International Journal of Engineering Sciences Paradigms and Researches (IJESPR)
(Vol. 43, Issue 01) and (Publishing Month: June 2017)
An Indexed and Referred Journal with Impact Factor: 2.80
ISSN (Online): 2319-6564
www.ijesonline.com

Unification of Zero Shot Learning and Supervised Learning

Manju[1] and Dr. Ankit[2]

[1]**Research Scholar, Department of Computer Science, Baba Mastnath University, Rohtak, Haryana (India)**
dr.manjumalik21@gmail.com

[2]**Assistant Professor, Department of Computer Science, Baba Mastnath University, Rohtak, Haryana (India)**
ankit524.in@gmail.com

Publishing Date: June 24, 2017

Abstract

Learning-based techniques for perceiving objects in regular pictures have gained vast ground throughout the most recent years. For unmistakable article classes especially in countenances and vehicles, there is dependable and productive finders are accessible which depends on the mix of capable low-level elements. This work has been finished by utilizing the swarm insight with SVM classifier. The Jaya calculation which is a parameter less calculation has been utilized to decide the important elements which are ordered by the SVM classifier. The improvement of the precision and other parameter is seen in the outcomes.

Keywords: *Swarm Intelligence, Jaya Algorithm, SVM Classifier, DAP, Accuracy.*

1. Introduction

Swarm Intelligence (SI) is an imaginative appropriated clever worldview for taking care of enhancement issues that initially took its motivation from the natural case by swarming, rushing and grouping wonders in vertebrates **[1]**. Swarm Intelligence additionally a computerized reasoning (AI) discipline, which is worried with the configuration of smart multi-specialist frameworks by taking motivation from the aggregate conduct of social creepy crawlies, for example, ants, termites, honey bees, and wasps, and from other creature social orders, for example, groups of winged animals or schools of fish.

2. Swarm Intelligence (Si) Models

Swarm knowledge models are alluded to as computational models enlivened by characteristic swarm frameworks. To date, a few swarm knowledge models in view of various common swarm frameworks have been proposed in the writing, and effectively connected in some genuine applications. Case of swarm knowledge models are: Particle Swarm Optimization **[2]**, Artificial Bee Colony **[3]**, Ant Colony Optimization **[4]**, Bacterial Foraging **[5]**, Cat Swarm Optimization **[6]**, Artificial Immune System **[7]** and Glowworm Swarm Optimization **[8]**.

3. JAYA Algorithm

Swarm Intelligence based algorithms are probabilistic algorithms and need controlling parameters for performance optimization like population size, number of generation etc. Some algorithm specific controlling parameters are also needed like crossover and mutation probability in genetic algorithm, inertia weight in the particle swarm optimization. The improper tuning of such parameters leads to performance degradation of the algorithm. The performance of various swarm intelligence based algorithms like particle swarm optimization, ant colony optimization, artificial bee colony, genetic algorithm, differential evolution etc. is controlled by these controlled by these controlling parameters. Recent development results in few parameters-less algorithms like teacher learning based algorithm (TLBO), jaya algorithm which needs only common parameters (no algorithm specific parameter needed). Jaya algorithm is simple as compared to the TLBO as it needs only one phase to complete its processing. This work will use the jaya algorithm described below.

Suppose O(p) is an objective function to be optimized (minimized or maximized). At any particular moment (say a^{th} iteration), s is the population size with each population consisting of m members. The population member obtaining the best value of O(p) say $O(p)_{best}$ is the best member while the member having lowest value is the worst member say $O(p)_{worst}$. Then,

$$p^m_{b,c,a} = p_{b,c,a} + r_{1,b,a}\left(p_{b,best,a} - |p_{b,c,a}|\right) - r_{2,b,a}\left(p_{b,worst,a} - |p_{b,c,a}|\right)$$

Where $p_{b,c,a}$ and $p^m_{b,c,a}$ are the original and modified value of b^{th} member of c^{th} population at a^{th} iteration respectively. $r_{1,b,a}$ and $r_{2,b,a}$ are variables having random value between 0 and 1. $p_{b,best,a}$ and $p_{b,worst,a}$ are best and worst b^{th} member at a^{th} iteration. The $r_{1,b,a}\left(p_{b,best,a} - |p_{b,c,a}|\right)$ coefficient of equation (1) moves the value of $p_{b,c,a}$ towards best member while the coefficient $-r_{2,b,a}\left(p_{b,worst,a} - |p_{b,c,a}|\right)$ moves the value of $p_{b,c,a}$ away from the worst value. The value of $p^m_{b,c,a}$ is accepted only if $O(p^m_{b,c,a}) > O(p_{b,c,a})$ for maximization problem and $O(p^m_{b,c,a}) < O(p_{b,c,a})$ for minimization problem. This process is applied to whole population to generate better solution until stopping criteria achieved. The procedure of jaya algorithm is given as:

1. Initiate population size(s), each population member(m), stopping criteria
2. Identify the best and worst population member say best and worst.
3. Modify current solution
$$p^m_{b,c,a} = p_{b,c,a} + r_{1,b,a}\left(p_{b,best,a} - |p_{b,c,a}|\right) - r_{2,b,a}\left(p_{b,worst,a} - |p_{b,c,a}|\right)$$
4. if $O(p^m_{b,c,a}) < O(p_{b,c,a})$ for minimization problem and $O(p^m_{b,c,a}) > O(p_{b,c,a})$ for maximization problem then
$$p_{b,c,a} = p^m_{b,c,a}$$
endif
5. if stopping criteria achieved then
    Optimized solution found
else
    Go to step 2.
endif

The above described procedure can be used to optimize any objective function.

4. SVM Classifier

The main focus of SVM classifier to determine the optimized solution by separating hyperplane can be given as $f(v_i) = w * \emptyset(v_i) + bias$, here w is weights, bias is the optimal bias, and ϕ is the nonlinear mapping applied to input vectors v.

The optimization is done by minimizing the w which results in maximized distance between the closest point of hyperplane and the hyperplane. It can be understood as:

$$\min\left(\emptyset(w)\right) = \frac{1}{2} * \|w\|^2 + c\sum_{i=1}^{P_n} e_i$$

Where c is the constant used for regularization and e is the normalized variation where e_i>=0 &$o(w * \emptyset(v_i) + bias) \geq 1 - e$ and i=1…P_n.
Applying Lagrangian method

$$\max L_1(a) = \sum_{i=1}^{P_n} a_i - \frac{1}{2}\sum_{i,j=1}^{P_n} a_i a_j o_i o_j \left(\emptyset(v_i) * \emptyset(v_j)\right)$$

Such that
$\sum_{i=1}^{P_n} a_i o_i = 0$ and $c \geq a_i \geq 0$ for i=1…P_n.
where a is the Lagrangian multiplier.

On solving the equation the classification can be given as:

$$L_e = \begin{cases} 0 \; if|o_i - f(v_i)| \leq e \\ |o_i - f(v_i) - e, |if|o_i - f(v_i)| > e \end{cases}$$

where e is the maximum allowed error.

Similarly the output of the ANN is given as$f(v_i) = \emptyset(w * v_i + bias)$. The minimum error allowed can be determined by the procedure explained above. This error is used as stopping criteria to perform the classification using ANN.

5. Supervised Learning Unified with Zero Hot Learning Algorithm

The process of the proposed work can be easily understood as follow:

1. Input object attribute say $a_y = (a_1^y, \dots \dots . a_m^y)$ be a vector of binary associations$a_m^y \epsilon \{0,1\}$.
2. Identify the best and worst population member say best and worst.
3. Modify current solution
$$p_{b,c,a}^m = p_{b,c,a} + r_{1,b,a}(p_{b,best,a} - |p_{b,c,a}|) - r_{2,b,a}(p_{b,worst,a} - |p_{b,c,a}|)$$
4. if $O(p_{b,c,a}^m) < O(p_{b,c,a})$ for minimization problem and $O(p_{b,c,a}^m) > O(p_{b,c,a})$for maximization problem then
$$p_{b,c,a} = p_{b,c,a}^m$$
endif
5. if stopping criteria achieved then
Optimized solution found
else
Go to step 2.
endif
6. new_Attribute_set=Step 2 to 5 determine the sufficient attributes for the processing
7. Divide new_Attribute_set into a cluster of attributes as discussed in section 3.1 using
$$p(z|x) \propto \prod_{m=1}^{M} \left(\frac{p(a_m|x)}{p(a_m)}\right) a_m^z$$
8. Classify the object on the basis of the attributes using SVM classifier as discussed in section 3.3 using
$$L_e = \begin{cases} 0 \; if |o_i - f(v_i)| \le e \\ |o_i - f(v_i) - e, | if |o_i - f(v_i)| > e \end{cases}$$

The above step using to classify the object on the basis of attributes. The implementation and analysis of the work is done in next section.

6. Simulation and Result Analysis

The test system utilized as a part of this work is MATLAB 7.0. The reproduction utilizes the MATLAB. Another m document is made to play out the coding of proposed calculation. This code utilizes the picture tool compartment of the MATLAB as capacities like imread are utilized to peruse the picture and these capacities are accessible because of picture tool compartment. The reproduction additionally needs neural system tool compartment of the MATLAB to utilize capacities like train to prepare the system. The code comparing to the proposed calculation is produced inside a m document and this m record is executed to get the outcomes.

a. Dataset Used

Besides the Animals with Attributes dataset we also perform experiments on two other datasets of natural images for which attribute annotations have been released. We briefly summarize their characteristics here. The aPascal-aYahoo dataset6 was introduced by Farhadi *et al.*. It consists of a 12,695 image subset of the PASCAL VOC 2008 dataset7 and 2644 images that were collected using the Yahoo image search engine. The PASCAL part serves a training data, and the Yahoo part as test data. Both sets have disjoint classes (20 classes for PASCAL, 12 for Yahoo), so learning with disjoint training and test classes is unavoidable. Attribute annotation is available on the image level: each image has been annotated with 64 binary attribute that characterize shape, material and the presence of important parts of the visible object. As image representation we rely on the precomputed color, texture, edge orientation and HoG features that the authors of extracted from the objects' bounding boxes (as provided by the PASCAL VOC annotation) and released as part of the dataset.

b. Performance Evaluation Parameters

The performance of the algorithm is analyzed by using various parameters discussed below with their respective values**.**

Accuracy

It is an indicator to show the accuracy of the classifier for selected subset of features. It can be given as:

$$Accuracy = \frac{1}{N}\sum_{i=1}^{N} Match(T_i, O_i)$$

Where Match is function which gives 1 if the target value T_i and output O_i is matched otherwise 0, N is the number of instance.

Sensitivity and Specificity

Sensitivity (Se) and specificity (Sp) describes how well a classifier discriminate the positive and negative classes. The sensitivity is an average of correctly classified positive elements for each class while the sensitivity is average of correctly classified negative elements for each class. It is given as:

$$Se = \frac{1}{c}\sum_{i=1}^{c} \frac{TP_i}{TP_i + FN_i}$$

$$Sp = \frac{1}{c}\sum_{i=1}^{c} \frac{TN_i}{FP_i + TN_i}$$

Where TP, FP, TN, FN are true positive, false positive, true negative, false negative respectively, c is the number of classes in any particular dataset.

c. Result Analysis

The analysis of the parameter discussed in previous section for the described two datasets has been given in the table 1 and table 2. The table 1 shows the comparison of the accuracy for the datasets with exiting [9].

Table 1: Comparison of Accuracy

Dataset		Existing Accuracy	Extended DAP Accuracy	Proposed Accuracy
AwA	Training	0.8405	0.8369	0.8514
	Test	0.7500	0.7261	0.9404
A-Pascal-yahoo	Training	0.9817	0.9817	0.9890
	Test	0.9167	0.9428	0.9740

The specificity and sensitivity generated using the formulae given in previous subsection using the values of confusion matrix is displayed in table 2.

Table 2: Comparison of Specificity (Sp) and Sensitivity (Se)

Dataset		Existing		Extended DAP		Proposed	
		Se	**Sp**	**Se**	**Sp**	**Se**	**Sp**
AwA	Training	0.7597	0.9426	0.7597	0.9344	0.7792	0.9426
	Test	0.8286	0.6939	0.5714	0.8367	0.9714	0.9184
A-Pascal-yahoo	Training	0.9856	0.9692	0.9904	0.9538	0.9952	0.9692
	Test	0.9327	0.8109	0.9770	0.7500	0.9902	0.7500

The comparison of the specificity and sensitivity is shown in table 2 which is also graphically compared using the figure 2 and 3. The improvement in the parameter can be easily observed.

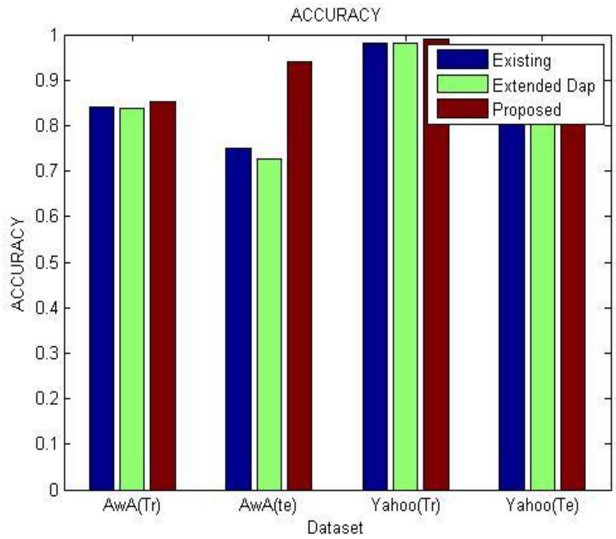

Figure 1: Comparison of Accuracy

The figure 1 shows the comparison of the accuracy values for the existing, extended Dap and the proposed algorithm over described datasets**.** The improvement in the accuracy can be analyzed by the graph.

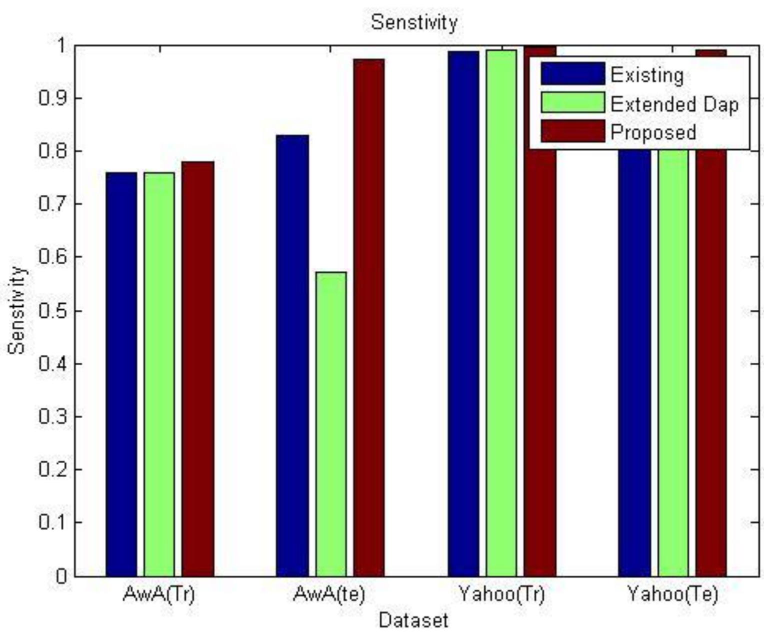

Figure 2: Comparison of Sensitivity

The figure 2 shows the comparison of the sensitivity values for the existing, extended Dap and the proposed algorithm over described datasets**.** The improvement in the sensitivity can be analyzed by the graph.

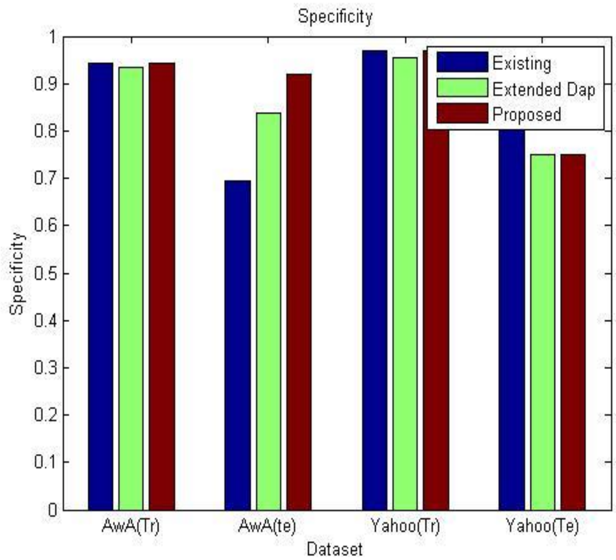

Figure 3: Comparison of Specificity

The figure 3 shows the comparison of the specificity values for the existing, extended Dap and the proposed algorithm over described datasets**.** The improvement in the specificity can be analyzed by the graph. The comparison shown in table 1 and 2 is done graphically shown in the figure 1 to 3. The improvement in the classification accuracy and other parameters can be easily determined by examine the table and the graph.

7. Conclusion

This work extends the Dap in its first part by using a gain based tree classification which improves the accuracy as well as the sensitivity and the specificity of the classification. The classification accuracy and the sensitivity, specificity has been analyzed on two datasets AwA and the yahoo pascal dataset. The analysis clarifies that the extended Dap performs better as compare to the existing work. The further extension of the work has been done by using the swarm intelligence with SVM classifier. The jaya algorithm which is a parameter less algorithm has been used to determine the relevant features which are classified by the SVM classifier. The improvement of the accuracy and other parameter is observed in the results.

References

[1] E. Bonabeau, M. Dorigo, and G. Theraulaz.Swarm Intelligence: From Natural to Artificial Systems. Oxford University Press, New York, NY, 1999.

[2] J. Kennedy and R. C. Eberhart.Particle Swarm Optimization.In Proceedings of IEEE International Conference on Neural Networks, Perth, Australia, pp. 1942–1948, 1995.

[3] D. Karaboga, An Idea Based On Honey Bee Swarm for Numerical Optimization, Technical Report-TR06,Erciyes University, Engineering Faculty, Computer Engineering Department, 2005.

[4] M. Dorigo and T. Stützle, Ant Colony Optimization. MIT Press, Cambridge, 2004. ISBN: 978-0-262-04219-2.

[5] K. M. Passino, Biomimicry of Bacteria Foraging for Distributed Optimization and Control, IEEE Control Systems Magazine, Vol. 22, 52–67, 2002.

[6] S.-C. Chu, P.-W.Tsai and J.-S. Pan, Cat swarm optimization, Proc. of the 9th Pacific

Rim International Conference on Artificial Intelligence, LNAI 4099, pp. 854-858, 2006.

[7] M. Bakhouya and J. Gaber, An Immune Inspired-based Optimization Algorithm: Application to the Traveling Salesman Problem, Advanced Modeling and Optimization, Vol. 9, No. 1, pp. 105-116, 2007.

[8] K.N. Krishnanand and D. Ghose, Glowworm swarm optimization for searching higher dimensional spaces. In: C. P. Lim, L. C. Jain, and S. Dehuri (eds.) Innovations in Swarm Intelligence. Springer, Heidelberg, 2009.

[9] C. H. Lampert, H. Nickisch, and S. Harmeling,"Learning to detect unseen object classes by between-class attribute transfer", in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2009.