2011 International Conference on Management and Artificial Intelligence
IPEDR vol.6 (2011) © (2011) IACSIT Press, Bali, Indonesia

KNOWLEDGE EXTRACTION IN HOPFIELD NETWORK

Saratha Sathasivam
School of Mathematical Sciences, Universiti Sains Malaysia, 11800 USM, Penang, Malaysia.
Email: saratha@cs.usm.my

***Abstract*- Deduction simplifies the knowledge representation without affecting the knowledge contents. Thus, deduction makes the clauses more compact and easier to be interpreted as a large amount of redundancy may obscure the meaning of the represented knowledge. In this paper we will look into how deduction is used in simplifying the induced rules without affecting the knowledge content in neural network. Computer simulations that had been carried out verified and validated the proposed theory. We prove that the deduced rules and the induced rules are similar in knowledge content.**

Keywords: Deduction, redundant, induced, neural network

I. INTRODUCTION

Abduction, deduction and induction have different merits and shortcomings. Yet the combination of all these three reasoning approaches provides researchers a powerful tool of inquiry. The logic of abduction and deduction contribute to our conceptual understanding of a phenomenon, while the logic of induction adds quantitative details to our conceptual knowledge. At the stage of abduction, the goal is to explore the data, to find a pattern, and to suggest hypotheses. On the other hand, deduction is to refine the hypotheses, based upon other premises.

The philosophical notions introduced by Peirce [1] are helpful for researchers in understanding the nature of knowledge and reality. According to Peirce, the logic of abduction and deduction enhance our conceptual understanding of a phenomenon, while the logic of induction provides us quantitative details to our conceptual knowledge. In this paper, we will concentrate on the implementation of deduction in neural symbolic integration.

II. DEDUCTION

Deduction starts with a general case, which is previously known facts or axioms and infers specific instances.

Deduction :
Major premise : All balls in the box are red
Major premise : These balls are from the box
Conclusion : These balls are red

In deduction, its conclusion is a logical consequence of the premises. The deduction is valid only in condition that the conclusion follow necessarily from the premises.

The advantage of deduction in the logic approach is that it can reduce interference effects due to common neuron sharing. Deduction can be applied to remove redundant clauses leading to a simplification of the clauses. A knowledge base is redundant if it contains parts that can be inferred from the rest of it. Deduction simplifies the knowledge representation without affecting the knowledge contents. Thus, deduction makes the clauses more compact and easier to be interpreted as a large amount of redundancy may obscure the meaning of the represented knowledge [2]. Moreover, with the decrease in the number of neuron in the network, the complexity of the network will be lower. This may then leads to some computational advantage. Therefore, deduction can be very useful especially in large data sets which tend to generate more redundant clauses. In the next section we will look into how deduction is used in simplifying the induced rules without affecting the knowledge content.

III. DEDUCTION ALGORITHM

The following steps are used:

i) List out the clauses with interference effects obtained from real life or simulated data sets using reverse analysis method [3]. Reverse analysis method is used to induce logical rules entrenched in data sets by using information about the connection strengths between the neurons in the network.
ii) By applying deduction, represent the clauses in a simpler manner.
iii) Check whether the clauses obtained from step (i) are similar with simplified clauses from step (ii). The consistency of the results can be checked by using the truth tables.
iv) Check that the resulting Hopfield neural network (by the deduced clauses) [5, 6, 7] produces the same global minima with the original network.

For an example, consider the case as illustrated below:

$$A \leftarrow B, C$$
$$C \leftarrow .$$

which is redundant due to repetition of neuron C.

By applying deductive technique, we expect that the set of clauses can be simplified to the clauses below:

$$A \leftarrow B$$
$$C \leftarrow .$$

To verify that the deduction above is valid, the consistency of the result is checked by using truth tables (Table 1 & Table 2).

TABLE I. TRUTH TABLE FOR $A \leftarrow B, C$ AND $C \leftarrow$

A	B	C	$A \leftarrow B, C$	$C \leftarrow .$	$A \leftarrow B, C$ *and* $C \leftarrow .$
1	1	1	1	1	1
1	1	-1	1	-1	-1
1	-1	1	1	1	1
1	-1	-1	1	-1	-1
-1	1	1	-1	1	-1
-1	1	-1	1	-1	-1
-1	-1	1	1	1	1
-1	-1	-1	1	-1	-1

TABLE II. TRUTH TABLE FOR $A \leftarrow B$ *and* $C \leftarrow$

A	B	C	$A \leftarrow B$	$C \leftarrow .$	$A \leftarrow B$ *and* $C \leftarrow .$
1	1	1	1	1	1
1	1	-1	1	-1	-1
1	-1	1	1	1	1
1	-1	-1	1	-1	-1
-1	1	1	-1	1	-1
-1	1	-1	-1	-1	-1
``-1	-1	1	1	1	1
-1	-1	-1	1	-1	-1

By inspection on Table 1 and Table 2, these two sets of clauses contain the same logical information. Sets of interpretation for both cases are similar, where there are 3 modals and 5 non-modals for both sets of program clauses. So, deduction works. The clauses, $A \leftarrow B, C$ *and* $C \leftarrow .$ can be simplified to the clauses, $A \leftarrow B$ *and* $C \leftarrow$. by applying the deductive technique.

An algorithm for automate deduction is proposed here. Before the implementation of deduction into an algorithm, we carried out the above deduction by inspection to study the pattern of the possible sets of clauses that can be deduced. In this project, we are only dealing with maximally three-atom Horn clauses neural network due to the reason [4, 5, 6]. Therefore, all the possible clauses that we will obtain from the real life or simulated data sets using reverse analysis method are as shown in the following Table 3.

TABLE III. ALL POSSIBLE CLAUSES FOR THREE-NEURON HORN CLAUSES NEURAL NETWORK (A, B, C)

First Order Clauses	Second Order Clauses	Third Order Clauses
$A \leftarrow .$	$A \leftarrow B$	$A \leftarrow B, C$
$B \leftarrow .$	$A \leftarrow C$	$B \leftarrow A, C$
$C \leftarrow .$	$B \leftarrow A$	$C \leftarrow A, B$

	$B \leftarrow C$ $C \leftarrow A$ $C \leftarrow B$	

Most of the sets of clauses discussed in Table 3 contain redundancy clauses due to common neuron sharing. So, we carried out analysis on all these sets of clauses and attempt to deduce it to a more simplified form. The consistency of the results is being checked by using the truth tables confirmations and the global minima of the resulting neural network. There should be an agreement between the neural network obtained using the original sets of clauses and neural network programmed with deduced clauses. The following Table 4 summarizes all the possible sets of two clauses that can be deduced and its' corresponding deduced clauses.

TABLE IV. THE SETS OF TWO CLAUSES THAT CAN BE DEDUCED AND ITS' DEDUCED CLAUSES

	Original Clauses	Deduced Clauses
Common neuron in the head of both clauses	$L \leftarrow .$ $L \leftarrow M$	$L \leftarrow .$
	$L \leftarrow .$ $L \leftarrow M.N$	$L \leftarrow .$
	$L \leftarrow M$ $L \leftarrow M, N$	$L \leftarrow M$
Common neuron in head of one of the clause and in body of another clause	$L \leftarrow .$ $M \leftarrow L$	$L \leftarrow .$ $M \leftarrow .$
	$L \leftarrow .$ $M \leftarrow L, N$	$L \leftarrow .$ $M \leftarrow N$
	$L \leftarrow M$ $N \leftarrow L, M$	$L \leftarrow M$ $N \leftarrow M$

Following we show how the clauses deduction can be carried out and the relevant truth table analysis.

Original Clauses : $L \leftarrow .$

$L \leftarrow M$

Deduced Clauses : $L \leftarrow .$

TABLE V. TRUTH TABLE FOR $(L \leftarrow .(L \leftarrow M$ and $L \leftarrow$.)

L	M	N	$L \leftarrow .(L \leftarrow M$	$L \leftarrow .$
1	1	1	1	1
1	1	-1	1	1
1	-1	1	1	1
1	-1	-1	1	1
-1	1	1	-1	-1
-1	1	-1	-1	-1

-1	-1	1	-1	-1
-1	-1	-1	-1	-1

By inspection on Table 5, these two sets of clauses contain the same logical information. Sets of interpretation for both cases are similar, where there are 4 modals and 4 non-modals for both sets of program clauses. So, deduction works.

The clauses, $L \leftarrow .$ and $L \leftarrow M$ can be simplified to the clause $L \leftarrow$. by applying the deduction technique.

IV. Simulation

The theory and studies of deduction that we obtained in the previous chapter are being implemented into a computer program. First of all, real life or simulated events data sets are being input each time the program runs. Then, we apply reverse analysis method to obtain the logical clauses entrenched in the network.

For those clauses with common neuron sharing, check whether the redundant neuron can be removed and lead to a more simplified form of clauses. Deduction is applied on the clauses that may be simplified according to our studies in previous section. Next, check if the deduced clauses obtained is consistent with the original clauses. The consistency of the results is checked using the truth tables. The resulting neural network using the deduced clauses should produce the same synaptic strength through Hebbian Learning with the network using the original clauses.

In the simulation process, we run the program for events with three neurons. We try to input different number of events with different combinations of neurons in each run to obtain a variety set of logical rules that may be deduced in a different way. This is to determine whether the deduction technique implemented in the program is efficient enough to simplify the logical rules generated from the simulated data sets. The following figure shows the maximum number of original clauses obtained and its minimum number of deduced clauses for different number of events input in our simulation.

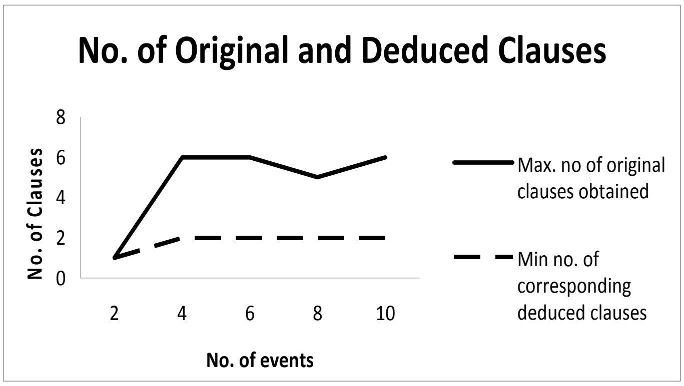

Figure 1. Maximum and minimum number of original clauses obtained

From Figure 1, it is clearly showed that the number of deduced clauses generated by the program is always smaller than or equal to the number of its original clauses obtained. From the simulation results that we obtain and Figure 1, it can be observed that deduction technique applied in the program simplify the original generated clauses which contain redundancy in knowledge representation i.e. which contain common neuron in the clauses.

V. Conclusion

By applying the deduction technique, the redundant neurons that may affect the interpretation of the clauses are being removed without affecting the knowledge contents. The consistency of the knowledge contents of both the original and deduced clauses have been confirmed by using the truth tables. Hence, the interference effect and logic redundancy in the logical clauses obtained from the simulated data sets can be reduced. Furthermore, the decrease in the number of neurons in a network will then decrease the complexity of the network as well. The complexity of the logical clauses will be lower by using the deduced clauses that are more compact and easier to be interpreted. Apart from that, this will lead to some computational advantage.

Acknowledgement

This research is partly financed by short term grant (304/ PMATHS/631006) from Universiti Sains Malaysia.

References

[1] Peirce, C. S. *Collected papers of Charles Sanders Peirce*. Cambridge: 1990. Harvard University Press.

[2] Ashok, K.G. A Neural Architecture For A Class Of Abduction Problems. *IEEE Transactions On Systems, Man, And Cybernetics*, **26(6), 1996,** pp 854-860.

[3] Wan Abdullah, W.A.T. . The Logic of Neural Networks. *Phys. Lett. A.*, **176**. 1993, pp 202-206.

[4] Hopcroft, J.E. and Ullmann, J.D. *Introduction to Automata Theory, Language and Computation*. 1979. Addison-Wesley Publ.

[5] Hopfield, J.J. Neural Networks and Physical Systems with Emergent Collective Computational Abilities. *Proceedings. Natl. Acad. Sci. USA.* , **79(8)**, 1982, pp 2554-2558.

[6] Hopfield, J.J..Neurons with Graded Response Have Collective Computational Properties like Those of Two-State Neurons. *Proceeding. Natl. Acad. Sci. USA.* , **81(10)**, 1984, pp 3088-3092.