Optimized Neural Network-Based Improved Multiverse Optimizer Algorithm For Automated Arabic Essay Scoring

Marwa M. Gaheen, Rania M. ElEraky, Ahmed A. Ewees

Abstract: The automated essay scoring is recognized as an automatic evaluation of essays or automated essay grading. Such methods are very helpful for assessing human graders and experts when evaluating a large volume of essays. In this paper, a new method is presented to score essays automatically. It uses particle swarm optimization to generate the initial population for the multiverse optimizer algorithm to train the classic Neural Network. It is called pMVO-NN. The proposed method is evaluated using 200 student's essays. These essays are scored by two human experts then they are passed to a pre-processing phase to be prepared and converted to a digit's matrix. The results are evaluated using a set of measures and it is compared with well-known optimization algorithms. The pMVO-NN outperformed all compared algorithms and obtained a correlation equals to 0.987 with the scores of the human experts.

Index Terms: Automated essay scoring, Natural language processing, Neural network, Multiverse optimizer algorithms, Particle swarm optimization.

1. INTRODUCTION

Automated essay scoring (AES) is a computer platform that evaluates essays which students write. Evaluation means that the computer platform can engage in the mission of scoring an article or setting a degree to it automatically. There is a growing need for such methods to assist students in evaluating their essays. The AES framework is also recognized as an automatic evaluation of essays and automated essay grading. There are different approaches employed in implementing AES platforms in general. Next, we briefly look at the approaches utilized in AES execution. Broadly these are Information Retrieval (IR), Machine Learning (ML), and Natural language processing (NLP). The IR is drawing a motivating level of research awareness in the domain of Computer Science. It is mechanisms of recovering relevant information rely on the user's query search [1]. The purpose of IR platform is to rank documents optimally given a query thus relevant documents would be ranked over non-relevant ones [2]. Through the years, numerous types of retrieval methods have been presented and developed, fundamentally: the Boolean methods, the Statistical methods, which contain the probabilistic retrieval methods and the vector space as well as the Knowledge-based and Linguistic methods [3]. The normal Boolean method has several weaknesses; for example, users find it troublesome to construct influential Boolean requests. While writing a query, the users turning to their awareness of English. As well, the common Boolean approach does not give a relevance rating of the retrieved texts. In probabilistic models and vector space, both utilize statistical information in the format of term frequencies to determine the relevance of essays regarding the query. Together schemes generate a list of essays ranked by their rated relevance [4]. The latent semantic analysis (LSA) relies on statistical retrieval methods. LSA compute distributional semantics among documents and expressions. It supposes there is some latent construction in the type of term usage across documents and that statistical methods can be utilized to estimate this construction. The merit of this approach is that essays can be retrieved even if they have no terms in common with the query. The model includes a deeper process within the construction. LSA algorithm generates a matrix that involves the terms in the rows and the essays in columns. Since the term-essays matrix is massive, it can be reduced by using singular value decomposition (SVD) [5]. Machine learning (ML) platforms automatically learn systems from data. It is widely used in many studies [6][7][8]. The merit of ML in the AES system, generally, is the efficiency to score different kinds of documents automatically. There are various ML mechanisms used in literature such as k-nearest-neighbor (k-NN), neural network, linear regression function [9], and adaptive-network-based fuzzy inference system [10][11].
Over the few years ago, there has been a considerable growth in the volume of NLP publications in terms of evaluation of the text and scoring student answers [4][12]. As considered an NLP implementation since typically the essence technology beyond the automated dissection of the student answer enlists NLP mechanisms. The student answers can cover essays, short answers, or spoken answers and the two most public kinds of AES are the automated valuation of content knowledge and writing quality. Both the feedback and scores relied on linguistic characteristics of the answers containing but not tied to (a) Lower-level mistakes in the answer, (b) The discourse construction and/or organization of the answer, (c) Relevance of the reply to the query that was asked [13]. One of the promising mechanisms used in NLP implementation is the neural network that has been proved to be capable of inducing semantic features and dense syntactic automatically, providing competitive outcomes to manually designed features for various tasks [14]. Neural network (NN) has been successfully utilized in many fields [15] [16] [17] especially in NLP implementations such as sequence labelling [18], clauses modelling [14], and sentimental analysis [19]. The record of Arabic NLP is comparatively recent parallel to that of English, and therefore it is lacking in many research areas including grading Arabic essays automatically. One of the reasons is the complication of the Arabic language which blocks the

- *Marwa M. Gaheen Department of Computer, Damietta University, Damietta, Egypt. E-mail: marwa_20@du.edu.eg*
- *Rania M. ElEraky. Department of Curricula and Teaching Methods, Faculty of Specific Education, Damietta University, Damietta, Egypt. E-mail: reraky@du.edu.eg*
- *A. A. Ewees. Department of Computer, Damietta University, Damietta, Egypt. E-mail: a.ewees@hotmail.com*

development of a power evaluation platform. For example, the Arabic language is distinguished by high ambiguity, compound morpho-syntactic agreement essentials, rich morphology, etc. Yet another operator is the existing trend in Arabic NLP where the generality of the research is directed towards tackling tweets [20]. There are various studies for essays in the English language [21], and for other languages, e.g. [22]. With quite a few works, the task on Arabic AES is a challenge. The authors of [23] suggested a hybrid AES system for Arabic essays assessment that makes use of reduced dimensionality for LSA (LSAD). The longitude of an essay ranged among 100-200 terms. The authors notified an accuracy of 96.72\% on the tested data. The correlation outcome among this system's score and the assessment by humans was 0.78. Another study [5] proposed a comparison among cosine correspondence and k-NN mechanism in the LSA technique to automatically score Arabic essays. Cosine correspondence with LSA led to preferable execution than using k-NN with LSA. The correlations reached 0.88 and 0.5, respectively. The authors of [24] suggested a web-based platform for Arabic AES that was relied on the vector space model (VSM). The correspondence measured using term frequency and inverse document frequency (tf-idf) and cosine correspondence term. The authors investigated the system using 30 written papers. They computed the correspondence of a written paper with the model responses. The paper was assigned the result corresponding to the highest similarity. In [25] the authors presented a method based on frequent neural networks to handle the task of AES. They examined a variety of neural network model architectures for AES and have obtained significant enhancements over a strong open-root baseline. The top system performed better than the baseline by 5.6\% for quadratic weighted Kappa. In [4] the researchers presented an AAES system that used LSA and RST for Arabic essays in Saudi Arabia. The results showed that 90\% of the test papers were correctly scored, and a correlation with the human expert obtained 0.756. In this paper, a new algorithm is proposed for Arabic AES. It utilizes the Particle swarm optimization (PSO) and multiverse optimizer algorithm (MVO) to improve the performance of the NN by training its weights to perform this task. The MVO [26] is an optimization method. It tries to simulate the multi-verse theory in physics and the universes' interaction. The MVO is successfully applied to solve different optimization problems [27] [28], but it may get trap in a local optimum, therefore, in this paper, the PSO is used to generate the initial population of the MVO to improve its population and increase its ability to get the optimal solution. The rest of this study is organized as follows, Section 2, contains a brief description for the used methods. Section 3 describes the proposed method. The experiment details are discussed in Section 4. The last section concludes the paper.

2 METHODS AND MATERIALS

2.1 Latent Semantic Analysis (LSA)

Detailed LSA is a mechanism that uses NLP and statistics in IR to obtain the semantic meaning in documents. Therefore, the LSA can detect the similarities between the texts even if they do not include similar words [29] [30]. The entire stages of the LSA mechanism are as follow. The texts that require to be matched should be symbolized in a Word-by-Context Matrix (WCM), where rows are the terms and columns are the documents (answers). The number of the term frequency is presented in a cell. The good practice is to collect the term in WCM when it is repeated more than one time [31]. The "Stop Words" are deleted before building the WCM because they are extremely common words with no valuable meaning. Subsequently, deleting them help in reducing the dimension of the WCM and speed up the processing [31].

Using the weight of each word in WCM instead of the frequency of this word is better to represent the significance of it (i.e. it gives higher weight to words that are more significant than others) [31]. The word weighting "W_i" can be specified by (1) it is known as TF-IDF.

$$W_i = tf_i \times idf_i \tag{1}$$

where, tf_i is i^{th} word frequency. idf_i denotes inverse document frequency of i^{th} word.

$$idf_i = \log\left(\frac{D}{df_i}\right) \tag{2}$$

where, $\log$ is the logarithm. D is the number of documents. df_i is the number of documents including term i.

LSA uses singular value decomposition (SVD), which is a type of factor analysis that decreases the matrix dimensionality [30]. The main matrix is integral to three sub-matrices satisfying (3).

$$X = U \sum V^T \tag{3}$$

where, X is WCM, every element (i,j) represents the number of occurrences of word i in essays j. U describes the rows as vectors for creating a square matrix. $\sum$ describes a non-negative diagonal matrix. V^T describes the columns as vectors.

The similarity's degree of the context is determined using Cosine Similarity $\cos(\theta)$ and is computed by (4).

$$Similarity = \cos(\theta) = \frac{A \bullet B}{\|A\|\|B\|} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n}(A_i)^2} \times \sqrt{\sum_{i=1}^{n}(B_i)^2}} \tag{4}$$

where, θ is the angle between A and B vectors. A is a text described as a vector in the WCM. B is a another text described as a vector in the WCM. The $\cos(\theta)$ lies among 1 and 0, it denoted solid similarity if it equals 1.

2.2 Feedforward Neural Network

The feedforward neural network (NN) is a mathematical form that is applied in several fields such as classification, regression, and forecasting [32]. The neural network simulates a structure of the connections of the neurons in the brain. The training method in the neural network is very important in the learning process and they are used to enhance the learning in numerous applications. The framework of the NN mainly consists of three layers: input, hidden, and output layers. These layers are linked by weights. The learning method aims to select the best weights' values that linked between the layers. If the units' number in the hidden layer is slight, learning is subsequently insufficient. On the contrary if the layer's number is so big the generalization will be poor. The input layer consists of input units $X_i = (1, \ldots, n)$. After the activation function is computed the value is transferred to the hidden layer units $Z_{i,j} = (1, \ldots, H)$, where the input unit value is doubled by the weight that attaches that unit with the hidden unit and totality all the weights attached to the hidden unit as next:

$$Z_{inj} = \sum_{i=0}^{n} x_i W_{ij} - \theta_j \tag{5}$$

where W_{ij} is the weight of the i^{th} unit in input layer j^{th}, the threshold is defined as θ_j and the i^{th} input unit is x_i. The activation function is computed by applying Eqs. (9) and (10), where Z_{inj} is the totality of all the input weights multiplied by the input unit value in the hidden layer:

$$f(Z_{inj}) = \frac{1}{1+\exp(-(Z_{inj}))} \tag{6}$$

$$Z_j = f(Z_{inj}) \tag{7}$$

The summation of the hidden unit's value $(Z_{inj}, j = 1, \ldots, H)$ multiplied by its corresponding weights w_{jk} minus its threshold θ_k is calculated with the next equation:

$$y_{ink} = \sum_{i=0}^{H} z_j W_{jk} - \theta_k \tag{8}$$

The next point is to compute the output unit value $(Y_k, k = 1, \ldots\ldots O)$ by using a sigmoid function:

$$y_k = f(y_{ink}) \tag{9}$$

$$f(y_{ink}) = \frac{1}{1+\exp(-(y_{ink}))} \tag{10}$$

The error E can be computed by the mean square error (MSE) measure to check the difference between the target value and the real value at each output unit:

$$E = \frac{1}{m} \sum_{i=0}^{m} (d_i - Y_i)^2 \tag{11}$$

where, $(d_i - Y_i)$ is the error among the desired output value and the active output value of the i^{th} output unit, m is the number of data. Thus, the fitness of every particle can be measured by:

$$fitness(X_i) = E(X_i) \tag{12}$$

2.3 Particle Swarm optimization (PSO)

PSO algorithm was introduced [33] to simulate the group (swarm) communication behavior in migrating, flocking, and hunting of the birds. Each bird in the group is called a particle. In PSO a particle can changes its position based on the neighbors and its experience. This algorithm uses a loop to perform the optimization task. It starts by generating random particles, these particles update their position in each loop based on a fitness function. This scenario is iterated until meeting the stop condition. For more details, this reference [33] can be checked. In this paper, the PSO is used to generate the initial population of the MVO algorithm.

2.4 Multi-Verse Optimizer (MVO)

Multi-Verse Optimizer (MVO) [26] is an optimization algorithm. The MVO is inspired by the multi-verse theory in physics and the interaction among universes. This algorithm utilizes three concepts namely white, black, and worm holes. As many optimization methods, the MVO initializes by creating a random population. This population updates its solution by starting loops, in each loop the best solution is checked and saved. This technique is iterated till reaching the stop condition. The mathematical form of this algorithm can be represented as follows. The symbol x represents the solution $(x_b^a, b = 1, 2, \ldots, N)$ where N is the number of solutions. a is the number of parameters.

$$x_b^a = \begin{cases} x_c^a & r1 < NI(Ui) \\ x_b^a & r1 \geq NI(Ui) \end{cases} \tag{13}$$

where x_b^a denotes the ath variable of bth solution, x_b define the bth solution, NI(x_b) is normalized fitness function values. r_1 is a random variable in the range [0, 1]. x_c^a is selected by a roulette wheel mechanism. In each loop x_b^a is updated based on the following equation:

$$x_b^a = \begin{cases} \begin{cases} X_b + TDR \times ((ub_b - lb_b) \times r4 + lb_b) & r3 < 0.5 \\ X_b - TDR \times ((ub_b - lb_b) \times r4 + lb_b) & r3 \geq 0.5 \end{cases} & r2 < WEP \\ x_b^a & r2 \geq WEP \end{cases} \tag{14}$$

where, X_b is the current best solution. Wormhole existence probability (WEP) and travelling distance rate (TDR) are coefficients and defined as in Eqs. 15-16, respectively. ub and lb define the upper and lower bounds, respectively. The random variables r4, r3, and r2 in the range [0, 1].

$$TDR = 1 - \frac{l^{1/p}}{L^{1/p}} \tag{15}$$

$$WEP = min + l \times \left(\frac{max - min}{L}\right) \tag{16}$$

where l and L are the current iteration and the iterations' length, respectively. p, min, and max are default values and equal 6, 0.2, and 1, respectively. More information about this algorithm can be found in [26].

3 PROPOSED METHOD

In this section, an optimized multi-layer perceptron (MLP) neural network is applied in a regression problem. The weights and biases of the NN are optimized using PSO and MVO algorithms. The PSO is used to generate the initial population of the MVO to improve its population and increase its ability to get the optimal solution. Therefore, the proposed method is called pMVO-NN. The pMVO-NN method starts to score the essays by a pre-processing phase to prepare the input essays to be used in the next phases and generate the WCM matrix. Then SVD is applied. The produced matrix represents the dataset. This dataset is divided to a training set and testing set. The optimization phase of pMVO-NN starts by generating random vectors to be evaluated in optimizing NN. These vectors are evaluated in each iteration using the NN objective function then the MSE measure, as in Eq. (9), is calculated to check if the current vector is the best one to be used as weights and biases or not. Based on the MSE value the pMVO-NN is adapted its behavior for saving the best solution to be used in the next iterations. Fig. 1 illustrates the phases of the pMVO-NN method.

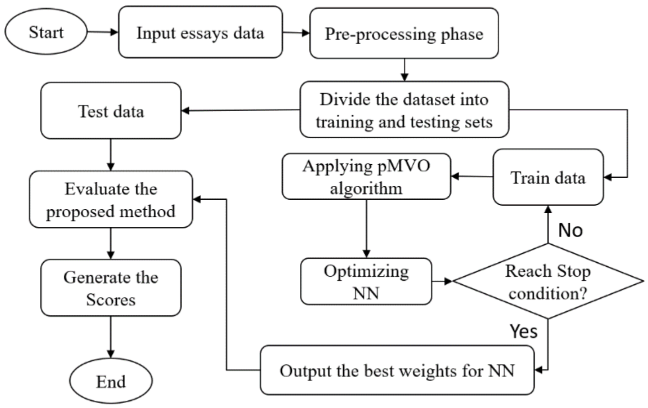


***Fig. 1**. The phases of the proposed method.*

4 EXPERIMENT AND RESULTS

4.1 Dataset description

A collection of 200 essays were collected from students' which written in Arabic. The query includes only one question, and the full mark was set to 10 degrees. The student's answers were corrected by two human assessors based on the course textbook to obtain the referential degrees. The average length of answers was 75 words per answer. The threshold answer was collected from the textbook to be added along with the student's responses. The dataset is randomly divided into 30% for training the proposed method (with grades vary from 0 to 10) and 70% for testing the proposed method.

4.2 Pre-processing phase

There are some pre-processing phases should be applied before training the proposed method as follow:

- Cleaning the content, removing all additional spaces, and all terms formation besides removing special characters (e.g. $, #, *, &,:, -, @).
- Unifying the term for the same word (text normalization).
- Spellchecking for all words in the dataset.
- Stemming the words, it is a significant and precise procedure where the complication of the Arabic language blocks the ability to build a robust system to execute this step, so stemming is done by methods that remove the suffix and the prefix of the term in the content for words more than three letters.
- Processing synonyms through exchanging them by synonyms dictionary to unify the famous terms with the same meaning in the content.
- Deletion of common terms "Stop Terms" does not greatly affect the substance of the content. To decrease the dimensions of the word-by-context matrix and try to focus on the core content within the text.
- After that, convert all current essays and the thresholds to vectors to create WCM matrix. Once the WCM is created, processing with LSA starts to determine the deserved grade of the current answer.
- Applying (1) and (2) for weighting calculation to produce WCM.

4.3 Pre-processing phase

The proposed method is evaluated using different measures namely Root mean square error (RMSE), Mean Absolute Error (MAE), Mean Absolute Relative Error (MARE), and Coefficient of Determination as in the following equations:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(k_i - K_i)^2} \quad (18)$$

where, k defines the output value and K defines the target value. n is the total number of data.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|k_i - K_i| \quad (19)$$

$$MARE = \frac{1}{n}\sum_{i=1}^{n}\left(\left|\frac{k_i - K_i}{k_i}\right|\right) \quad (20)$$

$$R2 = 1 - \frac{\sum(k_i - K_i)^2}{\sum(k_i - \mu_k)^2} \quad (21)$$

where, μ_k is the mean of the k values.

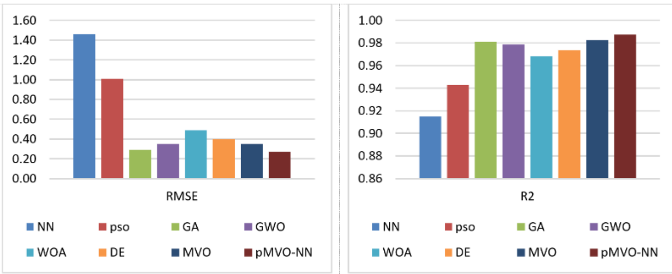
4.4 Results and Discussion

In this section, the pMVO-NN method is evaluated in scoring Arabic essays dataset. In this context, two experiments were conducted to evaluate the pMVO-NN over the results of the two human experts. the results of the proposed method were compared with the classic NN algorithm in addition to six optimization methods that used to optimize the NN namely PSO, GA, GWO, DE, MVO, and WOA. The parameters of all methods were taken from the original papers of them. The dataset was divided into two parts; the first one represents the 70% of the dataset for training and the rest was used for testing. The maximum number of iterations is set to 100 and the number of swarms is set to 25. The parameters of the algorithms are set as in their original studies. The statistical results are calculated after 30 runs. All experiments are performed using "MATLAB 2014" on "Windows 10" and "CPU Core I7" with 8GB of memory. Table 1 shows the results of the pMVO-NN in scoring Arabic essays based on the results of the first human expert along with all methods in terms of RMSE, MAE, MARE, R2 and time. From this table that can be noticed, the pMVO-NN is ranked first in all measures except for the time, whereas, it is ranked last. The pMVO-NN can achieve 98.7% of the correlation with the first human expert; it outperformed the classic NN by 7%. Based on the R2 measure, the MVO-NN came in the second rank with 98.2%. The rest of the algorithms are ranked as follows: GA, GWO, DE, WOA, and PSO, respectively. The fastest algorithm was the PSO, DE, and GWO, respectively. Fig. 2 summarizes these results based on RMSE and R2 measures.

TABLE 1

RESULTS OF THE PMVO-NN COMPARED TO THE OTHER METHODS IN TERMS OF THE SCORES OF THE FIRST HUMAN EXPERT

	Classic NN	PSO-NN	GA-NN	GWO-NN	WOA-NN	DE-NN	MVO-NN	pMVO-NN
RMSE	1.463	1.008	0.289	0.351	0.487	0.395	0.349	0.272
MAE	0.759	0.657	0.165	0.150	0.290	0.254	0.204	0.140
MARE	0.596	0.361	0.079	0.203	0.133	0.157	0.077	0.070
R2	0.914	0.942	0.980	0.978	0.967	0.973	0.982	0.987
Time	-	33.04	34.11	33.51	45.16	33.19	55.56	58.09



(a) (b)

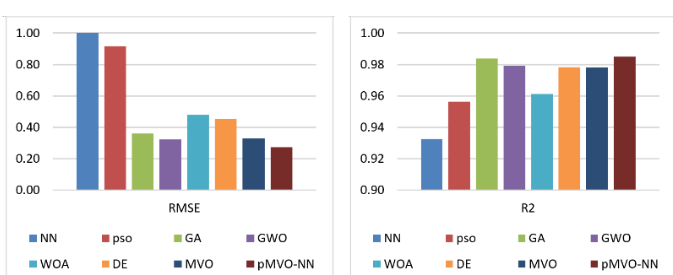
***Fig. 2**. Results based on the scores of the first human expert (a) RMSE results of the pMVO-NN compared to the other methods. (b) R2 results of the pMVO-NN compared to the other methods.*

Table 2 shows the results of the pMVO-NN in scoring Arabic essays based on the results of the second human expert along with all methods in terms of the performance measures. From this table that can be noticed, the pMVO-NN is also ranked second in all measures except for the time. The pMVO-NN can achieve 98.5% of the correlation with the second human expert; it outperformed the classic NN by 5%. The GA-NN and GWO-NN obtained the second and the third rank with 98.3% and 97.9%, respectively. The rest of the algorithms are ranked as follows: DE, MVO, and PSO, respectively. The fastest algorithm was the PSO, GWO, and DE, respectively, whereas, the pMVO-NN was obtained the last rank. Fig. 3 summarizes these results based on RMSE and R2 measures. The worst rank in time measure is due to the pMVO-NN takes more time to generate the population; however, this time is very closed to the classic MVO-NN and can be acceptable in automated scoring essays system.

TABLE 2

RESULTS OF THE PMVO-NN COMPARED TO THE OTHER METHODS IN TERMS OF THE SCORES OF THE SECOND HUMAN EXPERT

	Classic NN	PSO-NN	GA-NN	GWO-NN	WOA-NN	DE-NN	MVO-NN	pMVO-NN
RMSE	1.050	0.915	0.359	0.322	0.479	0.453	0.329	0.272
MAE	0.423	0.583	0.179	0.142	0.264	0.270	0.153	0.138
MRE	0.056	0.173	0.063	0.038	0.079	0.094	0.038	0.027
R2	0.932	0.956	0.983	0.979	0.961	0.978	0.978	0.985
Time	-	34.23	35.64	34.45	45.97	34.8	56.11	58.43



(a) (b)

***Fig. 3**. Results based on the scores of the second human expert (a) RMSE results of the pMVO-NN compared to the other methods. (b) R2 results of the pMVO-NN compared to the other methods.*

These results indicate that the pMVO-NN can optimize the NN's weight and bias effectively than the other methods. As well as it was able to score Arabic essays as the human expert to a large extent.

4.5 Statistical Analysis

In this section, the results of the pMVO-NN are analysis using a t-test as a statistical test to provide a statistical indicator to emphasis there is a significant difference between the pMVO-NN and the compared methods at the 5% significance level. Table 3 shows the results. From this table, that can be concluded, there is a significant difference between the pMVO-NN and the classic NN, PSO-NN, GWO-NN, WOA-NN, and MVO-NN at the 5% significance level in terms of RMSE measure. Whereas, there is no difference between GA-NN and MVO-NN because the results of these methods are similar to pMVO-NN to some extent.

TABLE 3

STATISTICAL RESULTS OF THE PMVO-NN COMPARED TO THE OTHER METHODS IN TERMS OF THE SCORES OF THE HUMAN EXPERTS

Measure	Classic NN	PSO-NN	GA-NN	GWO-NN	WOA-NN	DE-NN	MVO-NN
p-value Expert 1	< 0.05	<0.05	0.8814	< 0.05	< 0.05	<0.05	0.6958
p-value Expert 2	< 0.05	<0.05	0.5748	< 0.05	< 0.05	<0.05	0.7502

5 CONCLUSION

In this paper, a new method for automated essay scoring is proposed. It used particle swarm optimization to generate the initial population for the multiverse optimizer algorithm to train the classic neural network. It was called pMVO-NN. A set of 200 student's essays were used for evaluating the proposed method. These essays were scored by two human experts. The scored essays were passed to a pre-processing phase to be prepared and converted to a word-by-context matrix. The results were evaluated using five measures such as root mean square error (RMSE) and coefficient of determination (R2). The pMVO-NN outperformed all the compared optimization algorithms. It obtained a correlation equal to 0.987 with the scores of the first human expert and 0.985 with the second human expert. Due to the promising results of the proposed method, in the future, it will be evaluated using different size of essays and applied in different topics such as solving the feature selection and forecasting issues.

REFERENCES

[1] Jagdev Bhogal, Andrew MacFarlane, and Peter Smith. A review of ontology based query expansion. Information processing & management, 43(4):866–886, 2007.

[2] ChengXiang Zhai. Statistical language models for information retrieval. Synthesis Lectures on Human Language Technologies, 1(1):1–141, 2008.

[3] Manish Sharma and Rahul Patel. A survey on information retrieval models, techniques and applications. International Journal of Emerging Technology and Advanced Engineering, 3(11):542–545, 2013.

[4] Aqil M Azmi, Maram F Al-Jouie, and Muhammad Hussain. Aaee–automated evaluation of students' essays in arabic

language. Information Processing & Management, 56(5):1736–1752, 2019.
[5] AA Ewees, Mohamed Eisa, and MM Refaat. Comparison of cosine similarity and k-nn for automated essays scoring. International Journal of Advanced Research in Computer and Communication Engineering, 3(12), 2014.
[6] Marwa A Gaheen, Ahmed A Ewees, and Mohamed Eisa. Students head-pose estimation using partiallylatent mixture. In Emerging Trends in Electrical, Communications, and Information Technologies, pages 717–729. Springer, 2020.
[7] Marwa A Gaheen, Ahmed A Ewees, and Fifi Farouk. Face-pose estimation for learning systems. In 2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT), pages 1–6. IEEE, 2019.
[8] Ahmed A Ewees, Hend A ElLaban, and Rania M ElEraky. Features selection for facial expression recognition. In 2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT), pages 1–6. IEEE, 2019.
[9] Frank Hutter, Lars Kotthoff, and Joaquin Vanschoren. Automated machine learning-methods, systems, challenges, 2019.
[10] Khaled Ahmed, Ahmed A Ewees, and Aboul Ella Hassanien. Prediction and management system for forest fires based on hybrid flower pollination optimization algorithm and adaptive neuro-fuzzy inference system. In 2017 Eighth International Conference on Intelligent Computing and Information Systems (ICICIS), pages 299–304. IEEE, 2017.
[11] Liu Penghui, Ahmed A Ewees, Beste Hamiye Beyaztas, Chongchong Qi, Sinan Q Salih, Nadhir Al-Ansari, Suraj Kumar Bhagat, Zaher Mundher Yaseen, and Vijay P Singh. Metaheuristic optimization algorithms hybridized with artificial intelligence model for soil temperature prediction: Novel model. IEEE Access, 8:51884–51904, 2020.
[12] A. A. Bialy, M. A. Gaheen, R. M. ElEraky, A. F. ElGamal, and A. A. Ewees. Single arabic document summarization using natural language processing technique. In In Recent Advances in NLP: The Case of Arabic Language, pages 649–657. Springer, 2020.
[13] Nitin Madnani and Aoife Cahill. Automated scoring: Beyond natural language processing. In Proceedings of the 27th International Conference on Computational Linguistics, pages 1099–1109, 2018.
[14] Nal Kalchbrenner, Edward Grefenstette, and Phil Blunsom. A convolutional neural network for modelling sentences. arXiv preprint arXiv:1404.2188, 2014.
[15] Zakaria Alameer, Mohamed Abd Elaziz, Ahmed A Ewees, Haiwang Ye, and Zhang Jianhua. Forecasting gold price fluctuations using improved multilayer perceptron neural network and whale optimization algorithm. Resources Policy, 61:250–260, 2019.
[16] Ahmed T Sahlol, Yasmine S Moemen, Ahmed A Ewees, and Aboul Ella Hassanien. Evaluation of cisplatin efficiency as a chemotherapeutic drug based on neural networks optimized by genetic algorithm. In 2017 12th International Conference on Computer Engineering and Systems (ICCES), pages 682–685. IEEE, 2017.
[17] Ahmed T Sahlol, Ahmed A Ewees, Ahmed Monem Hemdan, and Aboul Ella Hassanien. Training feedforward neural networks using sine-cosine algorithm to improve the prediction of liver enzymes on fish farmed on nano-selenite. In 2016 12th International Computer Engineering Conference (ICENCO), pages 35–40. IEEE, 2016.
[18] Rie Johnson and Tong Zhang. Effective use of word order for text categorization with convolutional neural networks. arXiv preprint arXiv:1412.1058, 2014.
[19] David Zimbra, Manoochehr Ghiassi, and Sean Lee. Brand-related twitter sentiment analysis using feature engineering and the dynamic architecture for artificial neural networks. In 2016 49th Hawaii International Conference on System Sciences (HICSS), pages 1930–1938. IEEE, 2016.
[20] Nizar Y Habash. Introduction to arabic natural language processing. Synthesis Lectures on Human Language Technologies, 3(1):1–187, 2010.
[21] Wenxin Zhu. A study on the application of automated essay scoring in college english writing based on pigai. In 2019 5th International Conference on Social Science and Higher Education (ICSSHE 2019). Atlantis Press, 2019.
[22] Syed Latifi, Mark J Gierl, Andr´e-Philippe Boulais, and Andr´e F De Champlain. Using automated scoring to evaluate written responses in english and french on a high-stakes clinical competency examination. Evaluation & the health professions, 39(1):100–113, 2016.
[23] Mansour Alghamdi, Mohamed Alkanhal, Mohamed Al-Badrashiny, Abdulaziz Al-Qabbany, Ali Areshey, and Abdulaziz Alharbi. A hybrid automatic scoring system for arabic essays. Ai Communications, 27(2):103–111, 2014.
[24] Ayad R Abbas and Ahmed S Al-qazaz. Automated arabic essay scoring (aaes) using vectors space model (vsm) and latent semantics indexing (lsi). Engineering and Technology Journal, 33(3 Part (B) Scientific):410–426, 2015.
[25] Kaveh Taghipour and Hwee Tou Ng. A neural approach to automated essay scoring. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pages 1882–1891, 2016.
[26] Seyedali Mirjalili, Seyed Mohammad Mirjalili, and Abdolreza Hatamlou. Multi-verse optimizer: a nature-inspired algorithm for global optimization. Neural Computing and Applications, 27(2):495–513, 2016.
[27] Ahmed A Ewees, Mohamed Abd El Aziz, and Aboul Ella Hassanien. Chaotic multi-verse optimizerbased feature selection. Neural Computing and Applications, 31(4):991–1006, 2019.
[28] Mohamed Abd Elaziz, Diego Oliva, Ahmed A Ewees, and Shengwu Xiong. Multi-level thresholdingbased grey scale image segmentation using multi-objective multi-verse optimizer. Expert Systems with Applications, 125:112–129, 2019.
[29] Scott Deerwester, Susan T Dumais, GeorgeWFurnas, Thomas K Landauer, and Richard Harshman. Indexing by latent semantic analysis. Journal of the American society for information science, 41(6):391–407, 1990.
[30] Thomas K Landauer, Peter W Foltz, and Darrell Laham. An introduction to latent semantic analysis. Discourse processes, 25(2-3):259–284, 1998.
[31] Tuomo Kakkonen and Erkki Sutinen. Automatic assessment of the content of essays based on course materials. In ITRE 2004. 2nd International Conference Information Technology: Research and Education, pages 126–130. IEEE, 2004.
[32] Haydee Melo and Junzo Watada. Gaussian-pso with fuzzy reasoning based on structural learning for training a neural network. Neurocomputing, 172:405–412, 2016.
[33] Abdulaziz Shehab, Mahmoud Faroun, and Magdi Rashad. An automatic arabic essay grading system based on text similarity algorithms. International Journal of Advanced Computer Science and Applications (IJACSA), 9(3):263–268, 2018