ISSN 2229-5518

Multi agent system for classification task

Mohammed A. Mohammed, Ban N. Dhannoon

Abstract— These

Abstract— Multi-agent technology took a significant role in the field of decision-making and machine learning for solving the complex problems in the real world, they simulate human ability to decision-making where behave intelligently, autonomously, cooperatively, and socially to solve problems or to support human users. In this paper, an efficient classification system using multi agent's technology based on Neural Network is introduced, where each agent implements as a neural network (trained using back propagation learning algorithm). The system classifies a collection of datasets effectively with high degree of generalization, it reduces the time and effort to 1/n, where n is the number of classification agents. The developed system was tested using different standard datasets obtained from the UCI Machine Learning Repository for the empirical analysis of machine learning algorithms.

Keywords— Multi-agent technology; Neural Network (NN); Back propagation (BP); classification.

1 Introduction

Classification is one of the most frequently encountered decision making tasks of human activity. A classification problem occurs when an object needs to be assigned into a predefined group or class based on a number of observed attributes related to that object and many problems in business, science, industry, and medicine can be treated as classification problems [1]. Neural network is one of the intelligence methods based decision making and prediction systems where these methods are seemed to be successful to solve difficult and diverse problems by supervised training methods such as back-propagation (BP) algorithm [2]. The Multi-agent systems are computational systems in which two or more agents interact or work together to perform some set of tasks or to satisfy some set of goals. An agent in the system is considered a locus of problem-solving activity, it operates asynchronously with respect to other agents, and it has a certain level of autonomy. [3]. in our proposed system, there are multiple agents' implements as neural networks. These agents work together as multi-agent system to perform classification task on multiple datasets in optimal time and effort.

2 Multi agent systems

An intelligent agent can be defined as a piece of software which performs a given task using information gleaned from its environment to act in a suitable manner so as to complete the task successfully [2]. Agents are seldom stand-alone systems. In many situations they coexist and interact with other agents in several different ways. Such a system that consists of a group of agents that can potentially interact with each other is called a multiagent system (MAS), and the corresponding subfield of Artificial Intelligence (AI) that deals with principles and design of multiagent systems is called distributed AI [4].

3 Artificial Neural Network

Artificial neural networks, sometimes in context referred to only as neural networks, are information processing systems that have certain computational properties analogous to those which have been postulated for biological neural networks [5, 6]. The architecture of NN can be of single layer or multilayer. In single layer Neural Network, only one input layer and one output layer is there, while in multilayer neural network, there can be one or more hidden layer [7, 8]. The real power of a neural network comes from its pattern recognition capabilities. The neural network should be able to produce the desired output even if the input has been slightly distorted [9].

4 Learning Neural Network Models

After configuration input parameters of neural network models, next step is to train neural network models with these settings, the models are trained by using suitable learning algorithm since NNs have the ability to learn by example, the most commonly used NN model- the three-layer feed-forward NN model learning with Back propagation method (BPNN). In the BPNN architecture, each node at input and hidden layers receives input values, processes and passes to the next layer [10].

5 Learning and Generalization

Learning is a hyper surface reconstruction based on existing examples, while generalization means estimating the value on the hyper surface where there is no example. Mathematically, the learning process is a nonlinear curve-fitting process, while generalization is the interpolation and extrapolation of the input data. The goal of training neural networks is not to learn an exact representation of the training data itself, but rather to build a statistical model of the process which generates the data. the generalization capability of a network is jointly determined by the size of the training pattern set, the complexity of the problem, and the architecture of the network. There are many techniques to improve and measure generalization such as generalization by stopping criterion and generalization by regularization. Regularization is a reliable method for improving generalization. Training with a small amount of jitter in the input while keeping the same output can improve generalization. With jitter, the learning problem is

IJSER

equivalent to a smoothing regularization with the noise variance playing the role of the regularization parameter. Training with jitter thus allows regularization within the conventional layered feed forward network architecture [11].

6 The Proposed Classification System

The proposed system used four agents each one of them implements as a multi-layered feed forward neural networks with back propagation learning algorithm, three of them are used for classification task and the other agent (control agent) used to select the suitable agent from the three agents depending on data's features. Each neural network has one hidden layer. The number of nodes in input and output layers depends on the number of features and classes in datasets while the number of nodes in hidden layer is the average nodes of input and output layers. The training and testing of the system implemented on three real datasets where each classification agent trained on a single dataset while the control agent trained on all datasets to specify the corrected classified agent, this will reduces the testing time to 1/n where n is the number of classification agents. The proposed system has been established using Visual Basic.net programming language, and the tests have been conducted under the environment: Windows 7 (32 bit) operating system, laptop computer (Processor: AMD E_450 APU CPU, 1.65 GHz, and (3GB) RAM.

6.1 System Structure

The structure of the proposed system is illustrated in Fig 1.

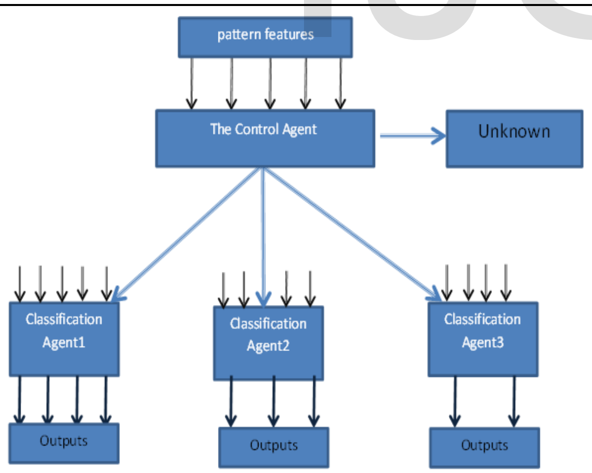

Fig. 1: the general structure of developed system.

The neural network that represent each agent composed of three layers; input layer contain z nodes, where z is the number of features in a class, output layer contain c nodes where c is the number of classes in dataset and hidden layer contained p nodes, where:

$$p = (z + c)/2 \quad \ldots\ldots\ldots\ldots\ldots (1)$$

The nodes of input layer in the control agent equal to the largest number of dataset's features, while its output layer equal to the number of classification agents.

6.2 Training the System

Three real datasets are used to train and test the system (User Knowledge Level, iris, and banknote authentication), one for each classification agent. Each agent is trained using back propagation with pattern mode as shown in Fig. 2 the training is stopped when network error is less than an accepted minimum error or the max number of iterations is exceeded. The control agent is trained on the training data of all classification agents.

- **Data Sets Information**

The attributes for each data set is illustrated in Table 1. The associated task of these data sets are classification and their attribute characteristics is real numbers, while the data characteristics is multivariate.

- **Training phase**

The results of training the system on the ratio of (70% and 80%) is shown in Table 2 and Table 3.

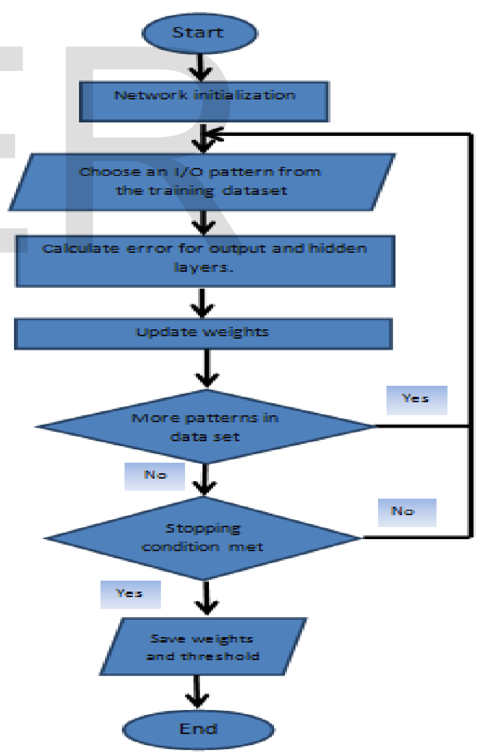

Fig. 2: BP Pattern mode training.

ISSN 2229-5518

TABLE 2

TRAINING THE SYSTEM (70 % TRAINING DATA).

The agent	Training dataset	Number of patterns	Training time	iteration	Net error	True rate	False rate
Classification agent1	User Knowledge Level	282	17.03s	2975	0.0094	97.52%	2.48%
Classification agent2	iris	105	19.64 s	11604	0.00059	98.1%	1.90%
Classification agent3	banknote authentication	960	0.15 s	14	0.00019	98.333%	1.667%

TABLE 3

TRAINING THE SYSTEM (80% TRAINING DATA).

The agent	Training dataset	Number of patterns	Training time	iteration	Net error	True rate	False rate
Classification agent1	User Knowledge Level	322	5.07 s	782	0.0098	98.14%	1.86%
Classification agent2	iris	120	7.15 s	3880	0.00069	98.33%	1.67%
Classification agent3	banknote authentication	1097	0.14 s	12	0.00012	100%	0%

IJSER

6.3 EXPERIMENTAL RESULTS

Experimental results show the efficiency of the proposed system for pattern classification in all tests' data where the control agent select the suitable agent to classify the data and this will reduces the time and effort to find the suitable agent.

- **TEST CLASSIFICATION AGENTS**

TABLE 4

HARD TESTING OF CLASSIFICATION AGENTS (30% TESTING).

dataset	True rate	False rate
User Knowledge Level dataset	96.69% (117 patterns)	3.31% (4 patterns)
iris dataset	100% (45 patterns)	0% (0 patterns)
banknote authentication dataset	100% (411 patterns)	0% (0 patterns)

The result of hard testing on non-trained (30% and 20%) data for each classification agent is shown in Table 4 and

TABLE 5

HARD TESTING OF CLASSIFICATION AGENTS (20% TESTING).

dataset	True rate	False rate
User Knowledge Level dataset	96.69% (117 patterns)	3.31% (4 patterns)
iris dataset	100% (45 patterns)	0% (0 patterns)
banknote authentication dataset	100% (411 patterns)	0% (0 patterns)

Table 5.

6.4 ESTIMATE THE GENERALIZATION

To estimate the generalization a noise added to the original data, the noise take range suitable to each dataset and generated randomly, each confused and original data is trained by the agent ,then test the original and confused tested data . Finally test the confused data (test data) by an agent trained with the original data to estimate the generalization. All results in tables bellow represent true rate of classification and each percent is an average of five experiments.

1. Estimate generalization of the classification agent1 (User Knowledge Level dataset) with 70% and 80% training data is shown in Table 6 and Table 7.

TABLE 1

DATA SET'S ATTRIBUTES.

characteristic	The User Knowledge Level dataset	iris dataset	banknote authentication dataset
Number of Instances	403	150	1371
Area	Education	Life	Computer
Number of Attributes	5	4	4
Number of Classes	4	3	2

TABLE 7

ESTIMATE THE GENERALIZATION OF CLASSIFICATION AGENT1 (80% TRAINING, 20% TESTING).

The noise ranges on training and testing data	Soft test	hard test	hard test with noise	Hard test with noise(training on original training data)
0_0.03	97.85%	95.30%	94.56%	94.56%
0_0.05	96.92%	95.30%	92.09%	91.85%
0_0.07	94.84%	95.80%	94.56%	91.85%
0_0.09	95.65%	95.80%	89.38%	87.65%
0_0.1	93.54%	94.81%	91.35%	89.13%

We conclude from the above results the improve generalization is required when the data is divided into 70% of data to train and 30% of data to test when the added noise is up to 0.07, while improve generalization is required when the added noise is up to 0.03 when the data is divided into 80% of data to train and 20% of data to test.

TABLE 8

ESTIMATE THE GENERALIZATION OF CLASSIFICATION AGENT2 (70% TRAINING, 30% TESTING).

The noise ranges on training and testing data	Soft test	hard test	hard test with noise	Hard test with noise(training on original training data)
0.05_0.1	96.38%	99.56%	98.22%	99.11%
0.05_0.2	96.28%	99.56%	97.25%	98.22%
0.05_0.3	93.31%	98.67%	93.33 %	93.33%
0.05_0.4	91.42%	89.33%	89.77%	94.66%
0.05_0.5	91.51%	95.55%	88.44%	92.00%

TABLE 9

ESTIMATE THE GENERALIZATION OF CLASSIFICATION AGENT2 (80% TRAINING, 20% TESTING).

The noise ranges on training and testing data	Soft test	hard test	hard test with noise	Hard test with noise(training on original training data)
0.05_0.1	96.25%	98.66%	98.66%	88.66%
0.05_0.2	95.83%	100%	98.66%	88.00%
0.05_0.3	96.41%	99.33%	76.66%	88.00%
0.05_0.4	95.75%	99.33%	70.00%	86.00%
0.05_0.5	93.00%	99.33%	66.66%	84.00%

2. Estimate generalization of the classification agent 2(iris dataset) with 70% and 80% training data is shown in Table 8 and 9.

TABLE 6

ESTIMATE THE GENERALIZATION OF CLASSIFICATION AGENT1 (70% TRAINING, 30% TESTING).

The noise ranges on training and testing data	Soft test	hard test	hard test with noise	Hard test with noise(training on original training data)
0_0.03	96.95%	96.19%	93.88%	94.71%
0_0.05	95.99%	94.87%	92.89%	94.38%
0_0.07	96.31%	92.23%	91.07%	91.90%
0_0.09	95.14%	93.71%	90.08	89.91%
0_0.1	94.75%	91.57%	88.59%	86.77%

We conclude from the above results the improve generalization is not required when the data is divided into

IJSER

TABLE 11

ESTIMATE THE GENERALIZATION OF CLASSIFICATION AGENT3 (80% TRAINING, 20% TESTING).

The noise ranges on training and testing data	Soft test	hard test	hard test with noise	Hard test with noise(training on original training data)
0_0.1	99.40%	99.34%	99.27%	99.92%
0_0.3	99.04%	99.27%	99.19%	99.78%
0_0.7	96.71%	96.49%	96.13%	99.48%
0_1	97.06%	97.00%	96.78%	98.32%
0_1.5	94.11%	95.69%	93.79%	95.62%

70% of data to train and 30% of data to test when the added noise is 0.5 and less, while improve generalization is required when the added noise is less than 0.2 when the data is divided into 80% of data to train and 20% of data to test.

3. Estimate generalization of the classification agent 3(Banknote dataset) with 70% and 80% training data is shown in Table 10 and Table 11.

We conclude from the above results the improve generalization is required when the data is divided into

TABLE 12

THE BEST DIVISIONS.

dataset	Best division
User Knowledge Level	70% training, 30% testing
iris	70% training, 30% testing
Banknote authentication	80% training, 20% testing

70% of data to train and 30% of data to test when the added noise is up to 0.7, while improve generalization is not required when the added noise is less than 1.5 when the

TABLE 13

TRAINING OF CONTROL AGENT.

Number of patterns	1484
True rate	100% (1484 patterns)
False rate	0% (0 pattern)
iterations	2034
time	48 s
Net error	0.00099

data is divided into 80% of data to train and 20% of data to test.

TABLE 14

HARD TESTING OF CONTROL AGENT.

Number of patterns	440
True rate	99.77% (439 patterns)
False rate	0.23% (1 pattern)

- **THE CONTROL AGENT**

The results above shown the best division for each data set, these divisions are shown in Table 12.

The result of training control agent is shown in table 13.

The result of hard test of control agent is shown in table 14.

In this paper, three classification agents were used; each one for a specific domain (data set) and one agent is used as a control agent. In the absence of the control agent, when we want to test any given pattern of unknown domain, it will be checked by the agents one by one. The first agent will either classified it or it will be declare it as an unknown pattern. If the first agent couldn't classify the pattern, the second agent will try to classify it, and so on until it will be classified or it will be declared as an unknown pattern.
The probability that any agent is the right agent is 1/n, where n is the number of classification agents. The aim of the control agent is to either direct the tested pattern to the dedicated agent or it declare it as an unknown pattern. The probability of selecting the right agent is 99.77%. Thus using the control agent will reduce the time and effort of selecting the suitable agent to classify the tested pattern.

7 CONCLUSIONS

The aim of this paper is to develop a pattern classification system based on multi-agent technology implemented as neural networks using back propagation as a learning algorithm. This system reduces the time and efforts to classify a collection of datasets to 1/n where n are the number of classification agents.

TABLE 10

ESTIMATE THE GENERALIZATION OF CLASSIFICATION AGENT3 (70% TRAINING, 30% TESTING).

The noise ranges on training and testing data	Soft test	hard test	hard test with noise	Hard test with noise(training on original training data)
0_0.1	99.18%	98.00%	98.00%	100%
0_0.3	99.23%	98.49%	98.05	99.65%
0_0.7	99.59%	98.92%	97.51%	98.63%
0_1	99.70%	99.22%	99.42%	97.61%
0_1.5	99.45%	99.22%	95.52%	93.81%

IJSER

8 REFERENCES

1. [1] Zhang, G.P., "Neural Networks for Classification: A Survey", Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on Vol. 30, No.4, Nov2000.
2. [2] Hanaa, M.M., Dr. Loay, E.G., Dr. Ban N.Dh.,"Agents Technology Based Cooperative Neural Networks for General Pattern Classification Model", International Journal of Scientific & Engineering Research, Vol. 5, No. 3, Mar2014.
3. [3] Victor R.L., Cooperative Multiagent Systems:A Personal View of the State of the Art, Knowledge and Data Engineering, IEEE, Vol. 11, No.1, pp. 133 – 142, 1999.
4. [4] Nikos Vlassis, A Concise Introduction to Multiagent Systems and Distributed Artificial Intelligence",Morgan & Claypool, Vol. 1, No. 1, Pp. 1-71, 2007
5. [5] Farooq A.,Biologically Inspired Modular Neural Networks, Ph.D. thesis, Virginia Polytechnic Institute and State University, 2000.
6. [6] Michael Negnevitsky,Artificial Intelligence a Guide to Intelligent Systems, Second Edition, Pearson Education Limited,2005.
7. [7] Kenji S.,Artificial Neural Networks Methodological Advances and Biomedical Applications, InTech , 2011.
8. [8] Amit G., KostaY.P., Gaurang P., Chintan G., Initial Classification Through Back Propagation In a Neural Network Following Optimization Through GA to Evaluate the Fitness of an Algorithm, International Journal of Computer Science & Information Technology (IJCSIT), Vol. 3, No. 1, Feb 2011.
9. [9] Jeff Heaton,Programming Neural Networks with Encog3 in Java,Heaton Research, Inc. St. Louis, MO, USA, October, 2011.
10. [10] Khaing W. M., Thinn T. N.,Optimum Neural Network Architecture for Precipitation Prediction of Myanmar, World Academy of Science, Engineering and Technology 48, 2008.
11. [11] Ke-Lin Du · M. N. S. Swamy., Neural Networks and Statistical Learning, _ Springer-Verlag London 2014.

IJSER