Deriving Qualitative Rules from Neural Networks in Environmental Science – Preliminary Report[1]

Franz Wotawa[2] and **Gerhard Wotawa**[3]

Abstract. As alternative to physical models, neural networks are a valueable forecast tool in environmental sciences. They can be used effectively due to their learning capabilities and their low computational costs. As far as the relevant variables of the system are measured and put into the network, it works fast and accurately. However, one of the major shortcomings of neural networks is that they do not reveal causal relationships between major system components and thus are unable to improve the explicit knowledge of the user. To overcome this problem, we introduce a novel approach for deriving qualitative informations out of neural networks. Some of the resulting rules can be directly used by a qualitative simulator for producing possible future scenarios. Because of the explicit representation of knowledge the rules should be easier to understand and can be used as starting point for creating models wherever a physical model is not available. We illustrate our approach using a Network for predicting surface ozone concentrations and discuss open problems and future research directions.

1 Introduction

Artificial Intelligence has been successfully applied to solve problems regarding physical and meteorological systems. Application areas include explanation of physical systems, forecasting, decision support, and modeling. The problem of giving meaningful explanations of the behavior of a physical system has been tackled by qualitative reasoning (QR) [16, 1]. Several approaches for representing physical systems in a qualitative way has been proposed. The most influencing techniques are qualitative simulation [8], confluences [2], and qualitative process theory [3]. Although, one of the goals of QR was to develop models allowing to reason about physical system and consequently to give a program some kind of physical problem solving capability, qualitative reasoning systems are also used in other areas, e.g., forecasting the qualitative behavior of a physical system. One requirement for using QR is the existence of a qualitative model. Sometimes such a model can directly be derived from a quantitative model, e.g., differential equations. However, such a model of a physical system is not always accessible, especially in domains where even experts in the field have no complete theory about the behavior and nature of a system and where observation data and their interrelationships can not be interpreted so far.

In the cases where no model is available, a partial model representing parameters of interest and their interaction has to be developed. Such a model needs not to capture all aspects of the system and is not indented to be a complete physical explanation of a phenomenon. Instead, the model must allow for deriving useful informations for solving a special task. In [5, 6] a model used by a decision support system has been introduced. [12] introduces a qualitative model for predicting severe rainfall events. Although, such models can be developed, modeling is not easy and time consuming. In some situations, when observations of the physical system are available, machine learning [11] can be applied to derive a model out of observational data. The resulting model is not necessarily a qualitative one. Several machine learning techniques have been proposed so far, including the induction of decision trees, rules, or neural networks.

Whenever AI forecast models are used in meteorology and environmental research, most frequently the neural network approach is chosen. In [7] the use of neural networks to prediction and data analyses in meteorology and oceanography has been suggested and compared with other empirical and statistical methods. There is also work [15] describing the induction of meteorological knowledge from observations in form of decision trees. An advantage of getting rules or a decision tree is that they give some deeper insights of the observations helping a human to understand interrelationships between model parameters. Neural network, however, have a implicit characterization of the derived knowledge. If they are only used for forecasting purposes this poses no problem. It has been shown in many areas that the forecast results obtained from neural networks may be as well or even better than results gathered by applying physical models, e.g., [17]. However, if connections are of interest the meteorological model provides more profound knowledge. To overcome this problem techniques for deriving rules out of trained neural networks have been proposed [14]. The result of rule extractions out of neural networks are rules of the form $X_1 \wedge \ldots \wedge X_n \stackrel{cf}{\rightarrow} O_j$ for networks with binary values, or $C_1 \wedge \ldots \wedge C_n \stackrel{cf}{\rightarrow} O_j$ with $C_j \equiv X_j \mathrm{op} \mu_j$ with op $\in \{<, >, \leq, geq\}$ in the general case. For both forms X_i denotes network inputs and O_j is an output node. The certainty factor cf is a measure to what extent the rule is valid.

Different to the output of traditional rule extraction techniques we are more interested in receiving qualitative models from neural networks. The qualitative models we have in mind describe relations between inputs and outputs not using quantities. An example is the rule *If the temperature increases then the ozone concentration increases as well (provided that all other parameters are stable)* or *Traffic and high temperature cause a high ozone level*. Both examples describe causal relationships between considered parameters. Although, such rules do not provide a quantitative value they allow for explaining connections and hence explaining how physical systems work. The

[1] The work described in this paper was partially funded by the Austrian Science Fund project N Z29-INF.

[2] Technische Universität Wien, Institut für Informationssysteme and Ludwig Wittgenstein Laboratory for Information Systems, Favoritenstraße 9–11, A-1040 Wien, Email: wotawa@dbai.tuwien.ac.at

[3] Universität für Bodenkultur, Insitut für Meteorologie und Physik, Türkenschanzstraße 18. A-1180 Wien, Email: wotawa@boku.ac.at

question now is how to derive such qualitative rules out of networks. In this paper we try to provide an answer by introducing an algorithm for this purpose.

There are other approaches for deriving qualitative knowledge from data. Sangüesa and Cortés [13] survey research done in deriving causal networks from data and introduce new algorithms. The question whether our approach or other approaches for learning qualitative knowledge perform better can not be answered at this stage of research. Future research must gain more experience by using different example data sets and directly comparing the outcome of the approaches.

This paper is organized as follows. In the next section we give some basic definitions regarding neural networks and qualitative reasoning. This section is followed by a section introducing the qualitative rule extraction algorithm and a section presenting some results when applying the algorithm to a real-world example. Finally, we discuss related work and future directions and conclude the paper.

2 Basic definitions

Like qualitative models of physical systems quantitative models (V, D, CM) are characterized by a set of variables $V = \{X_1, \ldots, X_n\}$ each variable X_i associated with a domain $d_i \in D$ and some constraints or rules CM describing the connections between variables. Differently from the quantitative case the domain for qualitative variables is finite, i.e., $dom_i = \{V_1, \ldots, V_m\} \in DOM$, and is totally ordered $V_1 < \ldots < V_m$. Since neural networks take input values from an infinite domain for computing output values, we have to define a mapping from the infinite domains used by neurons and the finite domain used in qualitative reasoning. For this purpose we assume a set of distinct intervals $I_i = \{h_1, \ldots, h_m\}$, with $\forall_{l,k} I_l \cap I_k = \emptyset$, for every variable $X_i \in V$ where each m_j corresponds directly to a value from the qualitative domain dom_i of variable X_i, i.e., h_1 corresponds to V_1,..., h_m corresponds to V_m.

In addition, we define a function $\eta_i : \mathbb{R} \mapsto DOM$ mapping values from the quantitative domain to the qualitative domain for each variable X_i. For example a model that predicts ozone concentration typically comprises variables temperature, wind speed, cloud cover and the ozone concentration from the day before $\{temp, speed, cloud, ozone\}$. A qualitative domain for temperature might be $\{very\_cold, cold, warm, hot\}$ where $very\_cold$ corresponds to $(-\infty, 0)$, $cold$ to $[0, 18)$, $warm$ to $[18, 30)$, and finally hot to $[30, \infty)$. Temperature values are given in $^\circ C$. The temperature 16° is mapped to $\eta_{temp}(16) = cold$.

The rules CM describing the behavior of the system represents causal relationships, e.g., *A high temperature coincides with a high ozone concentration*. Formally, such a rule can be stated by $(hot, temp) \leadsto (ozon, high)$ where $\leadsto$ denotes the causal direction. Generally, we define causal rules as $C \leadsto (O, V)$, where C is a logical formula consisting out of propositions of the form (X_i, V_i) connected by the logical operators $\wedge, \vee, \neg$. In order to allow for describing rules of the form *Increasing temperature coincides with an increase of the ozone concentration* we allow propositions of the form op(X_i) (op $\in \{+, -, \pm\}$ for increasing, decreasing, and stable) to be used in C and as consequent of the rule. Using this proposition we can express a M^+ (M^-) constraint used by [8], e.g., $M^+(X_i, X_j)$ is represented by the rule $+(X_i) \leadsto +(X_j)$ or rule

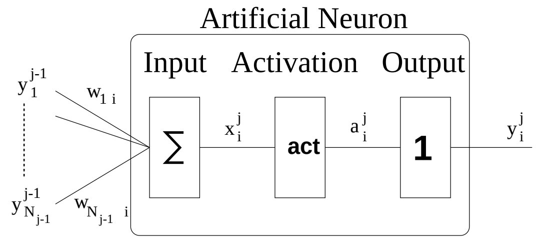

Figure 1. A single Neuron

$+(X_j) \leadsto +(X_i)$. The set of rules CM represents the model of a physical system.

Beside the formalization of the considered qualitative model we give a brief introduction into neural networks and how they are used. In this paper we only consider multi-layer feed-forward networks [4] with one input, one hidden, and one output layer. To be self contained we briefly recall the basics of neural networks. Neural networks can be seen in the sense of an abstract simulation of a biological neural system, such as the human brain. They are made up of many parallel working units, based upon single brain cells or neurons and the connections between them which simulate the axon interlaces of the brain.

The most important part of a neural network is the neuron. Data is processed in the neurons insofar as they accept incoming data, transform it into an activation signal and switch it to the neuron output. Individual neurons are combined in a single network through directed and weighted connections. Such networks take on external input through their own input neurons and propagate these through the neuron connections to the output neurons. Neural networks allow themselves to be placed in models with supervised and unsupervised training algorithms, whereby the data predictions mostly come into effect as supervised approaches.

Multi-layer Perceptrons (MLP) come about through the joining together of multiple non-linear perceptrons (see [4]) and are multilayered feed-forward networks. Figure 1 shows the formal representation of a single neuron used in MLPs, consisting of an input, an activation, and an output function. Usually the input function computes the sum of all inputs using the given weights, i.e., $x_i^j = \sum_{k=1}^{N_{j-1}} w_{ki} y_k^{j-1}$ where x_i^j denotes the i-th neuron in the j-th layer, w_{ik} the weight between the neuron i and k, and N_{j-1} the number of neurons in layer $j-1$. In most cases identity is used as the output function. Therefore this function is often ignored (as it is in our case). The activation function act takes the input value and computes the output value. The most popular activation function is the sigmoid function $y_i^j = \frac{1}{1+e^{x_i^j}}$ where y_i^j denotes the output value of the i-th neuron in the j-th layer. MLPs are normally trained through the Backpropogation algorithm by modifying the weights between the neurons.

Learning an MLP is usually done using the backpropagation algorithm (BP). The BP algorithm tries to minimize the output error function of a network by adapting the weights of the network connections in the direction of its negative gradient. The error function is therefore half the square of the network output error compared to the desired target output: $F = \frac{1}{2} \cdot \sum_i (t_i - y_i)^2$ where index i denotes the cells of the output layer, t_i represents the target output, and y_i the actual output of the neural network. The change in the weights runs parallel to the negative gradient of the error function $\Delta w_{ij} \propto -\frac{\partial F}{\partial w_{ij}}$.

The direct result is the back propagation learning rule distinguishing between different layers: $\Delta w_{ij} = \eta \delta_j y_i$, $\Delta \theta_j = \eta \delta_j$ with

1. **Output layer** $\delta_i^N = act'(x_i^N)(t_i - y_i^N)$
2. **Others** $\delta_i^j = act'(x_i^j) \sum_{k=1}^{N_{j+1}} \delta_k^{j+1} w_{ki}^{j+1}$

where η is the learning rate or learning coefficient and regulates the speed of the convergence of the algorithm, δ_i^j is also known as the error signal of a particular cell, and $act(x)$ is the activation function of the respective neuron. act' denotes the derivation of act.

3 Deriving qualitative rules

Rule extraction out of neural networks has been investigated by several researchers, e.g., [14]). In this paper we introduce an algorithm for extracting qualitative knowledge out of neural networks trained using observations from a physical system. We restrict neural networks to be multi-layer feed-forward networks with the aggregation function as input function, an monotonic increasing function as activation function, and the identity function as output function. Our algorithm can be characterized as black-box approach accordingly to [14]. It takes the trained neural network ANN, the variables $V = \{X_1, \ldots, X_n\}$, the function η_i for every variable X_i ($\mathfrak{E} = \{\eta_1, \ldots, \eta_n\}$), and a function $\Gamma = \{\Gamma_1, \ldots, \Gamma_n\}$ as input and computes causal rules. The function Γ_i returns the set of values for a variable X_i specified by their intervals I_i (corresponding to the values from the qualitative domain DOM_i, i.e., $\Gamma_i = \bigcup_{(lb,ub) \in I_i} \{lb, ub\}$. The values are of interest because they represent the interval boundary values.

Rule extraction is done in a two stage process. First, all possible rules using all combination of input values are computed. The input values of the network are given by the intervals corresponding to the qualitative domains of variables. In the second step the rules are summarized and minimized.

The following $qualitativeRules$ algorithm computes a set of qualitative rules.

Algorithm $qualitativeRules(ANN, V, \Gamma, \mathfrak{E})$

1. Let CM be the empty set.
2. For all elements d of $\Gamma_1 \times \ldots \times \Gamma_k$ given by Γ and where $\forall_{l=1\ldots k} X_l \in inputs(ANN)$, do the following:
   (a) Compute the value v_j of all output neurons $Y_j \in outputs(ANN)$ from ANN using the values d_i from d for all input neurons X_i.
   (b) Add the rule
   $$(X_1, \eta_1(d_1)) \wedge \ldots \wedge (X_k, \eta_k(d_k)) \leadsto (Y_j, \eta_j(v_j))$$
   to the set CM.
3. Return the set of rules CM.

It is easy to see that $qualitativeRules$ halts on every finite input. The termination criteria of the loops must be always met because of the finite input data. The complexity of the algorithm is of order $O(d^k)$, where d is the size of the largest domain and k is the number of input neurons. The space requirements for storing the resulting set of rules is also equal to $O(d^k)$. Although, the time and space requirements are high this poses no practical problem. First, the conversion process of the neural network into a set of qualitative rules can be done off-line after training the network. Hence, the use of the rules is separated from their computation. A second reason is a more pragmatic one. Since, the used networks comprises not so many neurons and the available computational power is high and still increasing, computation should be possible without any problem. However, as discussed previously the user is more interested in receiving deep knowledge, i.e., a small set of concise rules giving an explanation about the system's behavior.

Rule reduction 1: Reducing rules or combining them for a more compact representation can be done by applying logical considerations. Rules generated by $qualitativeRules$ can be seen as implications on propositions. So, we can use all techniques from propositional logic in order to reduce the number of implications. For example if we have two implications of the form $p \wedge R \rightarrow q$ and $\neg p \wedge R \rightarrow q$ with propositions p,q and a conjunction of propositions R, then we can combine them to $R \rightarrow q$. For our purposes this reduction rule must be reformulated. If we have rules of the form $(X_i, v_1) \wedge R \leadsto y, \ldots, (X_i, v_m) \wedge R \leadsto y$, with $\{v_1, \ldots, v_m\} = dom_i$, then we can combine them to $R \leadsto y$. Hence, the reduction rule can reduce m rules to one and retains correctness. However, the number of applications of the reduction rule depend on the given domains dom_i and the function η_i. Therefore, the rule may not be applicable at all. In order to overcome this problem we can relax the condition $\{v_1, \ldots, v_m\} = dom_i$ to $|\{v_1, \ldots, v_m\}| \geq |dom_i| - k$, where $k \in [0, |dom_i|]$ is an error value stating the allowed difference to the original condition.

Rule reduction 2: Another way of computing the relevant rules out of all is to check the probability that a variable having a value leads to the output having another value. The probability can be computed out of the given collection of rules. Formally, the second kind of reduction deals with computing rules of the form $(X_i, v) \overset{cf}{\leadsto} (Y, v')$, where cf denotes the probability. cf is defined by

$$cf =_{def} \frac{|\{r | r \equiv (\ldots \wedge (X_i, v) \wedge \ldots \leadsto (Y, v')), r \in CM\}|}{|\{r' | r' \equiv (L \leadsto (Y, v')), r' \in CM\}|}.$$

Using this definition we receive a set of rules M that have to be further minimized. The first reduction on M uses the following observation. Because, of the rule definition we know that cf represents the probability that the associated rule is correct, and that there are rules $r_1 \equiv (X_i, v_i) \overset{c}{\leadsto} (Y, v)$ and $r_2 \equiv (X_i, v_i) \overset{d}{\leadsto} (Y, v')$ with $v \neq v'$ in M. We can specify preference criteria for such rules r_1 and r_2. We say that r_1 is preferred over $r2$ ($r_1 > r_2$) iff $c > d$. Using this definition we reduce M by only considering preferred rules with respect to $>$. The second reduction makes sense only if a not to small bound for the probability value cf is used (preferable between 0.8 and 1).

Rule reduction 3: The third and last technique for deriving qualitative rules out of neural networks is different from the previous ones. Instead of deriving static knowledge, the third technique is for deriving rules specifying the dynamic behavior of a system. Using our formalism we search for rules of the form $+(X_i) \rightsquigarrow +(X_j)$ indicating that if X_i is increasing (decreasing) then X_j is increasing (decreasing) as well, or $-(X_i) \rightsquigarrow +(X_j)$ if there is an inverse relationship between variables X_i and X_j. For the following definitions we assume that $dom_i : v_i^1 < \ldots < v_i^k$ is the totally ordered domain of variable X_i having k distinct intervals as elements. Now, we can define the case where a rule $+(X_i) \rightsquigarrow +(X_j), i \neq j$ hold. Let CM be the (non-reduced) set of rules obtained from the neural net ANN. The rule $+(X_i) \rightsquigarrow +(X_j)$ can be derived from CM iff for all conjunctions of propositions L and R and the rules $L \wedge (X_i, v_i^l) \wedge R \rightsquigarrow (X_j, v_j^o)$ and $L \wedge (X_i, v_i^p) \wedge R \rightsquigarrow (X_j, v_j^q)$ in CM with $l < p$ follows that $o \leq q$. A similar definition holds for $-(X_i) \rightsquigarrow +(X_j)$ (only $l < p$ must be changed to $l > p$).

A rule $+(X_i) \rightsquigarrow +(X_j)$ may not be valid for all but for the most cases. To reflect this we relax the definition above and introduce a probability value cf, as done previously. The probability value is computed by dividing the number of cases where the rule hold with the number of all possible cases, i.e., the number of rules, i.e., $|CM|$. We write $+(X_i) \overset{cf}{\rightsquigarrow} +(X_j)$. Again we can reduce the number of rules having a probability value by only considering rules with a probability greater than a given boundary value, e.g., 0.9.

Summary: All three techniques proposed above for generating rules and reducing rule sets require the set of rules computed by $qualitativeRules$ as input. The first technique searches for elements within the antecedence of a rule that can be eliminated because of the fact that they do not influence the output. The result is a set of rule that is logically equivalent to the original set. The second technique is for computing rules that are only true to some extent. This technique computes rule of the form *If a value is within an interval, then the output is within another interval with a given probability*. The final technique detects functional relationships between inputs and outputs. The results of this technique can be used as constraints ($M+$ or $M-$) for a qualitative simulator [8].

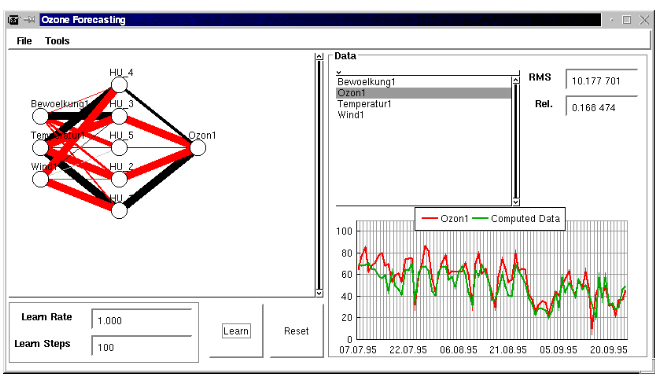

Figure 2. The user interface of the Ozone forecasting application

4 Results

We have implemented the described rule generation techniques in Smalltalk using VisualWorks 2.52 and applied to the prediction of surface ozone concentrations [17]. The user interface of our application is depicted in figure 2. **The performance results obtained are depicted in Table 1. Time is given in seconds and was measured on a Sun Ultra 1 workstation with 145 MHz under Solaris. We see that computing all rules was not very time consuming but the first rule reduction technique was. However, the results indicate that the proposed methods are applicable for larger neural networks.**

	Time [Sec]
Derive rules	0.5
Rule reduction 1	8.7
Rule reduction 2	1.1
Rule reduction 3	1.4

Table 1. Performance results for the ozone domain

The following rules were derived from a simple feed-forward network with 3 input, 5 hidden, and 1 output neuron. The network was trained on data from [17] specifying a learning rate of 1 and 100 training cycles using the available data from July to Septembers 1995. The quantitative domains of wind speed, temperature, cloud cover, and ozone were divided into five intervals (see figure 3). According to the complexity considerations of the algorithm $qualitativeRules$ we obtain 125 causal rules CM. In the first reduction step the rules were reduced to 111. This reduction was based on logical arguments without introducing uncertainty. By allowing a small deviation of one (two) to the original condition we received 83 (57) different rules. One obtained rule said that *If the cloud cover is small or medium and the temperature is high, then the ozone concentration is high as well*.

Using the same basic rule set CM and the second reduction technique we obtain the following rules with given certainty factors.

$$
\begin{gathered}
(temperature, low) \overset{0.76}{\rightsquigarrow} (ozone, low) \\
(temperature, low\_mid) \overset{0.48}{\rightsquigarrow} (ozone, low\_mid) \\
(temperature, mid\_high) \overset{0.56}{\rightsquigarrow} (ozone, low\_mid) \\
(temperature, high) \overset{0.68}{\rightsquigarrow} (ozone, mid) \\
(temperature, max) \overset{0.64}{\rightsquigarrow} (ozone, mid) \\
(wind, low) \overset{0.44}{\rightsquigarrow} (ozone, mid)
\end{gathered}
$$

It is interesting to note that the maximum ozone concentration is never mentioned. This indicates that no significant rule between maximum ozone and the considered meteorological parameters exists, and that maximum ozone is rather influenced by parameters not covered by our reductionist model. Note that other rules with a smaller certainty factor could be derived but are less relevant.

The final reduction technique delivering rules stating functional relationships delivered three results.

wind	[1.0,3.25),	low,
	[3.25,5.5),	low_mid,
	[5.5,7.75),	mid_high,
	[7.75,10.0)	high,
	[10.0,∞)	very_high
cloud	[0.0,0.25),	low,
	[0.25,0.5),	low_mid,
	[0.5,0.75),	mid_high,
	[0.75,1.0)	high,
	[1.0,∞)	very_high
temperature	[250,275),	low,
	[275,300),	low_mid,
	[300,325),	mid_high,
	[325,350),	high,
	[350,∞)	very_high
ozone	[0.0,25.0),	low,
	[25.0,50.0),	low_mid,
	[50.0,75.0),	mid_high,
	[75.0,100.0),	high,
	[100.0,∞)	very_high

Figure 3. Intervals and qualitative values for input variables

$$\begin{gathered} +(temperature) \rightsquigarrow +(ozone) \\ +(wind) \rightsquigarrow -(ozone) \\ +(cloude\_cover) \rightsquigarrow -(ozone) \end{gathered}$$

This result gives a good characterization of the influences between different input parameters. We see that the temperature influences the ozone concentration positively, while the other parameters have a negative influence. This result corresponds exactly to the observations. Increasing wind speeds prevent the accumulation of pollutants near the surface. An increase in cloud cover reduces the photochemically active radiation and thus ozone production.

5 Discussion

The proposed approach tries to get knowledge about physical system out of a trained neural network. In difference to previous research [14] in this direction the goal is not to get rules out of the network with the capability of replacing the network by a deduction system using the rule for computing output values based on given inputs. Instead we are interested in getting rules helping to understand physical systems. Research in qualitative reasoning has the aim of providing techniques for representing knowledge about physical systems that allow to reason about the world. Hence, our rules are intended to be qualitative (instead of quantitative). It is worth noting that the rules can not be derived directly from the available data. There are several arguments for that assumption. One argument is that the data is not complete meaning that there are situations explainable by a physical or ecological theory that are never observed so far. Furthermore observations are inaccurate due to measurement errors. All this finally suggests the use of neural networks. Because of the generalization capability of neural networks it is not necessary that all possible cases are to be used for the training session. The advantages of neural networks in meteorological applications have been discussed in detail in [7].

The qualitative rules and the techniques for compacting these rules as described here allow to gain more profound knowledge about physical systems. Because of their declarative nature they are understandable and can be used for deriving new knowledge, i.e., possible scenarios based on a given situation similar to qualitative simulation [8]. Consider our ozone forecasting example. If the temperature is increasing and wind and cloud cover is stable then we can derive that the ozone concentration is increasing as well. If wind speed is increasing several scenarios are possible. First, the ozone concentration is increasing as well. Second, the ozone concentration is stable, or finally, the ozone concentration is going to decrease. Using this example, we see that in difference to a quantitative simulation the result becomes ambiguous. However, a quantitative results require precise data which are not always available. This holds for ozone forecasting where weather predictions for temperature, wind speed, and cloud cover are used which are itself uncertain [17]. So, the result must be uncertain. Since, most ecological systems are very sensitive on input data, only slightly different inputs may lead to different outputs. Therefore, it is impossible to guarantee that real physical or ecological system will behave as expected. Additionally, for many applications, qualitative statements like *the surface ozone concentration is going to increase tomorrow* are enough. Furthermore, the ideas of deriving rules from data as described herein are especially interesting in situations where no explicit physical model is available to explain the observations.

Note that the results of our approach depend on several parameters: First, the choice of the qualitative domain and the intervals corresponding to qualitative values have an impact on the resulting rules. They should be carefully chosen depending on the given application. Second, the resulting knowledge obtained by the introduced methods, depends on the used neural network. Missing parameters which corresponds to a physical phenomenon, result in an incomplete representation of the real situation. However, this is a general problem. Inavailability of data also results in a incomplete or inaccurate model when using other learning techniques.

The paper is intended to give preliminary definitions and results in this field in order to foster research. There are many problems regarding our approach. The first open question is whether and to what extent it is possible to derive knowledge about systems dynamics. Many physical systems have a more or less important memory, i.e., values of system variables at a time point t depend on previous values at $t-1$ and so on. The second question is whether the approach can be extended to be applicable for other kind of neural networks, e.g., elman networks. Finally, we do not give a precise formal semantics of the resulting rules. Should they be treated as causal rules [9, 10] or as pure implications (with certainty factors). Although, answers to the open questions are required, we think that the proposed technique and the underlying idea is still of valid for the ecological domain where explicit models are hardly available but observations are. Possible application areas include decision support systems and helping researchers structuring knowledge. Deriving knowledge for decision support systems from available data can help to decrease costs for developing such systems and improve their output. The second application is to provide preliminary models of physical or ecological system that are not so well understood by humans.

6 Conclusion

In this paper we have given definitions and algorithms for deriving qualitative knowledge out of neural networks. We argued that the resulting rules can be seen as preliminary model of a physical or ecological system. The rules are automatically derived from the neural net and therefore no extra costs are required. Since, learning of neural networks is also automatically performed the user is only required to determine the inputs and needed outputs of the system. Therefore, the techniques as suggested are especially useful whenever no physical model is available to explain observational data. However, the quality of the output depends on the input. If an important input factor is missing, then the outcome can not be expected to represent a real model of the system. This, however, is rather a general problem of developing scientific theories out of limited and uncertain information then a shortcoming of our approach. The results of this paper should be taken as preliminary. Future research should include extensions regarding deriving knowledge about the behavior of system evolving over time, considering other types of neural networks, improving the used algorithms, and other. We believe that our approach can be especially useful for creating a rule base for decision support systems where qualitative results are more important than quantitative ones.

REFERENCES

[1] Philippe Dague, ‘Qualitative Reasoning: A Survey of Techniques and Applications’, *AI Communications*, **8**(3/4), (1995).

[2] Johan de Kleer and John Seely Brown, ‘A qualitative physics based on confluences’, *Artificial Intelligence*, **24**, 169–203, (1984).

[3] Kenneth D. Forbus, ‘Qualitative process theory’, *Artificial Intelligence*, **24**, 85–168, (1984).

[4] Simon Haykin, *Neural Networks – A Comprehensive Foundation*, Prentice Hall, 1999.

[5] Ulrich Heller and Peter Struss, ‘Transformation of Qualitative Dynamic Models – Application in Hydro-Ecology’, in *Proceedings of the 10^{th} International Workshop on Qualitative Reasoning*, (1996).

[6] Ulrich Heller and Peter Struss, ‘Conceptual Modeling in the Environmental Domain’, in *Proceedings of the 15^{th} IMACS World Congress on Scientific Computation, Modelling and Applied Mathematics*, Berlin, Germany, (1997).

[7] William W. Hsieh and Benyang Tang, ‘Applying Neural Network Models to Prediction and Data Analysis in Meteorology and Oceanography’, *Bulletin of the American Meteorological Society*, **79**(9), 1855–1870, (1998).

[8] Benjamin Kuipers, ‘Qualitative simulation’, *Artificial Intelligence*, **29**, 289–388, (1986).

[9] F. Lin, ‘Embracing causality in specifying the indirect effects of actions’, in *Proceedings* 14^{th} *International Joint Conf. on Artificial Intelligence*, Montreal, Canada, (August 1995). Morgan Kaufmann.

[10] N. McCain and H. Turner, ‘Causal theories of action and change’, in *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, (1997).

[11] Tom M. Mitchell, Paul E. Utgoff, and Ranan Banerjii, ‘Learning by experimentation: Acquiring and refining problem-solving heuristics’, in *Machine Learning — An Artificial Intelligence Approach*, eds., Ryszard S. Michalski, Jaime G. Carbonell, and Tom M. Mitchell, chapter 6, 163–190, Morgan Kaufmann, (1983).

[12] Satoru Oishi and Shuichi Ikebuchi, ‘Inference of local rainfall using qualitative reasoning’, in *Proceedings of the 10th International Workshop on Qualitative Reasoning, AAAI Technical Report WS-96-01*, pp. 181–189, (1996).

[13] Ramon Sangüesa and Ulises Cortés, ‘Learning causal networks from data: a survey and a new algorithm for recovering possibilistic causal networks’, *AI Communications*, **10**, 31–61, (1997).

[14] Ismail A. Taha and Joydeep Ghosh, ‘Symbolic Interpretation of Artificial Neural Networks’, *IEEE Transactions on Knowledge and Data Engineering*, **11**(3), 448–463, (1999).

[15] Walter Van de Velde, ‘Incremental Induction of Topological Minimal Trees’, in *Machine Learning: Proceedings of the Seventh International Converence*, Austin, Texas, (June 1990).

[16] *Readings in Qualitative Reasoning about Physical Systems*, eds., D. Weld and J. de Kleer, San Mateo, 1989.

[17] Dominik Wieland and Franz Wotawa, ‘Local Maximum Ozone Concentration Prediction Using Neural Networks’, in *AAAI-99 Workshop on Environmental Decision Support Systems and Artificial Intelligence (W7)*, (1999).