Proceedings of the 28th IEEE
EMBS Annual International Conference
New York City, USA, Aug 30-Sept 3, 2006

Improved Head Direction Command Classification using an Optimised Bayesian Neural Network

Son T. Nguyen[1], Hung T. Nguyen[1], Philip B. Taylor[1], James Middleton[2]

[1] Key University Research Centre for Health Technologies, Faculty of Engineering, University of Technology, Sydney, NSW, AUSTRALIA

[2] Rehabilitation Research Centre, The University of Sydney, NSW, AUSTRALIA

***Abstract*—Assistive technologies have recently emerged to improve the quality of life of severely disabled people by enhancing their independence in daily activities. Since many of those individuals have limited or non-existing control from the neck downward, alternative hands-free input modalities have become very important for these people to access assistive devices. In hands-free control, head movement has been proved to be a very effective user interface as it can provide a comfortable, reliable and natural way to access the device. Recently, neural networks have been shown to be useful not only for real-time pattern recognition but also for creating user-adaptive models. Since multi-layer perceptron neural networks trained using standard back-propagation may cause poor generalisation, the Bayesian technique has been proposed to improve the generalisation and robustness of these networks. This paper describes the use of Bayesian neural networks in developing a hands-free wheelchair control system. The experimental results show that with the optimised architecture, classification Bayesian neural networks can detect head commands of wheelchair users accurately irrespective to their levels of injuries.**

***Index Terms*— Bayesian neural networks, hands-free control, head movement, power wheelchairs.**

I. Introduction

SINCE many of disabled people have limited or non-existing control from the neck downward, alternative hands-free input modalities have become very important for them to access assistive devices. In hands-free control, head movement has been proven to be a very effective user interface as it can provide a comfortable, reliable and natural way to access the device.

The need to design challenging systems that are able to detect head direction commands of various forms of power wheelchair users leads to the use of neural networks [1-3]. In such a system, a trained multi-layer perceptron neural network is responsible for classifying input head movement signals into four output classes corresponding to forward, backward, left and right head movement in real-time. Since multi-layer perceptron neural networks trained using the Bayesian technique (Bayesian neural networks) can have superior generalisation properties than standard multi-layer perceptron neural networks in case of finite available training data, they have been used to develop effective head movement classifiers in our hands-free power wheelchair control systems [4, 5].

In this paper, we continue to optimise the classification Bayesian neural networks used to classify head direction commands of wheelchair users. The structure of the paper is organised as follows. The Section II provides an overview of a head movement-based wheelchair control system. In Section III, the formulation of classification Bayesian neural networks is briefly described. Section IV presents the selection of an appropriate architecture of classification Bayesian neural networks for this application by comparing the log evidence of different candidature architectures and how a disabled user-adaptive head movement classification can be performed. Section V provides a discussion and conclusion for this research.

II. System Overview

The wheelchair platform is based on a commercial powered wheelchair. The movement of the user's head is detected by analysing data from a dual-axis accelerometer installed in a cap worn by the user. The head movement data was collected with a sampling period of 100 ms. A pre-trained classification Bayesian neural network is used to classify four gestures in real-time, corresponding to commands for forward, backward, left and right.

The start of the head movement to control the travel direction of the wheelchair is determined to be the point where the deviation from the neutral position reached 25% of the maximum value on the relevant axis. The classification of the movement was determined as the first classification made by the Bayesian neural network after the start of the movement. The input to the Bayesian neural network is comprised of a window of 20 samples from each axis.

The computer interface module consists of the feedback to the user: real-time graphical displays of the accelerometer data allow the user to track the deviation of their head from the neutral position. Boolean outputs from the classifier and the numerical values of the Bayesian neural network inform the user of how the classifier is interpreting their head movement. The success of the system heavily depends on the network training.

III. Classification Bayesian Neural Networks

Bayesian learning of multi-layer perceptron neural networks is performed by considering Gaussian probability distributions of the weights which can give the best generalisation [6, 7]. In particular, the weights w in network X are adjusted to their most probable values given the training data D. Specifically, the posterior distribution of the weights can be computed using Bayes' rule as follows

$$p(w \mid D, X) = \frac{p(D \mid w, X) p(w \mid X)}{p(D \mid X)} \tag{1}$$

where $p(D \mid w, X)$ is the likelihood function, which contains information about the weights from observations and the prior distribution $p(w \mid X)$ contains information about the weights from background knowledge. The denominator, $p(D \mid X)$, is known as the evidence for network X.

Given a set of candidate networks X_i having different numbers of hidden nodes, the posterior probability of each network can be expressed as

$$p(X_i \mid D) = \frac{p(D \mid X_i) p(X_i)}{p(D)} \tag{2}$$

If the networks are assumed to be equally probable before any data is observed, then $p(X_i)$ is the same for all networks. Since $p(D)$ does not depend on the network, then the probable network is the one with the highest evidence $p(D \mid X_i)$. Therefore, the evidence can be used to compare and rank different candidature networks.

Regularisation can be used to prevent any weights becoming excessively large, which can lead to poor generalisation. For a multi-layer perceptron neural network classifier with G groups of weights and biases, a weight decay penalty term proportional to the sum of squares of the weights and biases is added to the data error function E_D to obtain the cost function

$$S = E_D + \sum_{g=1}^{G} \xi_g E_{W_g} \tag{3}$$

$$E_{W_g} = \frac{1}{2} \left\| w_g \right\|^2 \qquad (g = 1,...,G) \tag{4}$$

where S is called the cost function, ξ_g is a non-negative scalar, sometimes knows as a *hyperparameter*, ensuring the distribution of weights and biases in group g and w_g is the vector of weights and biases in group g.

In network training, the hyperparameters are initialised to be arbitrary small values. The cost function is then minimised using an advanced optimisation technique. When the cost function has reached a local minimum, the hyperparameter ξ_g ($g = 1,...,G$) must be re-estimated. This task requires computing the Hessian matrix of the cost function:

$$A = H + \sum_{g=1}^{G} \xi_g I_g \tag{5}$$

where H is the Hessian matrix of E_D and I_g is the identity matrix, which selects weights in the g th group. The number of 'well-determined' weights γ_g in group g is calculated based on the old value of ξ_g as follows [7]

$$\gamma_g = W_g - \xi_g tr\left(A^{-1} I_g\right) \qquad (g = 1,...,G) \tag{6}$$

The new value of the hyperparameter ξ_g is then re-estimated as [7]

$$\xi_g = \frac{\gamma_g}{2 E_{W_g}} \qquad (g = 1,...,G) \tag{7}$$

The hyperparameters need to be re-estimated several times until the cost function value ceases to change significantly between consecutive re-estimation periods. After the network training is completed, the values of parameters γ_g and ξ_g are then used to compute the log evidence of network X_i having M hidden nodes as follows [8, 9]

$$\begin{aligned} \ln Ev(X_i) = & -S + \sum_{g=1}^{G} \frac{W_g}{2} \ln \xi_g - \frac{1}{2} \ln|A| + \ln M! + M \ln 2 \\ & + \sum_{g=1}^{G} \frac{1}{2} \left(\frac{4\pi}{\gamma_g} \right) - G \ln(\ln \Omega) \end{aligned} \tag{8}$$

where W_g is the number of weights and biases in group g, and Ω is set to be 10^3 [9]. However, Ω is a minor factor because it is the same for all models and therefore does not effect to the relative comparison of log evidence of different network architectures. Equation (8) is used to compare different networks having different numbers of hidden nodes. The best network will be selected with the highest log evidence.

IV. Experiments and Results

A. Data Acquisition

Data was collected from eight adults, aged between 19 and 56, with approval from the UTS Human Research Ethics Committee and informed consent from the volunteers. Of these, four had high-level spinal cord injuries (C4 and C5) and were not able to use a standard joystick to control a wheelchair. The remaining four did not have conditions affecting their head movement. Data for each person was collected in two periods of ten minutes, with the user being prompted to give a specified movement every 6 seconds. Each specified movement was chosen randomly from the following: forward, backward, left and right. The extracted movement samples of those users are shown in Table I.

B. Optimisation of Network Architecture

The recorded movements of user 7 and 8 were randomly divided into two sets. Each set contained 20 samples of each movement. Training data was taken from the recorded movements of users 1, 2, 3, 4, 5 and 6, corresponding to 480 samples. Different Bayesian neural networks with varying numbers of hidden nodes were trained to select the optimal network architecture. These networks have the following specification:

- four hyperparameters ξ_1, ξ_2, ξ_3 and ξ_4 to constrain the magnitudes of the weights on the connection from the input nodes to the hidden nodes, the biases of the hidden nodes, the weights on the connection from the hidden nodes to the output nodes, and the biases of the output nodes;
- 41 inputs, corresponding to 20 samples from x axis, 20 samples from y axis and one augmented input with a constant value of 1;
- four outputs, each corresponding to one of the classes: forward, backward, left and right movement.

For a given number of hidden nodes, ten networks with different initial conditions were trained. The training procedure was implemented as follows:

1. The weights and biases in four different groups were initialised by random selections from zero-mean, unit variance Gaussians and the initial hyperparameters were chosen to be small values.
2. The network was trained to minimise the cost function S using the quasi-Newton algorithm [5].
3. When the network training had reached a local minimum, the values of the hyperparameters were re-estimated according to equation (6) and (7).
4. Steps 2 and 3 were repeated until the cost function value was smaller than a pre-determined value and did not change significantly in subsequent re-estimations.

The performances of the trained networks were tested using a test set taken from the first set of movement samples of user 7 and 8. As shown in Fig.1, the networks having three hidden nodes have the highest evidence. This means that three hidden nodes are sufficient to solve the problem. As shown in Fig.2, these networks also have low test errors (misclassification percentages). The optimisation of Bayesian neural network architecture is extremely important for on-line network training systems because incorporating with the quasi-Newton training algorithm it can contribute to the least network training time while still maintaining the best generalisation for the network.

TABLE I

Extracted Movement Samples

User	Forward	Backward	Left	Right	Injury Level
1	20	20	20	20	-
2	20	20	20	20	-
3	20	20	20	20	-
4	20	20	20	20	-
5	20	20	20	20	C5
6	20	20	20	20	C4
7	20	20	20	20	C4
8	20	20	20	20	C5

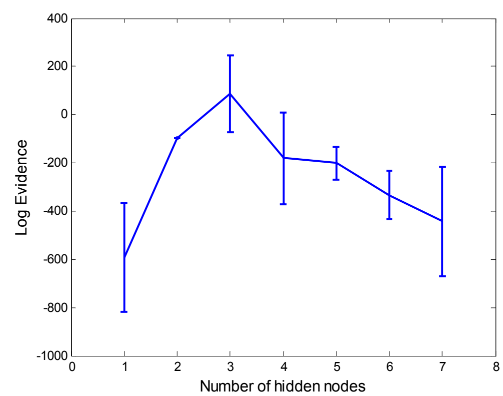

Fig.1. Log evidence versus number of hidden nodes: The solid curve shows the evidence averaged over the ten networks.

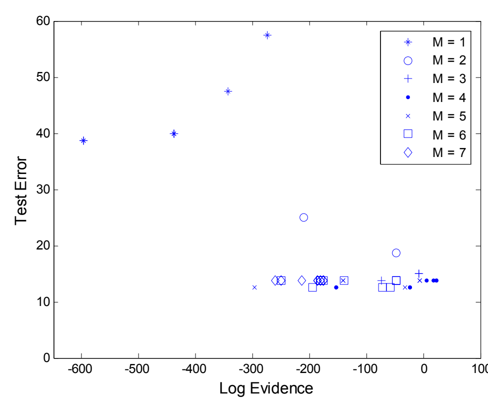

Fig.2. Test error versus log evidence: Every symbol appears ten times, once for each of the ten networks trained.

C. *Head Movement Classification*

1) Experiment 1:

A Bayesian neural network classifier having three hidden nodes was trained using the training data described in section IV-B. The performance of the trained network was tested using the first set of movement samples of user 7 and 8. The confusion matrix in Table II shows how accurately the trained network classifies each type of movement and the trained network can classify all samples in the test set with an accuracy of 85%.

2) Experiment 2:

More training data was taken from the second set of movement samples of user 7 and 8. The Bayesian neural network was trained using the same procedure in Experiment 1. The performance of the trained network was also tested using the first set of movement samples of user 7 and 8. Similarly, a confusion matrix used to evaluate the performance of the trained network as seen in Table III and the network can classify all samples in the test set with a success rate of 93.75%.

The classification results of Experiment 1 and 2 are summarised in Table IV. Especially, it can be seen that very high sensitivity (true positive) and specificity (true negative) have been achieved for Experiment 2. This means that the performance of the trained network has been significantly improved as more movement samples have been included to train the network.

TABLE II
CONFUSION MATRIX IN EXPERIMENT I

		Predicted Classification			
	Movement	Forward	Backward	Left	Right
Actual Classification	Forward	15	0	1	4
	Backward	0	17	3	0
	Left	4	0	16	0
	Right	0	0	0	20
	Accuracy (%)	85			

TABLE III
CONFUSION MATRIX IN EXPERIMENT II

		Predicted Classification			
	Movement	Forward	Backward	Left	Right
Actual Classification	Forward	18	0	1	1
	Backward	0	19	1	0
	Left	2	0	18	0
	Right	0	0	0	20
	Accuracy (%)	93.75			

TABLE IV
SENSITIVITY AND SPECIFICITY OF THE NETWORK

Experiment	Sensitivity	Specificity
1	0.85	0.95
2	0.9375	0.97917

V. DISCUSSION AND CONCLUSION

The results obtained show that Bayesian neural networks can be used to classify head movement accurately. The use of three hidden nodes is an optimal choice for the network architecture as it and the fast training quasi-Newton algorithm can make a significant reduction of on-line network training time. This number of hidden nodes guarantees the best generalisation of the trained network. When the Bayesian neural network was trained using head movement data of the six users, it can classify head movements of two new disabled persons with 85% accuracy. However, if the network was trained further with additional head movement samples of those two disabled persons, it can classify their head movement with a high accuracy of 93.75%. In other words, the network is able to provide an on-line adaptation to the head movement of new disabled wheelchair users.

ACKNOWLEDGMENT

This research was supported partly by the ARC LIEF grant (LE0454081), ARC Discovery grant (DP0666942) and Key University Research Centre for Health Technologies, UTS.

REFERENCES

[1] T. Joseph and H. Nguyen, "Neural network control of wheelchairs using telemetric head movement," *Proceedings of the 20th Annual International Conference of the IEEE, Engineering in Medicine and Biology Society*, vol. 5, pp. 2731 - 2733, 1998.

[2] P. B. Taylor and H. T. Nguyen, "Performance of a head-movement interface for wheelchair control," *Proceedings of the 25th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, vol. 2, pp. 1590 - 1593, 2003.

[3] H. T. Nguyen, L. M. King, and G. Knight, "Real-time head movement system and embedded Linux implementation for the control of power wheelchairs," *26th Annual International Conference of the Engineering in Medicine and Biology Society*, vol. 2, pp. 4892 - 4895, 2004.

[4] S. T. Nguyen, H. T. Nguyen, and P. Taylor, "Hands-Free Control of Power Wheelchairs using Bayesian Neural Networks," *Proceeding of IEEE Conference on Cybernetics and Intelligent Systems*, pp. 745 - 749, 2004.

[5] S. T. Nguyen, H. T. Nguyen, and P. Taylor, "Bayesian Neural Network Classification of Head Movement Direction using Various Advanced Optimisation Training Algorithms," *Proceedings of the first IEEE / RAS-EMBS International Conference on Biomedical Robotics and Biomechatronics, Tuscany Italy, 2006*, 2006.

[6] D. J. C. MacKay, "A practical Bayesian Framework for Backpropagation Networks," *Computation and Neural Systems*, vol. 4, pp. 448-472, 1992b.

[7] D. MacKay, "The Evidence Framework Applied to Classification Networks," *Neural Computation*, vol. 4, pp. 720 -736, 1992.

[8] W. D. Penny and S. J. Roberts, "Bayesian neural networks for classification: how useful is the evidence framework," *Neural Networks*, vol. 12, pp. 877--892, 1999.

[9] H. H. Thodberg, "A review of Bayesian neural networks with an application to near infrared spectroscopy," *IEEE Transactions on Neural Networks*, vol. 7, pp. 56 - 72, 1996.