PERFORMANCE ANALYSIS OF THE KNOWLEDGE SHARING NEURAL NETWORKS SYSTEM

SRINIDHI KATLA

ABSTRACT

Any organization will benefit the most when each agent has the knowledge of the learnings of other agents . This helps them to be aware of the ongoing activities in the other parts of organization and hence make informed decisions which would benefit the organization on the whole. Knowledge sharing also would enhance the "memory" of the organization and could ensure the continuity in the absence of any particular agent. One way to implement this is using artificial neural network model, since they can be modelled to accommodate the dynamics of an organizational network. We test the "learning" of the system by modifying its various parameters .

INTRODUCTION

An individual gains knowledge in various settings such as while pursuing interests, encountering new situations, making mistakes and interacting with others. An organization or a social setting will benefit at large if individuals share the gained knowledge as it increases the collective learning of the group, enhance coordination among the individuals and ensures that the learning is remembered for a longer duration. These would in turn aid in making informed decisions, maintain the continuity of the gained knowledge and reuse the learning when required , hence saving time and energy.

The same argument can be applied to an artificially intelligent multi-agent system environment. If the agents can share the learnings among each other, the overall time to train the system can be drastically reduced and the result would be a consistent knowledge system. One approach to this is to append the new learning in a centrally accessible location which is synced with the agents' local memory at regular intervals. This could lead to accumulation of large amounts of data in the central repository over a period of time. An alternate approach is to combine the learnings of the agents at regular intervals. Artificial neural networks provide an appropriate platform for such an implementation.

We briefly discuss the implementation of the above approach using artificial neural networks and report the performance of the systems under different training levels, learning rates and fuse rates.

TECHNICAL IMPLEMENTATION

The multi agent system is represented as an array of neural networks each of which is trained on a distinct concepts. After each block of training, the neural networks are fused together at "FUZE_RATE" which indicates the amount of learning shared between 2 neural networks.

Algorithm : (code provided by Dr. Michael Gashler)

Set the number of nodes , Nd; Learning rate of the neural networks , lr; Training block size ,TR ;number of iterations, k;

1. Initialize the neural network Nd nodes and lr learning rate;
2. In each iteration k: {
   Test the neural network model performance .
   Select the training set sample of size TR, {
   For each instance, train a subset of neural networks. (Do not train all the NNs for all the instances)
   }
3. Fuse the nodes such that each node shares knowledge with one other node.
   }
   The fuse method is as below:
4. Fuse(nn A, nn B){
   Update weight matrix of A, W_A and Weight Matrix of B, W_B as :

$$\mathbf{W_{A\ (fuzed)} = (1.0 - FUZE\_RATE) * W_{A\ (not\ fuzed)} + FUZE\_RATE * W_{B\ (not\ fuzed)}}$$

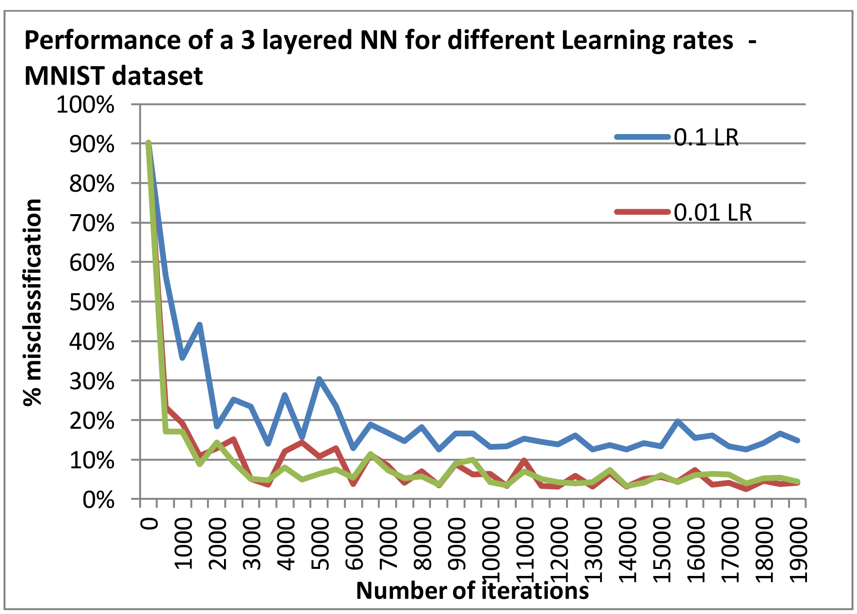
$$\mathbf{W_{B\ (fuzed)} = (1.0 - FUZE\_RATE) * W_{B\ (not\ fuzed)} + FUZE\_RATE * W_{A\ (not\ fuzed)}}\ \}$$

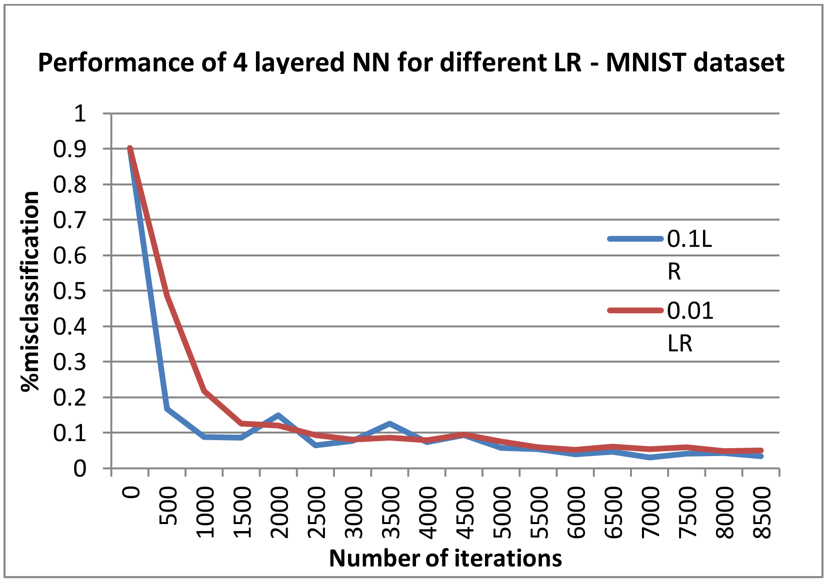
RESULTS

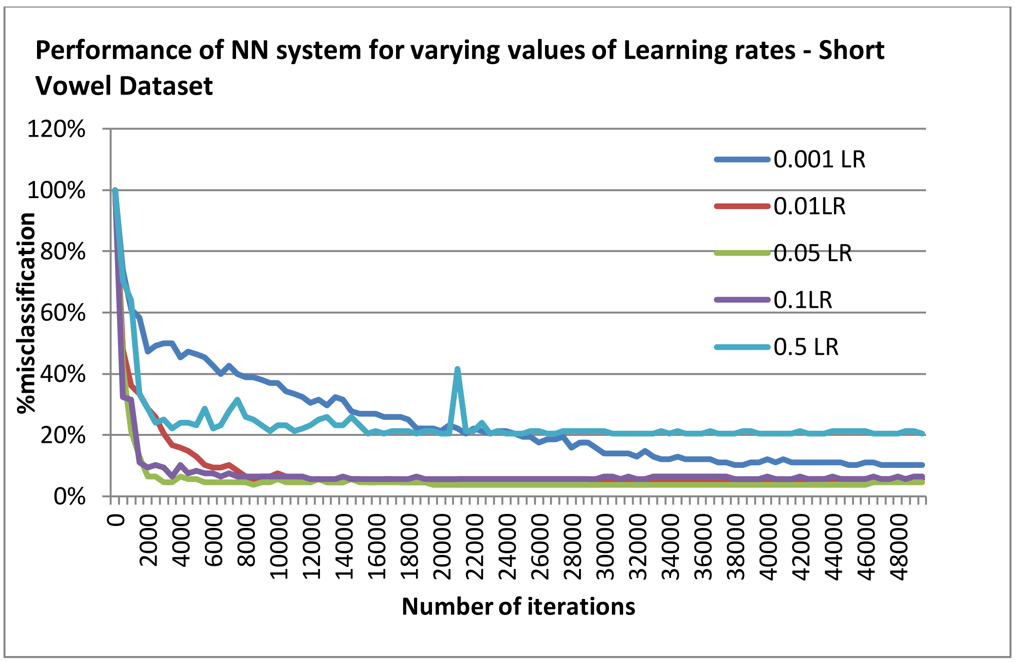
We included the neural network classes from the machine learning toolkit ,waffles [1] and performed the experiments on mnist and vowel datasets. For our experiments, we used the classical layer and default activation method. The performance was measured as the neural network model prediction error (sumsquared error) and misclassification error .

The mnist dataset has 784 numerical attributes and 10 classes, each representing a number while the vowel dataset has 10 numerical attributes and 11 classes. Some experiments were conducted on short_vowel dataset that is a subset of vowel dataset with 5 labels.

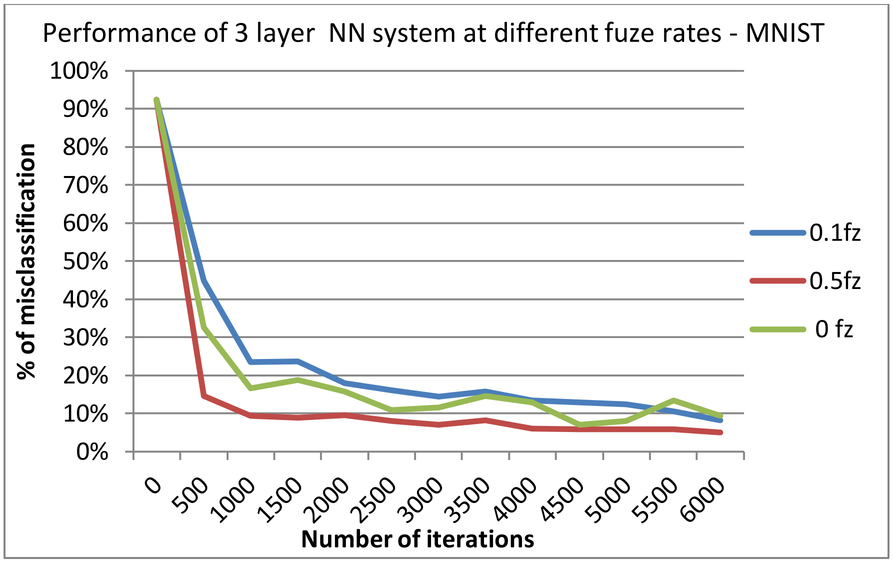
a. Learning rate: For both datasets, we found the performance to be optimal when the learning rate was between 0.01 – 0.1

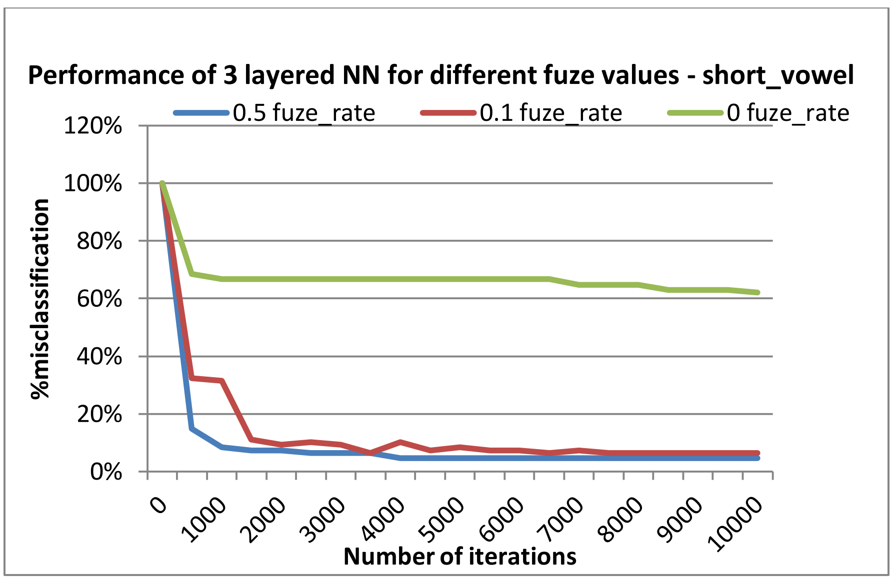






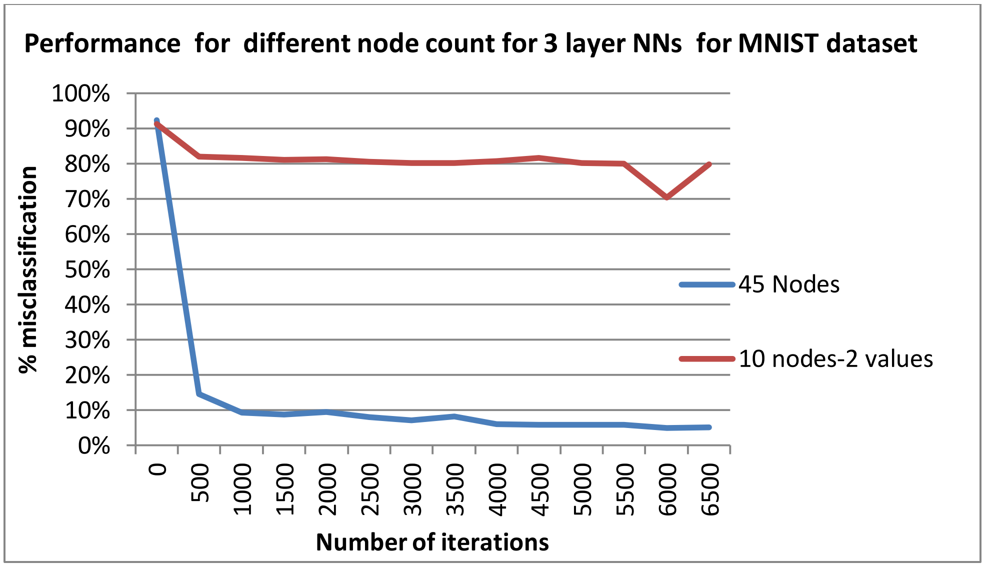
b. Fuze rate: The FUZE_RATE takes the value from 0 – 0.5. It indicates the amount of data being shared between the neural networks that are being fused. A fuze rate of 0.5 indicates that the neural network remember 50 % of its own learning and 50 % learning of the neural network node it is being fuzed with. Just like in real world, where the aggregate knowledge of the social setting is optimum if the knowledge shared by each individual is optimum, we found that for every case tested, the most optimal performance was obtained when the fuze rate was set to 0.5 as against any other value.

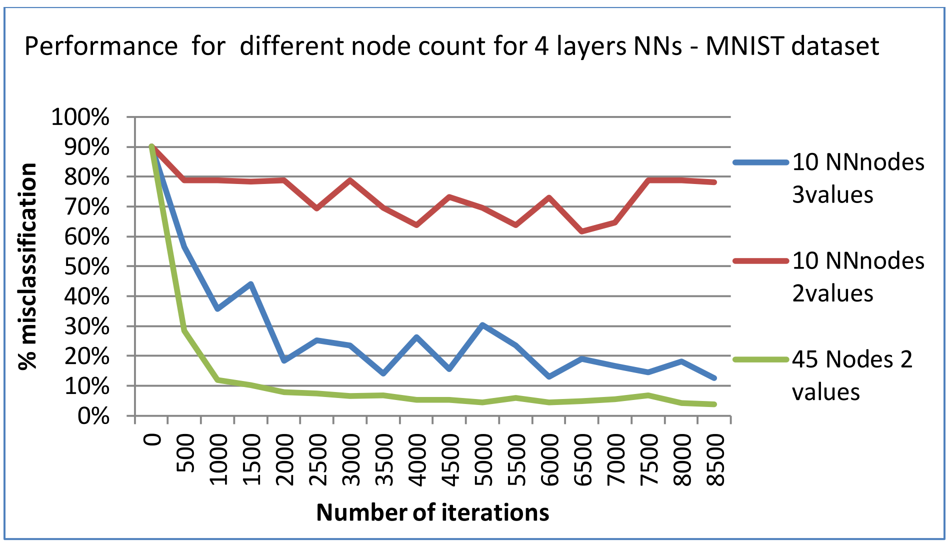




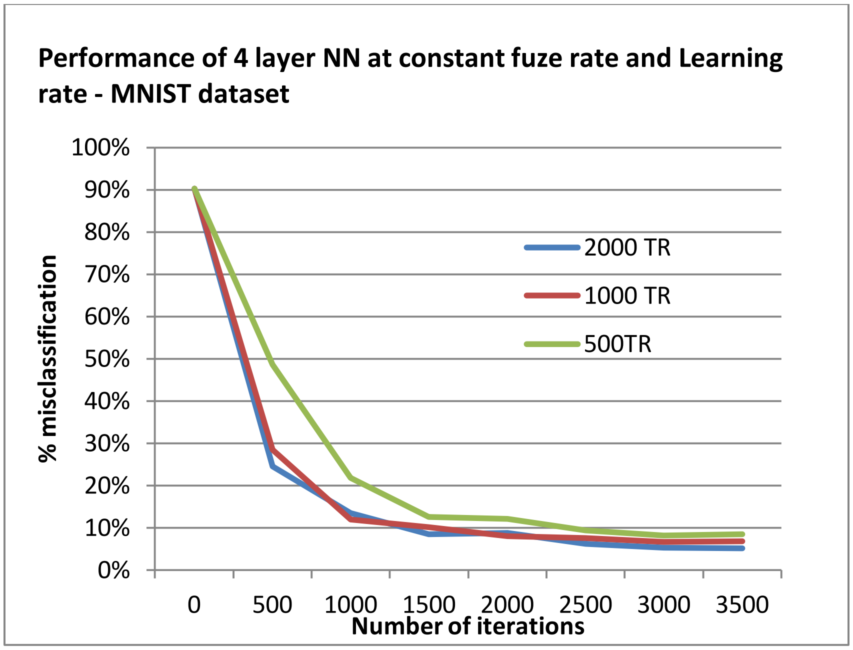
c. Number of Nodes : In the tests performed on mnist dataset, we found that the algorithm works better when large number of nodes are involved and if the nodes have at least some amount of common knowledge.

As the below graphs indicate, the rate of misclassification was considerably lower when 45 neural networks were used to train 10 labels where each NN was trained on 2 labels {(0,1) (0,2) ... } . When 10 neural networks were trained on 10 labels , each NN being trained on 2 labels {(0,1)(1,2)(2,3)..} the rate of miscalculation was very high. However, each of the NN was trained on 3 labels{(0,1,2), (1,2,3); (2,3,4)..}, the miscalculation % was lower than when trained on 2 nodes but higher than when 45 nodes were trained. Similar trend was seen for vowel dataset.





d. Training block size : As the training block size increased, the rate of misclassification and hence the neural network prediction was lower. But, the duration of training in each iteration would increase considerably for larger block size in particularly in larger datasets. The below graph depicts that the % of misclassification is only slightly higher (8%)in training set of 500 instances as against 5 % in training blocks of 2000 instances. However, it was noticed that the time taken for training 2000 instances for 3500 iterations was almost 4 times the time taken for 500 instances.



e. Number of layers : The optimum number of layers seemed depend on the type of dataset used. For mnist dataset, the 4 layered model seemed to perform better than 3 layered and 5 layered model. However, for vowel dataset, the 3layered system performed better than larger number of layers. This could be due to low feature to label ratio in vowel dataset.

Conclusion

By conducting experiments on MNIST and Vowel datasets we found that the system performs better for fuze rate settings of 0.5, the learning rate between 0.1 – 0.001 and large training sets. Also, when multiple neural networks are trained on same concept, the learning is better as against if the neural networks are trained on completely independent concepts.

References

DATASETS :

MNIST : http://uaf46365.ddns.uark.edu/data/mnist/

Vowel Dataset : http://statweb.stanford.edu/~tibs/ElemStatLearn/data.html

Machine Learning Toolkit

Waffles : http://uaf46365.ddns.uark.edu/waffles/