Usage of Particle Swarm Optimization to Improve the Performance of Supervised Classifiers

Bharanidharan N\textsuperscript{1}, Nivetha M\textsuperscript{2}, Balakrishna A\textsuperscript{3}, and Dhanaraju P\textsuperscript{4}
\textsuperscript{1,2,3,4} Department of ECE, VeltechRangarajanDr.Sagunthala R & D Institute of Science and Technology, Chennai

\textsuperscript{1}bharani2410@gmail.com, \textsuperscript{2}nivinivetha1525@gmail.com, \textsuperscript{3}arangibalakrishna87@gmail.com, \textsuperscript{4}dhanarajpalli123@gmail.com

Abstract. Representing the data appropriately will have a significant effect on the outcome produced by the classifier. Transforming the feature will help to represent the data points in a more suitable way for the classifier. Particle swarm optimization belongs to the Swarm Optimization techniques category and it is generally used for solving numerical optimization problems, weight updation in neural networks, and feature selection. This research work proposes a Particle swarm optimization-based transformation technique for increasing the classification metrics of popular classification algorithms namely K-Nearest Neighbor, Stochastic Gradient Descent Classifier, Decision trees, and linear discriminant analysis classifier. Experiments are conducted using the SONAR dataset and the highest accuracy of 82.69\% is attained for Stochastic Gradient Descent classifier when Particle swarm optimization is used as a transformation technique

Keywords- Particle swarm optimization; Binary classification; Swarm Intelligence; Machine learning; SONAR dataset

INTRODUCTION

There is a need for classification tasks in a variety of fields. Based on the past observations, the categorical class labels of new instances will be predicted through a supervised classification task. To choose the most appropriate algorithm to solve the problem it is always recommended to compare and try various algorithms’ performances. Adequate classification performance cannot be achieved by many classification/learning algorithms due to the poor quality data space representation. When facing difficult problems, the solution for this deficiency is the usage data transformation technique. To identify the implicit correlation of data points, new data points will be produced for enhancing the suitability of data points for the classification task. The constructed feature usually acts as function of the original low-level features [1-2].

Nowadays evolutionary techniques are widely used in various fields [3-4]. Swarm Intelligence (SI) algorithms belong to a sub-category of evolutionary techniques and they are based on the behavior of a group of animals [5]. Some of the SI algorithms include Particle Swarm Optimization (PSO), Dragonfly algorithm, Bat algorithm, Elephant herding optimization, etc. PSO is easier to implement, less expensive, can converge more quickly, and having very few parameters comparatively. In many complex problems, PSO has been successfully implemented as a powerful search technique. In this research work, PSO will be tested as a transformation technique on four different classifiers namely, K-Nearest Neighbor (KNN) classifier, Stochastic Gradient Decent (SGD) Classifier, Decision Trees (DT), and Linear Discriminant Analysis (LDA) classifier.
1.1 Particle Swarm Optimization

PSO technique tries to mimic the flying mechanism of bird flocks [6-7]. PSO is one of the iterative optimization algorithms and the velocity & position of birds (particles) are updated using equations (1) & (2).

\[ v_i(t+\tau) = \omega \cdot v_i(t) + c_1 \cdot r_1 \cdot (p_i(t) - x_i(t)) + c_2 \cdot r_2 \cdot (gbest - x_i(t)) \]  

\[ x_i(t+\tau) = x_i(t) + v_i(t) \]  

Here \( v_i \) and \( x_i \) denote the velocity & position of the \( i^{th} \) particle respectively; \( t \) represents the current iteration; \( p_i(t) \) specifies the individual finest position of the \( i^{th} \) particle; \( gbest \) signifies the over-all best
position (best position of all the particles); $w$, $c_1$, and $c_2$ denote the control parameters; $r_1$ and $r_2$ denote the random number in the range $[0,1]$.

### 1.2 KNN Classifier

KNN classifier [8] relies on a simple classification methodology based upon the distance and neighbors. The main advantage of the KNN classifier is it won’t make a prior assumption of data. But the prediction time is quite high as it has to find the distance between every data point. Different distance metrics like Euclidean distance, Chebyshev distance, Manhattan distance, etc. can be used in the KNN classifier. For example, Euclidean distance between two points $p$ and $q$ are found using the equation,

$$D(p, q) = \sqrt{(p1 - q1)^2 + (p2 - q2)^2 + \cdots + (pn - qn)^2}$$

### 1.3 SGD Classifier

Basically SGD [9] is an optimization function that tries to minimize the cost function by tuning the values of control parameters. It can be applied to various machine learning tasks.

### 1.4 DT Classifier

The decision tree [10] tries to build a tree architecture for modeling the classification or regression tasks. It will decompose the data points into sub-groups in an appropriate manner using a set of procedures.

### 1.5 LDA Classifier

LDA is a supervised classification method that is used to create machine learning models [11]. These models based on dimensionality reduction are used in the application, such as marketing predictive analysis and image recognition, amongst others. LDA is a simple and well-understood technique that is commonly used in classification ML models.

### 2 IMPLEMENTATION

Figure 2. represents the flowchart that depicts the methodology used in this research work. The features from the SONAR dataset will be directly given to PSO which acts as a transformation technique. The transformed data points from PSO’s output are given to any one of the four supervised classification techniques namely, KNN, SGD, DT, and LDA classifier. Finally, the performance will be assessed using five popular performance metrics namely Accuracy, Error Rate, Mathews Correlation Coefficient (MCC), Precision, Recall, and F1 score.
In general, if PSO is used for feature selection and solving an optimization problem, the particles will be initialized randomly. But to act as a transformation technique, the particles are initialized with features from the SONAR dataset. Fitness function is computed as the reciprocal of Euclidean distance from target to the current particle’s position. Depending upon the fitness values, global best and personal best is computed and then equations (1 & 2) are used to update the position of particles in each iteration.

The ideal values for the maximum number of iterations, target, three control parameters ($w$, $c_1$, and $c_2$) are found using the trial and error method. For instance, Figure 3 depicts the process of finding the ideal values for PSO control parameter $w$.

![Flowchart depicting the methodology used in this research work](chart.png)

**Figure 2.** Flowchart depicting the methodology used in this research work

![Computing optimized values for $w$ (control parameter of PSO)](graph.png)

**Figure 3.** Computing optimized values for $w$ (control parameter of PSO)
3. RESULT AND DISCUSSION

The performance metrics of four different supervised classifiers with and without PSO as transformation technique for the classification of test SONAR dataset are shown in Table 1. MCC can be considered as an efficient performance metric that gives an overall picture about the classification [10] and meaningful insights can be derived through MCC comparison for various classifiers and transforms as shown in Figure 4.

\begin{table}[h]
\centering
\begin{tabular}{|l|c|c|c|c|c|}
\hline
 & Accuracy (%) & Error rate (%) & MCC (%) & Precision (%) & Recall (%) & F1 score (%) \\
\hline
KNN & 75.00 & 25.00 & 49.59 & 74.19 & 82.14 & 77.96 \\
PSO-KNN & 80.76 & 19.23 & 62.37 & 76.47 & 92.85 & 83.87 \\
SGD & 76.92 & 23.07 & 53.57 & 78.57 & 78.57 & 78.57 \\
PSO-SGD & 82.69 & 17.31 & 65.94 & 78.79 & 92.86 & 85.25 \\
DT & 73.07 & 26.92 & 45.74 & 71.87 & 82.14 & 76.66 \\
PSO-DT & 76.92 & 23.08 & 55.29 & 72.22 & 92.86 & 81.25 \\
LDA & 78.85 & 21.15 & 57.93 & 75.76 & 89.29 & 81.97 \\
PSO-LDA & 80.77 & 19.23 & 62.37 & 76.47 & 92.86 & 83.87 \\
\hline
\end{tabular}
\caption{Performance metrics of various supervised classifiers with and without PSO as transformation technique}
\end{table}

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{mcc_comparison}
\caption{MCC comparison of various supervised binary classifiers with and without PSO as transformation technique}
\end{figure}

From Table 1, it can be witnessed that the accuracy of four different supervised classifiers namely KNN, SGD, DT, and LDA has accuracy in the range of 73-79% when no transformation techniques are used. If PSO is used as a transformation technique, the accuracy of all the four supervised algorithms is getting increased and the highest accuracy of 82.69% is attained for the PSO-SGD.
classifier. Not only accuracy, but all the performance metrics are also moving towards the ideal value when PSO is used as a transformation technique for the above mentioned four classifiers. From Figure 4, the increase in MCC due to PSO transform can be witnessed. Remarkably around a 26% increase in MCC can be observed for PSO-KNN classifier over KNN classifier.

4. CONCLUSION

Performance enhancement attained by four popular supervised classification algorithms using PSO as a transformation technique is analyzed in this research work. It is observed that PSO provides good performance improvement for all the four classifiers tested but the level of improvement varies with the classifier. For instance, a 26% increase in MCC is witnessed in KNN due to PSO usage and only an 8% increase in MCC is witnessed in LDA due to the usage of PSO. In the future, the efficiency of PSO as a transformation technique has to be studied for other datasets and other applications.

References

[1] Vafaie, H., DeJong, K., : Feature space transformation using genetic algorithms. IEEE Intelligent Systems, 13(2):57-65, (1998)
[2] Neshatian, K., Zhang, M., Johnston, M.: Feature construction and dimension reduction using genetic programming. Springer Lecture Notes in Computer Science, 4830, 160-170 (2007)
[3] Engelbrecht, A.P.: Computational intelligence: an introduction. Wiley (2007)
[4] Geem, Z., Kim, J., Loganathan, G.V.: A New Heuristic Optimization Algorithm. Harmony Search Simulation. 76, 60-68 (2001)
[5] Kennedy, J., Eberhart, R.: Swarm Intelligence. Morgan Kaufmann Publishers, San Francisco (2001)
[6] Kennedy, J., Eberhart, R.: Particle Swarm Optimization. IEEE International Conference on Neural Networks, Australia , 4, 1942-1948 (1995)
[7] Guo, Y., Li, W., Mileham, A., Owen, G.: Applications of particle swarm optimization in integrated process planning and scheduling. Robotics and Computer-Integrated Manufacturing, 25(2), 280-288 (2009)
[8] E. Ramprasath, P. Manojkumar, P. Veena “Analysis of Direct Current Motor in LabVIEW”, World Academy of Science, Engineering and Technology, 2015.
[9] Fasihul Kabir, Sabbir Siddique, Mohammed Rokibul Alam Kotwal, Mohammad Nurul Huda: Bangla text document categorization using Stochastic Gradient Descent (SGD) classifier. International Conference on Cognitive Computing and Information Processing(CCIP), Noida, India, DOI: 10.1109/CCIP.2015.7100687 (2015)
[10] Harsh Patel, Purvi Prajapati: Study and Analysis of Decision Tree Based Classification Algorithms. International Journal Of Computer Sciences And Engineering 6(10), 74-78 (2018)

[11] Alaa Tharwat, Tarek Gaber, Abdelhameed Ibrahim, Aboul Ella Hassanien: Linear discriminant analysis: A detailed tutorial. AI Communications 30(2), 169-190 (2017)