Proceedings of the 1988 Beijing International Workshop on Information Theory,
Beijing, China, July 1988

Information theory, expert systems and neural networks

Rodney M.F. Goodman and Padhraic Smyth
Department of Electrical Engineering
California Institute of Technology
Pasadena, CA 91125, USA

1988 Beijing International Workshop on Information Theory

Abstract

In this paper we focus on the problem of measuring the information content of a production rule. We define a rule in probabilistic terms and show that there is no unique way to measure the information content. We examine the information-theoretic properties of two such well-known measures and show clearly that, while one measure is inappropriate, the other possesses the necessary properties for a rule information measure. We outline the advantages of this measure and describe the ITRULE algorithm for automated knowledge-acquisition based on the measure. Furthermore we show how the information content can be used to guide the inference process in an expert system, and to define the connection weights in a neural network.

Background and Motivation

Rule-based expert systems are currently receiving attention for two primary reasons. Firstly, they allow for the representation of expert knowledge in a highly uniform and modular manner, considerably simplifying the task of programming such systems and allowing expert system designers to capture expert judgemental knowledge in the form of conceptually simple rules. The second reason for their widespread use is the considerable experimental evidence in cognitive science supporting rule-based representations as accurate models of the human reasoning process [1,2].

However there is no quantitative formal theory available for rule-based systems. Hence expert system design is currently more of an art-form than a science, relying on qualitative arguments and ad hoc procedures to achieve performance in particular domains. Indeed it can safely be said that while the basic idea of expert systems is well-founded, their implementation is hindered considerably by the lack of a well-defined theory. Our goal is to define a general theoretical framework from which to develop rigorous and consistent techniques for the design of expert systems, in particular leading to new techniques in automated knowledge-acquisition and inference strategies, with parallel neural network implementations.

The information content of a rule

A rule is considered to be of the form

If $\mathbf{Y}$=y then $\mathbf{X}$=x with probability p

where $\mathbf{X}$ and $\mathbf{Y}$ are two random variables with "x" and "y" being values in their respective discrete alphabets. We restrict the right-hand expression to being a single value-assignment expression while the left-hand side may be a conjunction of such expressions. This is the commonly accepted definition of a rule.

Hence we need to measure the information that the event $\mathbf{Y}= y$ yields about the variable $\mathbf{X}$. Let us denote this quantity as $m(\mathbf{X};\mathbf{Y}= y)$. Clearly we require that

$$E_y [\, m(\mathbf{X};\, \mathbf{Y} = y)\,] \;=\; I\,(\mathbf{X};\, \mathbf{Y}).$$

where $I(\mathbf{X};\, \mathbf{Y})$ is the standard measure of average mutual information between $\mathbf{X}$ and $\mathbf{Y}$. Blachman [3] has shown that $m(\mathbf{X};\mathbf{Y}= y)$ is not unique. We examine two such measures, to be referred to as the "i-measure" and the "j-measure".

Disadvantages of the "i-measure"

Consider $i(\mathbf{X};\mathbf{Y}=y)$ where

$$\begin{aligned} i(\mathbf{X};\mathbf{Y}=y) &= H(\mathbf{X}) - H(\mathbf{X}|\mathbf{Y}=y) \\ &= \sum_x p(x)\log(\frac{1}{p(x)}) - \sum_x p(x|y)\log(\frac{1}{p(x|y)}) \end{aligned}$$

We show that this measure contains undesirable properties as a measure of rule information, e.g. it is *not* necessarily non-negative. Among the other less desirable properties is the "information paradox" whereby if the posterior distribution of $\mathbf{X}$ is a permutation of the prior distribution, $i(\mathbf{X};\mathbf{Y}=y) = 0$. This occurs because the "i-measure" measures the decrease in entropy of $\mathbf{X}$ due to the event $\mathbf{Y}=y$. We show that (unlike the non-conditional case), under a conditional constraint, decrease in entropy is *not* equal to the average information recieved.

Advantages of the "j-measure"

The "j-measure" is much more commonly accepted than the "i-measure" as the information that the event $\mathbf{Y}=y$ yields about the variable $\mathbf{X}$ and is defined as

$$j(\mathbf{X};\ \mathbf{Y}=y) \ = \ \sum_x p(x|y).\log(\frac{p(x|y)}{p(y)})$$

(This measure is also known as the conditional mutual information and is a special case of cross-entropy). We show that $j(\mathbf{X};\mathbf{Y}=y)$ posesses many desirable properties, e.g. as a non-negative information measure it is unique. The "j-measure" takes the value $\log\frac{1}{p(x)}$ if $p=1$ (the rule transition probability), and takes the value zero if and only if the *a posteriori* distribution of $\mathbf{X}$ is exactly the same as the *a priori* distribution. Hence the "j-measure" converges to the appropriate quantities in limiting cases.

Average information content of a rule

Henceforth we refer to $j(\mathbf{X};\mathbf{Y}=y)$ as the *instantaneous information* while the *average information content* is defined as

$$J\ (\mathbf{X};\mathbf{Y}=y) \ = \ p(y)\ .\ j\ (\mathbf{X};\mathbf{Y}=y).$$

Note that this measure is an average in the sense there is an implicit assumption that the instantaneous information from the other "Y-terms" is zero, which is consistent with the cognitive science approach to production rules where essentially we can only draw inferences about certain events and not others. We will see later how the concept of average, rather than instantaneous, information is used by the learning algorithm. In an intuitive sense, the average measure relates to the average value of the rule, while the instantaneous measure can be used to rank rules after the event $\mathbf{Y}=y$ has occured.

Properties of the J-measure in terms of rules

The J-measure is the average change in information required to specify the variable $\mathbf{X}$ having learned that $\mathbf{Y}=y$. The J-measure can be used to rank rules in an induction algorithm. In particular it can easily be shown [4] that the measure satisfies the basic requirements of an induction measure as defined by Holland et al. [2], i.e. it supports the basic inductive mechanisms of condition-simplifying generalisation, instance-based generalisation, and specialisation. As an example consider specialisation. Let J_g and J_s be the information content of a general rule and a more specialised version of the same rule, i.e. the same rule with an extra condition. The cognitive science approach uses heuristics to determine which rule of the two is the better. The advantage of using the J-measure is that it implicitly determines the better rule in a quantitative and consistent fashion. Consider an example in the animal domain. The relevant features are "has.wings","can.fly", and "is.a.penguin", and feature

values are restricted to the set $\{true, false\}$. The general rule might be: *If has.wings is true then can.fly is true with probability 0.9*; the more specialised version might be: *If has.wings is true and is.penguin is false then can.fly is true with probability 1.0.* It is not intuitively obvious which rule is "better" on the average. If we specify $prob(can.fly = true) = 0.27$ and $prob(has.wings = true) = 0.3$ then we find that $J_s = 0.47$ bits and $J_g = 0.38$ bits, i.e. the more specialised rule is better. But without the J-measure it would be very difficult if not impossible to rank rules in this fashion. The ability to rank hypotheses forms the basis of an automatic induction algorithm we call ITRULE.

The ITRULE (Information-Theoretic Rule Induction) algorithm takes as input a set of feature vectors (where the N features are restricted to being discrete-valued) and it produces as output a set of K probabilistic rules [5]. The rules are the K most informative rules available from the data as ranked by the J-measure (the parameter K is defined by the user). As previously mentioned, the rules are restricted to conjunctive expressions on the left, and a single expression on the right.

rule ranking	premise clause	conclusion	probability/j-measure
rule(1)	If has.legs = 0	then is.a.snake = 1	with p = 1.0, j = 0.53
rule(2)	If is.a.snake = 1	then has.legs = 0	with p = 1.0, j = 0.53
rule(3)	If has.wings = 1	then is.a.bird = 0	with p = 1.0, j = 0.48
rule(4)	If is.a.bird = 1	then has.wings = 0	with p = 1.0, j = 0.48
rule(5)	If has.legs = 1	then is.snake = 0	with p = 1.0, j = 0.40
rule(6)	If is.a.snake = 0	then has.legs = 1	with p = 1.0, j = 0.40
rule(7)	If is.a.reindeer = 1	then has.antlers = 1	with p = 1.0, j = 0.39
rule(8)	If has.antlers = 1	then is.a.reindeer = 1	with p = 1.0, j = 0.39
rule(9)	If is.dangerous = 1 and is.a.dog = 0	then has.legs = 0	with p = 0.94, j = 0.36
rule(10)	If is.dangerous = 1 and is.a.dog = 0	then is.a.snake = 1	with p = 0.94, j = 0.36

Figure 1 : the 10 most important rules for the animal data

As an example of ITRULE in use we show in figure 1 the rules generated from some simulated data relating binary features of animals. There were nine features in all (e.g. has.wings, is.a.dog, etc) and 1,000 feature vectors were generated. The probabilities of some feature-values (e.g. is.a.snake, is.dangerous) were set higher in order to make the data more interesting. Necessarily the information in the induced rules is restricted by the particular choice of features. Nonetheless the most informative rules are quite consistent with human intuition, e.g. "birds have wings" etc.

Probabilistic inference using the J-measure

In this section we show how a set of rules, as generated by ITRULE, can form an inference model and how the J-measure can guide the use of these rules so that the system can display some form of rational behaviour. We would like our rule-based system to use probabilistic considerations in its search, to backward chain along paths which lend the most support for the hypothesis, to query for attributes which are likely to be of most use in inference, to forward chain selectively and to keep track of context. The optimal solution to this problem is computationally prohibitive — we would choose the best decision (about what to do next) averaged over every possible state of the external environment. We propose, more realistically, to use a greedy strategy. As we shall now outline, the J-measure lends itself to a natural interpretation as an appropriate greedy measure for choosing the best rule, for either forward or backward chaining.

For backward chaining, given a particular goal $\mathbf{X}$, the proposition $\mathbf{Y} = y$ which maximises $J(\mathbf{X}; \mathbf{Y} = y)$ is *by definition* the proposition which provides the most information on average about $\mathbf{X}$. Hence, to continue backward chaining, we choose $\mathbf{Y} = y$ as a sub-goal, and backward chain on that proposition. Keeping a list of ranked J-measures for all rules which contain the current sub-goals (in the consequent clauses) enables the system to locally optimise its backward chaining strategy.

When a probability statement for a particular proposition becomes available from the external environment, we can insert on a forward chaining agenda all rules whose antecedent clause contains this proposition. The rule with the highest J-measure denotes the rule which, if fired, will yield the most information by probability

propagation, i.e., the J-measure can be used for conflict resolution. By combining the forward and backward chaining candidate rules in a single rule agenda we can implement a mixed strategy, where the rule with the highest J-measure, whether forward or backward, is chosen. In this manner the system can determine the best strategy in a dynamic manner.

Probability propagation by forward chaining will cause the J-measures of associated rules to change — hence the J-measures are context dependent. For example, let $\mathbf{X}$ be the goal attribute, and let $\mathbf{Y} = y$ and $\mathbf{Z} = z$ be two possible candidates for backward chaining on $\mathbf{X}$. Let $J(\mathbf{X}; \mathbf{Y} = y)$ be the higher J-measure. Backward chaining on $\mathbf{Y} = y$ we may find some piece of evidence $p(\mathbf{E} = e) = 1$ which, when forward chained, results in $p(\mathbf{Z} = z) = 0$. $J(\mathbf{X}; \mathbf{Z} = z)$ will become zero with the result that the search strategy will no longer consider $\mathbf{Z} = z$ as a backward chaining candidate in the context of knowing $\mathbf{E} = e$.

Parallel inference using a neural network

We note that the rules derived by ITRULE form a network or graph. There is thus a striking similarity between this network and the interconnections of a connectionist machine or neural network. The nodes or "neurons" in the neural network correspond to the propositions or attributes in our model. The "activation" of a node may be interpreted as the degree of belief in the proposition. A rule in our model is equivalent to the "links" between neurons. The "weight" of a particular link is indicative of the influence of a LHS neuron on the RHS neuron, given that the LHS neuron is fired or activated to some degree. Thus we can interpret these weights as resembling the J-measures of our model. The "activation" recieved by the RHS neuron is the product of the LHS neuron activation and the corresponding link weight value. Each RHS neuron updates its output activation by summing the incoming activations and thresholding these according to a suitable activation function. This corresponds to local inference with a conditional independence assumption. To perform parallel inference we input some new activation value for one or more of the propositions and allow the network to "relax" to a new set of activations which represent our posterior beliefs in the propositions. Intuitively, we see the nodes exchanging "information" by firing rules. As each neuron updates locally and independently of all the others we have true parallel inference.

Conclusions

In this paper we have outlined a framework for the use of information theory in expert systems design. We have shown a coherent pathway all the way from raw data, through automated rule induction, to probabilistic inference, and a fast neural network implementation. There are many subtleties to this framework, and these are the subject of our on-going research.

Acknowledgements This work was supported in part by Pacific Bell, and by Caltech's Program in Advanced Technologies, sponsored by Aerojet General, General Motors, and TRW.

References

1. A.Newell and H.A.Simon, *Human Problem Solving*, Englewood Cliffs, N.J : Prentice Hall, 1972.
2. J.H.Holland, K.J.Holyoak, R.E.Nisbett, P.R.Thagard, *Induction: Processes of Inference, Learning and Discovery*, Cambridge, MA : MIT Press, 1986.
3. N.M.Blachman, 'The amount of information that y gives about X', *IEEE Trans. Information Theory*, vol.IT-14, no.1, Jan. 1968, pp.27-31.
4. R.M.F.Goodman and P.Smyth, 'Information measures for induction', 1988 International Symposium on Information Theory, Kobe, Japan, June 1988.
5. R.M.F.Goodman and P.Smyth, 'Information Theoretic rule induction', *Proceedings of the 1988 European Conference on Artificial Intelligence*, Munich, 1988.