Knowledge Extraction from Self-Organizing Neural Networks

A.Ultsch

Department of Computer Sience, University of Dortmund
P.O: Box 500 500
D-4600 Dortmund 50, Germany

Abstract:

In this work we present the integration of neural networks with a rule based expert system. The system realizes the automatic acquisition of knowledge out of a set of examples. It enhances the reasoning capabilities of classical expert systems with the ability of generalise and the handling of incomplete cases. It uses neural nets with unsupervised learning algorithms to extract regularities out of case data. A symbolic rule generator transforms these regularities into PROLOG rules. The generated rules and the trained neural nets are embedded into the expert system as knowledge bases. In the system«s diagnosis phase it is possible to use these knowledge bases together with human experts« knowledge bases in order to diagnose a unknown case. Furthermore the system is able to diagnose and to complete inconsistent data using the trained neural nets exploiting their ability to generalise.

1. Introduction

Knowledge acquisition for expert systems poses many problems. Expert systems depend on a human expert formulate knowledge in symbolic rules. It is almost impossible for an expert to describe knowledge entirely in the form of rules. In particular it is very difficult to describe knowledge acquired by experience. An expert system may therefore not be able to diagnose a case which the expert is able to. The question is how to extract experience from a set of examples for the use of expert systems.

Machine Learning algorithms such as "learning from example" claim that they are able to extract knowledge from experience. Symbolic systems as, for example, ID3 [Quinlan 84] and version-space [Mitchell 82] are capable to learn from examples. Connectionist systems claim to have advantages over these systems in generalisation and in handling noisy and incomplete data. Queries to expert systems often contain inconsistent data. For every data set the rule based systems have to find a definite diagnosis. Inconsistent data can force symbolic systems into an indefinite state. In connectionist networks a distributed representation of concepts is used. The interference of different concepts allows networks to generalize [Hinton et al. 86a]. A network computes for every input the best output. Due to this connectionist networks perform well in handling noisy and incomplete data. They are also able to make a plausible statement about missing components. A system that uses an rule based expert system with an integrated connectionist network could benefit of the described advantages of connectionist systems.

2. System Overview

Figure 1 gives an overview of our system. Case data that are presented to an expert system are (usually) stored in a case database. A data transformation module encodes such cases in a

suitable way in order to be learned by neuronal networks. This module performs as follows: first it transforms the data so that the components have equal scopes. One of the possibilities is the use of a z-transformation.The second task of the transformation module is to encode the data into a binary input pattern because some neural networks, as, for example, the competitive learning model [Rumelhart/Zipser 85], only processes binary inputs. To do this, the intervals of the components are subdivided into different ranges. These ranges are adapted according to the distribution of the components. So every component of a vector is represented by the range its value belongs to. Depending on the kind of representation, the ranges could be encoded locally, locally-distributed or distributed.
With the so transformed data different neuronal networks with unsuperwised learning algorithms, such as Competitive Learning [Rumelhart/Zipser 85], ART [CARPENTER/GROSSBERG 87] and Kohonen [Kohonen 84], are trained. These Networks have the ability to adapt their internal structures (weights) to the structure of the data.In a Rule Generation module the structures learned by the neuronal networks are detected, examined and transormed into expert systems rules. These rules can be inspected by a human expert an added to an expert system.

Figure 1: Overview of the System

When a case is presented to the expert system, the system first tries to reason with the rules that have been aquired from an expert in order to produce a suitable diagnosis. If this fails to produce a diagnosis, the new rules produced by the process described above can be used. If the case can be handled in such a way, all steps of the reasoning process may be inspected by and explained to a user of the system.

If the system, however, is not able to produce a suitable diagnosisin this way. Be it, that data is missing, or the input is erroneous. Or no rule fits the data, since such a case has not been considerd while building the knowledge base, the expert system can turn the case over to the networks. The networks, with their aibility to associate and generalize, search for a most suiting case that has been learned before. The diagnosis, that has been associated with that case is then returned as a possible diagnosis.

3. Detecting Structures

One of the network we used was a Kohonen network consisting of two layers. The input layer has n units representing the n components of a data vector. The output layer is a two dimensional array of units arranged on a grid. The number of the output units is determined experimentally. Each unit in the input layer is connected to every unit in the output layer with a weight associated. The weights are initialized randomly taking the smallest and the greatest value of each component (of all vectors) as boundaries. They are adjusted according to Kohonen's learning rule [Kohonen 84]. The applied rule uses the Euclidean distance and a simulated mexican-hat function to realize lateral inhibition. In the output layer neighbouring units form regions, which correspond to similar input vectors. These neighbourhoods form disjoint regions, thus classifying the input vectors.

The automatic detection of this classification is difficult because the Kohonen algorithm converges to an equal distribution of the units in the output layer. So a special algorithm, the so called U-matrix method was developed in order to detect classes that are in the data [Ultsch/Siemon 90]. For example, using a data set containing blood analysis values from 20 patients (20 vectors with 11 real-valued components) selected from a set of 1500 patients [Deichsel/Trampisch 85], the following stucture could be seen:

Figure 2: U-Matrix of acidosis data

In figure 2 three major classes casn be distinguished: the upper right corner the left part and the lower right part. The latter two may be subdivided furthermore into tho subgroups each. It turned out that this clustering corresponded nicely with the different patient's diagnoses, as Figure 3 shows.

Figure 3: U-matrix with diagnoses.

In summary, by using this method, stucture in the data can be detected as classes. These classes represent sets of data that have something in common. The discovery of the properties of each class and its reformulation in a symbolic form is described in the next chapter.

4. Rule Generation

As a first approach to generate rules from the classified data (see chapter 3) we used a well known machine learning algorithm: ID3 [QUINLAN 84, Ultsch/Panda 91]. While being able to generate rules this algorithm has a serious problem [Ultsch 91]: it uses a minimalization criterion that seems to be unnatural for a human expert. Rules are generated, that use only a minimal set of decisions to come to a conclusion. This is not was must be done, for example, in a medical domain. Here the number of decisions is based on the type of the disease. In simple cases, i.e. where the symptoms are unanymous, very few tests are made, while in difficult cases a diagnosis must be based on all awailable information.

In order to solve this problem we have developed a rule generation algorithm, called sig*, that takes the significane of a symptom (range of a component value) into account [Ultsch 91]. One of the data sets we tested the algorithm with, was the diagnosis of iron deficiency. We had a test set of 242 pations with 11 cinical test values each. The rules generated with sig* showed, first,a high degree of coincidence with expert's diagnosis rules and, second, exhibited knowledge not prior known to us while making sense to the experts [Ultsch 91].

We have tested this algorithm in another way. The data set was randomly divided into two subsets and only one of them was used to generate rules. The other subset was used to compare the diagnosis generated by the rules with the actual diseases.

disease
#
patients
correct diagnoses
wrong
diagnoses

normochr.
23
23
0

hypochrome
15
15
1

Eisenmangel
22
20
2

echter EM
16
16
0

chron.Prozeß
1
1
0

sideroachr.
1
1
1

hyperchr.
5
5
1

Polyglobul.
0
0
0

o.B.
38
38
6

sum
121
119
11

Figure 5: sig*-generated diagnoses vs. true diagnoses

In 119 out of 121 cases (98%) the system produced the right diagnosis. In eleven cases a wrong diagnosis was given. Most of this cases were additional wrong diagnosis. For example in six cases "o.B."-patients (meaning healthy) were also cosidered to have a slight form of iron deficiency. In other cases, like "sideroacrestic", the data set that has been learnned by the network was to small (one case) in order to produce a meaningful diagnostic rule.

5. Conclusion

The implementation of our system demonstrates the usefulness of the combination of a rule-based expert system with neural networks [Ultsch et al. 91b,a, Ultsch/Panda 91]. Unsupervised leaning neural networks are capable to extract regularities from data. Due to the distributed subsymbolic representation, neural networks are typically not able to explain inferences. Our system avoids this disadvantage by extracting symbolic rules out of the network. The acquired rules can be used like the expert's rules. In particular it is is therefore possible to explain the inferences of the connectionist system.

Such a system is useful in two aspects. First the system is able to learn from examples with a known diagnosis. With this extracted knowledge it is possible to diagnose new unknown examples. Another abilityis to handle a (large) data set for which a classification or diagnosis is unknown. For such a data set classification rules are proposed to an expert. The integration of a connectionist module realizes "learning from examples". Furthermore the system is able to handle noisy and incomplete data. First results show that the combination of a rule based expert system with a connectionist module is not only feasible but also useful. Our system is one of the possible ways to combine the advantages of the symbolic and subsymbolic paradigms. It is an example to equip a rule based expert system with the ability to learn from experience using a neural network.

Acknowledgements

We would like to thank all members of the student research groups PANDA (PROLOG And Neural Distributed Architectures) for their tremendous job in implementing a preliminary version of the system. H.-P. Siemon has implemented the Kohonen algorithm on a transputer.

This work has been supported in part by the Forschungspreis Nordrhein-Westfalen.
References

[CARPENTER/GROSSBERG 87] CARPENTER, G.A., GROSSBERG,S.: Self-Organization of Stable Category Recognition Codes for Analog Input Patterns, Applied Optics, Vol. 26, S. 4.919-4.930, 1987

[Deichsel/Trampisch 85] Deichsel G., Trampisch H.J.:Clusteranalyse und Diskriminanzanalyse,Gustav Fisher Verlag, Stuttgart 1985

[Kohonen 84] Kohonen T.: Self-Organization and Associative Memory, Springer Verlag, Berlin 1984

[Michalski et al. 84] Michalski R., Carbonell J.G., Mitchell T.M.: Machine Learning - An artificial intelligence approach,Springer Verlag, Berlin 1984

[Mitchell 82] Mitchell T.M.: Generalization as Search, in: Artificial Intelli-gence 18, p. 203-226

[Quinlan 84] Quinlan J.R.: Learning Efficient Classification Procedures and their Application to Chess End Games in: [Michalski et al. 84]

[Rumelhart/McClelland 86] Rumelhart D.E.,McClelland,J.L.:Parallel Distri-buted Processing: Explorations in the Microstructure of Cognition, Volume 1: Foundations, MIT Press, Cambridge (Massachusetts) 1986

[Rumelhart/Zipser 85] Rumelhart D.E., Zipser D.: Feature discovery by Com-petitive Learning, in: Cognitve Science (1985) Vol. 9, pp.75 - 112

[Siemon/Ultsch 90] Siemon,H.P., Ultsch,A.: Kohonen Networks on Transputers: Implementation and Animation, Proc. Intern. Neural Networks, Kluwer Academic Press, Paris, 1990, pp 643 - 646

[Ultsch 91] Ultsch, A.: Konnektionistische Modelle und ihre Integration mit wissensbasierten Systemen, Habilitationsschrift, Univ.Dortmund, 1991.

[Ultsch et al. 91b] Ultsch, A., Halmans, G., Mantyk, R.: CONCAT: A Connectionist Knowledge Ackquisition Tool, Proc. IEEE International Conference on System Sciences, January 9-11, Hawaii, 1991, pp 507 - 513.

[Ultsch et al. 91a] Ultsch, A.,Halmans,G.,Mantyk, R.:A Connectionist Knowledge Acquisition Tool: CONCAT, Proc. International Workshop on Artificial Intelligence and Statistics, January 2-5, Ft. Lauderdale FL, 1991.

[Ultsch/Panda 91] Ultsch, A., Panda, PG.: Die Kopplung konnektionistischer Modelle mit wissensbasierten Systemen, Tagungsband Expertenystemtage Dortmund, Februar 1991 VDI Verlag, pp 74-94.

[Ultsch/Siemon 90] Ultsch, A., Siemon, H.P.: Kohonen's Self Organizing Feature Maps for Exploratory Data Analysis, Proc. Intern. Neural Networks, Kluwer Academic Press, Paris, 1990, pp 305 - 308