

Automated feature selection using improved migrating birds optimization for enhanced medical diagnosis

Naoual El Aboudi, Youness Riouali, Ahmed Reda Maminou, Hassane Jabri, Laila Benhlma

Ecole Mohammadia d'ingénieurs, Mohammed V University in Rabat, Rabat, Morocco

Article Info

Article history:

Received Oct 21, 2023

Revised Jan 14, 2024

Accepted Jan 15, 2024

Keywords:

Automated pipeline

Feature selection

Machine learning

Medical diagnosis

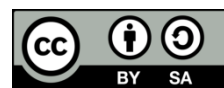
Migrating birds optimization

MLOps

ABSTRACT

The feature selection task is a crucial phase in data analysis, aiming to identify a minimized set of relevant features for the target class, thereby eliminating irrelevant and redundant attributes used for model training. While population-based feature selection approaches offer prominent solutions for classification performance, their computational time can be prohibitive. To mitigate delays and optimize resource utilization, this study adopts machine learning operations (MLOps). MLOps involves the seamless transition of experimental machine learning models into production, serving them to end users and automating the feature selection phase. This paper introduces a novel feature selection method based on improved migrating bird optimization and its automated variant integrated into MLOps. Experiments conducted on six medical datasets validate the effectiveness of our proposed feature selection method in improving the outcomes of medical diagnosis systems. The results showcase satisfactory performance in terms of classification compared to concurrent feature selection algorithms.

This is an open access article under the [CC BY-SA](#) license.



Corresponding Author:

Naoual El Aboudi

Ecole Mohammadia d'ingénieurs, Mohammed V University in Rabat

Rabat, Morocco

Email: n.el022021@gmail.com

1. INTRODUCTION

Data preprocessing and model training are pivotal stages in the machine learning process, demanding significant time and effort [1]. Raw data often harbors imperfections such as missing, noisy, and redundant information, underscoring the importance of preprocessing steps to enhance prediction quality [2], [3]. In this context, feature selection emerges as a critical aspect of data preprocessing, aiming to identify and retain relevant features while eliminating those independent of the target class. Population-based feature selection approaches offer a promising solution despite their computational intensity. Inspired by natural phenomena, such as animal behavior in colonies, birds, and fish, these techniques utilize optimization algorithms. They iteratively refine initial solutions while attempting to avoid local minima, aiming to produce near-optimal solutions based on a defined objective function. Among population-based methods, migrating bird optimization (MBO) stands out, especially when applied to the feature subset problem [4].

Recent advancements have witnessed the application of population-based feature selection approaches to enhance medical diagnosis. Noteworthy methods include a stacked genetic-based approach for heart disease diagnosis and the utilization of cat swarm optimizer (SCSO) and binary memory-based SCSO (BMSCSO) on 21 benchmark disease datasets [5]. Xue presented a new population-based feature selection method based on genetic algorithm combined to relief [6]. A new feature selection method based on particle swarm optimization was proposed to tackle privacy protection problem [7]. In another study, feature selection using genetic algorithm and support vector method is adopted for coronavirus disease (COVID-19)

diagnosis [8]. Meenachi applied a variety of metaheuristics methods-based feature selection to handle cancer classification [9]. However, the prevailing manual pipeline process, especially in the feature selection phase, incurs substantial operational costs and introduces delays, directly impacting overall system performance. In response, the intersection of development operations (DevOps) and machine learning (ML), known as machine learning operations (MLOps), emerges as a compelling avenue [10]. MLOps, an extension of DevOps principles tailored for ML [11], offers advantages like reduced development cycles, increased release frequency, and automation potential for feature selection phases. The automation of Feature selection task through MLOps has never been done before to the best of authors knowledge.

This study aims to contribute by proposing a new feature selection method based on improved migrating birds optimization and automating it within the MLOps lifecycle, minimizing computational costs, and ensuring satisfactory results. More specifically, key contributions of the present paper include the following: i) Proposing a new population-based feature selection approach based on improved migrating birds optimization within a manual machine learning process; ii) Automating the migrating birds optimization-based feature selection approach through the MLOps pipeline, a novel endeavor compared to other feature selection methods; and iii) Conducting experiments on six datasets designed to improve medical diagnosis using the proposed improved migrating birds optimization-based feature selection.

Section 2 provides a background of population-based feature selection methods and machine learning operations and introduces an automation pipeline in machine learning for prediction, outlining key components and processes. Section 3 presents our proposed method based on improved migrating birds optimization for feature selection and its automation through the MLOps pipeline. Experiments on medical datasets are presented and discussed in section 4. Finally, section 5 concludes the paper, summarizing key findings and offering recommendations for future research.

2. BACKGROUND

2.1. Feature selection

The feature selection task is a crucial phase in data analysis, aiming to identify a minimized set of relevant features for the target class [12]. It involves eliminating irrelevant and redundant attributes used for model training [13]. Feature selection approaches can be categorized into four types: filters, wrappers, hybrids, and embedded methods. Filters algorithms select features based on statistical criteria, eliminating those with scores below a predefined threshold. Wrapper approaches, on the other hand, use a classification method to compute performance accuracy for candidate feature subsets. Hybrid methods combine the strengths of both filter and wrapper approaches [14], [15]. Wrapper approaches are further categorized based on search methods, including exponential complexity methods, sequential selection algorithms, and population-based methods. Exponential complexity approaches explore all possible feature subsets, being time-consuming. Sequential selection methods, like greedy algorithms, involve a hill climbing process to find the optimal feature subset [16]. Sequential methods consist of forward and backward approaches. Sequential forward selection (SFS) starts with an empty feature set, progressively improving classification performance by adding features until no further enhancement is possible. In contrast, sequential backward selection (SBS) discards features based on initial subset classification performance, eliminating redundant features. However, these methods become time-consuming with a high number of attributes [17], [18]. In the following section, we explore population-based feature selection methods, a distinct category of approaches designed to effectively tackle feature selection challenges.

2.2. Population based feature selection methods

Population-based methods yield effective solutions within reasonable processing time, leveraging a technique inspired by natural evolution. The process commences with the generation of an initial population, consisting of a set of solutions that are iteratively updated to enhance fitness according to a predefined criterion. This iterative process persists until a stop criterion is met, signaling the completion of execution and the provision of an optimal solution. Genetic algorithms (GA), particle swarm optimization (PSO), and migrating birds optimization (MBO) are prominent examples of population-based techniques widely adopted in the realm of feature selection [19]–[21]. In the subsequent section, we will closely examine MBO method due to its relevance and effectiveness in this context. To automate machine learning operations, a machine learning automated pipeline is adopted, with its components detailed in the following section.

2.3. Automation pipeline in machine learning for prediction

Deploying ML models into production with confidence and trust is a challenging task, requiring seamless integration with organizational processes and practices [22], [23]. Often, organizations invest substantial effort in building ML models and deploying them behind application programming interface (API) endpoints.

However, this practice overlooks critical tasks such as accessing and preparing data in production, connecting models with online business applications, and delivering continual enhancements. Moreover, manual development and deployment processes incur additional time and resource costs for production releases [24].

To initiate a ML process, it is essential to establish extract-transform-load (ETL) steps. This process focuses on compiling a dataset from diverse sources, undertaking tasks such as data cleaning, labeling, and ensuring that the analyzed data is ready for subsequent stages. This pipeline is responsible for the construction, testing, and evaluation of a model. At this stage, tasks like feature discovery and model evaluation are efficiently managed. Subsequently, the serving pipeline takes charge of model deployment and monitoring. This involves transferring the trained model to end-users and establishing monitoring systems.

The model-building process initiates by defining multiple models with the objective of identifying the most suitable and high-performing one. These models undergo assessment and training, iterating through a set of parameters. During the testing stage, the optimal model is selected. The trained model is rigorously tested against various metrics, utilizing testing datasets in the evaluation step. At this stage, the model's performance is compared to the one produced. Subsequently, the model is chosen for deployment into production if it attains a satisfactory test score. Finally, the selected model is stored. Once the model is built and evaluated, the next phase involves its deployment into the production environment. Subsequently, the model is made accessible to end-users through microservices or front-end applications, ensuring access to a validated and reliable model for predictions. The model's performance is continuously monitored, prompting the re-execution of the pipeline when necessary. In the following section, we propose a new feature selection method based on an improved MBO algorithm, followed by its automated integration into the MLOps pipeline.

3. RESEARCH METHOD

In the pursuit of an optimized feature selection methodology, Section 3 unfolds the proposed approach. This section encompasses both the development of improved MBO-based feature selection (section 3.1) and its automation within the MLOps Pipeline (section 3.2). The improved feature selection is designed to enhance the overall efficiency of the process, while its integration into the MLOps Pipeline ensures a streamlined and automated workflow for improved functionality.

3.1. Improved migrating birds optimization-based feature selection

3.1.1. Migrating birds optimization

The migrating birds optimization (MBO) was first introduced by Duman in [25], is a population-based neighborhood search technique. In contrast to other population-based methods, MBO algorithm is a population-based approach that incorporates a neighborhood function in its principle, enabling a comprehensive exploration of the search space. It has been applied to address several NP-hard optimization problems, including the quadratic assignment problem (QAP) [26], and has demonstrated efficacy in enhancing the performance of artificial neural network (ANN) classifiers [27].

The MBO algorithm emulates the V-formation flight of migrating birds, a strategy that allows them to conserve energy [28]. In MBO method, each bird, except the leader in the V-flight formation, utilizes the upward air vortices produced by the preceding one to fly efficiently. Similarly, the MBO algorithm initiates by selecting a leader, and the other solutions are divided into two groups to mimic the V-flight formation. Concerning the chosen neighborhood structure, each solution generates a predefined number of solutions. Subsequently, the MBO algorithm compares each solution with its neighbors in terms of the fitness function, starting with the leader solution. If a solution's fitness exceeds that of the candidate solution, it is replaced. Moreover, the current solution leverages the best-unused neighbors of the preceding solution, aiming to enhance its quality. In essence, a specified solution seeks improvement by considering the best candidate from a group of its own neighbors, while some of the best neighbors inherit from the previous solution. This iterative process occurs over multiple tours until a specified number of iterations is reached. Eventually, one of the subsequent solutions assumes the role of the leader, and the previous steps are repeated. The MBO algorithm is parameterized by a predefined number of initial solutions (n), producing k neighbor solutions for the leader solution. Additionally, each candidate solution should share x neighbor solutions with the furthest solution. The number of tours (m) is user-defined, and an iteration limit (K) is set accordingly.

3.1.2. Migrating bird's optimization-based feature selection

In this subsection, we introduce the foundation of our approach, wherein each solution of MBO is represented by a binary vector (X) with the size of the feature set. The value '1' indicates that the specified feature is selected, while '0' denotes the exclusion of the candidate feature. The proposed method introduces a new variant of MBO tailored for handling feature subset problems. The feature selection component comprises four subcomponents:

- Generating initial population: There are four techniques of initialization, forward initialization (inspired by SFS), backward initialization (based on SBS), merged initialization (which combines SFS and SBS methods), and random initialization. In our study, we adopt merged initialization due to its satisfactory results. This method involves splitting all solutions into two groups, where the first group comprises solutions initialized using forward initialization, and the second group includes solutions initialized by backward initialization.
- Generating neighbors: As our proposal is a neighborhood search technique, designing an effective neighborhood is crucial for exploring the entire search space and identifying the best solution. Our neighborhood is based on symmetric uncertainty (SU), which has been used to identify correlated features to class labels or between them through nonlinear relationships.
- Evaluation: The neighbor solutions undergo evaluation through a learning algorithm to determine the best solution.

The pseudocode for the MBO with SU-based feature selection algorithm (MBOSU-FS) is provided in Algorithm 3. It is built upon the concepts presented in Algorithm 1 (identifying relevant features) and Algorithm 2 (identifying irrelevant features), detailed in Algorithms 1 and 2, respectively.

Algorithm 1. Identifying relevant features

Input: The original feature set, $F=(f_1, f_2, \dots, f_d)$, the set of class labels, C , S_{\min}
Output: The set F_{relevant}
Compute the $SU(f_i, C)$, for $i=1, 2, \dots, d$;
for $i=1$ to d do
 if $SU(f_i, C) \geq S_{\min}$ then
 Save the i -th feature into the set F_{relevant} ;
 End if
End for
return F_{relevant}

Algorithm 2. Identifying irrelevant features

Input: The original feature set, $F=(f_1, f_2, \dots, f_d)$, the set of class labels, C , S_{\max}
Output: The set $F_{\text{irrelevant}}$
Compute the $SU(f_i, C)$, for $i=1, 2, \dots, d$;
for $i=1$ to d do
 if $SU(f_i, C) \leq S_{\max}$ then
 Save the i -th feature into the set $F_{\text{irrelevant}}$;
 End if
End for
return $F_{\text{irrelevant}}$

Algorithm 3. Migrating birds optimization with symmetric uncertainty based feature section (MBOSU-FS)

Input: Initial set of features X , random number j
Output: Neighbor solution X'
 $X' \leftarrow X$
Choose random j entries from X to be changed;
foreach i from selected entries do
 if $((X_i=1) \text{ and } (f_i \in F_{\text{irrelevant}}))$ then
 $X'_i=0$;
 else if $((X_i=0) \text{ and } (f_i \in F_{\text{relevant}}))$ then
 $X'_i=1$;
 End if
End foreach
Compute $f(X')$
return X' , $f(X')$

In our proposed MBOSU-FS method, each solution is represented by a binary vector (X), where d is the size of the entire feature space. Here, (X_i) is a binary entry representing the selection status of the i^{th} feature in the dataset. The candidate feature (F_i) corresponds to the i^{th} element of the feature set. We design a neighborhood function that introduces changes to (j) randomly chosen entries in the current solution. The selection of these entries is performed among those that have the potential to be altered. Our neighborhood structure relies on symmetric uncertainty (SU). The newly obtained solution replaces the current solution if its fitness score surpasses the score of the current solution.

3.2. Automated feature selection based migrating birds optimization within MLOps pipeline

This section focuses on automating the feature selection procedure to ensure the continuous delivery of the best model. As discussed in the previous section, the feature selection component aims to propose the most prominent features for building the model in the MLOps pipeline. Figure 1 illustrates the MLOps

pipeline deployment using automated feature selection steps. The MLOps pipeline consists of several components:

- a. ETL process: This phase, synonymous with data preprocessing, involves the extraction, transformation, and loading of raw data to prepare it for subsequent modeling stages [29]. During this stage, original data, characterized by diverse dimensions and formats, is extracted from various sources. Subsequently, the data undergoes cleaning and labeling, addressing issues such as misspellings, inconsistencies, duplicates, conflicts, or missing information. Following this, the data is transformed into a standardized format, allowing aggregation with other sources into its final business-ready format, and ultimately stored in production databases. The subsequent subcomponents play crucial roles in the data preprocessing stage:
 - Data cleaning: This component involves detecting, filling, correcting, modifying and removing noisy, corrupted, duplicated, incorrectly formatted, missing, and irrelevant data.
 - Data imputation: Missing values can have a big impact on the conclusions and results drawn from data and if they are not handled properly. Therefore, after collecting the data, the next step is to examine the missing values and use data imputation techniques to extrapolate them.
 - Data integration: This step involves merging data from multiple sources that can be homogeneous or heterogeneous into a unified and coherent dataset.
 - Data normalization: It is used for scaling the values to fit into a predetermined range and aggregating the data according to the needs of the data set.
- b. Model building pipeline: In this phase, various machine learning algorithms are implemented using the preprocessed data to train diverse models. Notably, the improved migrating birds optimization (MBOSU-FS) is applied to generate the most effective ML model. The output of this step is a trained and validated model.
- c. Model deployment and serving: The validated model is deployed into a production environment for making predictions in real-world scenarios.
- d. Model monitoring: Continuous monitoring of the predictive performance ensures the model's effectiveness over time, prompting a new iteration in the ML process if necessary.
- e. Automated triggering: This phase is executed automatically based on a predetermined schedule or triggered in response to observed degradation in the production environment. It initiates a new iteration of feature selection and, consequently, a new cycle of pipeline execution.

With a comprehensive understanding of our methodology's core components, we proceed to experimental results in subsequent section.

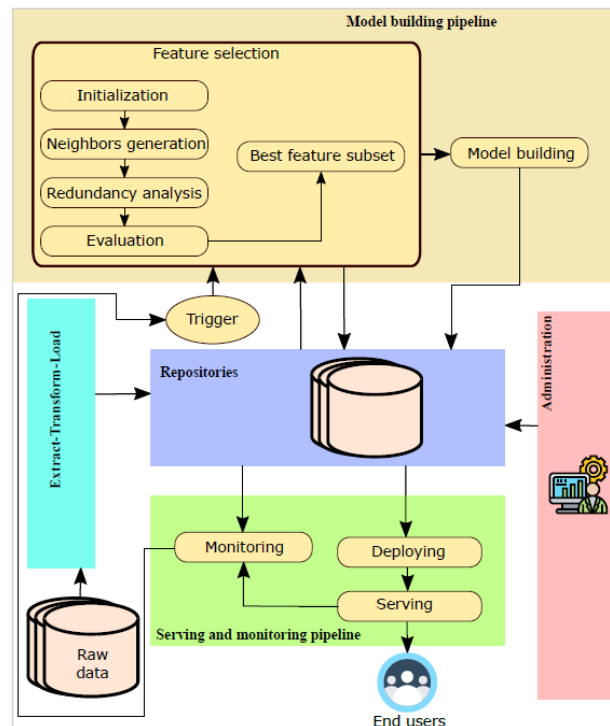


Figure 1. Automated feature selection process within the ML pipeline

4. RESULTS AND DISCUSSION

In the context of medical diagnosis, the judicious selection of features is paramount for constructing an effective predictive model. The performance of MBO-based feature selection method combined with SU was evaluated using six medical datasets from UCI [30]. These datasets were divided into a 70% training set and a 30% testing set, with continuous attribute datasets being discretized. The K-nearest neighbor (k-NN) classifier [31] was employed, a widely adopted classification method in artificial intelligence applications.

In the context of medical diagnosis, selecting a relevant set of features is crucial for constructing an efficient predictive model. To assess the performance of MBO-based feature selection method combined with SU, we chose six medical datasets from UCI [30]. These datasets were divided into a 70% training set and a 30% testing set, with continuous attribute datasets being discretized. Our experiments utilize the k-NN classifier [31], a classification method widely adopted in applications such as artificial intelligence.

Our paper employs a 10-fold cross-validation method on the training set to evaluate the resulting solutions. The testing classification error rate is then calculated based on the testing dataset. Table 1 illustrates the characteristics of the considered datasets.

In this experiment, MBOSU-FS initiates with 11 initial solutions ($n=11$) and undergoes 500 iterations ($K=500$). The leading solution generates 3 solutions ($k=3$), where each candidate solution shares one neighbor solution with the subsequent solution ($x=1$). The number of tours is set at 3 ($m=3$), and 2 denotes the number of entries to be changed. MBOSU-FS is systematically compared with concurrent feature selection methods: Particle swarm optimization for feature selection (PSOFS) [32] and genetic algorithm-based feature selection (GAFS) [19]. To ensure a fair comparison, all experiments adhere to the same number of fitness evaluations [26]. Additionally, all methods employ an identical number of initial solutions ($n=11$).

In the case of PSOFS, the ($c1$) and ($c2$) learning components are fixed at 2 ($c1=c2=2$). (v_{max}) and (v_{min}) bounds of velocities were set to -4 and 4. For GAFS, the probability of the crossover operator and the probability of the mutation operator were set to be 0.6 and 0.02, respectively. The computations were conducted on an HP ZBook Studio 15 laptop with a Core i7-7820HQ 2.90 GHz CPU and 32 GB of RAM. The results demonstrate that MBOSU-FS consistently outperformed concurrent algorithms, yielding similar or superior outcomes. Across the majority of datasets, MBOSU-FS achieved enhanced performance. For instance, on the Arrhythmia dataset, the proposed MBOSU-FS method yielded an average accuracy of 48.81%. Notably, it outperformed all other methods on the Lung dataset, achieving an average accuracy of 81.34%. Table 2 presents the classification performance results for the utilized feature selection methods, while Table 3 details the outcomes of our proposal and other methods in terms of the number of selected features. The results of feature selection methods were obtained from 20 independent runs, and the reported values adhere to a 95% confidence interval.

The computational time results reported in Table 4 indicate that MBOSU-FS has an affordable computational cost when the k-NN classifier is adopted. The smaller amount of time spent when using the k-NN classifier is expected, given that the complexity of the MBOSU-FS method is quadratic. The complexity of the k-NN algorithm is $O(knd)$, where d is the dimension of each solution (i.e., the number of features). The average number of selected features by MBOSU-FS reflects its ability to discern relevant features efficiently.

Table 1. Datasets used for experiments

Datasets	Features	Instances	Classes
Pima	8	768	2
Lung	56	32	3
Arrhythmia	279	452	16
WBCD	30	569	2
Hepatitis	19	155	2
Heart	13	303	2

Table 2. Accuracy of feature selection methods

Accuracy (%)	Feature selection methods		
	GAFS	PSOFS	MBOSU-FS
Pima	69.65%	68.71%	70.83%
Lung	80.15 %	81.59%	81.65%
Arrhythmia	47.23%	48.06%	48.87%
WBCD	93.55%	93.87%	94.23%
Hepatitis	81.5%	81.33%	81.66%
Heart	97%	94.6%	96.21%

Table 3. The average number of selected features for each database

Number of Selected Features	MBOSU-FS
Pima	8
Lung	16
Arrhythmia	83
WBCD	9
Hepatitis	11
Heart	7

Table 4. Computational time of MBOSU-FS (seconds)

Datasets	MBOSU-FS
Pima	0.145
LUNG	0.383
Arrhythmia	0.761
WBCD	0.349
Hepatitis	0.221
Heart	0.33

Our study included a comparative analysis with contemporary feature selection methods, namely PSOFS and GAFFS. MBOSU-FS consistently outperformed these methods, reaffirming its effectiveness. The performance gains achieved by MBOSU-FS are not only in terms of accuracy but also in the number of selected features, striking a balance between classification performance and computational efficiency.

The superior performance of MBOSU-FS across diverse datasets holds promising implications for medical diagnosis applications. The method's ability to discern relevant features while maintaining computational efficiency positions it as a valuable tool in the development of predictive models. The notable accuracy on datasets like Lung further underscores its potential in addressing intricate healthcare challenges.

5. CONCLUSION

In this paper, we proposed a novel and enhanced MBO-based feature selection method to improve outcomes of medical diagnosis. Then, we have underscored the significance of automating machine learning operations, emphasizing the acceleration of processes by automating our proposed feature selection method through MLOps pipeline. While our current focus has been on the feature selection process, our ongoing efforts are directed toward implementing specific components for the full automation of this process. Looking ahead, our primary future goal is to extend automation to encompass a broader spectrum of machine learning components, thereby ensuring a more efficient machine learning lifecycle.

REFERENCES





- [1] M. Z. Al-Taie, S. Kadry, and J. P. Lucas, "Online data preprocessing: A case study approach," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 9, no. 4, pp. 2620–2626, Aug. 2019, doi: 10.11591/ijece.v9i4.pp2620-2626.
- [2] K. Maharana, S. Mondal, and B. Nemade, "A review: Data pre-processing and data augmentation techniques," *Global Transitions Proceedings*, vol. 3, no. 1, pp. 91–99, Jun. 2022, doi: 10.1016/j.glt.2022.04.020.
- [3] E. A. Felix and S. P. Lee, "Systematic literature review of preprocessing techniques for imbalanced data," *IET Software*, vol. 13, no. 6, pp. 479–496, Dec. 2019, doi: 10.1049/iet-sen.2018.5193.
- [4] E. Duman, M. Uysal, and A. F. Alkaya, "Migrating birds optimization: A new metaheuristic approach and its performance on quadratic assignment problem," *Information Sciences*, vol. 217, pp. 65–77, Dec. 2012, doi: 10.1016/j.ins.2012.06.032.
- [5] A. Qtaish, D. Albashish, M. Braik, M. T. Alshammari, A. Alreshidi, and E. J. Alreshidi, "Memory-based sand cat swarm optimization for feature selection in medical diagnosis," *Electronics (Switzerland)*, vol. 12, no. 9, Apr. 2023, doi: 10.3390/electronics12092042.
- [6] Y. Xue, H. Zhu, and F. Neri, "A feature selection approach based on NSGA-II with ReliefF," *Applied Soft Computing*, vol. 134, p. 109987, Feb. 2023, doi: 10.1016/j.asoc.2023.109987.
- [7] Y. Hu *et al.*, "A federated feature selection algorithm based on particle swarm optimization under privacy protection," *Knowledge-Based Systems*, vol. 260, p. 110122, Jan. 2023, doi: 10.1016/j.knsys.2022.110122.
- [8] Wiharto, E. Suryani, S. Setyawan, and B. P. Putra, "The cost-based feature selection model for coronary heart disease diagnosis system using deep neural network," *IEEE Access*, vol. 10, pp. 29687–29697, 2022, doi: 10.1109/ACCESS.2022.3158752.
- [9] L. Meenachi and S. Ramakrishnan, "Metaheuristic search based feature selection methods for classification of cancer," *Pattern Recognition*, vol. 119, Art. no. 108079, Nov. 2021, doi: 10.1016/j.patcog.2021.108079.
- [10] A. Mishra and Z. Otaiwi, "DevOps and software quality: A systematic mapping," *Computer Science Review*, vol. 38, Nov. 2020, doi: 10.1016/j.cosrev.2020.100308.
- [11] D. Kreuzberger, N. Kuhl, and S. Hirschl, "Machine learning operations (MLOps): Overview, definition, and architecture," *IEEE Access*, vol. 11, pp. 31866–31879, May 2023, doi: 10.1109/ACCESS.2023.3262138.
- [12] H. Mamdouh Farghaly and T. Abd El-Hafeez, "A high-quality feature selection method based on frequent and correlated items for text classification," *Soft Computing*, vol. 27, no. 16, pp. 11259–11274, Jun. 2023, doi: 10.1007/s00500-023-08587-x.
- [13] A. Alzahrani and M. A. A. Bhuiyan, "Feature selection for urban land cover classification employing genetic algorithm," *Bulletin of Electrical Engineering and Informatics*, vol. 11, no. 2, pp. 793–802, Apr. 2022, doi: 10.11591/eei.v11i2.3399.
- [14] N. Singh and P. Singh, "A hybrid ensemble-filter wrapper feature selection approach for medical data classification,"

Automated feature selection using improved migrating birds optimization for ... (Naoual El Aboudi)





- Chemometrics and Intelligent Laboratory Systems*, vol. 217, Oct. 2021, doi: 10.1016/j.chemolab.2021.104396.
- [15] O. M. Alyasiri, Y. N. Cheah, A. K. Abasi, and O. M. Al-Janabi, "Wrapper and hybrid feature selection methods using metaheuristic algorithms for english text classification: A systematic review," *IEEE Access*, vol. 10, pp. 39833–39852, 2022, doi: 10.1109/ACCESS.2022.3165814.
- [16] O. O. Akinola, A. E. Ezugwu, J. O. Agushaka, R. A. Zitar, and L. Abualigah, "Multiclass feature selection with metaheuristic optimization algorithms: a review," *Neural Computing and Applications*, vol. 34, no. 22, pp. 19751–19790, Aug. 2022, doi: 10.1007/s00521-022-07705-4.
- [17] Abdullah, I. Faye, and M. R. Islam, "EEG channel selection techniques in motor imagery applications: A review and new perspectives," *Bioengineering*, vol. 9, no. 12, p. 726, Nov. 2022, doi: 10.3390/bioengineering9120726.
- [18] F. Asdaghi and A. Soleimani, "An effective feature selection method for web spam detection," *Knowledge-Based Systems*, vol. 166, pp. 198–206, Feb. 2019, doi: 10.1016/j.knsys.2018.12.026.
- [19] B. Oluyele, A. Leisa, J. Leng, and D. Dean, "A genetic algorithm-based feature selection," *International Journal of Electronics Communication and Computer Engineering*, vol. 5, pp. 899–905, 2014.
- [20] A. G. Gad, "Particle swarm optimization algorithm and its applications: A systematic review," *Archives of Computational Methods in Engineering*, vol. 29, no. 5, pp. 2531–2561, Apr. 2022, doi: 10.1007/s11831-021-09694-4.
- [21] E. Ulker and V. Tongur, "Migrating birds optimization (MBO) algorithm to solve knapsack problem," *Procedia Computer Science*, vol. 111, pp. 71–76, 2017, doi: 10.1016/j.procs.2017.06.012.
- [22] A. Paleyes, R. G. Urma, and N. D. Lawrence, "Challenges in deploying machine learning: A survey of case studies," *ACM Computing Surveys*, vol. 55, no. 6, pp. 1–29, Dec. 2022, doi: 10.1145/3533378.
- [23] L. E. Lwakatare, A. Raj, I. Crnkovic, J. Bosch, and H. H. Olsson, "Large-scale machine learning systems in real-world industrial settings: A review of challenges and solutions," *Information and Software Technology*, vol. 127, Art. no. 106368, Nov. 2020, doi: 10.1016/j.infsof.2020.106368.
- [24] M. S. Rahman, E. Rivera, F. Khomh, Y.-G. Guéhéneuc, and B. Lehnert, "Machine learning software engineering in practice: An industrial case study," *arxiv.org/abs/1906.07154*, Jun. 2019.
- [25] A. Zakaryazad, E. Duman, and A. Kibekbaev, *Profit-based artificial neural network (ANN) trained by migrating birds optimization: A case study in credit card fraud detection*. 2015.
- [26] M. M. Tahroodi and A. Payan, "A method to rank the efficient units based on cross efficiency matrix without involving the zero weights," *International Journal of Data Analysis Techniques and Strategies*, vol. 11, no. 2, pp. 101–114, 2019, doi: 10.1504/IJDATS.2019.098821.
- [27] S. Ramírez-Gallego, B. Krawczyk, S. García, and F. W. M. Herrera, "A survey on data preprocessing for data stream mining: Current status and future directions," *Neurocomputing*, vol. 239, pp. 39–57, May 2017, doi: 10.1016/j.neucom.2017.01.078.
- [28] N. Cheung, *Machine learning techniques for medical analysis*. Atlanta: School of Information Technology and Electrical Engineering, 2001.
- [29] A. K. Hamoud, M. K. Hussein, Z. Alhilfi, and R. H. Sabr, "Implementing data-driven decision support system based on independent educational data mart," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 11, no. 6, pp. 5301–5314, Dec. 2021, doi: 10.11591/ijece.v11i6.pp5301-5314.
- [30] B. Xue, M. Zhang, and W. N. Browne, "A comprehensive comparison on evolutionary feature selection approaches to classification," *International Journal of Computational Intelligence and Applications*, vol. 14, no. 2, 2015, doi: 10.1142/S146902681550008X.
- [31] M. Memik, S. H. Liu, D. Karaboga, and M. Črepinšek, "On clarifying misconceptions when comparing variants of the Artificial Bee Colony Algorithm by offering a new implementation," *Information Sciences*, vol. 291, no. C, pp. 115–127, Jan. 2015, doi: 10.1016/j.ins.2014.08.040.
- [32] B. Xue, M. Zhang, and W. N. Browne, "Particle swarm optimisation for feature selection in classification: Novel initialisation and updating mechanisms," *Applied Soft Computing Journal*, vol. 18, pp. 261–276, May 2014, doi: 10.1016/j.asoc.2013.09.018.

BIOGRAPHIES OF AUTHORS






Naoual El Aboudi     attained her Ph.D. from the Mohammadia Engineering School, Mohammed V University in Rabat, solidifying her commitment to advancing knowledge in her field. In 2010, she earned her degree in computer science engineering from the same institution. Her research endeavors encompass the dynamic realms of machine learning, big data analytics, and data mining. Through her contributions, she seeks to unravel the complexities of these domains, leaving an indelible mark on the landscape of computer science. For inquiries, she can be contacted at n.el022021@gmail.com.






Youness Riouali     earned his Ph.D. in Engineering Sciences from The Mohammadia School of Engineers, Mohammed V University, Morocco. With over 12 years of experience in the IT industry, he has cultivated a rich background in various domains. His research focuses on traffic road modeling and prediction, machine learning, artificial intelligence, software engineering, software process improvement, and digital transformation. For any inquiries, he can be reached via email at youness.riouali@hotmail.fr.






Ahmed Reda Maminou    is a Ph.D. student at Mohammadia Engineering School, Mohammed V University in Rabat. He earned his degree in computer science engineering in 2023. His research includes data science, machine learning, big data analytics, and deep learning. For inquiries, he can be contacted at ahmedredamaaninou@gmail.com.



Hassane Jabri    earned his degree in computer science engineering in 2023. His research encompasses data science, machine learning, big data analytics, and deep learning. For inquiries, he can be contacted at mrhassan.jabri@gmail.com.



Laila Benhlime    is a full Professor of Computer Science at the Mohammadia Engineering School of Mohammed V University, Morocco. She has worked during decades on Data Management and Knowledge Discovery. With an extensive academic background, she serves as program committee member for several conferences and journals. She is co-author of the book « L'essentiel du Génie logiciel », Pearson Education, 2010 and has more than 50 published papers in various domains including Knowledge and Data discovery, Personalization, Data Quality and Data Governance, applied to e-Gov, mobility and e-Health. For further communication, she can be reached via email at benhlime@emi.ac.ma.