xuất mô hình học máy sử dụng thuật toán chuyên biệt để đánh giá mật độ phương tiện với độ chính xác cao hơn so với các thuật toán truyền thống. Không sử dụng học sâu, mô hình đạt độ chính xác cao nhất 91% khi áp dụng thuật toán Random Forest..

Rõ ràng, rất khó xác định phương pháp nào tối ưu nhất trong việc dự đoán lưu lượng giao thông, do các kiến trúc được phát triển khác nhau, cũng như các nghiên cứu thường không công bố đầy đủ mã nguồn. Bài báo này tiến hành nghiên cứu và thử nghiệm năm mô hình học sâu phổ biến gồm Transformer, GRU, LSTM, MLP, RNN dự đoán lưu lượng giao thông trên dữ liệu thời gian thực. Kết quả thực nghiệm cho thấy mô hình đạt độ chính xác cao nhất là Transformer với độ đo R^2 là 93%.



Phần tiếp theo trình bày một số mô hình học sâu ứng dụng trong dự đoán lưu lượng giao thông. Phần 3 trình bày về bộ dữ liệu. Phần 4 là kết quả thực nghiệm. Cuối cùng, Phần 5 là kết luận.

1. MỘT SỐ MÔ HÌNH HỌC SÂU

1.1. Mô hình LSTM

Long Short-Term Memory (LSTM) là một biến thể của mạng nơ-ron hồi quy (RNN) được giới thiệu để giải quyết các vấn đề mất mát gradient trong mạng RNN truyền thống [10]. Không giống với các mạng truyền thẳng, LSTM có chứa các kết nối phản hồi, sử dụng để xử lý các điểm dữ liệu đơn lẻ và toàn bộ chuỗi dữ liệu. LSTM có kiến trúc gồm các cổng giúp kiểm soát thông tin cụ thể như sau. Cổng quên (Forget Gate) có vai trò quyết định thông tin được giữ lại hay bỏ đi, được tính theo công thức $ft = \sigma(Wf \cdot [ht - 1, xt] + bf)$. Cổng nhập (Input Gate) quyết định thông tin nào được cập nhật khi ghi nhớ, được tính theo công thức $it = \sigma(Wi \cdot [ht - 1, xt] + bi)$ và $Ct = \tanh(WC \cdot [ht - 1, xt] + b)$. Cổng đầu ra (Output Gate) quyết định thông tin được sử dụng để tạo đầu ra với công thức $ot = \sigma(Wo \cdot [ht - 1, xt] + bo)$ và $ht = ft \cdot Ct - 1 + it \cdot Ct$. Kiến trúc này giúp LSTM ghi nhớ thông tin tốt hơn, phù hợp với các bài toán xử lý ngôn ngữ tự nhiên, nhận dạng hành động, phân tích giám sát và chuỗi thời gian như dự đoán lưu lượng giao thông. LSTM bao gồm nhiều lớp ẩn kết hợp Dropout để tránh overfitting.

1.2. Mô hình GRU

Mô hình Gated Recurrent Unit (GRU) là một biến thể cải tiến của LSTM với số lượng tham số ít hơn, giúp giảm thời gian huấn luyện mà vẫn giữ được khả năng ghi nhớ thông tin chuỗi thời gian [11]. GRU sử dụng hai cổng chính là Update Gate và Reset Gate cụ thể như sau. Cổng Update Gate xác định phần nào của trạng thái cũ cần được giữ lại với công thức $zt = \sigma(Wz \cdot [ht - 1, xt] + bz)$. Cổng Reset Gate quyết định bao nhiêu thông tin từ trạng thái cũ có thể bỏ đi, được tính theo công thức $rt = \sigma(Wr \cdot [ht - 1, xt] + br)$. Với cấu trúc hai cổng, GRU điều chỉnh dòng chảy thông tin, đơn giản hóa kiến trúc tốt hơn so với LSTM nhưng vẫn đảm bảo hiệu quả trong các bài toán chuỗi thời gian. GRU thường được ưu tiên sử dụng khi tài nguyên tính toán bị giới hạn hoặc khi cần tối ưu tốc độ huấn luyện.

1.3. Mô hình MLP

Multi-Layer Perceptron (MLP) là một mạng nơ-ron truyền thẳng với nhiều lớp ẩn. Mô hình này sử dụng các lớp Dense cùng với hàm kích hoạt phi tuyến như Relu hoặc Sigmoid để học các đặc trưng dữ liệu [12]. Tuy nhiên, do MLP không có cơ chế ghi nhớ chuỗi thời gian nên mô hình này thường không đạt độ chính xác cao bằng LSTM hoặc GRU trong các bài toán



dự đoán chuỗi thời gian. MLP thường được sử dụng như một baseline để so sánh với các mô hình tiên tiến hơn trong học sâu.

1.4. Mô hình RNN

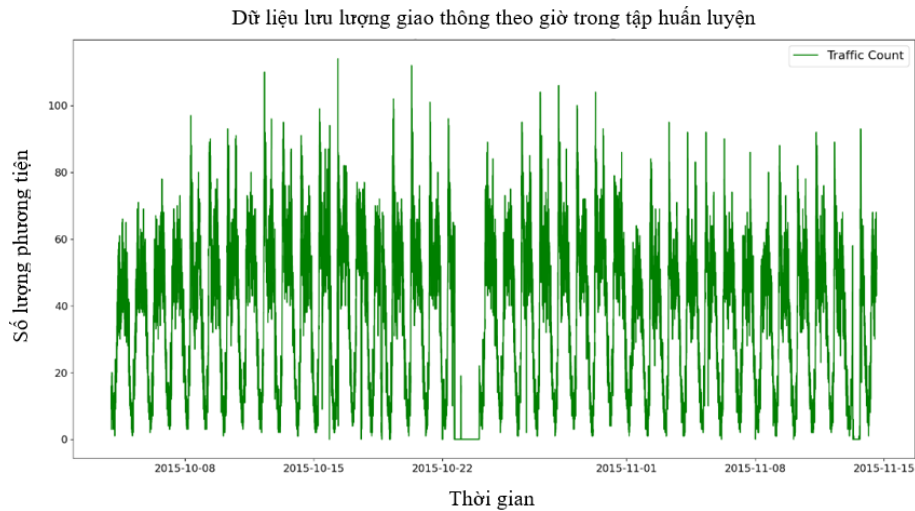
Recurrent Neural Network (RNN) là một loại mạng nơ-ron được thiết kế để xử lý dữ liệu chuỗi và dữ liệu có tính tuần tự, chẳng hạn như văn bản, âm thanh, chuỗi thời gian và dữ liệu chuỗi sinh học. Không giống như mạng nơ-ron truyền thẳng, RNN có khả năng ghi nhớ thông tin từ các bước trước đó nhờ cơ chế hidden state, giúp mô hình có khả năng học và dự đoán các phụ thuộc theo thời gian [13]. RNN có thể được sử dụng cho bài toán dự đoán lưu lượng giao thông, nhưng nhược điểm gặp phải là hiện tượng mất mát gradient khi số bước thời gian tăng lên. Điều này khiến RNN gặp khó khăn khi xử lý chuỗi dài và ảnh hưởng đến độ chính xác của việc dự đoán.

1.5. Mô hình Transformer

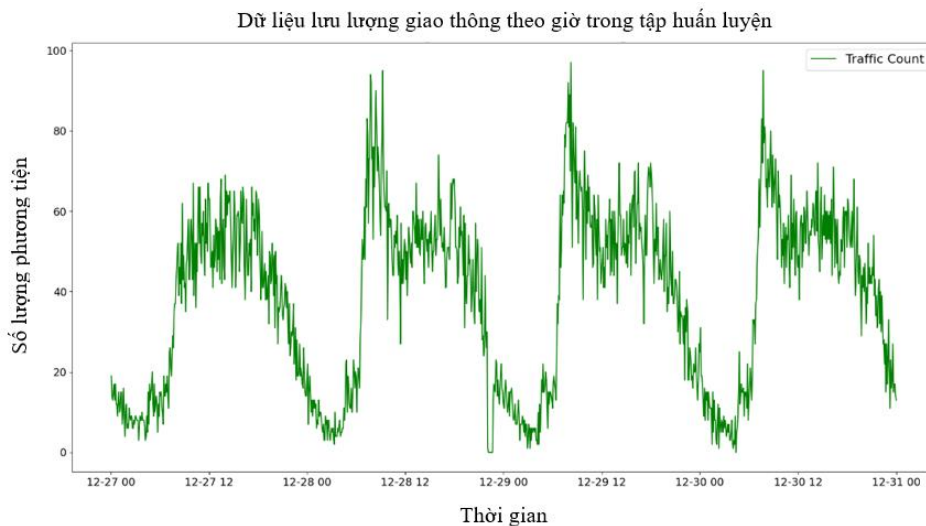
Transformer là một mô hình Deep Learning được giới thiệu trong bài báo [14]. Mô hình này thiết kế để xử lý dữ liệu tuần tự một cách hiệu quả, đặc biệt là trong các bài toán xử lý ngôn ngữ tự nhiên, sau đó được mở rộng sang nhiều lĩnh vực khác, bao gồm dự đoán chuỗi thời gian. Mô hình này dựa trên cơ chế Self-Attention cho phép nó nắm bắt mối quan hệ giữa các phần tử trong một chuỗi mà không cần dựa vào các bước tuần tự như LSTM hay RNN. Kiến trúc chính của mô hình bao gồm các thành phần Positional Encoding, Attention, Feed Forward Layer, Normalization & Residual Connection. Với cơ chế Attention, mô hình có thể học được quan hệ dài hạn trong dữ liệu mà không gặp vấn đề mất dần thông tin như RNN/LSTM.

2. Cơ sở dữ liệu

Trong bài báo này, chúng tôi sử dụng bộ dữ liệu của Coplin với 26479 bản ghi thể hiện số lượng phương tiện trên đường phố, với cập nhật mỗi năm phút một lần [15]. Dữ liệu được thu thập liên tục trong khoảng thời gian bốn tháng, từ 00:00 ngày 04/10/2015 đến 23:55 ngày 03/01/2016 với hai cột thông tin, cột thứ nhất ghi lại thời gian thu thập dữ liệu, cột thứ hai là số lượng phương tiện tung ứng tại thời điểm đó. Dữ liệu được tiền xử lý và phân chia thành hai phần huấn luyện và kiểm thử. Hình 1 và Hình 2 minh họa sự phân bố của dữ liệu huấn luyện và dữ liệu kiểm thử. Theo đó, dữ liệu huấn luyện được lấy từ 08/10/2015 đến 15/11/2015, dữ liệu kiểm thử được lấy từ 27/12/2015 đến 31/12/2015. Dữ liệu sau khi phân chia được tiền xử lý bằng kỹ thuật Min/Max scaling để chuẩn hóa và giúp tất cả các đặc trưng có cùng quy mô trong khoảng từ 0 đến 1, tăng tốc độ huấn luyện cũng như cải thiện khả năng học của mô hình.



Hình 1. Phân bố số lượng phương tiện trong tập dữ liệu huấn luyện từ ngày 08/10/2015 đến ngày 15/11/2015



Hình 2. Số lượng phương tiện trong tập dữ liệu kiểm thử từ ngày 27/12 đến ngày 31/12

3. Kết quả thực nghiệm

Trong phần này, chúng tôi trình bày kết quả huấn luyện và kiểm thử các mô hình học sâu đã nghiên cứu để dự đoán lưu lượng giao thông. Các mô hình được xây dựng và huấn luyện trên Google Colab với cấu hình GPU Tesla T4, RAM 12GB, Python 3.6, TensorFlow với Learning Rate to $1e-4$, $batch_size=64$, 200 epoch, $time_step = 24$.

Để làm đánh giá hiệu quả của mô hình, chúng tôi thực hiện thống kê và so sánh ba giá trị sau khi kiểm thử các mô hình bao gồm R^2 , MSE và RMSE. Cụ thể, R^2 là hệ số dùng để đánh giá độ chính xác của một mô hình hồi quy, thể hiện tỷ lệ phần trăm của tổng biến thiên các giá trị dự đoán so với thực tế, với các giá trị thực tế được tính theo công thức (1). Giá trị R^2 càng gần 1, thì mô hình đạt độ chính xác càng cao. Mean Squared Error (MSE) là một chỉ số thống

kê được sử dụng để đo lường mức độ trung bình của sai số giữa các giá trị dự đoán và giá trị thực tế trong mô hình hồi quy. MSE giúp đánh giá độ chính xác của một mô hình dự báo bằng cách tính trung bình của bình phương các sai số giữa giá trị dự đoán và giá trị thực tế, được tính theo công thức (2). Khi giá trị MSE càng thấp cho thấy mô hình dự đoán càng chính xác, với các giá trị dự đoán sát gần hơn với giá trị thực tế. RMSE là căn bậc hai của Mean Squared Error (MSE), thể hiện trung bình của MSE, được tính theo công thức (3).

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2} \quad (1)$$

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad (2)$$

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (3)$$

Tại các công thức trên, y_i là giá trị thực tế tại điểm dữ liệu thứ i , \hat{y}_i là giá trị dự đoán của mô hình tại điểm dữ liệu thứ i , \bar{y} là giá trị trung bình của các giá trị thực tế y_i . Các giá trị R^2 , MSE, RMSE của năm mô hình được so sánh trong Bảng 1.

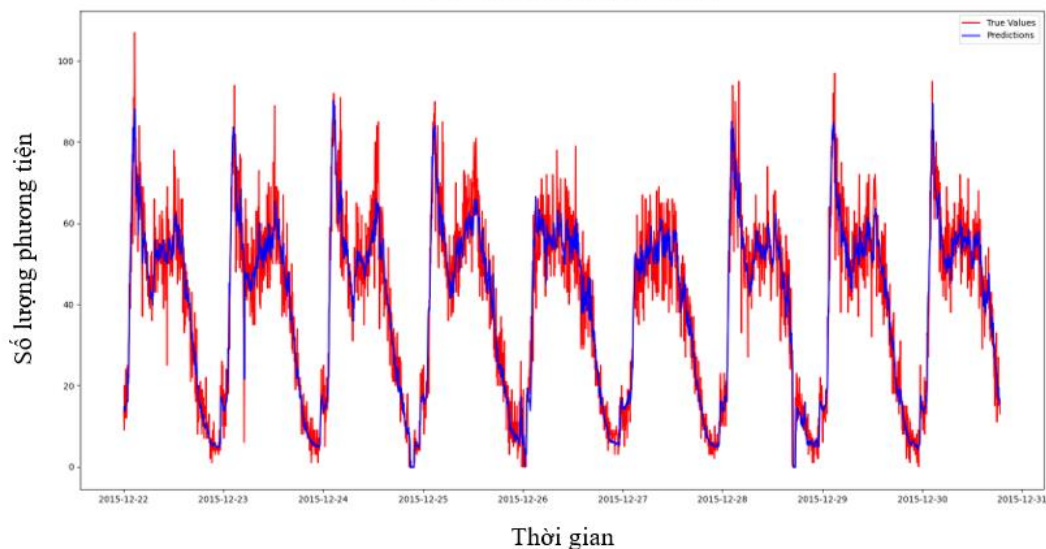
Bảng 1. So sánh độ chính xác giữa các mô hình

Mô hình	R ² -score	MSE	RMSE
GRU	0.9173	43.4	6.59
LSTM	0.9136	45.4	6.74
MLP	0.81	68.24	8.6
Transformer	0.93	41.27	6.42
RNN	0.89	52.66	7.26

Theo đó, Transformer đạt hiệu suất tốt nhất với $R^2=0.93$, $MSE = 41.27$, $RMSE = 6.42$, nhờ cơ chế Self-Attention giúp nắm bắt quan hệ dài hạn, từ đó giảm đáng kể sai số dự báo so với các mô hình khác. GRU có $R^2=0.9173$, $MSE = 43.4$, $RMSE = 6.59$, nhỉnh hơn LSTM với $R^2=0.9136$, $MSE = 45.4$, $RMSE = 6.74$, cho thấy GRU có lợi thế về việc sử dụng số lượng tham số ít, nhưng vẫn giữ được khả năng ghi nhớ thông tin dài hạn, giúp tăng tốc độ huấn luyện và giảm tài nguyên tính toán. RNN có $R^2=0.89$, $MSE = 52.66$, $RMSE = 7.26$, thấp hơn GRU và LSTM, phản ánh hạn chế của mô hình RNN truyền thống trong việc xử lý dữ liệu chuỗi dài do vấn đề mất mát gradient, khiến mô hình khó học được thông tin quan trọng ở các bước thời gian

xa và dẫn đến dự báo thiếu chính xác hơn. MLP có hiệu suất kém nhất với $R^2=0.81$, $MSE = 68.24$, $RMSE = 8.6$, mô hình này không phù hợp với bài toán dự báo chuỗi thời gian do bản chất là mạng nơ-ron truyền thẳng, nên không có cơ chế lưu trữ trạng thái hay ghi nhớ thông tin tuần tự, làm giảm khả năng nắm bắt các mối quan hệ trong dữ liệu thời gian thực, dẫn đến độ chính xác thấp hơn và sai số dự báo cao hơn so với các mô hình có kiến trúc hồi quy. Bảng 3 cho thấy kết quả dự đoán từ mô hình Transformer là tốt nhất. Để trực quan hóa việc dự đoán của mô hình Transformer, chúng tôi biểu diễn thông qua Hình 3. Theo đó trực hoành biểu diễn thời gian từ ngày 22/12/2015 đến 31/12/2015, trục tung biểu diễn số lượng phương tiện, với đường màu đỏ thể hiện giá trị thực tế và đường màu xanh đại diện cho giá trị dự báo từ mô hình. Biểu đồ cho thấy hai đường có sự tương quan cao, đặc biệt là dự báo được chính xác các dao động theo chu kỳ ngày, đêm, phản ánh rõ ràng xu hướng biến động của lưu lượng phương tiện trong khoảng thời gian nghiên cứu, tuy nhiên, Transformer dự đoán vẫn tồn tại một số sai lệch đáng kể tại các đỉnh và đáy, cho thấy mô hình có thể gặp khó khăn trong việc dự báo các đột biến lớn về lưu lượng giao thông. Tuy nhiên xét về tổng thể, Transformer vẫn thể hiện độ chính xác khá tốt khi duy trì được xu hướng chung và phản ánh được tính chất tuần hoàn của lưu lượng giao thông theo thời gian, chứng tỏ mô hình có tiềm năng cao trong các ứng dụng dự báo ngắn hạn nhằm hỗ trợ tối ưu hóa hệ thống giao thông và quản lý hạ tầng đô thị hiệu quả.

Dự báo lưu lượng giao thông



Hình 3. Kết quả dự đoán của mô hình Transformer

4. Kết luận

Dự đoán lưu lượng giao thông ngày càng trở nên cần thiết trong bối cảnh đô thị hóa nhanh chóng hiện nay. Hiểu rõ các mô hình và phân tích dữ liệu giao thông có thể mang lại những thông tin quý giá cho việc lập kế hoạch, phát triển cơ sở hạ tầng và quản lý tắc nghẽn. Bài báo



đã trình bày, đánh giá các mô hình học sâu GRU, LSTM, MLP, Transformer và RNN trong dự đoán lưu lượng giao thông trên dữ liệu thời gian thực. Kết quả cho thấy mô hình Transformer cho độ chính xác cao nhất với R^2 -score đạt 0.93. Trong tương lai, chúng tôi sẽ tiếp tục nghiên cứu và phát triển các phương pháp mới để cải thiện độ chính xác của mô hình, đồng thời áp dụng kết quả nghiên cứu vào việc xây dựng các ứng dụng nhằm phát triển hệ thống giao thông thông minh và quy hoạch đô thị tại Việt Nam.

Tài liệu tham khảo

- [1] Alfonso Navarro-Espinoza, et al, “*Traffic Flow Prediction for smart Traffic Lights Using Machine Learning Algorithms*”, Technologies, 2022.
- [2] Nazirkar Reshma Ramchandra, C. Rajabhushanam, “*Machine learning algorithms performance evaluation in traffic flow prediction*”, Materials Today: Proceedings, 51, 1, 1046-1050, ISSN 2214-7853, 2022.
- [3] Azzedine Boukerche, Jiahao Wang, “*Machine Learning-based traffic prediction models for Intelligent Transportation Systems*”, Computer Networks, 181, 107530, ISSN 1389-1286, 2020.
- [4] Junyi Li, et al, “*Transferability improvement in short-term traffic prediction using stacked LSTM network*”, Transportation Research Part C: Emerging Technologies, 124, 2021.
- [5] Tian C, et al. “*Spatial-temporal attention wavenet: A deep learning framework for traffic prediction considering spatial-temporal dependencies.*” IET Intell Transp Syst. 15: 549–561, 2021;
- [6] Pan, Z., Sharma, et al. “*Ising-Traffic: Using Ising Machine Learning to Predict Traffic Congestion under Uncertainty.*” Proceedings of the AAAI Conference on Artificial Intelligence. 37, 8 (6.2023), 9354-9363. 2023.
- [7] Y. Gu, W. Lu, et al, “*An Improved Bayesian Combination Model for Short-Term Traffic Prediction With Deep Learning,*”, IEEE Transactions on Intelligent Transportation Systems, 21, 3, 1332-1342, 2020.
- [8] Monireh Abdoos, Ana L.C. Bazzan, “*Hierarchical traffic signal optimization using reinforcement learning and traffic prediction with long-short term memory*”, Expert Systems with Applications, , 171, 114580, ISSN 0957-4174, 2021.
- [9] G. Meena, D. Sharma, M. Mahrishi, “*Traffic Prediction for Intelligent Transportation System using Machine Learning,*” 3rd International Conference on Emerging Technologies in Computer Engineering: Machine Learning and Internet of Things (ICETCE), Jaipur, 145-148, 2020.



- [10] Hochreiter, S., & Schmidhuber, J. “*Long short-term memory*”. *Neural Computation*, 9(8), 1735–1780, 1997.
- [11] Cho, K., et al. “*Learning phrase representations using RNN encoder-decoder for statistical machine translation*”. arXiv preprint arXiv:1406.1078, 2014.
- [12] Rumelhart, D. E., Hinton, G. E., & Williams, R. J. “*Learning representations by back-propagating errors*”. *Nature*, 323(6088), 533–536. 1986.
- [13] Elman, J. L. . “*Finding structure in time*”. *Cognitive Science*, 14(2), 179–211, 1990
- [14] Vaswani, A., et al. “*Attention is all you need*”. *Advances in Neural Information Processing Systems*, 30. 2017.
- [15] Coplin, DataSet(Traffic flow), 2018.