THE PRINCIPLE OF MULTIPLE KNOWLEDGE

INFORMATICA 2/91

Keywords: knowledge representation, learning, classification

Matjaž Gams and Viljem Križman
Jožef Stefan Institute
Jamova 39, Ljubljana

ABSTRACT The principle of multiple knowledge is presented and it's implications are analyzed in real-life classification tasks. Empirical measurements, simulated computer models and analogy with humans strongly indicate that better classification accuracy is obtained when constructing and using multiple knowledge bases instead of one knowledge base alone.

POVZETEK V članku je predstavljen princip mnogoterega znanja in njegov pomen v realnih klasifikacijskih domenah. Empirične meritve, simulirani računalniški modeli in primerjava z ljudmi kažejo, da je v splošnem klasifikacijska točnost več baz znanja boljša kot najboljše baze izmed njih.

1 Introduction

Real-life domains are characterized by nonexistence of exact computational model and consequently, traditional computing approach is often inadequate. Expert systems enabled a significant step ahead yet different studies, e.g. (Keyes 89), report that they are successful only when the application is relatively small, simple, well understood and when experts agree with each other. When coping with more difficult problems, existing ES methodology lacks mechanisms for dealing with

- nonexistence of a single compact body of encodable knowledge,
- nonexistence of isolated static body of knowledge, independent of time, related events and environments,
- nonexistence of perfect exact single algorithm for applying the knowledge.

It is surprising that AI literature hardly mentions problems with multiple [1], contradictory and redundant knowledge with the purpose to increase performance by exploiting these properties. Recent keyword search of 50.000 abstracts from AI literature over 10 years (Clark 90) indicates that only a few refer to this problem. Indeed, computer methods regularly tend to use only one single body of knowledge in the form of one computer procedure.

People usually take quite different approach in real life. They tend to form expert groups, verify results independently and cross-check information in order to find the best solution. Practical experiences show that such approach in general produces better results than when relying on one man or one source of knowledge alone. In other words - people inherently and successfully use multiple knowledge in most of difficult tasks without paying much attention to that phenomenon.

2 Related work

Many well known AI systems like DENDRAL, MYCIN, PROSPECTOR and DIPMETER already enable the use of multiple knowledge to a certain degree. Multiple knowledge is also present in qualitative modeling (Murthy 88), e.g., when combining qualitative and quantitative models, and in the second generation expert systems.

Probably the most relevant work regarding multiple knowledge was reported in empirical learning or learning from examples. Different authors argue that a

[1] In this paper we don't distinguish between multiple methods, multiple systems, multiple knowledge or multiple knowledge bases.

proper use of multiple knowledge enables better possibilities to analyze the domain and better classification accuracy. Catlett & Jack (87) reported of important increase in classification accuracy in several domains when one knowledge base in the form of a decision tree was constructed for each class instead of classifying all classes with one tree. Similarly, Buntine (89) reported of important increase when 10 different decision trees were designed each time with different attribute at the root. Another interesting approach was taken by Schlimmer (87) enabling multiple view on the same domain by using different algorithms each with differently preprocessed input data. Brazdil & Torgo (90) used different data and different algorithms and achieved very significant improvements. Slightly different approach was taken in GINESYS (Gams 78; Gams, Drobnič & Petkovšek 91) where two methods were applied, one AI and one statistical, on the same data and combined together.

In the process of knowledge acquisition different views of experts sometimes result in construction of multiple knowledge bases (Boose et al 89; Clark 90). Other systems like BLIP (Emde 89) are capable of representing several competing hypotheses.

3 The principle of multiple knowledge

The principle of multiple knowledge is shown at the level of the basic definition of learning (Charniak & McDermott 85): Learning can be seen as modifying knowledge base according to the experience in the past for the use in the future. Performances of a knowledge base are measured on specific tasks to evaluate the successfulness of learning. The improvement over older definitions is based on the recognition that learning can be successful only when much is already known. This learning schema can be further modified according to the way how humans use multiple knowledge in real-life tasks:

	LEARNING		APPLYING	
old multiple knowledge bases	$\Longrightarrow$	modified multiple knowledge bases	$\Longrightarrow$	

On the basis of modification of the learning schema the principle of multiple knowledge (further on referred also as Principle) can be defined:

In order to achieve better performances it is generally better to construct and use multiple knowledge bases than one knowledge base alone, as long as they reasonably cooperate.

The emphasis in the definition of the Principle is on the word **reasonable**. Each multiple method should have as good performance as possible and methods should be multiple to a reasonable degree. The Principle should be understood in the following way: For a difficult real-life problem the user can in most cases obtain better results if instead of one method several methods are used together. Practical experiences and theoretical models described in the paper can provide some useful advice, however, the burden of finding intelligent combination of methods in a specific application is left to the user and is domain dependent.

Attempts to verify the advantages of multiple knowledge were performed in three areas, with:

- simulated models,
- empirical measurements on two real-life domains and
- an experiment on the Winston's arch problem.

4 Models of multiple knowledge

Multiple methods were simulated and analyzed with computer models simulating a general classification process. Typically, 10000 classification tasks were generated and average accuracy was measured as the percentage of correct predictions. Each method classified with a predefined average accuracy which was nearly always set to 0.50. By default, all methods were completely independent of each other in the sense that the probability of correct classification by any method was independent of predictions of all other methods. No actual description of the given task was given, rather, each method classified as a random generator with additional specifications.

Each method also estimated confidence factor of its classification. Confidence factor was a real number between 0 and 1 and was generated randomly with a predefined specifications. It didn't necessary represent probability, meaning there was no constraint on the sum of confidence factors of all predictions. The default value was 0.50. Classification of methods combined together was performed on the basis of confidence factors. There were two basic classification schemata:

- **best-one**, where the method with the best confidence factor was chosen for classification and
- **majority**, where all methods added their confidence factors for and against correct prediction.

Best-one schema classified correctly if a method with the best confidence factor classified correctly. Majority schema classified correctly if the sum of confidence factors for the correct prediction was bigger than the sum of confidence factors against it.

No additional information was used. For example, when having many classifications it could be possible to assume that the most probable class would be the one most commonly predicted. No such or other additional mechanism was embedded into models and in most cases parameters were deliberately set to lower levels than expected in real life. The idea is that if even such uninformed system produces better results it is even easier to reproduce the gains in real life.

Three parameters were varied:

- CONNECT - the percentage of cases where the confidence factor was verified to randomly fall into subinterval between 0.50 and 1.00 if the method predicted correctly and between 0.00 and 0.50 if the method failed.
- DECREASE - the decrease of average classification accuracy of each next method.
- DEPEND - the percentage of classifications of j-th method which were checked to classify the same as the first method.

The following is an example with CONNECT = 0.10, DECREASE = 0, DEPEND = 0, best-one voting schema, average 0.50 accuracy and 0.50 certainty factor. Methods are completely independent of each other. The relation between the number of methods and classification accuracy is observed. The meaning of rows is as follows: (1) index of the current method, (2) classification accuracy of the current method, (3) classification accuracy of 1..current method together, (4) confidence factor of correct predictions of the current method and (5) of 1..current method together.

Results of over 2000 similar tests with different combinations of parameters indicate that multiple knowledge can significantly improve overall classification accuracy under the following conditions: (a) certainty factor is positively correlated to the accuracy of each method, (b) classification accuracy of each multiple method is not much smaller then the best one, (c) methods are not too similar. The increase is especially noticeable when having a small number of multiple methods, e.g., 2-5.

5 Empirical measurements

Two real-life domains were chosen for benchmarking available systems. They represent descriptions of patients and their diagnoses mainly obtained by autopsy. Basic data: lymphography - 150 examples, 9 classes, 18 attributes; primary tumor - 339 examples, 22 classes, 17 attributes (Gams 89). Over 3 years around 20 AI and statistical systems were tested in several thousands of tests each time randomly dividing data into learning and test data and adding additional noise, varying the percentage of learning data etc. The best two methods were combined together in a multiple system GINESYS which achieved the best overall classification accuracy in more than 95 percentage of measurements, where one meaurement consisted of averaging classification accuracy over 10 tests.

Although the improvements of classification accuracy over the second the best system were not sufficient to be proven by the significance tests, they were typically at the level of 1% on the average. The chosen statistical measure was the T-test (Jamnik 87) which demands more or less permanent improvements in order to be fulfilled. In our measurements, the standard deviation was usually around 5%, meaning that the distributions of testing and learning data varied quite a lot. Therefore, while the improvements can not be statistically proven as significant when comparisons are made on the bases of one test, they happen nearly always when average over 10 tests is compared. For example, in (Gams 89), where these tests are described, from 164 averages there were only 3 cases where GINESYS had

1	2	3	4	5	6	7	8	9	10	20	30
50.1	50.1	50.2	50.0	50.1	50.0	49.7	50.2	50.3	50.5	50.1	50.0
50.1	52.3	53.6	54.0	54.5	54.2	54.5	54.5	54.8	54.9	55.2	55.7
52.3	52.5	52.4	52.7	52.5	52.8	53.5	52.7	52.8	52.5	52.0	52.4
52.3	68.0	75.7	80.8	84.0	86.3	88.0	89.2	90.3	91.1	95.2	96.8

Table 1: Classification accuracy and confidence factors of simulated models

not achieved the best classification accuracy. Additionally, although the 1% improvement of classification accuracy may seem unimportant, one should have in mind that in our measurements the difference between the classification accuracy of the best and reasonably worst system was typically from 10 to 15%.

GINESYS is a freely available scientific system which was, together with the benchmarking data, sent to over 50 researchers mainly in the developed countries. No major inconsistency was found.

6 Experiment with Winston's arches

It is difficult to measure the gains of multiple knowledge in exact domains. Most of exact deterministic procedures obviously don't need any multiple knowledge to perform theirs tasks. But there might still be many interesting tasks and domains where multiple knowledge can offer additional possibilities. As an example we have chosen the well known problem of Winston's arch (Winston 75).

The task of the system is to learn the concept "arch" from examples. In (Winston 75) there are two positive examples of arches and two negative examples of near misses presented in Figure 1.

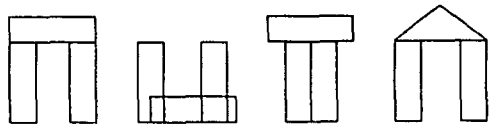


Figure 1: Learning examples for the Winston's arch problem.

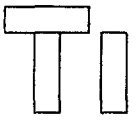
This case example was re-examined by many researches (e.g. Quinlan 90). The descriptions of objects and the solution can use the following relations: supports, left_of, touches, brick, wedge, parallelepiped all with predefined number of arguments. Negation is allowed. Although this description language is more limited that Winston's, the four learning examples are the same and similar problems can be observed in both approaches. The solution by Winston's system in this limited language is the following:

(S1)
arch(A,B,C) ←
    supports(B,A),
    supports(C,A),
    not (touches(B,C)).

Given the same learning examples, Quinlan's system FOIL constructed the following solution:

(S2)
arch(A,B,C) ←
    supports(B,A),
    not (touches(B,C)).

which covers not only the two positive examples, but also



The difference between the two solutions is commented by Quinlan: "Since FOIL looks for concepts in a general-to-specific fashion, it only discovers descriptions that are minimally sufficient to distinguish tuples in the relation from other tuples, which Dietterich and Michalski refer to as maximally general description. In this case, FOIL never formulates a requirement that the lintel be either a block or a wedge, or that both sides support it, because it has never seen near misses in Winston's terms, that make these properties relevant. FOIL would require a description of many more objects in order to elaborate its definitions of the relation arch."

In a different way, Winston devotes especial attention to different kinds of relations: "Humans, however, have no trouble identifying the forth example as an arch because they know that the exact shape of the top object in an arch is unimportant. On the other hand, no one fails to reject the second example because the support relation of the arch is crucial. Consequently, it seems that a description must indicate which relations are mandatory and which are inconsequential before that description qualifies as a model. This does not require any descriptive apparatus not already on hand."

In other words, Winston's solution (S1) contains only "MUST-BE" relations, which corresponds to FOIL's most general solution (S2). Other relations, also constructed by Winston's system are not obligatory, for example that A is a brick. Winston's solution not only contains more information, it is also the one expected by humans. Unfortunately, his system heavily depends on the proper order of examples and is domain dependent whenever near misses differ from positive examples in more than one relation.

FOIL is independent of the order of examples, is very quick and general. It has also a mechanism for handling noise and has found solutions of several interesting problems.

Both systems are very well known within the artificial intelligence community. However, they still more or less follow the formal logic approach which, for example, typically constructs only one solution.

Let us re-examine the Quinlan's solution by the multiple-knowledge approach. To do this, it is necessary to transform the description of examples from relations to attributes and its values. Theoretically this is possible because this domain is finite and practically it is possible because the domain is small. The transformation was performed by the LINUS system (Lavrač, Džeroski & Grobelnik 91).

LINUS enables transformation of problems presented in first-order logic without recursion into attribute-value language. Then, any of empirical learning systems like NEWGEM (Mozetič 85), ASSISTANT (Cestnik et al 87) or GINESYS can be chosen for intermediate knowledge construction. Finally, LINUS transforms results back into first-order logic. LINUS already enables the use of multiple knowledge since the user can apply different methods, compare the results and choose the best of them. Additionally, GINESYS alone enables construction of large sets of rules sorted by user-chosen preference criterion.

By applying GINESYS and using different preference functions two solutions emerge as the most reasonable, one being already presented and the other:

```
arch(A,B,C) ←
      supports(C,A),
      not (touches(B,C)).
```

The solution of Winston's system is also constructed but is somehow lost in tens of rules with similar values of preference functions. When another two near misses eliminating both too general solutions were presented, LINUS produced the wanted solution with ASSISTANT or NEWGEM and GINESYS showed there isn't any solution of similar preference. GINESYS enables yet further analyses. In one of the modes it produced the following solution (this time in an Algol-like notation):

```
if not (supports(B,A)) then not (arch(A,B,C)) else
  if not (supports(C,A)) then not (arch(A,B,C)) else
    if touches(B,C) then not (arch(A,B,C))
                    else arch(A,B,C)
```

and further analyses showed that first three rules can be freely interchanged.

With this example we have tried to illustrate additional possibilities enabled by multiple knowledge in exact domains. They include:

- finding all solutions of the same or similar preference
- analyzing different possible solutions by different preference functions
- analyzing why hasn't the system produced the intended solution
- analyzing the importance of subparts of solution and the form of the solution.

The approach of abstract logic has shown certain disadvantages even in the simple real-life task of learning the concept "arch" from four examples. First, most general solution was not the one that would be normally designed by humans using common-sense knowledge. Second, only one solution was proposed while solutions of the same preference were not even shown to the user. And third, several interesting solutions could be singled out by using different preference functions, but again, that is not common practice in the formal deterministic approach. Therefore, the debate about the appropriateness of the formal logic approach to real-life problems might bear some weight (BYTE, september 1990, 63 of the World's Most Influential People in Personal Computing Predict the Future, Analyze the Present), although much more elaborate analyses should be performed than our simple example.

Whatever the case, multiple solutions, i.e. solutions by the multiple-knowledge approach, seem to give better possibilities to analyze the properties of the domain and present more valuable information than systems constructing only one solution.

7 Discussion

There are many reports of different authors that confirm the practical advantages of multiple knowledge. Here presented simulated models strongly indicate the same conclusion and, furthermore, it seems that people inherently use multiple knowledge.

The Principle can be compared to other basic approaches to computing. Under the assumption that the Principle is valid the following can be argumented: (a) several basic principles like the use of redundant information in the information theory (Shannon & Weaver 64) or hierarhical decomposition promote very similar

approach; (b) Bayesian inductive procedure could be modified (Cheeseman 89) to capture the Principle; (c) Occam's razor (Blumer et al 86) and some conclusions in pattern-recognition theory should be accepted only in the pure form - their simplifications or generalizations sometimes confront not only the Principle but empirical observations as well.

Finally, we argue that if this Principle enables as important improvements in real-life domains as the first empirical measurements and simulations indicate, then people should use many knowledge bases, or in other words many different computer procedures, for most practical difficult tasks.

References

Blumer A., Ehrenfeucht A., Haussler D., Warmuth M.K. (1986) Occam's Razor, UCSC-CLR-86-2, USA.

Boose J., Bradshaw J., Kitto C. & Shema D. (1989) From ETS to Aquinas : Six Years of Knowledge Acquisition Tool Development, Proc. of EKAW 1989.

Brazdil P.B., Torgo L. (1990) Knowledge Acquisition via Knowledge Integration, Proc. of EKAW 1990.

Buntine W. (1989) Learning Classification Rules Using Bayes, Proc. of the International Workshop on Machine Learning, Ithaca, New York.

Catlett J. & Jack C. (1987) Is it Better to Learn Each Class Separately ?, Technical Report, Sydney.

Cestnik B., Kononenko I., Bratko I. (1987) ASSISTANT 86 : a Knowledge - Elicitation Tool for Sophisticated Users, Progress in Machine Learning, Bratko I., Lavrač N. (Ed.), Sigma Press.

Cheeseman P. (1989) On Finding the Most Probable Model, Technical Report.

Charniak E., McDermott D. (1985) Introduction to Artificial Intelligence, Eddison-Wesley.

Clark P. (1990) Reasoning with Differing Expert Opinions: A Novel Application of Expert System Technology, Technical Report, Turing Institute.

Emde W. (1989) An Inference Engine for Representing Multiple Theories, Knowledge Representation and Organization in Machine Learning, K. Morik (Ed.), Springer Verlag.

Gams M. (1987) Unifying Principles in Automatic Learning, Ph.D. thesis, Ljubljana.

Gams M. (1989) New Measurements Highlight the Importance of Redundant Knowledge, Proc. of EWSL 89, Montpellier.

Gams M., Drobnič M., Petkovšek M. (1991) Learning from Examples - a Uniform View, International Journal of Man-Machine Studies.

Jamnik R. (1978) Verjetnostni račun, Mladinska knjiga, Ljubljana.

Keyes J. (1989) Why Expert Systems Fail, AI Expert, 4, 11.

Lavrač N., Džeroski S., Grobelnik M. (1991) Learning Nonrecursive Definitions of Relations with LINUS, Machine Learning - EWSL91, Y.Kodratoff (Ed.), Springer Verlag.

Mozetič I., NEWGEM - program for learning from examples, Technical documentation, University of Illinois at Urbana - Champaign, USA.

Murthy S.S. (1988) Qualitative Reasoning at Multiple Resolutions, Proc. of Qual. Physics Workshop, France.

Quinlan J.R. (1990) Learning Logical Definitions from Relations, Machine Learning, Vol. 5, No. 3.

Schlimmer J.C. (1987) Learning and Representation Change, Proc. of AAAI 87, USA.

Shannon C.E. & Weaver W. (1964) The Mathematical Theory of Communications, Urbana, Illinois, University of Illinois Press.

Winston P.H. (1975): "Learning Structural Descriptions From Examples", The Psychology of Computer Vision, McGraw-Hill.