■Ngày nhận bài: 25/10/2023 ■Ngày sửa bài: 17/11/2023 ■Ngày chấp nhận đăng: 26/12/2023

Xây dựng mô hình học máy được tối ưu hóa bằng thuật toán jellyfish search để dự báo năng suất lao động trên công trường

Developing a machine learning model optimized with the jellyfish search algorithm for predicting labor productivity on construction sites

> KS VÕ HUỲNH KIM CHI[1], TS TRƯƠNG ĐÌNH NHẬT[2*], TS NGUYỄN THANH PHONG[3], THS LÊ THỊ THÙY LINH[4]

[1]HVCH Ngành Quản lý xây dựng, Khoa Sau Đại học, Trường Đại học Mở TP.HCM; Email: chivhk.218m@ou.edu.vn

[2*]GV Khoa Xây dựng, Trường Đại học Kiến trúc TP.HCM; Email: nhat.truongdinh@uah.edu.vn

[3]Bộ môn QLDA&XD, Khoa Xây dựng, Trường Đại học Mở TP.HCM; Email: phong.nt@ou.edu.vn

[4]GV Khoa Sư phạm Công nghiệp, Trường ĐH Sư phạm Kỹ thuật, Đại học Đà Nẵng; Email: lttlinh@ute.udn.vn

TÓM TẮT

Các hoạt động xây dựng phụ thuộc rất nhiều vào năng suất lao động bởi các tác động trực tiếp của nó đến hiệu quả kinh tế và tiến độ của dự án. Vì vậy, nâng cao năng suất lao động trên công trường luôn là mục tiêu hàng đầu của các doanh nghiệp và các chuyên gia quản lý xây dựng. Nghiên cứu này trình bày các so sánh và đánh giá hiệu suất của các mô hình học máy, bao gồm bốn mô hình đơn ANN, SVR, LR, CART và ba mô hình hỗn hợp Voting, Bagging, Stacking. Kết quả thu được cho thấy mô hình hỗn hợp Bagging-ANN mang lại hiệu quả cao nhất. Các tham số của mô hình được chọn sẽ được tối ưu hóa bằng thuật toán Jellyfish Search để nâng cao hiệu suất mô hình. Kết quả cuối cùng được so sánh với các đề xuất trước đó cho thấy hiệu suất vượt trội của mô hình JS-Bagging-ANN.

Từ khóa: Jellyfish Search; năng suất lao động; mô hình học máy; tối ưu hóa; dự báo.

ABSTRACT

Construction activities are significantly dependent on labor productivity due to its direct impact on economic efficiency and project progress. Therefore, enhancing labor productivity on construction sites remains a top priority for businesses and construction management experts. This study presents a comparative evaluation of the performance of various machine learning models, including four individual models: ANN, SVR, LR, CART, and three ensemble models: Voting, Bagging, and Stacking. The results demonstrate that the Bagging-ANN ensemble model yields the highest efficiency. The model's parameters are optimized using the Jellyfish Search algorithm to improve its performance. The final results are compared with literature, revealing the superior performance of the JS-Bagging-ANN model.

Keywords: Jellyfish Search; labor productivity; machine learning models, optimization; prediction system.

1. GIỚI THIỆU

Đối với các dự án xây dựng, tổng năng suất dự án bị phụ thuộc rất nhiều vào năng suất lao động trên công trường trong giai đoạn thi công xây dựng. Tuy nhiên, việc dự đoán năng suất lao động trên công trường luôn là vấn đề thách thức đối với các nhà quản lý xây dựng vì tính không ổn định cũng như phụ thuộc rất nhiều vào quy mô, công suất và địa điểm xây dựng [1, 2]. Ngoài ra có rất nhiều yếu tố ảnh hưởng đến năng suất lao động như quá trình lao động, kỹ năng lao động, vật liệu và công cụ, điều kiện địa điểm, ... Việc nắm bắt sớm xu hướng của năng suất lao động có thể giúp người quản lý đưa ra các quyết định kịp thời trong việc điều chỉnh nhân lực, phương án thi công xây dựng hay hướng quản lý phù hợp.

Ngày nay, với sự phát triển mạnh mẽ của khoa học kỹ thuật, trí tuệ nhân tạo (AI) ngày càng chứng tỏ là một công cụ mạnh mẽ để nâng cao hiệu quả xây dựng và mang lại cơ hội giải quyết các vấn đề kỹ thuật phức tạp [3, 4]. Các mô hình đơn cũng như mô hình hỗn hợp sẽ được so sánh để đưa ra các đánh giá cũng như lựa chọn mô hình tối ưu nhất. Các mô hình phổ biến nhất và được các nhà nghiên cứu đánh giá cao trong các mô hình dự báo thường gặp sẽ được sử dụng trong nghiên cứu này lần lượt là Artificial Neural Network (ANN), Support Vector Regression (SVR), Linear Regression (LR), Classification and Regression Tree (CART).

Đồng thời, việc kết hợp các mô hình đơn để tạo ra các mô hình hỗn hợp nhằm nâng cao hiệu suất của của mô hình. Các mô hình mới sẽ được so sánh và đánh giá để chọn ra mô hình có hiệu suất

tốt nhất. Vì độ chính xác của dự đoán năng suất là rất quan trọng để lập kế hoạch dự án xây dựng, mô hình có hiệu suất càng tốt sẽ mang lại lợi ích đáng kể. Mô hình được chọn sẽ được tối ưu hóa tăng cường bởi thuật toán Jellyfish Search (JS) [5].

2. TỔNG QUAN NGHIÊN CỨU

2.1 Định nghĩa năng suất lao động trong xây dựng

Năng suất có thể được định nghĩa là mối quan hệ giữa đầu ra được tạo ra bởi một hệ thống tổ chức nhất định và đầu vào được hệ thống sử dụng để tạo ra đầu ra [6]. Có thể hiểu, định nghĩa về năng suất có nghĩa là tỷ lệ giữa số lượng đầu vào với số lượng đầu ra và được đo lường bằng công thức sau:

$$\text{Năng suất} = \frac{\text{Đầu ra}}{\text{Đầu vào}} \quad (1)$$

Trong đó:

Đầu vào: nguyên liệu đầu vào, điện năng, công cụ, lao động, chi phí và thời gian.

Đầu ra: khối lượng hoặc số lượng sản phẩm.

2.2 Yếu tố ảnh hưởng đến năng suất lao động

Năng suất bị ảnh hưởng bởi các yếu tố khác nhau tại công trường. Các yếu tố có thể được chia thành hai loại, đó là yếu tố bên trong và yếu tố bên ngoài.

Các yếu tố bên trong: công nghệ, tổ chức, quản lý, các vấn đề tài chính, v.v.

Các yếu tố bên ngoài: thời tiết, chính trị địa phương, sự thay đổi của nhu cầu, mối quan hệ giữa tất cả các bên liên quan, v.v.

Các yếu tố ảnh hưởng đến năng suất trong xây dựng đã là chủ đề của nhiều nghiên cứu. Tuy nhiên, đây là một vấn đề khá khó khăn trong thực tế bởi tính bất ổn định của chúng, cũng là tính duy nhất của mọi dự án xây dựng. Việc nghiên cứu các yếu tố ảnh hưởng đến năng suất lao động là cần thiết vì tầm ảnh hưởng của nó đến tính kinh tế của dự án. Vì vậy, d tìm ra một mô hình dự báo có hiệu suất cao để dự báo năng suất lao động trên công trường là rất cần thiết.

2.3 Một số nghiên cứu tương tự về năng suất lao động

Một số nghiên cứu trong nước về năng suất lao động ở Việt Nam cũng rất phong phú. Theo đó hiểu được tình hình kinh tế ở Việt Nam và các doanh nghiệp hoạt động ở lĩnh vực xây dựng, một số nghiên cứu trước cũng dự đoán tình hình năng suất xây dựng.

Đến năm 2016, PGS.TS Đinh Tuấn Hải và Hoàng Văn Trình đã trình bày một nghiên cứu trên tạp chí Kinh tế xây dựng số tháng 02/2016 với nội dung "Giải pháp nâng cao năng suất lao động trong xây dựng" nhằm đánh giá thực trạng năng suất lao động của ngành Xây dựng ở Việt Nam và mở rộng ra thế giới [7]. Từ đó đề xuất một số giải pháp nhằm thúc đẩy nâng cao chất lượng quản lý Nhà nước. Cụ thể là nâng cao năng lực quản lý, kỹ năng điều hành của nhà thầu và tư vấn giám sát, nâng cao năng suất con người và máy móc thiết bị, nâng cao hiệu quả trong kỹ thuật lựa chọn nhà thầu mục tiêu nâng cao chất lượng với tác động môi trường lao động.

Đến năm 2017, TS Lê Văn Cư và cộng sự đã trình bày một nghiên cứu về "Thực trạng một số giải pháp nhằm nâng cao năng suất lao động ngành Xây dựng", nghiên cứu này được đăng trên tạp chí Kinh tế xây dựng số tháng 02/2017 [8]. Do số liệu ngành Xây dựng không được theo dõi và ghi chép thường xuyên. Do vậy bài nghiên cứu dựa trên các số liệu thống kê bổ sung được khảo sát và thu thập theo định kỳ nhằm xác định năng suất lao động ngành Xây dựng. Nghiên cứu xác định rằng các yếu tố tổng hợp (TFP) là một trong các chỉ tiêu quan trọng để đánh giá tăng trưởng kinh tế và có vai trò quan trọng đối với việc xác định tốc độ tăng GDP. Kết quả nghiên cứu cho thấy dựa vào TFP của ngành Xây dựng người ta có thể có được những thông tin chuẩn xác và kịp thời nhằm phục vụ công tác chỉ đạo ngành Xây dựng hiện nay.

Năm 2021, Min-Yuan Cheng và các cộng sự [9] đã ứng dụng trí tuệ nhân tạo (AI) để dự báo năng suất lao động của một dự án, Mô hình kết hợp giữa máy vec tơ hỗ trợ bình phương nhỏ nhất (LSSVM), tìm kiếm sinh vật cộng sinh (SOS) cùng với phương pháp lựa chọn tính năng (FS) tạo thành một mô hình lai SOS-LSSVN-FS. Kết quả thống kê của phương pháp xác nhận chéo 10 lần chỉ ra mô hình SOS - LSSVN - FS đạt được độ chính xác cao nhất với dự đoán năng suất 3,6% sai số phần trăm tuyệt đối trung bình (MAPE), tốt hơn ít nhất 19,6% so với mô hình AI khác.

Tiếp đó năm 2022, một nghiên cứu được thực hiện bởi tác giả Dinh-Nhat Truong và Jui-Sheng Chou [10] đã kết hợp logic mờ (FA) và trình tối ưu hóa JS xây dựng thuật toán tối ưu hóa cục bộ. Thuật toán được đề xuất so sánh với những trình tối ưu hóa khác. Sau đó FAJS sử dụng tối ưu hóa các siêu tham số của hệ thống xếp chồng (SS) liên quan đến năng suất lao động, cường độ nén kết cấu nhà, cường độ dọc trục bê tông và khả năng chịu cắt. Kết quả nghiên cứu cho thấy FAJS-SS dự đoán chính xác hơn các hệ thống học máy khác trong tài liệu và được đề xuất để cung cấp trong giai đoạn lập kế hoạch và thiết kế.

3. THU THẬP VÀ XỬ LÝ CÁC SỐ LIỆU

Số liệu được thu thập từ bộ dữ liệu gốc trước đây được sử dụng từ nghiên cứu của Min-Yuan Cheng và các cộng sự [9]. Theo đó bộ dữ liệu tính toán được xử lý và thu thập với 220 bộ dữ liệu từ 02 dự án bất động sản tại Montreal, Canada trong hơn 30 tháng từ 09/2004 đến 06/2004. Bảng 1 dưới đây mô tả các tham số mô tả biến của tập dữ liệu.

4. PHƯƠNG PHÁP NGHIÊN CỨU

4.1. Phương pháp học máy

4.1.1 Mô hình đơn

Mô hình mạng nơ-ron nhân tạo ANN: Mạng nơron nhân tạo (Artificial Neural Network - ANN) là một mô hình tính toán được lấy cảm hứng từ cấu trúc và hoạt động của hệ thần kinh trong não người. ANN bao gồm một tập hợp các đơn vị tính toán gọi là "nơron" hoặc " nơron nhân tạo" được tổ chức thành các lớp (layers) và được sử dụng để mô phỏng quá trình học và xử lý thông tin. Mục tiêu của nó là tối ưu hóa trọng số để đưa ra các dự đoán hoặc đầu ra chính xác dựa trên dữ liệu mới.

Mô hình vec-tơ hỗ trợ hồi quy SVR: Mô hình Vec-tơ Hỗ trợ Hồi quy (Support Vector Regression - SVR) là một phương pháp trong machine learning được sử dụng để giải quyết bài toán hồi quy, nghĩa là dự đoán một giá trị số thực dựa trên dữ liệu đầu vào. SVR được xây dựng trên cơ sở của mô hình Vec-tơ Hỗ trợ (Support Vector Machine - SVM), mà ban đầu được phát triển cho bài toán phân loại. Theo đó điểm quan trọng trong SVR là tìm ra một hàm giả thuyết (hypothesis function) sao cho sai số giữa giá trị thực tế và giá trị dự đoán (được gọi là lỗi hồi quy) là nhỏ nhất. SVR hoạt động bằng cách tối ưu hóa khoảng cách giữa các điểm dữ liệu (được biểu diễn bởi các vectơ đặc trưng) và siêu phẳng (hyperplane) mà hàm giả thuyết xác định.

Mô hình hồi quy tuyến tính LR: Mô hình hồi quy tuyến tính (Linear Regression - LR) là một trong những mô hình cơ bản trong machine learning và thống kê, được sử dụng để mô hình hóa mối quan hệ tuyến tính giữa biến đầu vào và biến mục tiêu (đầu ra) trong bài toán hồi quy. Mô hình hồi quy tuyến tính được xem là phiên bản hoàn thiện và nâng cao so với hồi quy đơn giản. Hồi quy tuyến tính được xem là phương pháp thống kê hồi quy dữ liệu nhằm xác định mối liên kết giữa một biến phụ thuộc (có giá trị liên tục) với hai hoặc nhiều biến độc lập.

Bảng 1: Bảng thống kê bộ dữ liệu [9]

Nhóm		Biến số	Tên biến	Diễn giải	Giá trị nhỏ nhất	Giá trị lớn nhất	Giá trị trung bình	Độ lệch chuẩn
Biến đầu vào	Thời tiết	X_1	Nhiệt độ (°C)	Giá trị trung bình trong 08 giờ làm việc/ ngày	-26	25	4,04	12,05
		X_2	Độ ẩm (%)	Giá trị trung bình trong 08 giờ làm việc/ ngày	18	97	66,46	15,60
		X_3	Mưa gió	Không mưa = 0; Mưa nhỏ = 1; Mưa = 2 và Tuyết = 3	0	3	0,28	0,60
		X_4	Tốc độ gió (km/h)	Giá trị trung bình trong 08 giờ làm việc/ ngày	3	43	15,41	8,47
	Lao động	X_5	Số lượng Công nhân		8	24	16,02	5,08
		X_6	Tỷ lệ lao động (%)	Tỷ lệ lao động (công nhân phổ thông)	29	47	35,50	3,79
	Hạng mục	X_7	Loại công việc	Loại hoạt động: sàn = 1; tường = 2	1	2	1,43	0,50
		X_8	Tầng xây dựng		1	17	11,38	3,75
		X_9	Phương pháp làm việc	Dạng gỗ truyền thống = 1; dạng bay = 2	1	2	1,44	0,50
	Hoạt động	X_{10}	Công việc trực tiếp		53	86	71,13	7,45
		X_{11}	Công việc hỗ trợ		2	13	5,85	2,03
		X_{12}	Trì hoãn công việc	Ngày không hoạt động	6	43	23,04	8,12
Biến đầu ra		Y	Năng suất hàng ngày (m^2/giờ lao động)	Số lượng hoàn thành trong ngày chia cho tổng số giờ làm việc	0,82	2,53	1,58	0,35

Mô hình cây phân loại và hồi quy CART: Mô hình Cây phân loại và hồi quy (Classification and Regression Trees - CART) là một phương pháp học máy được sử dụng cho cả bài toán phân loại (classification) và bài toán hồi quy (regression). CART sử dụng cấu trúc cây để tạo ra một mô hình dự đoán dựa trên các quy tắc điều kiện. Nói cách khác, mô hình cây phân loại và hồi quy được xem như một cây nhị phân và được kết hợp và liên kết của nút gốc ban đầu với các nút quyết định và nút cuối. Trong đó nút gốc và mỗi nút đại diện cho một đặc tính. Mỗi nhánh được đại diện cho kết quả thử nghiệm. Mô hình CART là một công cụ mạnh mẽ và linh hoạt có thể được sử dụng trong nhiều tình huống khác nhau, từ phân loại đến hồi quy, nhưng điều quan trọng là cần kiểm soát overfitting và tối ưu hóa các tham số để đảm bảo hiệu suất tốt trên dữ liệu mới.

4.1.2. Mô hình hỗn hợp

Mô hình hỗn hợp Voting: Mô hình hỗn hợp Voting (Voting Ensemble) là một kỹ thuật trong machine learning được sử dụng để kết hợp dự đoán từ nhiều mô hình máy học khác nhau để tạo ra dự đoán cuối cùng. Mục tiêu của mô hình hỗn hợp Voting là tận dụng sự đa dạng của các mô hình để tạo ra dự đoán tốt hơn so với việc sử dụng một mô hình duy nhất. Mô hình này được xem là phương pháp đơn giản nhất để kết hợp nhiều mô hình phân loại đơn lẻ, Voting sử dụng quy tắc meta để kết hợp các giá trị đầu ra của mô hình đơn lẻ.

Mô hình hỗn hợp Bagging: Mô hình hỗn hợp Bagging (Bootstrap Aggregating) là một phương pháp thường được sử dụng để kết hợp nhiều mô hình cơ bản để cải thiện tính ổn định và hiệu suất dự đoán. Kỹ thuật này thường được áp dụng trong bài toán phân loại và hồi quy. Mô hình Bagging thực hiện việc sao chép các mẫu dữ liệu một cách ngẫu nhiên để thay thế tập dữ liệu ban đầu, mỗi mô hình hồi quy dự đoán các giá trị từ các mẫu dữ liệu đều độc lập.

Mô hình hỗn hợp Stacking: Mô hình hỗn hợp Stacking (Stacking Ensemble) là một kỹ thuật trong machine learning được sử dụng để kết hợp dự đoán từ nhiều mô hình cơ bản khác nhau để tạo ra dự đoán cuối cùng. Stacking là một trong những phương pháp kết hợp phức tạp hơn so với Bagging và Voting. Lợi ích chính của mô hình hỗn hợp Stacking bao gồm việc cải thiện hiệu suất dự đoán bằng cách kết hợp sự mạnh mẽ của nhiều mô hình cơ bản và tạo ra một mô hình tổng hợp có khả năng tổng quát hóa tốt hơn so với các mô hình cơ bản. Ngoài ra, Stracking rất linh hoạt trong việc lựa chọn kiểu mô hình và cấu trúc tổng hợp.

4.1.3. Các chỉ số hiệu suất đánh giá mô hình

Nghiên cứu này sửa dụng năm chỉ số hiệu suất thông dụng để đánh giá và dự đoán độ chính xác của mô hình, lần lượt là hệ số tương quan tuyến tính R, sai số trung bình tuyệt đối MAE, sai số bình phương trung bình RMSE, phần trăm sai số trung bình tuyệt đối MAPE và chỉ số tổng hợp SI.

$$R = \frac{n\sum y.y' - \left(\sum y\right)\left(\sum y'\right)}{\sqrt{n\left(\sum y^2\right)\left(\sum y\right)^2}\sqrt{n\left(\sum y'^2\right)\left(\sum y'\right)^2}} \quad (2)$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|y - y'\right| \quad (3)$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(y' - y\right)^2} \quad (4)$$

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y - y'}{y}\right| \quad (5)$$

$$SI = \frac{1}{m}\sum_{i=1}^{m}\left(\frac{P_i - P_{min,i}}{P_{max,i} - P_{min,i}}\right) \quad (6)$$

4.1.4. Lý thuyết tối ưu hóa jellyfish search

Tối ưu hóa Jellyfish Search (JS) được công bố năm 2021 bởi Chou và Trương [10] là một phương pháp tối ưu hóa lấy cảm hứng từ thiên

nhiên dựa trên bản năng của loài sứa trong tự nhiên. Theo đó ban đầu khi thực hiện hành vi kiếm thức ăn chúng dựa vào cảm nhận từ dòng chảy của đại dương, sau đó đi theo dòng hải lưu nhằm tìm kiếm thức ăn là các loại sinh vật phù du. Với khoảng thời gian nhất định chúng tụ hợp lại ngày càng di chuyển trong bầy sứa. Kết thúc thời gian, chuyển động của sứa sẽ chuyển sang chuyển động thụ động và chuyển động tích cực bên trong bầy để khai thác. Về cuối, quá trình sứa nở hoa diễn ra, đó gọi là giai đoạn tối ưu. Mặc dù là một thuật toán mới, JS đã nhanh chóng chứng minh tính hữu dụng trong việc giải quyết các bài toán tối ưu hóa liên tục và rời rạc.

Dòng hải lưu: dòng hải lưu được thu hút bởi đàn sứa do có chứa một lượng lớn chất dinh dưỡng. Để xác định hướng của dòng hải lưu (trend) bằng cách lấy trung bình tất cả các vectơ từ mỗi con sứa trong đại dương đến các con sứa ở những vị trí tốt nhất.

Bầy sứa: theo nguyên lý hoạt động của bầy sứa, ban đầu khi hình thành chúng đều hoạt động thụ động (loại A), theo thời gian chúng sẽ di chuyển trong bầy và hoạt động một cách chuyển động (loại B). Khi đó chuyển động loại A là sự chuyển động của sứa xung quanh vị trí của chúng. Vị trí cập nhật tương ứng của mỗi con sứa được hình thành theo công thức tính sau:

Cơ chế kiểm soát thời gian: để điều chỉnh sự chuyển động của sứa giữa việc di chuyển theo dòng hải lưu và bên trong bầy sứa, thì cơ chế điều khiển thời gian bao gồm chức năng điều khiển thời gian c(t) và hằng số Co. Theo đó khi giá trị c(t) < Co, sứa sẽ di chuyển bên trong bầy và ngược lại c(t) > Co sứa di chuyển theo dòng hải lưu. Với Co là giá trị không được xác định trước và việc kiểm soát thời gian thay đổi ngẫu nhiên từ 0 đến 1. Khi chọn Co = 0,5 là giá trị trung bình của 0 và 1.

4.1.5. Sơ đồ thích ứng mô hình lai JS-Bagging-ANN

Mô hình lai dự đoán được phát triển trong Matlab cho phép người dùng vẽ đồ thị các hàm, thao tác với ma trận, thực hiện các thuật toán. Nhờ khả năng thực hiện các phép tính toán nâng cao, Matlab được sử dụng ở nghiên cứu để tích hợp thuật toán JS với mô hình tốt nhất trong các sơ đồ cơ sở và tổng hợp, đó là đóng gói ANN (Bagging-ANN), để phát triển một mô hình hỗn hợp, cụ thể là JS-Bagging-ANN (Hình 1) cho thấy cấu trúc của mô hình lai này, tương ứng như sau:

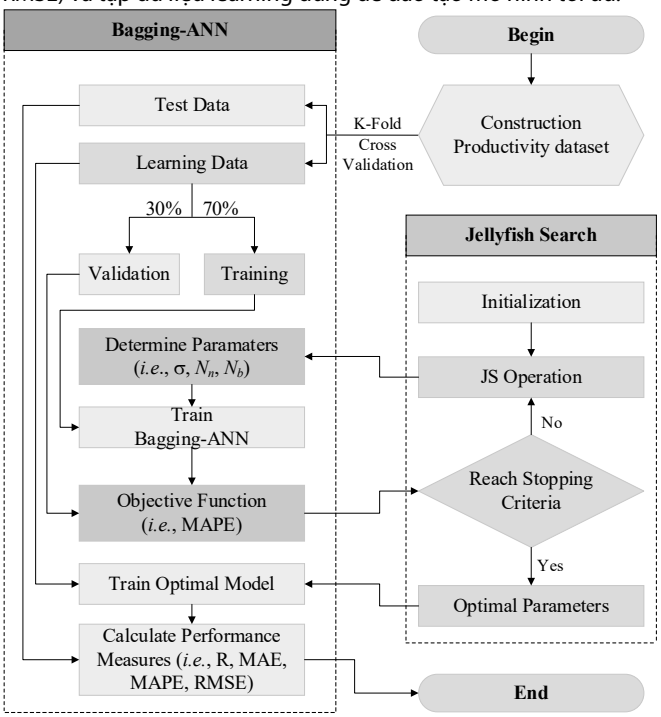
Quy trình chạy mô hình: Phần này mô tả quá trình thích ứng mô hình được đề xuất và mỗi phần sẽ được giải thích theo từng bước như hình 1 cho thấy sơ đồ chi tiết của cả quá trình cho đến mô phỏng mô hình mới. Quy trình chia thành hai phần gồm: phần đầu tiên chạy mô hình với bảy bước và phần hai ứng dụng thuật toán JS với bốn bước. Theo đó, hai cả hai phần đều có sự liên kết lẫn nhau để tạo ra mô hình mới tối ưu.

Xử lý dữ liệu: Giai đoạn này cơ sở dữ liệu được chuẩn hóa làm đầu vào trước khi sử dụng cho mô hình suy luận để giảm biên lỗi trong hiệu suất dự đoán. Cơ sở dữ liệu từ đầu vào được chuyển đổi theo cách tỷ lệ tuyến tính thành một phạm vi từ [0-1]. Hàm được dùng để chuẩn hóa dữ liệu theo công thức. Bộ dữ liệu xử lý sau đó thực hiện thông qua xác thực chéo 10 lần.

Bộ dữ liệu learning và dữ liệu test: Theo nghiên cứu này, xác nhận chéo 10-lần sử dụng để giảm thiểu sự sai lệch tiềm ẩn trong phân vùng dữ liệu và cho kết quả 2 tập dữ liệu, dữ liệu test và dữ liệu learning. Với bộ dữ liệu learning để làm giảm vượt trội quá mức của mô hình do đó việc tạo ra tập dữ liệu với 90% dữ liệu được chuẩn hóa được tạo ra từ 10 bộ dữ liệu khác nhau lấy một cách ngẫu nhiên (10-fold). 10% dữ liệu còn lại được sử dụng cho bộ dữ liệu test nhằm xác định hiệu quả và độ chính xác của mô hình.

Bộ dữ liệu training: Bộ dữ liệu đào tạo nhận được từ đầu ra của phân vùng dữ liệu bằng cách xác thực chéo k-lần và được áp dụng để thiết lập mô hình suy luận. Theo đó bộ dữ liệu đào tạo sử dụng để tiến hành thực hiện chạy mô hình và xác định thông số thông qua xác thực chéo từ bộ dữ liệu training được lấy ngẫu nhiên tương ứng cho 70% bộ dữ liệu.

Bộ dữ liệu validation: Tương tự 30% tập dữ liệu còn lại được sử dụng cho việc xác định thông số như là MAPE. Với tập dữ liệu test được sử dụng để tính toán các biện pháp thực hiện (như là MAPE, RMSE) và tập dữ liệu learning dùng để đào tạo mô hình tối ưu.



Hình 1. Sơ đồ mô hình JS- Bagging-ANN

Thuật toán jellyfish search: Để tìm kiếm các thông số tối ưu của mô hình học máy, JS tìm tham số này bằng việc sử dụng hàm mục tiêu. Theo đó với mỗi vòng lặp sẽ tìm kiếm một bộ tham số phù hợp, bộ tham số không phù hợp sẽ được loại bỏ và thay thế. Quá trình trên diễn ra lặp đi lặp lại để thuật toán JS ghi nhớ và tạo ra giá trị mục tiêu ở vị trí khác. Do tính chất diễn biến liên lục để tạo ra một mô hình tối ưu với độ chính xác cao.

Hàm mục tiêu: Theo đó hàm mục tiêu được xác định bởi các tham số thuật toán của JS bao gồm số cá thể = 30, số lần lặp tối đa = 40. Các hàm mục tiêu được trình bày dưới dạng công thức:

$$f(N_n, N_b) = \text{MAPE}^{\text{Training process}}_{\text{Validation data}} \quad (7)$$

Trong đó:

N_n: số lượng tế bào thần kinh.

N_b: số lượng túi.

4.1.6. Thông số mô hình dự đoán JS-Bagging-ANN

Từ hàm mục tiêu mô hình dự đoán JS-Bagging-ANN được lập trình và tính toán trong Matlab với các thông số tương ứng ở bảng 2 như sau:

Bảng 2: Cài đặt thông số mô hình JS-Bagging-ANN

Mô hình	Tham số	Giá trị
JS-Bagging-ANN	Số vòng lặp	40
	Số lượng dân số	30
	Số *lượng túi* (N_b)	[1, 10]
	Số lượng tế bào thần kinh (N_n)	[1, 100]

Dựa vào bảng 2 nghiên cứu tính toán được thông số tối ưu và đánh giá kết quả sau khi xử lý nhằm xác nhận giải pháp tốt nhất cho mô hình dự đoán năng suất xây dựng.

5. KẾT QUẢ MÔ HÌNH

5.1. So sánh và đánh giá các chỉ số hiệu suất của mô hình đơn và mô hình hỗn hợp

Phần mềm mã nguồn mở Waikato Environment for Knowledge Analysis (Weka) được sử dụng để phân tích các bộ dữ liệu cho các mô hình đơn và hỗn hợp, các kết quả sẽ được so sánh là lựa chọn để tìm ra mô hình phù hợp nhất thông qua các chỉ số hiệu suất (bảng 3). Dựa trên mô hình đơn và mô hình hỗn hợp các kết quả cho thấy mô hình hỗn hợp bagging-ANN cho ra hiệu suất cao với R = 0,976, MAE = 0,060, RMSE = 0,077, MAPE = 20,790%, SI = 0,454 và Rank =2.

Bảng 3: Bảng kết quả tổng hợp các mô hình tính toán

STT	Mô hình	R	MAE	RMSE	MAPE (%)	SI	Rank
I. Mô hình đơn - Single							
1	**ANN**	**0,969**	**0,068**	**0,088**	**23,766**	**0,595**	**10**
2	SVR	0,953	0,083	0,109	28,922	0,876	22
3	LR	0,956	0,085	0,104	29,439	0,858	20
4	CART	0,947	0,089	0,114	31,140	0,979	23
II. Mô hình hỗn hợp - Voting							
5	ANN+SVR	0,955	0,083	0,105	28,926	0,853	18
6	ANN+LR	0,973	0,065	0,082	22,717	0,529	4
7	ANN+CART	0,971	0,066	0,085	22,924	0,556	8
8	SVR+LR	0,955	0,083	0,105	28,926	0,853	18
9	SVR+CART	0,962	0,076	0,097	26,284	0,727	16
10	LR+CART	0,962	0,076	0,096	26,508	0,727	15
11	ANN+SVR+LR	0,969	0,069	0,088	23,980	0,602	11
12	**ANN+LR+CART**	**0,973**	**0,065**	**0,082**	**22,675**	**0,528**	**3**
13	ANN+SVR+CART	0,972	0,065	0,083	22,767	0,538	5
14	SVR+LR+CART	0,962	0,076	0,096	26,422	0,726	14
15	ANN+SVR+CART+LR	0,971	0,067	0,085	23,481	0,566	9
III. Mô hình hỗn hợp - Bagging							
16	**ANN**	**0,976**	**0,060**	**0,077**	**20,790**	**0,454**	**2**
17	SVR	0,954	0,082	0,108	28,517	0,858	21
18	LR	0,956	0,083	0,104	28,999	0,844	17
19	CART	0,968	0,071	0,090	24,797	0,631	12
IV. Mô hình hỗn hợp - Stacking							
20	ANN - (ANN+SVR+CART+LR)	0,949	0,093	0,116	32,214	1,000	24
21	SVR - (ANN+SVR+CART+LR)	0,972	0,067	0,084	23,263	0,555	7
22	**LR - (ANN+SVR+CART+LR)**	**0,973**	**0,066**	**0,083**	**23,008**	**0,539**	**6**
23	CART - (ANN+SVR+CART+LR)	0,961	0,074	0,098	25,897	0,723	13
V. Mô hình lai							
24	**JS-Bagging-ANN**	**0,984**	**0,043**	**0,009**	**2,707**	**0,000**	**1**

Bảng 4: Bảng so sánh hiệu suất của mô hình đề xuất và các nghiên cứu đã công bố

Mô hình	R	MAE	RMSE	MAPE (%)	SI	Rank
SOS-LSSVM$_{FS}$ [9]	0,979	0,056	0,072	3,670	1,00	3
FAJS-SS$_{LSSVR}$ [10]	0,984	0,045	0,009	2,790	0,07	2
SOM [11]	-	-	-	25,050	-	4
JS-Bagging ANN (Mô hình đề xuất)	**0,984**	**0,043**	**0,009**	**2,707**	**0,00**	**1**

5.2. Tối ưu hóa mô hình được chọn bằng JS và xác thực chéo để giảm thiểu các sai lệch tiềm ẩn

Với kết quả thu được, mô hình Bagging ANN được đánh giá là mô hình tối ưu nhất trong tổng số 23 mô hình đã phân tích (chưa bao gồm mô hình lai). Thuật toán tối ưu hóa JS được sử dụng để tối ưu hóa các tham số nằm cải thiện hiệu suất của mô hình Bagging ANN. Để giảm thiểu sự sai lệch tiềm ẩn trong phân vùng dữ liệu, xác nhận chéo 10 lần được sử dụng. Trong nghiên cứu này, 10% tập dữ liệu được sử dụng làm dữ liệu xác nhận để kiểm tra và 90% tập dữ liệu còn lại được sử dụng để đào tạo. Quá trình này sau đó lặp lại 10 lần và kết quả trung bình của chúng sử dụng làm kết quả cuối cùng. Kết quả thể hiện ở bảng 3 cho thấy giai đoạn test cho ra các chỉ số đánh giá hiệu suất của mô hình lai đề xuất JS-Bagging ANN lần lượt là: R = 0,984; MAE = 0,043; RMSE = 0,009; MAPE = 2,707%.

5.3. Đánh giá và so sánh kết quả của mô hình đề xuất với các nghiên cứu trước

Để đánh giá hiệu suất mô hình đề xuất JS-Bagging ANN, kết quả cuối cùng của mô hình được sử dụng để so sánh với các kết quả từ các nghiên cứu đã công bố trước đây (bảng 4).

Kết quả cho thấy mô hình đề xuất JS-Bagging-ANN được xem là mô hình lai tối ưu nhất so với ba mô hình nghiên cứu còn lại, theo đó với mô hình JS-Bagging-ANN thấp nhất với chỉ số đánh giá hiệu suất R = 0,984; MAE = 0,043; RMSE = 0,009; MAPE = 2,707% đứng thứ nhất, xếp vị trí thứ hai FAJS-SS_{LSSVR} với R = 0,984; MAE = 0,045; RMSE = 0,009; MAPE = 2,790% và vị trí thứ ba mô hình SOS-$LSSVM_{FS}$ với R = 0,979; MAE = 0,056; RMSE = 0,072; MAPE = 3,670%. Giá trị của mô hình SOM quá cao khi MAPE = 25,050% vì vậy nghiên cứu được bỏ qua không tiến hành tính toán hệ số SI.

6. KẾT LUẬN

Kết quả nghiên cứu trình bày 04 mô hình đơn và 03 mô hình hỗn hợp lần lượt gồm: ANN, SVR, LR, CART và Voting, Bagging, Stacking về năng suất lao động được thực hiện thông qua 220 bộ dữ liệu thu thập được trong hai dự án ở Montreal Canada. Qua đó mô hình hỗn hợp Bagging ANN với các giá trị R, MAE, RMSE, MAPE thấp nhất cho cả quá trình thử nghiệm (test) với giá trị lần lượt là R = 0,976; MAE = 0,060; RMSE = 0,077; MAPE = 20,790% là mô hình tối ưu nhất và ổn định nhất với chỉ số SI = 0,454 so với 22 mô hình còn lại trong quá trình thử nghiệm.

Mô hình có hiệu suất cao nhất sẽ được kết hợp với thuật toán tối ưu hóa JS. Xác thực chéo 10 lần được triển khai để đánh giá hiệu suất của mô hình đã phát triển. Các kết quả khả quan với các chỉ số hiệu suất lần lượt là R = 0,984; MAE = 0,043; RMSE = 0,009; MAPE = 2,707% và chỉ số SI = 0,000 đã chứng tỏ được sự vượt trội của mô hình JS-Bagging ANN.

Kết quả nghiên cứu sẽ được đánh giá lại một lần nữa bằng cách so sánh tương đương với các nghiên cứu đã được công bố trước đây. Mô hình đề xuất JS-Bagging ANN đã chứng tỏ được hiệu suất tính toán tối ưu khi đạt kết quả tốt nhất, điều này chỉ ra rằng mô hình JS-Bagging ANN rất phù hợp dùng để dự báo năng suất lao động trên công trường.

Các nghiên cứu trong tương lai có thể ứng dụng kết quả mô hình JS-Bagging ANN để so sánh, mở rộng thêm mô hình hoặc tích hợp tính năng, phương pháp học máy và các thuật toán khác nhằm giải quyết các bài toán quan trọng trong xây dựng như: dự báo tiến độ của dự án, dự đoán chi phí dự phòng v.v.

TÀI LIỆU THAM KHẢO

[1] Grau, D., et al., *Assessing the impact of materials tracking technologies on construction craft productivity.* Automation in Construction, 2009. **18**(7): p. 903-911.

[2] Cheng, M.-Y., et al., *Predicting productivity loss caused by change orders using the evolutionary fuzzy support vector machine inference model.* Journal of Civil Engineering and Management, 2015. **21**: p. 881-892.

[3] Gong, M., et al., *Gradient boosting machine for predicting return temperature of district heating system: A case study for residential buildings in Tianjin.* Journal of Building Engineering, 2020. **27**: p. 100950.

[4] Saleem, M., *Assessing the load carrying capacity of concrete anchor bolts using non-destructive tests and artificial multilayer neural network.* Journal of Building Engineering, 2020. **30**: p. 101260.

[5] Chou, J.-S. and D.-N. Truong, *A novel metaheuristic optimizer inspired by behavior of jellyfish in ocean.* Applied Mathematics and Computation, 2021. **389**: p. 125535.

[6] Hannula, M., *Total productivity measurement based on partial productivity ratios.* International Journal of Production Economics, 2002. **78**(1): p. 57-67.

[7] Hải, Đ.T. and H.V. Trình, *Giải pháp nâng cao năng suất lao động trong xây dựng*. Tạp chí Kinh tế Xây dựng, 2016(02), Trang 36-40.

[8] Cư, L.V., L.V. Long, and V.Q. Thắng, *Thực trạng một số giải pháp nhằm nâng cao năng suất lao động ngành xây dựng*. Tạp chí Kinh tế Xây dựng, 2017(02), Trang 35-41.

[9] Cheng, M.-Y., M.-T. Cao, and A.Y. Jaya Mendrofa, *Dynamic feature selection for accurately predicting construction productivity using symbiotic organisms search-optimized least square support vector machine.* Journal of Building Engineering, 2021. **35**: p. 101973.

[10]Truong, D.-N. and J.-S. Chou, *Fuzzy adaptive jellyfish search-optimized stacking machine learning for engineering planning and design.* Automation in Construction, 2022. **143**: p. 104579.

[11] Oral, E.L. and M. Oral, *Predicting construction crew productivity by using Self Organizing Maps.* Automation in Construction, 2010. **19**(6): p. 791-797.