SOCAR Proceedings

Refining & Petrochemistry

journal home page: http://proceedings.socar.az

METAHEURISTIC-ASSISTED DL AND ML APPROACHES INTEGRATED WITH DEA MODELS TO PREDICT THE PERFORMANCE OF PETROLEUM REFINERIES

Fereshteh Koushki*[1], Mona Naghdehforoushha[2]

[1]*Islamic Azad University, Qazvin, Iran*

[2]*Islamic Azad University, Takestan, Iran*

A B S T R A C T

Performance evaluation of oil exploitation centers is crucial in both economic and environmental aspects. Data Envelopment Analysis (DEA), as a powerful mathematical optimization model, is commonly used to measure the efficiency of multi-input/output decision making units (DMUs). However, computational limitations in solving large-scale evaluation problems have motivated this research to combine DEA models with artificial intelligence (AI) techniques for estimating the efficiency scores of oil refineries. Machine learning (ML) and deep learning (DL) methods can be employed to address the challenges associated with large-scale mathematical optimization models. Additionally, metaheuristic methods can be utilized to improve hyperparameters of ML and DL algorithms. In this study, metaheuristic-assisted DL and ML approaches were integrated with mathematical linear programming to estimate the performance of oil refineries. First, the efficiency scores of oil refineries calculated by DEA model were used as training and testing datasets for the DL and ML models - Long Short-Term Memory (LSTM) and Multi-Layer Perceptron (MLP)-. Then, the metaheuristic algorithms - Grid Search and Particle Swarm Optimization (PSO) - were used to optimize the hyperparameters of the LSTM and MLP models. The results indicated that tuning the hyperparameters of the MLP and LSTM algorithms significantly reduced prediction errors. Additionally, LSTM-based algorithms had higher prediction accuracy compared to MLP-based algorithms. Furthermore, the LSTM-PSO approach predicted the efficiency scores with the highest accuracy value of 96%.

Keywords: oil refinery; performance evaluation; Data Envelopment Analysis (DEA); Machine Learning (ML); Deep learning (DL); Metaheuristics.

Date submitted: 28.08.2025 **Date accepted:** 26.01.2026

1. Introduction

Performance evaluation of the oil exploitation sector helps identify and improve operational inefficiencies, especially in the consumption of natural resources and the production of pollutant emissions. Efficient processing of oil and gas products is also crucial in countries with petroleum-based economies. DEA is a nonparametric approach used to measure the efficiency of multi-input/output DMUs [1]. However, computational limitations in solving large-scale evaluation problems have motivated this research to employ AI techniques. This study combines ML and DL approaches with a DEA model to address these computational limitations and predict the efficiency scores of oil refineries.

Metaheuristic methods are commonly utilized to improve hyperparameters of ML and DL algorithms. Optimizing hyperparameters directly impacts the accuracy and generalization of the model. Hyperparameters tuning enhances the reliability and functionality of ML algorithms [2]. In this paper, a metaheuristic algorithm - PSO- is used to enhance the hyperparameters of MLP and LSTM algorithms.

The contributions of this study are as follows:

I. Evaluating the performance of oil exploitation centers is crucial in both economic and environmental aspects. Implementing the methods proposed in this paper offers valuable managerial insights to enhance both economic and environmental performance effectively.

II. This study utilizes ML and DL approaches to address the computational limitations of DEA models and accurately estimate the efficiency scores of oil refineries.

III. This paper applies metaheuristics to enhance the hyperparameters of ML and DL algorithms. Adjusting hyperparameters directly impacts the accuracy and generalization of these algorithms.

IV. This paper, for the first time, combines a DEA method with metaheuristic-assisted ML and DL algorithms to estimate the efficiency of oil refineries.

In the following section, the recent studies that eval-

**E-mail:* f_koushki13@iau.ac.ir

http://dx.doi.org/10.5510/OGP20260201205

uated oil refineries using DEA approaches are reviewed. Preliminaries for the DEA, MLP, LSTM, Grid Search, and PSO approaches are described in Section 3. Additionally, this section details the combination of the predictive algorithms with a DEA model to estimate the efficiency scores of oil refineries. Section 4 applies the DEA model results as datasets for the MLP, LSTM, and their metaheuristic-improved versions to predict the efficiency scores of oil refineries in Iran.

2. Literature review

Processing crude oil to produce petroleum products involves complex operations, significant costs, and high energy consumption. Therefore, it is crucial to develop basic DEA models that can adapt to real-world situations for optimizing the performance of oil refineries. Recent studies have combined DEA methods with other mathematical models, including multi-objective and regression analysis. Additionally, basic DEA models have been customized to evaluate oil exploitation plants across multiple time periods. Furthermore, researchers have considered the presence of imprecise data and introduced fuzzy DEA models. Below are recent studies that have explored different types of DEA model development to assess the performance of oil refineries.

Li et al. [3] evaluated the sustainability of oil enterprises using technical and relative efficiency scores obtained by solving the input-oriented DEA model. Tavana et al. [4] used a multi-objective dynamic network DEA approach to calculate the efficiency of oil refineries in fuzzy environments. Mohammed Atris [5] utilized DEA approaches to rank oil and gas refineries according to their efficiency measures. Sánchez Robles et al. [6] examined the efficiency of European oil companies using DEA methods based on structural, environmental, financial, and economic factors. Oliveira et al. [7] applied DEA along with the Malmquist Index to assess oil refineries.Ngeti and Mafukidze [8] used the DEA-based efficiency scores in a regression model to assess oil companies across structural, environmental, and financial dimensions. Hasanvand et al. [9] proposed a fuzzy network DEA model to assess the oil exploitation centers in Iran.

The complexity of real-life production systems results in large-scale mathematical models for performance evaluation. To address computational issues, recent studies have explored combining DEA models with ML approaches [10-14]. This study, for the first time for refineries and the data configuration, combines a DEA model with metaheuristic-powered ML and DL methods to estimate the efficiency scores of oil refineries. By applying metaheuristic approaches to enhance the hyperparameters of ML and DL algorithms, their functionality, particularly prediction accuracy, is improved.

3. Methodology

3.1. DEA model

An oil refinery is considered a DMU. The input and output vectors of DMU_j (j=1,...,n) are denoted by X_j and Y_j, respectively. The classic input-oriented DEA model to calculate the efficiency score of DMU_o is shown as Model (1) [1].

$$\min \theta$$
$$s.t. \sum_{j=1}^{n} \lambda_j X_j \le \theta X_o, \quad \sum_{j=1}^{n} \lambda_j Y_j \ge Y_o, \quad \lambda \ge 0 \tag{1}$$

DEA models have limitations in assessing the efficiency scores of numerous DMUs. The computational workload increases significantly when determining the score of a new DMU, as it requires recalculating the efficiency scores of all existing DMUs. Instead, the DEA model is solved for a limited number of DMUs, and then the results are used as datasets in ML and DL algorithms to predict the score of a new DMU. In this paper, ML and DL algorithms, along with their metaheuristic-enhanced versions, are utilized to predict efficiency scores.

3.2. MLP, LSTM, Grid Search, and PSO algorithms

1. In the MLP approach, data transferred from input layer to the hidden layers is processed by applying an activation function [15]. The processed data is then used by the output layer to produce the final output [16].
2. LSTM networks are DL models that are particularly suitable for sequence and time-series data. They are capable of learning long-term dependencies and retaining information across time steps, making them superior to standard feed-forward networks in many predictive tasks [17].
3. The Grid Search method systematically tries all combinations of parameter values to find the one that gives the best performance. The hyper parameter values are generally independent of one another. Therefore, the search process can be made parallel [18].
4. PSO navigates a multi-dimensional search space with a swarm of particles to obtain the optimal solution. The algorithm begins with a population of particles, which are randomly generated within the search space [19]. In each iteration, the position (solution) of particles is updated based on the Personal Best and Global Best [20].

4. Numerical results

4.1. Configuring and optimizing the hyper-parameters of the ML algorithms by metaheuristic methods

The MLP and LSTM are strong ML and DL algorithms used to estimate efficiency scores. The PSO model is applied for hyper parameter tuning of the aforementioned algorithms. Traditional hyper-parameter optimization methods such as Grid Search and Random Search are computationally intensive and often inefficient in exploring large, complex search spaces [21]. To address these limitations, metaheuristic optimization algorithms, particularly PSO, have gained traction due to their ability to efficiently navigate high-dimensional and nonlinear spaces with fewer function evaluations [22]. Compared to PSO, other metaheuristic algorithms such as Genetic Algorithm (GA) and Ant Colony Optimization (ACO) offer diverse search mechanisms but often exhibit instability in deep learning hyper-parameter tuning due to the complexity of configuring parameters such as mutation rate, selection rate, and pheromone intensity [23]. Similarly, although Simulated Annealing (SA) and Differential Evolution (DE) can escape local minima, they generally suffer from slower convergence and less consistent performance relative to PSO [24, 25].

Unlike traditional grid-based or manual tuning techniques, PSO explores the continuous hyper-parameter space more effectively by balancing exploration and exploitation, thereby avoiding local optima and converging toward near-optimal solutions. Its simplicity, minimal parameter tuning requirements, and rapid convergence make it particularly well-suited for hyper parameter optimization in machine learning. Moreover, the continuous nature of the hyper parameter search space in models like MLP and LSTM complements the inherent characteristics of the PSO algorithm [26]. In our implementation, the PSO algorithm was configured with a population size of 20, a maximum of 30 iterations, cognitive and social coefficients each set to 1.5, and an inertia weight fixed at 0.7. In parallel, Grid Search was also employed to fine-tune the hyperparameters of the MLP model. Despite its computational cost, Grid Search conducts an exhaustive search over a predefined grid of parameter values, ensuring that the best combination within the grid is identified. This process uses 3-fold cross-validation (cv=3) to evaluate each parameter configuration. The final model is selected based on the lowest mean squared error and is subsequently assessed on the test dataset. The key parameters of the PSO and Grid Search algorithms are described in tables 1 and 2. To avoid bias arising from unequal search effort, both PSO and Grid Search were applied to the same hyper parameter space. PSO explores this space continuously using multiple particles and iterations, whereas Grid Search evaluates a discretized grid sampled from the same parameter ranges. Consequently, the comparison focuses on the search strategy rather than the absolute number of evaluated configurations.

MLP includes one input layer, one hidden layer with 20 neurons and ReLU activation function, and one output layer with a linear activation function. L2 regularization is used to prevent overfitting. LSTM consists of one layer with 50 units and one output layer. Dropout regularization with a rate of 0.2 is applied to prevent overfitting. The optimizer 'Adam' with a rate of 0.001 is used in both MLP and LSTM.

The following table summarizes the hyper-parameters and corresponding search ranges used in the PSO-optimized and Grid Searched-optimized variants of the MLP and LSTM models (table 3).

4.2. Estimating efficiency scores using MLP, LSTM, PSO-MLP, and PSO-LSTM algorithms

Model (1) is solved to generate the efficiency scores of oil refineries in Iran for 297 weeks (9 oil refineries over 33 weeks). Inputs and outputs of an oil refinery are listed below:

- Inputs:
  x_1: Staff (Person)
  x_2: Energy payment cost ($10)
  x_3: Other costs ($10,000)
  x_4: Oil (Barrels)
  x_5: Gas (Liters)
- Outputs:
  y_1: Pure oil (Barrels)
  y_2: Pure gas (Liters)
  y_3: Pollutant emissions (Kilograms)

Pollutant emissions are undesirable outputs and we consider the inverse of them as desirable outputs. Model (1) was run on the GAMS software platform, and the efficiency scores were used as a dataset.

Temporal validation & leakage control:

- Splitting was done over time: the first 27 weeks (for each refinery) – train/val; the last 6 – test (no shuffling).
- For LSTM, a rolling-origin scheme: trained on [1…t], tested on t+1, step Δ; metrics aggregated.
- The DEA frontier for test weeks was computed without using test observations (recomputed on train sets) to avoid target leakage.

The testing dataset is shown in table 4.

The efficiency scores related to the testing dataset and their predicted values by the MLP, LSTM, MLP-Grid Search, MLP-PSO, LSTM-Grid Search, and LSTM-PSO algorithms are shown in table 5. The algorithms were run on the Python software platform.

The Accuracy (<10%), defined as the proportion of predictions within a 10% deviation from actual values, Mean Absolute Error (MAE), R-squared (R^2), and Mean Squared Error (MSE) values of the aforementioned algorithms related to the testing data are presented in table 6.

Table 1

Key Parameters of the PSO Algorithm

Parameter	Description
Number of particles	The number of individuals in the swarm (population size)
Number of iterations	The number of times the swarm updates its position to search for optima
Inertia weight (w)	Controls how much of the previous velocity is retained in the new update
Cognitive coefficient(c1)	Represents the particle's own tendency to return to its personal best position
Social coefficient(c2)	Represents the particle's tendency to follow the global best position found by the swarm

Table 2

Key Parameters of the Grid Search Algorithm

Parameter	Description
estimator	The machine learning model whose hyperparameters will be optimized
Param_grid	Dictionary of parameters and possible values to search over
n_jobs	Number of CPU cores to use
refit	It true, refits the best model on the entire dataset after search
verbose	Helps monitor progress for large searches

Table 3

Models' hyper parameters

Model	Hyper-parameter	Range / Value	Description
MLP	hidden_layer_sizes	50	Number of neurons in the hidden layer
MLP	Max iterations	2000	Maximum number of training iterations
MLP	learning_rate_init	0.001	Initial learning rate for weight updates
MLP	alpha	0.0001	L2 regularization term (to prevent overfitting)
MLP	solver	'Adam'	Specifies the optimization algorithm used to update weights during training.
MLP	activation	'ReLU'	Determines the activation function for the hidden layers
MLP	random_state	42	Controls the randomness for weight initialization, data shuffling, and other stochastic components
MLP-PSO	hidden_layer_sizes	[20, 150]	Optimized number of neurons in hidden layer
MLP-PSO	learning_rate_init	[1e-4, 1e-2]	Optimized learning rate
MLP-PSO	alpha	[1e-6, 1e-3]	L2 regularization
MLP-GridSearchCV	hidden_layer_sizes	{20, 50, 100, 150}	Number of neurons
MLP-GridSearchCV	learning_rate_init	{1e-4, 5e-4, 1e-3, 5e-3, 1e-2}	Initial learning rate
MLP-GridSearchCV	alpha	{1e-6, 1e-5, 1e-4, 1e-3}	L2 regularization
LSTM	Learning_rate	0.001	Learning rate used in the optimizer in keras platform
LSTM	Units	64	The number of memory units (cells) in the LSTM layer
LSTM	Optimizer	'Adam'	The optimization algorithm used for training
LSTM	Dropout	0.2	Dropout rate
LSTM-PSO	Units	[20, 150]	The number of memory units (cells) in the LSTM layer
LSTM-PSO	Learning_rate	[1e-4, 1e-2]	Learning rate used in the optimizer in keras platform
LSTM-PSO	Dropout	[0.1,0.5]	Dropout rate
LSTM-GridSearchCV	Units	{32, 64, 96, 128}	Number of LSTM cells per layer
LSTM-GridSearchCV	Dropout	{0.1, 0.3, 0.5}	Dropout rate
LSTM-GridSearchCV	Learning_rate	{1e-4, 5e-4, 1e-3, 5e-3, 1e-2}	Learning rate or the optimizer

Table. 4

Data related to the testing dataset

Week	x_1	x_2	x_3	x_4	x_5	y_1	y_2	y_3	Score
1	270	229	4547	20616352	14762	9582053	1247	15405	.664
2	173	85	3661	14686386	12103	9353053	1241	12071	1
3	173	85	3724	16729560	12186	9486386	1245	13705	.880
4	177	89	3731	14880503	12396	9514720	1245	14071	.846
5	270	219	4548	20622642	14594	9586387	1247	15738	.671
6	250	180	4146	19352201	14311	9553053	1245	15171	.711
7	253	139	4251	20748428	13971	9649387	1247	16938	.705
8	198	170	4190	18855346	13408	9683053	1256	17538	.736
9	173	85	3717	14811321	12180	9469720	1245	13405	.900
10	265	200	4708	19477987	13421	9920054	1267	21739	.741
11	206	201	4709	18855346	13922	9925921	1267	22235	.726
12	257	202	4710	18861635	13994	9913354	1267	22505	.709
13	218	203	4711	18867925	13870	9933354	1267	22739	.722
14	177	89.5	3738	14911950	12396	9519720	1245	14438	.823
15	281	226	4745	20603774	14792	9548386	1245	14805	.671
16	329	257	5601	22630675	15997	16640091	1586	417522	1
17	173	85	3703	14748428	12150	9437686	1241	12904	.935
18	269	273	4700	18798742	13813	9753054	1264	19062	.713
19	200	197	4701	18805031	13470	9773387	1264	19405	.738

Table. 4 (continued)

Data related to the testing dataset

Week	x_1	x_2	x_3	x_4	x_5	y_1	y_2	y_3	Score
20	173	85	3689	14716981	12182	9419720	1241	12171	.991
21	209	204	4712	18874214	13925	9953054	1268	22739	.726
22	260	205	4713	19509434	13888	9986387	1268	23072	.719
23	271	216	4714	18886792	13427	9986387	1268	23406	.743
24	212	200	4715	18893082	13428	10019721	1268	23432	.752
25	283	242	4670	20660377	14709	9686387	1256	17738	.668
26	269	220	4360	20666667	14950	9716720	1264	18072	.688
27	277	215	4220	18786164	13821	9719720	1264	18392	.717
28	244	199	4716	18899371	13429	10026387	1268	23406	.746
29	214	199	4717	19974843	13440	10053054	1268	23406	.753
30	233	210	4718	20610063	13431	10053054	1269	23406	.748
31	216	211	4719	19547170	13932	10086388	1269	23406	.732
32	217	221	4720	19622642	13413	10086388	1269	23406	.755
33	173	85	3668	14691824	12179	9361733	1241	12071	1
34	193	122	4859	16020102	13223	11076689	1302	60347	.882
35	226	221	4729	19610063	14452	10353055	1269	30073	.723
36	277	202	4730	19371069	13443	10386388	1269	30739	.765
37	298	213	4731	18993711	13994	10419721	1269	31399	.736
38	202	126.5	4922	16173435	13328	11367389	1302	73700	.898
39	180	143.5	5160	16707103	13663	12519723	1375	127344	.971
40	179	136.5	5062	16471436	13583	12053056	1346	104010	.944
41	180	115.5	4768	15822435	13092	10653055	1301	41846	.853
42	189	120	4831	15940102	13092	10953055	1302	53680	.874
43	207	129	4957	16256769	13328	11553056	1338	77166	.910
44	216	133.5	5020	16366769	13369	11853056	1342	90500	.931
45	271	205	4704	20710692	13417	9819720	1266	20405	.736
46	180	139	5097	16566770	13663	12219723	1342	113910	.951
47	205	189	5797	19704773	14976	14005726	1462	280067	.963
48	178	113.5	4740	15736768	13086	10519721	1301	33340	.844
49	189	170.5	5538	18466772	14333	13853059	1419	218781	1
50	180	142	5139	16666770	13663	12419723	1366	124010	.964
51	204	127.5	4936	16213436	13328	11453056	1302	73833	.904
52	190	179.5	5664	19066772	14666	14086392	1442	248310	1
53	210	130.5	4978	16303436	13328	11636389	1342	80500	.916
54	184	117.5	4796	15875102	13092	10786388	1301	50346	.862

Table. 5

Actual and predicted efficiency scores related to the data shown in table 3

Week	DEA-score	MLP	MLP-GridSearch	MLP-PSO	LSTM	LSTM-GridSearch	LSTM-PSO
1	.664	1	1	.800	.698	.723	.686
2	1	.891	.951	.962	.985	.994	.999
3	.880	.845	.799	.836	.869	.901	.924
4	.846	.783	.784	.797	.841	.873	.877
5	.671	1	.862	.754	.712	.711	.667
6	.711	.924	.851	.833	.745	.743	.713
7	.705	.765	.772	.721	.728	.713	.710
8	.736	.625	.783	.755	.751	.764	.735
9	.900	.815	.811	.888	.885	.910	.899
10	.741	.658	.783	.774	.766	.769	.738
11	.726	.817	.742	.734	.752	.759	.730
12	.709	.714	.754	.746	.734	.748	.725

Table. 5 (continued)

Actual and predicted efficiency scores related to the data shown in table 3

Week	DEA-score	MLP	MLP-GridSearch	MLP-PSO	LSTM	LSTM-GridSearch	LSTM-PSO
13	.722	.763	.738	.736	.747	.744	.731
14	.823	.792	.774	.888	.826	.879	.876
15	.671	1	.942	.754	.712	.711	.668
16	1	.934	.982	.984	.992	.999	1
17	.935	.798	.854	.902	.921	.946	.944
18	.713	.878	.877	.783	.736	.744	.719
19	.738	.810	.741	.801	.763	.754	.735
20	.991	.812	.854	.972	.972	.986	.990
21	.753	.871	.746	.768	.778	.783	.749
22	.726	.808	.735	.718	.751	.749	.733
23	.719	.737	.773	.767	.774	.751	.721
24	.743	.598	.779	.775	.768	.764	.742
25	.752	.785	.705	.801	.777	.776	.741
26	.668	1	.918	.879	.692	.689	.683
27	.688	.997	.887	.816	.744	.727	.701
28	.717	.765	.824	.809	.740	.737	.711
29	.746	.655	.722	.784	.771	.770	.742
30	.748	.854	.768	.799	.773	.771	.736
31	.732	.845	.752	.744	.757	.756	.736
32	.755	.871	.775	.733	.780	.779	.744
33	1	.923	.976	.989	.988	.997	1
34	.882	.908	.909	.900	.906	.903	.885
35	.723	.867	.777	.762	.748	.735	.730
36	.765	.611	.712	.763	.789	.788	.764
37	.736	.683	.743	.741	.758	.751	.745
38	.898	.936	.923	.913	.923	.921	.897
39	.971	1	1	1	.995	1	1
40	.944	.987	.969	.964	.968	.959	.947
41	.853	.871	.879	.871	.878	.883	.849
42	.874	.911	.891	.893	.899	.909	.880
43	.910	.924	.931	.920	.935	.933	.913
44	.931	.968	.961	.945	.956	.950	.938
45	.736	.683	.749	.744	.762	.755	.730
46	.951	1	1	1	.976	.987	.959
47	.963	.993	1	1	.975	.973	.969
48	.844	.845	.829	.851	.869	.864	.853
49	1	1	1	1	1	1	1
50	.964	1	1	1	.989	1	1
51	.904	.940	.934	.917	.928	.922	.894
52	1	1	1	1	1	1	1
53	.916	.932	.933	.924	.941	.940	.920
54	.862	.906	.896	.891	.887	.888	.868

Table 6

Comparison of metrics related to the testing data

Metric	MLP	MLP-Grid Search	MLP-PSO	LSTM	LSTM-Grid Search	LSTM-PSO
MSE	0.0292	0.0078	0.0053	0.0028	0.0011	0.00078
MAE	0.1183	0.0613	0.0507	0.0392	0.0301	0.0225
Accuracy (<10%)	0.4815	0.799	0.8412	0.9074	0.9382	0.9611
R^2	–1.51	–0.22	0.48	0.781	0.881	0.921

4.3. Interpretation of results

1) The DEA model was used to calculate the efficiency scores of oil refineries within a specific time period.

2) Subsequently, the MLP and LSTM algorithms were employed to address the computational issues of DEA models. The scores obtained in the previous step served as training and testing datasets for these algorithms.

3) The PSO and Grid Search methods were then used to optimize the hyperparameters of the MLP and LSTM algorithms.

4) The accuracy and errors of the MLP and LSTM methods, as well as their improved versions were utilized as metrics to determine the best-performing algorithm. The results shown in table 6 indicate that:

I. Hyperparameter tuning significantly improved prediction accuracy by applying the PSO model compare to the Grid Search method.

II. LSTM-based algorithms had lower prediction errors compared to MLP-based algorithms.

III. The LSTM-PSO model outperformed others in accuracy and error reduction. Such outstanding performance is primarily attributed to two factors: (i) the inherent capability of LSTM networks to effectively capture temporal dependencies and complex sequence-based patterns in data, and (ii) the superior global search ability of the PSO algorithm in fine-tuning high-dimensional and nonlinear hyper-parameter spaces.

Based on the discussion above, MLP-based models performed worse compared to LSTM-based models. The MLP configuration exhibited the weakest predictive performance among all models, which can be attributed to the absence of hyper-parameter tuning, its relatively shallow architecture, and its limited capacity for capturing nonlinear data relationships. Notably, even the un-optimized LSTM model outperformed both the optimized and unoptimized MLP models, further emphasizing the architectural advantages of LSTM networks in modeling complex dynamic interactions within time-series data. Additionally, the LSTM-PSO approach has the best performance and can accurately predict the efficiency score of the oil refinery for the next week. In contrast, using the network DEA model involves recalculating the efficiency scores of all existing oil refineries, each of which consumes a large amount of computations.

4.4. Ranked-based performance evaluation and uncertainty analysis

The results indicate that the LSTM-PSO model achieves a Spearman's rank correlation coefficient of ρ=0.9874, reflecting an almost perfect monotonic relationship between the predicted efficiency scores and those obtained from DEA. This demonstrates that increases or decreases in the predicted values closely mirror the directional trends observed in the DEA outcomes. In addition, the Kendall's τ coefficient of 0.9222 further confirms a very high degree of pairwise agreement between the model's predictions and the actual rankings, implying that the majority of Decision-Making Unit (DMU) pairs maintain consistent ordering under both the predictive model and the DEA methodology.

To assess the statistical reliability and uncertainty of these rank correlation measures, a non-parametric bootstrap procedure was applied. Specifically, 5000 bootstrap resamples were generated from the test dataset, and Spearman's ρ and Kendall's τ were recalculated for each iteration. The resulting 95% confidence interval for Spearman's ρ was estimated as [0.9688, 0.9931], while Kendall's τ exhibited a corresponding 95% confidence interval of [0.8817, 0.9558]. The narrow width and high lower bounds of these intervals indicate that the observed ranking concordance is statistically robust, showing minimal sensitivity to sampling variability or individual DMU observations. This procedure also highlights the non-parametric strength of the analysis, ensuring its reliability without assuming any specific data distribution.

These rank-based findings are fully consistent with the point-wise prediction accuracy metrics reported in Table 6. The low MAE of 0.0225 and MSE of 0.00078 indicate that the predicted efficiency scores are numerically very close to the DEA-derived values. Moreover, an Accuracy(<10%) of 96.11% shows that over 96% of the predicted scores deviate from the corresponding DEA scores by less than 10%, reinforcing the reliability of the predictions. Importantly, the rank correlation results confirm that this high numerical accuracy translates into robust preservation of the relative efficiency positions among DMUs, which is essential for DEA-based performance assessment.

Taken together, the combined evidence from error-based metrics, rank correlation coefficients, and bootstrap confidence intervals demonstrates that the proposed LSTM-PSO model serves as a statistically dependable and operationally robust surrogate for DEA in ranking DMUs. This surrogate model is particularly advantageous in large-scale or dynamic contexts, where repeated DEA re-optimization can be computationally intensive, yet maintaining the integrity of efficiency rankings is crucial for informed decision-making and benchmarking.

5. Conclusion and directions for future research

The ML and DL methods can be employed to address the challenges associated with large-scale mathematical optimization models. In this paper, a DEA model was used to measure the efficiency scores of a limited number of DMUs. A part of the results was used to train MLP and LSTM algorithms, which are powerful ML and DL algorithms, respectively. These trained algorithms were then employed to estimate the scores of DMUs in the testing dataset. To improve the prediction accuracy, the metaheuristic PSO and Grid Search algorithms were utilized to optimize the hyperparameters of the MLP and LSTM models. The MLP and LSTM algorithms, along with their improved versions, were used to forecast the efficiency scores of oil refineries in Iran. The findings indicated that implementing the PSO method improved the accuracy of predictions more than the Grid Search method. Furthermore, the LSTM-based approaches, as DL methods, outperformed the MLP-based methods, which are ML models. Moreover, the MLP and LSTM-PSO methods demonstrated the highest and lowest prediction errors, respectively.

Limitations of this study include a restricted number of input and output items available for evaluating refineries. An oil refinery has a complex multi-stage structure with various inputs and outputs. Future research could involve applying network DEA models that incorporate connections among the sub-divisions and considering more input and output items. Additionally, for a comprehensive and reliable analysis, evaluating the gas and oil exploitation sector should be conducted over a long time period, involving the application of methods such as dynamic approaches and the Malmquist Index in evaluation models.

References

1. Charnes, A., Cooper, W. W., Rhodes, E. (1978). Measuring the efficiency of decision-making units. *European Journal of Operational Research*, 2, 429–444.

2. Bahrami, A., Rakhshaninejad, M., Ghousi, R., Atashi, A. (2025). Enhancing machine learning performance in cardiac surgery ICU: Hyperparameter optimization with metaheuristic algorithm. *PLoS One*, 20(2), e0311250.

3. Li, H., Dong, K., Sun, R., et al. (2017). Sustainability assessment of refining enterprises using a DEA-based model. *Sustainability*, 9(4), 620.

4. Tavana, M., Khalili-Damghani, K., Santos Arteaga, F. J., Hosseini, A. (2019). A fuzzy multi-objective multi-period network DEA model for efficiency measurement in oil refineries. *Computers & Industrial Engineering*, 135, 143-155.

5. Mohammed Atris, A. (2020). Assessment of oil refinery performance: Application of data envelopment analysis-discriminant analysis. *Resources Policy*, 65, 101543.

6. Sánchez Robles, B., Herrador-Alcaide, T., Montserrat, H-S. (2022). Efficiency of European oil companies: an empirical analysis. *Energy Efficiency*, 15, 63.

7. Oliveira, M. S. D., Lizot, M., Siqueira, H., et al. (2023). Efficiency analysis of oil refineries using DEA window analysis, cluster analysis, and malmquist productivity index. *Sustainability*, 15(18), 13611.

8. Ngeti, K. B. D. J. R., Mafukidze, B. S. (2024). DEA analysis of national oil companies. *Open Access Library Journal*, 11(12).

9. Hasanvand, M., Taleghani, M., Fathi-Vajargah, B. (2025). Evaluating performance of national oil and gas production facilities: A fuzzy network approach to manage undesirable outputs. *International Journal of Applied Operational Research*, 13(1), 17-38.

10. Zhu, N., Zhu, Ch., Emrouznejad, A. (2021). A combined machine learning algorithms and DEA method for measuring and predicting the efficiency of Chinese manufacturing listed companies. *Journal of Management Science and Engineering*, 6(4), 435-448.

11. Kaur, G., Rajni, Sivia, J. S. (2024). Integrating data envelopment analysis and machine learning approaches for energy optimization, decreased carbon footprints, and wheat yield prediction across North-Western India. *Journal of Soil Science and Plant Nutrition*, 24, 1424–1447.

12. Koushki, F., Naghdehforoushha, M. (2025). AI-Powered selection and classification of resilient suppliers: a hybrid approach using fuzzy DEA and ML techniques and its application in the textile industry. *Advances in Industrial Engineering*, 59(2), 307-319.

13. Koushki, F., Naghdehforoushha, M. (2025). Integrating a novel fuzzy network DEA method with machine learning algorithms to predict the performance of production systems. *Asia-Pacific Journal of Operational Research*. https://doi.org/10.1142/S0217595925500320.

14. Koushki, F., Naghdehforoushha, M. (2025). Predicting the sustainability of supply chains by integrating a novel network DEA model with ML techniques. *Journal of Industrial and Management Optimization*, 21(8), 5326-5347.

15. Al Bataineh, A., Kaur D., Jalali, S. M. J. (2022). Multi-layer perceptron training optimization using nature inspired computing. *IEEE Access*, 10, 36963-36977.

16. Bahtiyar, H., Soydaner, D., Yüksel, E. (2022). Application of multilayer perceptron with data augmentation in nuclear physics. *Applied Soft Computing*, 128, 109470.

17. Ghojogh, B., Ghodsi, A. (2023). Recurrent neural networks and long short-term memory networks: Tutorial and survey. https://arxiv.org/abs/2304.11461

18. Liashchynskyi, P., Liashchynskyi, P. (2019). Grid search, random search, genetic algorithm: a big comparison for NAS. https://arxiv.org/abs/1912.06059

19. Houssein, E. H., Gad, A. G., Hussain, K., Suganthan, P. N. (2021). Major advances in particle swarm optimization: Theory, analysis, and application. *Swarm and Evolutionary Computation*, 63, 100868.

20. Gad, A. G. (2022). Particle swarm optimization algorithm and its applications: A systematic review. *Archives of Computational Methods in Engineering*, 29, 2531–2561.

21. Li, L., Jamieson, K., Rostamizadeh, A., et al. (2021). A system for massively parallel hyperparameter tuning. *Communications of the ACM*, 64(12), 72–81.

22. Agrawal, S., Chakraborty, A. (2021). A comparative study of particle swarm optimization and its variants for hyperparameter tuning in deep learning. *Procedia Computer Science*, 192, 1722–1731.

23. Singh, A., Singh, S. (2021). Recent advances in metaheuristic algorithms for hyperparameter tuning: A review. *Swarm and Evolutionary Computation*, 62, 100840.

24. Rashedi, E., Nezamabadi-pour, H., Saryazdi, S. (2020). A comprehensive review on metaheuristic algorithms for hyperparameter optimization. *Applied Soft Computing*, 91, 106260.

25. Gupta, A., Khandelwal, S. (2022). Comparative study of metaheuristic optimization techniques in deep learning: Challenges and future directions. *Journal of Computational Science*, 59, 101537.

26. Shiomi, H., Takahashi, M., Tanaka, T. (2025). Hybrid PSO-deep learning model for hyperparameter optimization in time series forecasting. *Expert Systems with Applications*, 225, 120218.