International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys. 10, No.1 (Apr-2021)

http://dx.doi.org/10.12785/ijcds/100155

NMF-DCA: An Efficient Dendritic Cell Algorithm Based on Non-Negative Matrix Factorization

Mourad Belhadj[1], Foudil Cherif[2] and Mohamed Cheriet[3]

[1,2] LESIA Laboratory, University of Biskra, Biskra, Algeria
[3]ETS, University of Québec, Montréal, Canada

Received 11 Sep. 2020, Revised 18 Feb. 2021, Accepted 26 Feb. 2021, Published 21 Apr. 2021

Abstract: We discuss in this paper the challenge of enhancing the dendritic cell algorithm preprocessing phase. In short, to minimize data dimensionality we propose a new dendritic cell algorithm based on Non-negative Matrix Factorization. The aim of this method is to extract latent features from lower rank data transformation. The proposed method was divided in two steps. The first step is a factorization of the original data. Secondly, the new reduced space should be assigned to its respective signal category. Experimental findings show that the preprocessing step of the dendritic cell algorithm is significantly improved with respect to its execution and a higher accuracy rate. Our algorithm is also compared to other classification algorithms particularly MLP, SVM, and KNN. The comparison shows that the actual rate of current algorithms is outperformed by the proposed algorithm.

Keywords: Dendritic Cell Algorithm, Non-Negative Matrix Factorization, Artificial Immune System

1. INTRODUCTION

The natural immune system is one of the exciting fields because of its robust characteristics, it is an artificial organ responsible to defend organism from threats [1]. The second-generation artificial immune systems have shown its effectiveness in various applications and one such popular algorithm: the Dendritic Cell Algorithm (DCA) [2] that solves anomaly detection problems inspired by the natural vertebrate immune system. The DCA is an imitation of the natural dendritic cell and based on the controversial immune theory, the Danger model [3]. The preprocessing phase has shown its effectiveness and performance on DCA contrary to manual processing which requires the intervention of an expert, the process performed in two steps: feature selection and signal categorization [4].

The first step selects the most significant and essential features (applying one of feature selection techniques). In contrast, the second step allows categorizing each selected feature resulting from the previous step to its specific signal category (Pathogen-Associated Molecular Pattern (PAMP), Safe Signal (SS), and Danger Signal (DS)). The first work [2] requires an expert knowledge intervention to select significant and informative features and classify them in their appropriate signal according to their importance. Other works proposed automating the preprocessing phase to improve the DCA [4] without the need for expert domain knowledge.

The main objective of this work is to investigate methods for improving preprocessing runtime to keep DCA as lightweight as possible. Non-negative matrix factorization allowed a considerable improvement by reducing dimensionality space.

The remainder of this paper is organized as follows: in section 2, we present an overview of related works considering the DCA's automated preprocessing phase. Section 3 highlights an overview of the original DCA, and in section 4, we present the concept of the NMF. Section 5 contains a description of the suggested study. The Experimental evaluation performed in section 6, this section also includes results and discussion—finally, the conclusion drawn in the last section.

2. RELATED WORKS

The first versions of the DCA process manually and require expert domain users to extract knowledge, and the process becomes more challenging to manage in large dimensions and according to the applied domain. In [4] authors applied Principal Component Analysis (PCA) to automate the preprocessing phase based on its relevance

E-mail: mourad.belhadj@univ-biskra.dz, cherif.foudil@univ-biskra.dz, mohamed.cheriet@etsmtl.ca

and significance [5], to maximize the variance of selected essential features and thus reduce the noise and redundancy in data. This first step is the feature selection followed by the categorization of signals (PAMP, SS, DS), by ranking the PCA element according to their variability, compared with correlation analysis and information gain. The PCA allows automation with an acceptable accuracy rate [4]. Using PCA has the drawback of projection to the lower-dimensional space measurements from the original features.

A newer version of DCA presented in [6] to handle the DCA drawbacks and leakage of the PCA, by introducing the concept of rough set theory [7] REDUCT and CORE and present the RST-DCA [6,8]. The concept was applied to select features in the preprocessing phase of DCA by keeping the most informative features and discarding unnecessary ones using a rough set concept. The REDUCT preserves the same classification of the original data, and it assigns convenient signals by using one feature assigned to both PAMP and SS as they represent the most significant signals. The remaining features designated as DS (Danger Signals)

In the same perspective, to select informative features [6,9] proposed the RC-DCA and modify the categorization process by assigning each feature to its specific signal category. It shows in [6] that RST-DCA and RC-DCA performed unnecessary calculation and consumed time and resources to deal with this leakage, the author showed that *one-reduct* is sufficient to reduce data, as defined in original DCA, it supposed to be a lightweight algorithm. So QR-DCA dealt with this by implementing the *Quickreduct* algorithm to reduce the time to gain considerably and optimizing more resources.

The cited solutions are performed in terms of accuracy and highlight the improvement in DCA. Original DCA designed to be a lightweight algorithm. Whereas, using the rough or the fuzzy set in the preprocessing phase takes a considerable time [11,12] and affects the lightweight benefits of this algorithm. On the other hand, in real-time, data may be gradually refined, and information regarding the problem domain may actively add or remove knowledge, and the nature of data changed. And selected features may change their influence on data through time.

3. DENDRITIC CELL ALGORITHM

The dendritic cell algorithm is an abstraction of the biological dendritic cell. It is a population-based collection of dendritic cells (DC) with their sample memory collected randomly from the tissue. The DCA accepts two sorts of data, antigens, and signals, where antigens are data identifiers to be classified and signals represent data values as appeared in equation (1). These signals are categorized into three types of signals (PAMP, danger signal, and safe signal) and the inflammation signal Inflammation, which is denoted by *I*.

Figure 1 summarizes how the DCA processes provide an anomaly context. The dendritic cell obtains the previous signals from the preprocessing phase, which is crucial to transform data into one of three signals. It is composed of two steps: feature selection and signal categorization. In other terms, this phase transforms the original data into three signals related to the antigens, where the tissue contains these antigens.

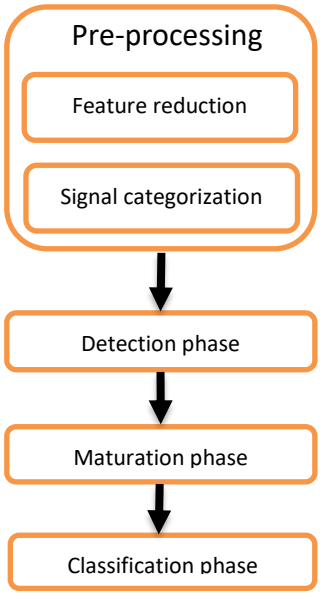

Figure 1. Dendritic cell algorithm phases

After achieving the preprocessing phase, the detection phase is the next step by combining signals with a predefined weight matrix, shown in TABLE 1, to get three outputs: Costimulatory signal (Csm), Semi-mature signals and mature signals. Every cell randomly samples and accumulates the three output values of antigens until the Csm value exceeds the migration threshold, and so preventing the long lifespan of DC. By comparing semi-mature and mature output values, we can determine if the cell goes under an abnormal state. We find that mature output is more significant than semi-mature, and otherwise, the DC is in its normal state.

$$C = \left(\frac{(W_1 * \sum_i PP_i) + (W_2 * \sum_i SS_i) + (W_3 * \sum_i DS_i)}{(W_1) + (W_2 *) + (W_3)}\right) * (1 + I) \quad \textbf{(1)}$$

The final step in this algorithm is known as the classification phase. It is from the obtained values of migrated dendritic cells and calculated using equation (1), where it determines the number of matured DCs presented in a value called: Mature Context Antigen Value (MCAV). It shows the anomaly degree of each antigen, when MCAV is close to one means antigen is possibly anomalous that

$$MCAV = \frac{matured - cells}{presented - cells} \quad (2)$$

antigen is abnormal, for this reason, we have to calculate MCAV for each antigen according to the equation (2) and compare it to the anomaly threshold, which is obtained automatically [2] from the original data and is equal to the abnormal data item divided by the total number of data or it is generally user-defined value. The last step is the classification phase which defines dangerous antigen if it is greater than the anomalous threshold and healthy otherwise.

TABLE 1.DEFAULT WEIGHT MATRIX [13].

	Csm	Mature	Semi-mature
PAMP	2	1	2
DS	0	0	3
SS	2	1	-1

4. NON-NEGATIVE MATRIX FACTORIZATION

Non-negative matrix factorization (NMF) is a new modern feature - extraction algorithm that offers a non - negative sparse and partial reproduction of the original data. This process aims to uncover latent low - dimensional factorizations within a high - dimensional data space [14]. The idea behind it is similar to known Lower Upper Decomposition [15], such as Principal Component Analysis (PCA) [16] and Vector Quantization [17] (VQ). From

Figure 2, let us take an initial data matrix $V = v_1, v_2, \dots, v_n \in R^{n\times m}$. The purpose of NMF is to find two non-negative matrices $W = w_{ik}\ \mathbb{R}^{n\times r}$. $h_{kj} \in \mathbb{R}^{r\times m}$ that approximate the original matrix V. We can model the factorization WH in terms of vectors as $v \approx Wh$ where v and h are the corresponding column vectors of V and H. The linear combination of columns of W weighted by the components of H represents the approximation of the corresponding column v, seen W as containing a basis vectors optimized of the linear approximation of V [18] (Matrix W is called basis matrix and encoding matrix H Coefficient matrix).

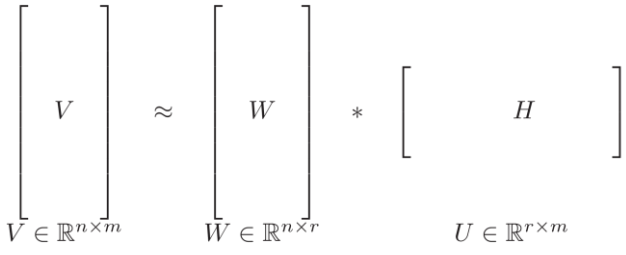

Figure 2. NMF Factorization

The NMF is a relatively new way of reducing dimensionality over data into linear combination bases W and weight H. The benefit of using NMF instead of another low-rank factorization such as Singular Value Decomposition (SVD) or Principal Component Analysis (PCA) is due to its advantages [19]. Firstly, NMF is easily interpretable. According to [18], the constraint of non-negativity produces an additive part representation, and in consequence, the factors W H are in general naturally sparse. Secondly, NMF interprets its factors remarkably. As a result of the first advantage, compared to other factorization techniques, the NMF allows saving more storage in terms of dense factors [20].

Matrix factorization is not a unique solution. So, it cannot be solved analytically in general, the solution is a numerical approximation, and it consists of solving a non-linear optimization problem in equation (3):

$$min \parallel V_{(n\times m)} - W_{(n\times r)}H_{(r\times m)} \parallel^2 / V, W, H \geq 0 \quad (3)$$

The rank r is defined by user empirically following the general rule $r < \frac{nm}{n+m}$ [21,22] and the cost function to measure the error between the original matrix V and its factors W and H as mentioned in equation (4) refers to the Frobenius norm (square Euclidean distance between two matrices) [23], with a possibility to use other alternative cost functions [18,24,25].

$$ER = \parallel V - WH \parallel^2 = \sum_{i,j}\left(x_{i,j} - \sum_{r=1}^{r} w_{ir}h_{rj}\right)^2 \quad (4)$$

To find factorization $V \approx WH$ we can use one of the iterative approaches following rules proposed in [26]. Iterations continue until the convergence to the local minimum satisfying the cost function (equation (5)).

$$W_{ir} = W_{ir}\frac{[VH^T]_{ir}}{[WHH^T]_{ir}} \quad H_{rj} = H_{rj}\frac{[W^TV]_{rj}}{[W^TWH]_{rj}} \quad (5)$$

Initially, the NMF can be slow to converge to a global minimum [27], for large scale data V as W and H are not unique. The minimization problem is non-convex, works like [22,28,29] came to overcome drawbacks and better improve this method. The NMF was applied successfully in domains [30] like hyper-spectral imaging, image processing, text mining, and other applications, including document clustering, recommendation systems, music analysis, computational biology, and environment.

5. NMF-DCA PROPOSED SOLUTION

Having reviewed related work, we now present the main body of our research. We introduce a model based on non-negative matrix factorization to automate the DCA's preprocessing phase. For this, the solution includes both steps, feature transformation, and signal categorization. On top of that, human capabilities show their limits when about a significant amount of data occur, extensive effort, and research to handle and support humans in analyzing and extracting useful and hidden information. An automatic mechanism called dimensional reduction is evident and necessary for grappling with the

curse of dimensionality and for encouraging people to reduce uncertainty.

The dimensionality reduction process obtains a smaller dimension of the original data, which reduces the storage of space with less time, increases the classification model's performance, and reduces the complexity of data.

Dimensionality reduction has shown its effectiveness in different domains. Two dominant techniques exist in dimensionality reduction: feature selection and feature extraction. The feature selection is the process of selecting a subset for most informative and relevant features from the original data. The new reduced space may better describe the original one, and this process reduces computational costs while keeping the right balance between efficiency and classification accuracy, which is highly beneficial.

On the other hand, we use feature reduction methods to find the minimum lowly correlated or uncorrelated factors and extract hidden properties of data. This technique can also better describe original data adding to the former method caption of hidden proprieties. The feature reduction is the process of transforming original data into a new transformed and reduced space composed of new value features. While reducing computational complexity in terms of time and storage space, furthermore to the discovery hidden latent structure in data, and so getting the advantage of transformed dimensionality reduction.

Additionally, to the advantages mentioned above, NMF avoids overfitting in a prediction context. This advantage is essential to reduce false positive/negative, which implies that NMF expresses better the scalability and sparsity problem improving prediction performance. The NMF was mainly used to factorize the data matrix under non-negative constraints. Consequently, it can automatically extract sparse and easily interpret factors. Compared with PCA, NMF does not remove the mean of data, which leads to non-negative values. Therefore, it can preserve more information than PCA [31].

The non-negative matrix factorization allows us to overcome runtime limitation. It improves the quality of the original data and reduces it without affecting or losing parts. Further, NMF does not need any statistical assumption on the original data but relies on the non-negative constraint, which is more convenient to represent data [18].

A. Data transformation

In many situations, when trying to model a particular phenomenon, negative values are merely insignificant. For example, a negative height for people or a negative number of visits to a website are just some of the discrete-continuous or discrete variables that are utterly absurd in terms of the final estimate.

NMF gives better data projection while keeping preserved original data structure. After performing data preparation comparing with other factorization methods because of its non-negative constraints, it decomposes the data into the product of two non-negative lower-ranking matrices W and H. The basis NMF matrix contained in the sub-matrix W and matrix H contains weights (coefficients). The NMF algorithm begins to modify W and H until achieving convergence. The algorithm guarantees that the two matrices W and H are non-negative and approach original data by mapping it to the product, considering that outliers can significantly impact the factorization. To improve the matrix factorization while benefiting from normalization, we need to decrease the error tolerance.

Initially, NMF must start with a seed (i.e., an initial value for W_0 and/or H_0) indicating the beginning for iterations because there is no global minimum and the high dimensionality of data, it is crucial to have appropriate initialization to obtain meaningful results. We initiate the seed randomly so that W and H take a uniform distribution.

The choice of the low factorization rank r is user-defined empirically, using expert knowledge, or simply using the trial and error [22]. We tested r chosen from a range starting from $2\ to\ min(m,n)$ where m is the number of features and n number of samples from the data. In each picked value, we calculate the Euclidean distance cost function between the original data and product of its factors W and H (equation (3)), we select the minimum distance value.

B. Signal categorization

The second step is to assign the transformed space into three signals to their appropriate signal category. After transforming original features using NMF reduction in the preprocessing phase, and as highlighted in the previous section, where the new data resulted from its original into a lower-rank matrix. This process reflects the importance of each resulted column to its right signal (PAMP, DS or SS), reminding the following vital points for existing signals in the DCA algorithm:

- PAMP signals: Anomalies are likely to exist when these signals are present
- Safe signals: indicate that the system is safe and no anomalies are present
- Danger signals as these signals are less important to others. Damaged cells send danger signals, their presence increases an anomalous situation but has lower potency than PAMP.

According to the amount of the danger signal DS, it is not enough to define a final context for an antigen as normal or abnormal because its presence may mean an anomaly but with lower probability as it may be classified as normal. However, the concentration of two other PAMP and SS signals can define a concluding context of antigen,

equation (1) translates the importance of each signal if we consider the default weight matrix in [2]. TABLE 1 shows that weights in the first column (costimulatory output) containing high weights are given to Safe and PAMP signals while the weight of danger signal is less important. That gives PAMP and Safe signal more essential to decide if there is an abnormality or not.

The basis matrix W is the reduced space of the original data. It represents the essential information weighted by the coefficient matrix H, considering that matrix as a hidden layer. So, the first column represents a high concentration part of the original data. In other words, the first vector is the most informative, so we assign this vector to both Safe signal and PAMP as they represent significant informative signals.

The algorithm computes the Csm threshold automatically to generate concentration for PAMP and SS, compared to each item of matrix W. If its value is less than the threshold, we attribute PAMP concentration to the corresponding value, and we attribute zero to the safe signal. If it is greater than the threshold, the Safe signal takes the value one, and PAMP takes the value zero.
The importance of the other remaining columns in the basis matrix is less critical than the first one, and we combine the remaining columns of new transformed features to form the Danger signal (DS).

6. EXPERIMENTAL EVALUATION

The main objective of this work is to show that the proposed solution, using matrix factorization in the preprocessing phase for the DCA algorithm, can optimize and improve the DCA's detection capability within less time. The experiment was carried out on binary class datasets selected from the WBC dataset [32], and other datasets with positive numerical values listed in TABLE 2. The experiment was performed on Dell Intel(R) Xeon(R) CPU E5-2670 v2 @ 2.50GHz GHz, coded in Python 3.6/64 bit.

We kept the experiment environment the same as the state-of-art works [4,10]. The experiment performed on a population of one hundred cells, and ten DCs sample the antigen vector each cycle. The maturation threshold (Csm) is generated automatically for each DC every cycle. We also used the predefined weights matrix used for signal transformation of the DCA same as defined in the state of the art (see TABLE 1)

To show the efficiency of NMF in the DCA preprocessing phase, we compared it with one of the quickest rough set algorithms: The *Quickreduct* (QR-DCA [10]). Our main objective of this study is to optimize the runtime of the preprocessing phase while increasing the classification rate of the DCA. To give more meaning to this study, we considered it necessary to compare with another method of the state of the art which was presented in [4]. The processing time using the PCA method is not applicable due to this method's nature, which is semi-automatic.

TABLE 2. DATASET DESCRIPTION

Dataset name	Ref	Instance	Attribute
Breast cancer Wisconsin (Original)	Breast 10	699	10
Breast cancer Wisconsin (Diagnostic)	Breast 32	569	32
SPECTF Heart	Spectf	267	44
Red wine quality	Red wine	1599	12
Sonar, mines vs. Rocks	Sonar	208	60
Congressional voting	Vote	435	16
Statlog (Heart)	heart	270	16
Liver disorders	Bupa	345	7

We start preprocessing data. This step needs to perform normalization and scaling functions because of the non-negative constraints of the NMF algorithm. The normalization function starts by calculating the basis matrix (new low dimension matrix), then using trial and error concept to find low-rank r, the best approximation of the distance between the original data and factorized product matrix, which is chosen by computing $min\ \|V - WH\|^2$.

The next step consists in defining for each new feature from new data its suitable signal category (PAMP, SS or DS). The presence of the signal PAMP or safe signal indicates an anomaly in the tissue, and the first column is assigned to one of the signals: PAMP or safe signal, and the other columns are combined from the danger signal.

Because of the importance of the first vector of the transformed data, it represents mostly the original data, so it is obvious to represent PAMP or SS.

Then, we use obtained new data from the former process, and we perform the multiplication process to get prepared multiple signals merged with antigen labels to be used in the maturation step. Filtering thresholds are generated automatically from the original dataset. For more accuracy, we average results (Accuracy, sensitivity, specificity, and execution time) set on more than ten runs. The performance of NMF-DCA is evaluated in terms of accuracy, sensitivity, specificity, and execution time:
$Sensitivity = TP/(TP + FN)$,
$Specificity = TN/(TN + FP)$ and
$Accuracy = (TP + TN)/(TP + TN + FN + FN)$

TABLE 3. COMPARATIVE RESULTS

	Accuracy			Sensitivity			Specificity			Time (secs)	
	NMF	QR	PCA	NMF	QR	PCA	NMF	QR	PCA	NMF	QR
Breast 10	98.92	79.80	89,95	99.50	78.61	98,24	98.88	82.26	85,6	3.06	11.62
Breast 32	98.85	90.50	97,28	98.96	93.99	94,96	99.00	74.52	98,66	4.15	44.08
Vote	96.09	93.97	89,45	98.4	98.97	72,16	94.09	89.94	99,71	1.46	3.84
Bupa	94.73	99.42	68,11	90.97	98.62	81,5	97.26	99.01	49,65	1.10	1.20
Stat heart	91.08	75.36	58,14	86.31	83.52	11,66	93.49	65.35	95,33	1.27	3.34
Red wine	89.63	79.92	84,35	92.12	74.72	58,96	87.37	90.75	99,43	2.10	22.54
Spectf	84.81	91.98	86,08	94.04	69.68	63,78	85.04	91.64	80,4	1.05	6.58
Sonar	67.31	80.23	85,06	93.67	97.94	61,79	59.12	97.30	98,87	0.69	8.56

(TP: true positive, TN: true negative, FP: false positive and FN: false negative).

To determine the effectiveness of non-negative matrix factorization on the dendritic cell algorithm as a preprocessing phase and overcome the limitation of runtime in former works. We have made a matrix factorization on binary class datasets (class 1, followed by class 2). This transformation was based on non-negative NMF matrix factorization and used the new reduced data as an input signal (Safe signal, PAMP, and danger signal) to the DCA.

For the first dataset, breast10 from results in TABLE **3**, it shows a high rate of accuracy with more than 98% in 3 seconds. The sensitivity and specificity are more than 98%. Matching to QRDCA, the accuracy was about 79,80% less than the NMF in terms of accuracy, processed in 11,62 seconds to complete, remarkably the PCA outperformed QRDCA accuracy with more than 89%. The accuracy of the breast32 dataset exceeds 98% with 97.68% sensitivity and 99% specificity in a total time of 4.15 seconds, while QRDCA took over 44 seconds with 90,50% accuracy against 97,28% PCA.

Vote dataset also gives a consistent result with 96.06% accuracy and 98.40% sensitivity. At the same time, the specificity decreased. We also noticed that it took 1.46 seconds to achieve the whole process, comparing the QR that made it in 3.84 seconds and QRDCA accuracy exceeded PCA accuracy with 93,97% vs 89.45%. Unlike the Bupa dataset, the QRDCA accuracy outperformed, both algorithms (the NMFDCA and PCA), with 99.42%, 98.62% sensitivity, and 99;01% specificity in 1.20 seconds, while the NMFDCA accuracy was less than the former with 94.73% bypassing the QRDCA in 1.10 seconds, PCA accuracy was about 68.11%.

The Redwine dataset's accuracy was about 89.63%, 92.12% sensitivity, and 87.37% specificity processed in 2.10 seconds. Moreover, accuracy was about 79.92% for QRDCA achieved in 22.54 seconds, PCA Accuracy was better than QRDCA with 84,35% and specificity with 99,43% but does not performed in sensitivity with 58,96%. Remarkably, the Spectf dataset rates using QRDCA overpasses NMFDCA and PCA with an accuracy rate of 91.98% against 84.81, and 85.04% against 91.64% specificity for NFMDCA, and 86,08% accuracy, 63,78% sensitivity, and 80% specificity for PCA. However, NMFDCA sensitivity was more important with 94.04% unlike QRDCA with 69.68%.

In the first dataset, the excellent accuracy of breast10 with 98,54% translated by the high quality of factorization, where it gives a very close representation of the original data, which means that the error is insignificant. Compared with the rough set [10], the feature selection process using the rough set took more than 11.62 seconds to select relevant features with an accuracy rate of 79.80%. In comparison, NMF took only 3.06 seconds to achieve the whole process. In this case, the process of feature selection was not able to categorize selected features to the appropriate signal, especially the safe signal and PAMP most significant signal and crucial to final results.

Because QRDCA selects the first feature issued by the QR feature selection algorithm as the safe signal and the second as PAMP, it does not mean that we cannot affect other features to PAMP or safe signal, so, the process of feature selection was not enough to categorize selected features. Moreover, PCA outperformed QRDCA and got better accuracy rate the process using PCA[4] based on biplot feature directions to categorize them, this process is relatively semi-automatic, so we cannot measure the runtime.

The breast32 took the same way as the first one with a reasonable accuracy rate but took more time than one second compared with the first one because of the feature number. In contrast, it took more than 44.08 seconds to get

feature selection over the QR process, which is a significant improvement to NMFDCA. This dataset shows increased incredibly with more than 22 seconds, which slows down the DCA, and derives from its main character

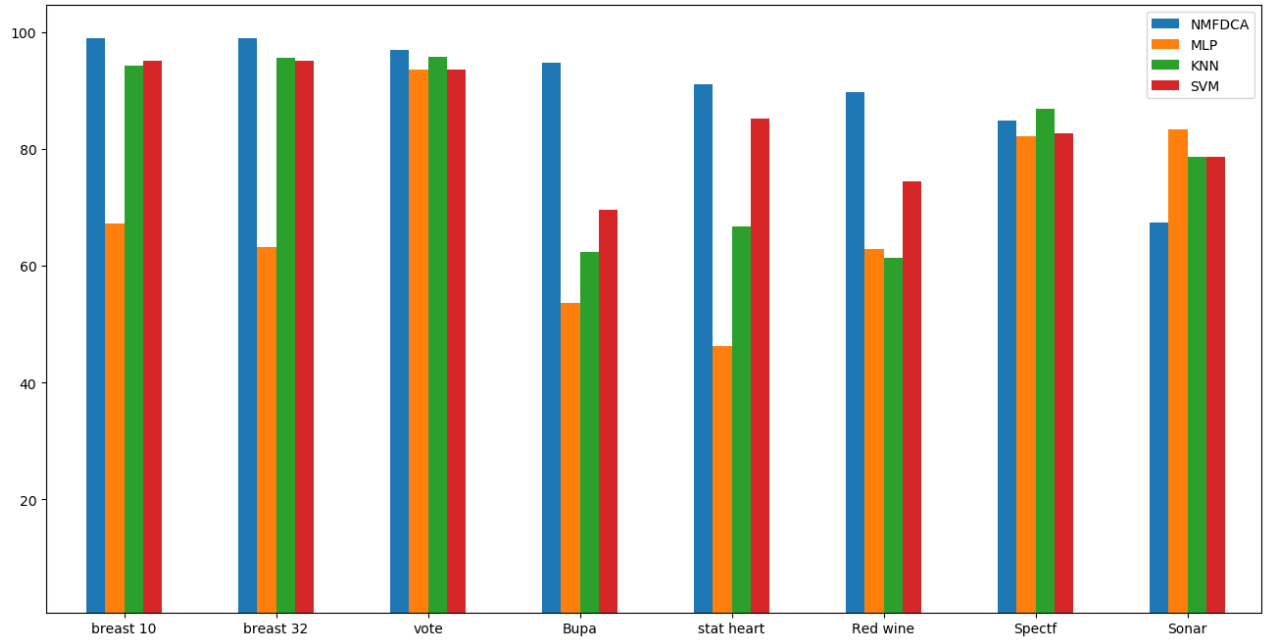

Figure 3. Comparison with SVM, KNN, and MLP classifiers

a significant difference in processing time, considering the number of features and also in terms of accuracy with lower than the other data, due to the quality of the matrix factorization. However, keeping a better result than other classifiers.

98.92% against 90.50%. The NMF factorization gives an almost identical image to the original data with reduced space so that the categorization process will be affected by the quality of factorization. Comparing with PCA, we found that the result was very close to NMFDCA with a better assignment of attributes to their categories, unlike QRDCA, which failed to categorize them properly.

As to the red wine with 1599 instances and 12 features, the accuracy was about 89.63% in 2.10 seconds. The number of instances was considerably more significant than the previous datasets. Over and above, less quality of factorization was observed in results. The reason is due to the ill-posed problem. In consequence, the factorization error affected the quality of the categorization process. It decreased the accuracy rate compared to the cancer breast datasets, which are considered well-posed problems with a good quality of factorization and optimized error.

Notice that the NMF factorization is not a unique solution, and it suffers from being highly ill-posed [22], the accuracy can drastically decrease in the case of an ill-posed problem. To get better accuracy, we have to transform it into a well-posed problem and also have to minimize $\|V - WH\|^2$.

However, in terms of execution time, when applying the feature selection process using rough set method time as a lightweight algorithm. In some cases, the QRDCA can give a better accuracy rate comparing to NMFDCA according to TABLE **3**, it shows that the feature selection process applying to Spectf and Sonar datasets using rough set can be efficient to categorize selected features to its appropriate category, but took a considerable time to achieve the selection process. Whereas the NMF factorization gave a less accuracy rate to the cited two datasets, we observe that preprocessing time was clearly in less time. Thus, the use of non-negative matrix factorization allows keeping the lightweight character of the DCA. We can improve the NMF efficiency to preprocess DCA when we properly handle the ill-posed problems. When applying PCA to DCA, the accuracy can be better than QRDCA in most cases. Still, this method suffers from its difficulty as it is based on observation and projection of features on the biplot.

Besides comparison to the QR-DCA, our experiments confirm the improvement of running time while keeping a reasonable accuracy rate. To give more objectivity to our study, we compared our method with three classifiers: multi-layer perceptron (MLP), K-nearest neighbors (KNN), and support vector machine (SVM). We used 5-folds cross-validation to test classifiers.

As shown in Figure **3**, the accuracy rate of the breast 10 using MLP was about 67.14%, 94.28% using KNN, and SVM was 95%. Whereas NMF was about 98.92%, this result is interpreted by the good factorization quality. The weak accuracy of the MLP may result from many points like the number of neurons, the number of hidden layers, the weight initialization, or the activation function. In

return, KNN and SVM gave an acceptable accuracy rate. The results of the breast 32 dataset were approximately the same as the first in terms of accuracy rate with the observation that the MLP still registers a low accuracy rate.

By testing the classifiers on the vote dataset, we observe that MLP wins over SVM and approaches the results of KNN. The NMF accuracy rate was about 96% higher than other classifiers. The accuracy rate of the Bupa dataset was Stat heart dataset with an accuracy rate of 91.08% NMF, 46.29% MLP, 66.66% KNN, and 86.18% SVM, which is reflected by the fact that the factorization error is minimized, the good quality of the factorization and therefore very close to the original data. Whereas with other classifiers, the accuracy rate was lower.

We are aware that our research may have limitations. When the data are an ill-posed problem for factorization, the error $\|V - WH\|^2$ increases and gives lowered rates, where we observe that the classifiers exceed our method: Spectf and Sonar datasets. To remedy this problem, we need to handle and regularize the ill-posed data.

7. CONCLUSION

The proposed algorithm uses the NMF method to convert the original data in space reduction using matrix factorization and categorizes the new data into its corresponding signal. This process optimizes the preprocessing phase and retains the dendritic cell algorithm efficiency. The optimized new transformed NMF-based data enables us to obtain information which will help us achieve a high accuracy. In comparison with other state-of-the-art approaches, experimental evaluations verify the suggested approach's efficiency comparing to QR-DCA and DCA-PCA in terms of runtime and accuracy rate. It has been shown that the new approach outperforms other classifiers (MLP, SVM, and KNN). In perspective, a comparative analysis between the dendritic cell algorithm and the perceptron algorithm would continue to be studied in depth to understand the learning process in both algorithms.

REFERENCES

[1] J. Brownlee, *Clever Algorithms Nature-Inspired Programming Recipes Ii*, 2011.

[2] J. Greensmith, U. Aickelin and S. Cayzer, Introducing dendritic cells as a novel immune-inspired algorithm for anomaly detection, in International Conference on Artificial Immune Systems, 2005, pp. 153–167.

[3] P. Matzinger, *Tolerance, danger, and the extended family*, Annu. Rev. Immunol. 12 (1994), pp. 991–1045.

[4] F. Gu, J. Greensmith, R. Oates and U. Aickelin, *Pca 4 dca: The application of principal component analysis to the dendritic cell algorithm*, arXiv Prepr. arXiv1004.3460 (2010), .

[5] F. Song, Z. Guo, D.M.-2010 international conference on and undefined 2010, *Feature selection using principal component analysis*, ieeexplore.ieee.org .

[6] Z. Chelly and Z. Elouedi, *A survey of the dendritic cell algorithm*, Knowl. Inf. Syst. 48 (2016), pp. 505–535.

[7] S. Akama, T. Murai and Y. Kudo, *Rough Set Theory*, in *Reasoning with Rough Sets: Logical Approaches to Granularity-Based Framework*, Springer International Publishing, Cham, 2018, pp. 7–50.

[8] Z. Chelly and Z. Elouedi, RST-DCA: a dendritic cell algorithm based on rough set theory, in International Conference on Neural Information Processing, 2012, pp. 480–487.

[9] Z. Chelly and Z. Elouedi, Rc-dca: A new feature selection and signal categorization technique for the dendritic cell algorithm based on rough set theory, in International Conference on Artificial Immune Systems, 2012, pp. 152–165.

[10] Z. Chelly and Z. Elouedi, Qr-dca: A new rough data pre-processing approach for the dendritic cell algorithm, in International Conference on Adaptive and Natural Computing Algorithms, 2013, pp. 140–150.

[11] M. Kopczyński and J. Stepaniuk, *Hardware Implementations of Rough Set Methods in Programmable Logic Devices*, in *Rough Sets and Intelligent Systems - Professor Zdzisław Pawlak in Memoriam: Volume 2*, Springer Berlin Heidelberg, Berlin, Heidelberg, 2013, pp. 309–321.

[12] Z. Chelly and Z. Elouedi, A fuzzy-rough data pre-processing approach for the dendritic cell classifier, in European Conference on Symbolic and Quantitative Approaches to Reasoning and Uncertainty, 2013, pp. 109–120.

[13] J. Greensmith, *The dendritic cell algorithm*, Citeseer, 2007.

[14] M.W. Berry, M. Browne, A.N. Langville, V.P. Pauca and R.J. Plemmons, *Algorithms and applications for approximate nonnegative matrix factorization*, Comput. Stat. Data Anal. 52 (2007), pp. 155–173.

[15] N. Gillis, L.T.K. Hien, V. Leplat and V.Y.F. Tan, *Distributionally Robust and Multi-Objective Nonnegative Matrix Factorization*, (2019), .

[16] S. Wold, K. Esbensen and P. Geladi, *Principal component analysis*, Chemom. Intell. Lab. Syst. 2 (1987), pp. 37–52.

[17] R. Gray, *Vector quantization*, IEEE Assp Mag. 1 (1984), pp. 4–29.

[18] D.D. Lee and H.S. Seung, *Learning the parts of objects by non-negative matrix factorization*, Nature 401 (1999), pp. 788–791.

[19] C.D. Meyer, *Matrix Analysis and Applied Linear Algebra*, Vol. 71, Siam, 2000.

[20] R. Albright, J. Cox, D. Duling, A.N. Langville and C. Meyer, *Algorithms, initializations, and convergence for the nonnegative matrix factorization*, 2006.

[21] O.G. Okun, Non-negative matrix factorization and classifiers: experimental study, in Proceedings of the 4th IASTED International Conference on Visualization, Imaging, and Image Processing (VIIP'04), 2004, pp. 550–555.

[22] N. Gillis, *Sparse and unique nonnegative matrix factorization through data preprocessing*, J. Mach. Learn. Res. 13 (2012), pp. 3349–3386.

[23] P. Paatero and U. Tapper, *Positive matrix factorization: A non-negative factor model with optimal utilization of error estimates of data values*, Environmetrics 5 (1994), pp. 111–126.

[24] S. Sra and I.S. Dhillon, Generalized nonnegative matrix approximations with Bregman divergences, in Advances in neural information processing systems, 2005, pp. 283–290.

[25] P. Paatero, *Least squares formulation of robust non-negative factor analysis*, Chemom. Intell. Lab. Syst. 37 (1997), pp. 23–35.

[26] D.D. Lee and H.S. Seung, *Algorithms for Non-negative Matrix Factorization*, in *Advances in Neural Information Processing Systems 13*, T.K. Leen, T.G. Dietterich and V. Tresp, eds., MIT Press, 2001, pp. 556–562.

[27] C. Boutsidis and E. Gallopoulos, *SVD based initialization: A head start for nonnegative matrix factorization*, Pattern Recognit. 41 (2008), pp. 1350–1362.

[28] S. Nakano, K. Yamamoto and S. Nakagawa, Fast NMF based approach and improved VQ based approach for speech recognition from mixed sound, in (APSIPA ASC), 2012 Asia-Pacific, 2012, pp. 1–4.

[29] J. Kim and H. Park, *Fast nonnegative matrix factorization: An active-set-like method and comparisons*, SIAM J. Sci. Comput. 33 (2011), pp. 3261–3281.

[30] Y.-X. Wang and Y.-J. Zhang, *Nonnegative matrix factorization: A comprehensive review*, IEEE Trans. Knowl. Data Eng. 25 (2013), pp. 1336–1353.

[31] B. Ren, L. Pueyo, G. Ben Zhu, J. Debes and G. Duchêne, *Non-negative Matrix Factorization: Robust Extraction of Extended Structures*, Astrophys. J. 852 (2018), pp. 104.

[32] *{UCI} Machine Learning Repository, URL: http://archive.ics.uci.edu/ml*. 2017.

Mourad Belhadj has 11 years of experience in academics, teaching undergraduate computer science students at the University of Ouargla. Currently, a Ph.D. student at the University of Biskra, research scholar in Computer Science in area of the artificial immune system and machine learning.

Professor Foudil Cherif is a full professor at the University of Biskra, Algeria. He received his Ph.D. degree in computer science from the University of Biskra, Algeria, in 2006. He supervised several Ph.D. and Magister these which have been successfully defended these last years. His current research interest is in Crowd simulation, artificial life, Artificial intelligence, RFID security, software engineering, and formal verification of security protocols.

Professor Mohemed Cheriet received the B.Sc. CE (Bab Ezzouar University, Algiers), DEA, and Doctorate of the University of Paris 6 (Paris 6, France).

As a scientist and educator, Prof. Cheriet has taken an active role in publishing technical papers and authoring books. He has published 70 international journal papers and 135 international conference papers and has delivered 17 invited talks.