Supplement material for the manuscript “Advancing RVFL networks: Robust classification with the HawkEye loss function"

Mushir Akhtar[1], Ritik Mishra[1], M. Sajid[1], A. Quadir[1], M. Tanveer[1]*, and Mohd. Arshad[1]

Indian Institute of Technology Indore, Simrol, Indore, India
{phd2101241004,phd2301241003,phd2101241003,mscphd2207141002,
mtanveer,arshad}@iiti.ac.in

S.I Related Work

In this section, we provide a brief overview of the structure and mathematical formulation of the standard RVFL network.

S.I.A Random vector functional link (RVFL) network

RVFL network is a specialized variant of the SLFN, comprising three distinct layers: input, hidden, and output. Each layer contains interconnected neurons, linked by specific weights. The weights between the input and hidden layers are randomly generated within a predetermined range and remain fixed throughout the training process. In contrast, the output weights are computed using the least squares method or the Moore-Penrose inverse, ensuring optimal performance. A key feature of the RVFL network is the inclusion of direct connections between the input and output layers. This integration of original features enhances the network's generalization capabilities. The standard architecture of the RVFL framework is illustrated in Figure S.1. This structure allows the RVFL network to leverage both the fixed random weights for initial transformations and the computed output weights for final predictions, resulting in a robust and efficient learning model.

Let $X = [x_1, x_2, x_3, \ldots, x_n]^\intercal$, $x_i \in \mathbb{R}^m$, be a m dimensional training dataset, and $Y = [y_1, y_2, y_3, \ldots, y_n]^\intercal$, $y_i \in \mathbb{R}^c$ be the target matrix. Here n represents the number of samples, m denotes the number of features, and c is number of classes. The mathematical formulation of the standard RVFL can be written as:

$$\underset{\beta}{\operatorname{argmin}} \frac{1}{2}\|\beta\|_2^2 + \frac{1}{2}C\|T\beta - Y\|_2^2. \tag{1}$$

Here

$$T = \begin{bmatrix} X \; H \end{bmatrix}_{n\times(m+h_l)},$$

[*] Corresponding Author

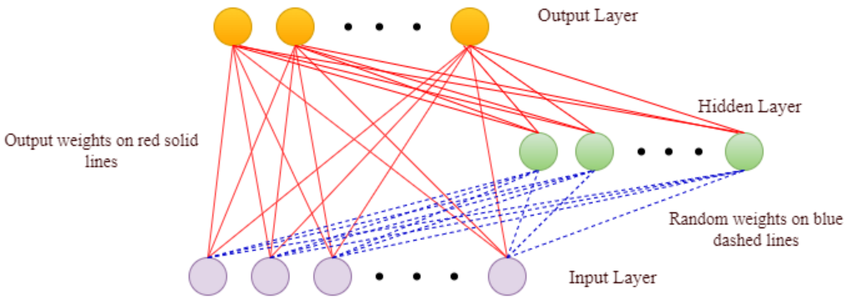
Output Layer
Hidden Layer
Output weights on red solid lines
Random weights on blue dashed lines
Input Layer

Fig. S.1: The architecture of the standard RVFL network.

where,

$$X = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1m} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{nm} \end{bmatrix}_{n \times m},$$

$$H = \begin{bmatrix} \Phi\left(w_1 \cdot x_1 + \sigma_1\right) & \Phi\left(w_2 \cdot x_1 + \sigma_2\right) & \ldots & \Phi\left(w_{h_l} \cdot x_1 + \sigma_{h_l}\right) \\ \vdots & \ddots & \vdots & \vdots \\ \Phi\left(w_1 \cdot x_n + \sigma_1\right) & \Phi\left(w_2 \cdot x_n + \sigma_2\right) & \ldots & \Phi\left(w_{h_l} \cdot x_n + \sigma_{h_l}\right) \end{bmatrix}_{n \times h_l},$$

$$\beta = \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_{(m+h_l)} \end{bmatrix}_{(m+h_l) \times c}, \text{and} \quad Y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}_{n \times c}.$$

Here, X and H are input matrix and hidden layer matrix respectively, also h_l represents the number on nodes in hidden layer. The matrix T is obtained by column-wise concatenation of input and hidden layer matrix. The output weight vector $\beta_p = [\beta_{p1}, \beta_{p2}, \cdots, \beta_{pc}]$ connects the p^{th} input or hidden node to the output nodes, where $1 \leq p \leq m + h_l$. Similarly, the weight vector $w_j = [w_{j1}, w_{j2}, ..., w_{jm}]$, connects the j^{th} hidden node to the input nodes, where j ranges from 1 to h_l. The input samples and target output are represented by $x_i = [x_{i1}, x_{i2}, ..., x_{im}]$ and $y_i = [y_{i1}, y_{i2}, ..., y_{ic}]$, respectively and $1 \leq i \leq n$. Additionally, Φ is a non-constant activation function, and σ_i is the bias term of the i^{th} hidden node.

The optimal solution of the RVFL is obtained as:

$$\beta = \begin{cases} \left(T^\intercal T + \frac{1}{C} I\right)^{-1} T^\intercal Y, & (m + h_l) \leq n, \\ T^\intercal \left(T T^\intercal + \frac{1}{C} I\right)^{-1} Y, & n < (m + h_l). \end{cases} \tag{2}$$

The equation (2) involves a regularization parameter C, which requires tuning, and an identity matrix I, of suitable dimension. Notably, the matrices $T^{\intercal}T$ and $TT^{\intercal}$ are symmetric and positive semi-definite, and since C is a positive value, the resulting matrices in the equation are guaranteed to be positive definite, therefore, $\left(T^{\intercal}T + \frac{1}{C}I\right)$ and $\left(TT^{\intercal} + \frac{1}{C}I\right)$ are non-singular matrix.

S.II Complexity analysis

In this section, we provide an analysis of the computational complexity of the NAG algorithm presented in Algorithm 1. Let k and m denote the number of samples in the mini-batch and features in the training dataset; h_l refers to the number of hidden nodes; and I represents the number of iterations. At each iteration, the gradient calculation involves an inner loop over k mini-batch samples, where each iteration computes an exponential and vector multiplication with a complexity of $\mathcal{O}(m+h_l)$. The computational complexity associated with updating the variable v can be expressed as $\mathcal{O}\left((I+m+h_l)k^2\right)$. Similarly, the complexity for updating the parameter $\hat{\beta}$ is $\mathcal{O}(Ik)$, lastly the complexity of updating the parameter α is $O(I)$ [7]. Therefore, the total computational complexity over I iterations can be summarized as $\mathcal{O}\left((I+m+h_l)k^2\right)$.

S.III Experimental Evaluation

A detailed overview of the UCI and KEEL datasets utilized for evaluating the proposed H-RVFL model against baseline models is presented in Table S.I. The detailed results of the proposed H-RVFL model against the baseline models on each of the 33 (Group A) datasets are presented in Table S.II.

S.III.A Experimental setup and hyperparameters selection

All the experiments are implemented using MATLAB R2023a on window 10 running on a PC with configuration Intel(R) Core(TM) i7-6700 CPU @ 3.40GHz, 3408 Mhz, 4 Core(s), 8 Logical Processor(s) with 16 GB of RAM. We utilized a 4-fold cross-validation method combined with a grid search to fine-tune the model's hyperparameters. For each set of hyperparameters, we independently calculated the testing accuracy for each fold. The mean of these four-fold accuracies was then taken as the average testing accuracy for each set of hyperparameters. Now, we present the parameter selection ranges for the proposed H-RVFL and the baseline models. For all the models, the regularization parameter (C) is selected from the set $\{10^i \,|\, i = -6, -4, \ldots, 4, 6\}$. For SVM and IF-RVFL, the kernel parameter (σ) is selected from the set $\{10^i \,|\, i = -6, -4, \ldots, 4, 6\}$. For each model except SVM, the number of hidden layer nodes is selected from the range $[3:20:203]$. For NF-RVFL, the number of fuzzy nodes in the fuzzy layer is selected from the range $[5:10:45]$, and the standard deviation is taken as 1. For the proposed H-RVFL, loss hyperparameters a, λ, and ε are selected from $[0.1:$

Table S.I Description of UCI and KEEL datasets used to evaluate the models.

Dataset	Number of samples	Number of features
acute_inflammation	120	6
blood	748	4
breast_cancer	286	9
breast_cancer_wisc	699	9
chess_krvkp	3196	36
congressional_voting	435	16
conn_bench_sonar_mines_rocks	208	60
cylinder_bands	512	35
fertility	100	9
haberman_survival	306	3
heart_hungarian	294	12
hepatitis	155	19
hill_valley	1212	100
horse_colic	368	25
ilpd_indian_liver	583	9
ionosphere	351	33
molec_biol_promoter	106	57
monks_1	556	6
musk_1	476	166
pittsburg_bridges_T_OR_D	102	7
planning	182	12
spambase	4601	57
spect	265	22
statlog_australian_credit	690	14
tic_tac_toe	958	9
titanic	2201	3
vertebral_column_2clases	310	6
crossplane130	130	2
crossplane150	150	2
haber	306	3
shuttle-6_vs_2-3	230	9
vowel	988	10
yeast2vs8	483	8
adult	48842	14
connect_4	67557	42
magic	19020	10
miniboone	130064	50
musk_2	6598	166
ringnorm	7400	20
twonorm	7400	20

$0.2 : 5]$, $[0.1 : 0.2 : 2]$, and $\{0.1, 0.01, 0.001\}$, respectively. The parameters for the NAG algorithm are set as follows: (i) initial model parameter $\hat{\beta}_0 = 0.01$, (ii) initial velocity $v_0 = 0.01$, (iii) initial learning rate $\alpha = 0.01$, (iv) learning decay

Table S.II Performance comparison of the proposed H-RVFL model against the baseline models on 33 (Group A) UCI and KEEL datasets.

Dataset	SVM [1]	RVFLwoDL [3]	RVFL [5]	NF-RVFL [6]	IF-RVFL [4]	H-RVFL†
	Acc.±Std.	Acc.±Std.	Acc.±Std.	Acc.±Std.	Acc.±Std.	Acc.±Std.
acute_inflammation	100±0	100±0	100±0	100±0	100±0	100±0
blood	76.34±10.6	76.34±12.06	76.47±11.48	76.74±9.21	77.01±12.16	77.67±12.46
breast_cancer	70.14±47.65	70.14±47.65	70.14±47.65	69.26±10.19	73.01±26.87	95.49±9.03
breast_cancer_wisc	90.28±2.67	86.85±6.37	86.7±6.27	85.99±6.49	87.13±5.29	69.97±8.98
chess_krvkp	95.65±6.2	74.56±7.85	78.6±6.38	84.11±3.04	77.03±7.2	95.65±6.2
congressional_voting	61.84±3.19	61.61±1	62.31±2.45	62.53±2.98	61.84±2.17	61.38±1.75
conn_bench_sonar_mines_rocks	28.37±48.18	49.52±7.59	57.21±14.42	60.1±15.65	68.27±8.95	96.63±6.73
cylinder_bands	60.94±15.23	64.84±4.94	65.82±6.64	68.95±4.87	65.63±4.73	66.8±3.91
fertility	88±7.3	89±8.25	89±8.25	88±7.3	89±6	90±5.16
haberman_survival	73.51±4.03	73.51±4.03	73.51±4.03	75.46±5.23	73.51±4.03	74.5±3.89
heart_hungarian	63.85±47.48	63.85±47.48	57.7±35.21	72.46±11.66	78.55±10.65	91.89±16.22
hepatitis	79.25±14.08	84.51±9.13	84.51±7.26	80.63±8.79	85.83±4.84	82.46±13.26
hill_valley	67.9±3.53	79.37±4.74	82.1±5.88	87.21±6.07	84.16±6.99	54.87±3.97
horse_colic	66.3±4.07	85.87±2.66	84.78±3.66	83.97±5.64	85.6±3.12	68.48±2.81
ilpd_indian_liver	71.36±5.58	71.36±5.58	71.36±5.58	70.85±5.49	71.7±6.04	71.53±5.8
ionosphere	64.77±23.66	88.06±7.08	89.76±6.87	89.2±6.78	89.19±4.38	73.01±18.61
molec_biol_promoter	50±57.74	63.18±12.23	62.22±11.97	80.13±8.59	74.61±13.14	100±0
monks_1	86.51±8.15	82.91±2.96	82.19±5.71	74.82±7.29	82.19±9.33	55.4±7.5
musk_1	56.51±51.33	63.24±10.75	65.55±10.29	72.9±8.01	68.49±47.28	93.49±13.03
pittsburg_bridges_T_OR_D	86.15±14.87	86.15±14.87	86.15±14.87	83.27±10.62	89.19±6.86	90.15±6.99
planning	72.48±7.13	71.38±7.69	71.38±7.69	69.19±6.73	71.39±7.65	73.57±6.75
spambase	89.39±20.99	86.7±5.11	87.35±4.6	88.2±3.53	90.31±1.59	92.76±14.48
spect	60.36±7.06	66.07±5.43	66.43±2.6	68.3±4.95	69.45±3.73	64.54±3.5
statlog_australian_credit	67.83±1.44	67.97±1.28	68.26±1.45	66.67±3.19	68.26±1.39	68.55±1.71
tic_tac_toe	71.48±32.39	59.36±41.55	51.74±34.95	62±5.35	70.75±27.41	90.31±19.38
titanic	67.7±21.72	77.1±26.04	77.1±26.04	79.23±22.48	79.23±22.48	82.6±16.19
vertebral_column_2clases	70.73±27.76	65.18±25.35	65.83±25.9	77.12±7.28	82.28±9.71	93.27±13.46
crossplane130	56.71±9.89	42.23±6.16	42.23±6.16	42.23±6.16	50.05±10.88	57.77±6.16
crossplane150	61.93±8.98	61.93±8.98	61.93±8.98	61.93±8.98	61.93±8.98	62.61±7.65
haber	73.51±4.03	73.84±4.65	73.84±4.65	73.84±3.66	73.84±3.66	74.17±3.54
shuttle-6_vs_2-3	100±0	100±0	100±0	100±0	100±0	100±0
vowel	94.33±2.27	95.95±1.65	96.05±1.42	96.46±1.46	96.66±1.63	97.57±0.99
yeast2vs8	97.52±1.17	97.93±0.82	97.93±0.82	98.14±1.04	97.93±1.43	98.14±1.04
Avg. Acc.±Avg.Std.	73.38±15.77	75.17±10.66	75.34±10.31	77.27±6.63	78.61±8.81	**80.76±7.31**
Avg. Rank	4.12	3.82	3.45	3.21	2.39	**2.18**

† represents the proposed model.

Here, Avg., Acc., and Std. are acronyms used to represent average, accuracy, and standard deviation, respectively.

The boldface and underline indicate the best and second-best models, respectively.

factor $\eta = 0.1$, (v) momentum parameter $r = 0.6$, (vi) two distinct minibatch sizes based on the dataset size: $k = 2^3$ for datasets with fewer than 100 samples and $k = 2^6$ for datasets with 100 or more samples, (vii) maximum iteration number $I = 500$, and (viii) convergence tolerance $\theta = 10^{-6}$. The activation function is a crucial element in neural network architecture, introducing non-linearity and allowing networks to capture complex patterns. With this in mind, we also fine-tune the activation function. We evaluate six activation functions: Sigmoid, Sine, Radbas, Tribas, Tansig, and ReLU. For all the experiments, we normalized the data in the range of $[0, 1]$.

S.III.B Statistical analysis

To further affirm the effectiveness of the proposed H-RVFL model, a comprehensive statistical evaluation was conducted on 33 (Group A) UCI and KEEL datasets. This evaluation included three distinct analyses: a ranking test, the Nemenyi post hoc test, and a win-tie-loss sign test [2].

Ranking test: To gain a comprehensive evaluation of the proposed H-RVFL performance, we implemented a ranking approach. Relying solely on average accuracy can be misleading, as exceptional results on a few datasets might conceal subpar performance on others. To address this limitation, we ranked each model separately for each dataset, enabling a thorough examination of their strengths and weaknesses. In the ranking test, the model that performs the worst on a particular dataset receives a higher ranking, while the top-performing model is assigned a lower ranking. Suppose we have $\mathscr{M}$ models being evaluated across $\mathscr{D}$ distinct datasets. The performance ranking of the m^{th} model on the d^{th} dataset is denoted by $\rho(m, d)$. We calculate the average rank of the m^{th} model as follows:

$$\rho(m) = \frac{\sum_{d=1}^{\mathscr{D}} \rho(m, d)}{\mathscr{D}}.$$

The proposed H-RVFL model achieves the lowest average rank of 2.18 among all the evaluated models. In comparison, the baseline models SVM, RVFLwoDL, RVFL, NF-RVFL, and IF-RVFL have average ranks of 4.12, 3.82, 3.45, 3.21, and 2.39, respectively (see Table S.III). This ranking test clearly demonstrates the superior performance of the H-RVFL model over the baseline models.

Nemenyi post hoc test: To investigate the performance difference between the proposed H-RVFL model and the baseline models, a pairwise comparison is conducted using the Nemenyi post-hoc test. This test determines whether the disparity in performance between two models is statistically significant by examining the difference in their average ranks. If the difference in average ranks exceeds a certain threshold value, known as the critical difference (CD), the model with the higher average rank is considered statistically superior. The CD is computed as follows:

$$CD = q_\alpha \sqrt{\frac{\mathscr{M}(\mathscr{M} + 1)}{6\mathscr{D}}}, \tag{3}$$

where q_α is derived from the studentized range statistic divided by $\sqrt{2}$, also known as critical value for the two-tailed Nemenyi test. From distribution table of two-tailed Nemenyi test, the value of q_α at 0.1 significance level is 2.589. Thus the value of CD is calculated as 1.19. As presented in Table S.IV, the average rank differences between the proposed H-RVFL model and the baseline models SVM, RVFLwoDL, RVFL, NF-RVFL, and IF-RVFL are 1.94, 1.64, 1.27, 1.03, and 0.21, respectively. These differences highlight the notable superiority of the proposed H-RVFL model over SVM, RVFLwoDL, and RVFL, as these differences exceed the critical difference (CD) threshold. Although the rank difference of the proposed H-RVFL from NF-RVFL and IF-RVFL is lower than the CD, the overall evaluation results still demonstrate the superior performance of the H-RVFL model. This superiority is evident not only in the average ranking but also in the model's consistent performance across various datasets.

Win-tie-loss sign test: It is a widely recognized pairwise statistical test that evaluates whether there is a significant difference between the performance outcomes of two models. Under the null hypothesis, it is assumed that the two models perform equivalently, with each model expected to win on $\mathscr{D}/2$ out of $\mathscr{D}$ datasets. The models are deemed significantly different at a 5% level of significance if one model achieves at least $\mathscr{D}/2 + 1.96\sqrt{\mathscr{D}/2}$ wins. In cases of ties, the score is evenly distributed between the two models. In our case, we have $\mathscr{D} = 33, \mathscr{D}/2 + 1.96\sqrt{\mathscr{D}/2} = 22.19$. Thus two models will be considered significantly different if one of them wins at least on 22 datasets. Table S.V shows the pairwise win-tie-loss test results of the proposed H-RVFL against the baseline models. Clearly, the proposed H-RVFL wins on 26 (w.r.t SVM), 23 (w.r.t. RVFLwoDL), 23 (w.r.t. RVFL), 21 (w.r.t. NF-RVFL), and 22 (w.r.t. IF-RVFL) among 33 datasets. After evenly distributing the ties between the models, it is evident that the proposed H-RVFL model demonstrates superior performance compared to all the baseline models.

Table S.III Rank comparison of the proposed H-RVFL model against the baseline models on each of the 33 (Group A) UCI and KEEL datasets.

Dataset	SVM [1]	RVFLwoDL [3]	RVFL [5]	NF-RVFL [6]	IF-RVFL [4]	H-RVFL[†]
acute_inflammation	1	1	1	1	1	1
blood	6	5	4	3	2	1
breast_cancer	5	3	3	6	2	1
breast_cancer_wisc	1	3	4	5	2	6
chess_krvkp	1	6	4	3	5	2
congressional_voting	4	5	2	1	3	6
conn_bench_sonar_mines_rocks	6	5	4	3	2	1
cylinder_bands	6	5	3	1	4	2
fertility	5	2	2	5	2	1
haberman_survival	3	4	4	1	4	2
heart_hungarian	5	4	6	3	2	1
hepatitis	6	2	2	5	1	4
hill_valley	5	4	3	1	2	6
horse_colic	6	1	3	4	2	5
ilpd_indian_liver	3	4	4	6	1	2
ionosphere	6	4	1	2	3	5
molec_biol_promoter	6	4	5	2	3	1
monks_1	1	2	3	5	3	6
musk_1	6	5	4	2	3	1
pittsburg_bridges_T_OR_D	3	4	4	6	2	1
planning	2	4	4	6	3	1
spambase	3	6	5	4	2	1
spect	6	4	3	2	1	5
statlog_australian_credit	5	4	2	6	3	1
tic_tac_toe	2	5	6	4	3	1
titanic	6	4	4	2	2	1
vertebral_column_2clases	4	6	5	3	2	1
crossplane130	2	4	4	4	3	1
crossplane150	2	3	3	3	3	1
haber	6	4	4	2	2	1
shuttle-6_vs_2-3	1	1	1	1	1	1
vowel	6	5	4	3	2	1
yeast2vs8	6	3	3	1	3	1
Avg. Rank	4.12	3.82	3.45	3.21	2.39	**2.18**

[†] represents the proposed model.

Table S.IV Differences in the rankings of the proposed H-RVFL model against baseline models on 33 (Group A) UCI and KEEL datasets.

Models	Average Rank	Rank difference	Significant difference (As per Nemenyi post hoc test)
SVM [1]	4.12	1.94	Yes
RVFLwoDL [3]	3.82	1.64	Yes
RVFL [5]	3.45	1.27	Yes
NF-RVFL [6]	3.21	1.03	No
IF-RVFL [4]	2.39	0.21	No
H-RVFL[†]	2.18	-	N/A

[†] represents the proposed model.

Table S.V Pairwise win-tie-loss test results on 33 (Group A) UCI and KEEL datasets. Here, in [a, b, c], a, b, and c correspond to the count of wins, ties, and losses of the row model over column method, respectively.

	SVM [1]	RVFLwoDL [3]	RVFL [5]	NF-RVFL [6]	IF-RVFL [4]
RVFLwoDL [3]	$[18, 2, 13]$				
RVFL [5]	$[18, 2, 13]$	$[13, 14, 6]$			
NF-RVFL [6]	$[18, 3, 12]$	$[19, 4, 10]$	$[18, 4, 11]$		
IF-RVFL [4]	$[23, 2, 8]$	$[25, 6, 2]$	$[21, 7, 5]$	$[19, 5, 9]$	
H-RVFL [†]	$[26, 2, 5]$	$[23, 2, 8]$	$[23, 2, 8]$	$[21, 3, 9]$	$[22, 2, 9]$

[†] represents the proposed model.

Bibliography

[1] C. Cortes and V. Vapnik. Support-vector networks. *Machine Learning*, 20 (3):273–297, 1995.

[2] J. Demšar. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, 7(1):1–30, 2006. URL `http://jmlr.org/papers/v7/demsar06a.html`.

[3] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew. Extreme learning machine: theory and applications. *Neurocomputing*, 70(1-3):489–501, 2006.

[4] A. K. Malik, M. A. Ganaie, M. Tanveer, P. N. Suganthan, and for the Alzheimer's Disease Neuroimaging Initiative. Alzheimer's disease diagnosis via intuitionistic fuzzy random vector functional link network. *IEEE Transactions on Computational Social Systems*, pages 1–12, 2022. `https://doi.org/10.1109/TCSS.2022.3146974`.

[5] Y.-H. Pao, G.-H. Park, and D. J. Sobajic. Learning and generalization characteristics of the random vector functional-link net. *Neurocomputing*, 6(2): 163–180, 1994.

[6] M. Sajid, A. K. Malik, M. Tanveer, and P. N. Suganthan. Neuro-fuzzy random vector functional link neural network for classification and regression problems. *IEEE Transactions on Fuzzy Systems*, 2024. `https://doi.org/10.1109/TFUZZ.2024.3359652`.

[7] J. Tang, B. Liu, S. Fu, Y. Tian, and G. Kou. Advancing robust regression: Addressing asymmetric noise with the BLINEX loss function. *Information Fusion*, page 102463, 2024.