Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

Poster Abstracts

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

Separation of logical and calculation capabilities in a problem solving task

Jean-Bernard Auriol - Jean-Louis Dessalles
Département Informatique - ENST
46, rue Barrault, 75013, Paris - France
auriol@inf.enst.fr - dessalles@enst.fr

ABSTRACT
We present herein a model based on a strict separation between logical and calculation capabilities, designed to mimic aspects of human problem solving behaviour. Our model has been designed to be simple and psychologically plausible. We have tested our approach on the Tower of Hanoi task by comparing the results provided by our model with the performance of novice subjects. We also compared these results with the performance of a few other computational models. These comparisons are quite promising.

Keywords
problem solving, logical knowledge, procedural knowledge, calculation.

INTRODUCTION
In (Johnson-Laird and Byrne, 1991), convincing evidence is presented that seems to undermine the existence of human logical capabilities. Mental models (Johnson-Laird, 83) would explain experimental results on logical problem solving tasks much better than logical models do. Evidence from the observation of natural conversations (Dessalles, 1993) suggest however that the ability we have to argue with each other in everyday verbal interactions relies on genuine logical capabilities. Our hypothesis is that the same logical capabilities are involved in problem solving. We propose that the problem solving behaviour of subjects can be partly explained by the joint operation of two separate sets of capabilities: logical and calculation capabilities.

THE MODEL

Calculation Capabilities

Calculation Knowledge Representation: Operators
The representation of calculation knowledge is based on operators. An operator takes the following form:

(State 1, Operation, State 2)

where **State 2** results from the application of **Operation** to **State 1**. Operators are able to propose in sequence all existing legal steps from a given situation. An operator can be applied recursively, up to a given search depth, by taking one of the resulting states it has proposed as a new starting state.

Preference
We postulate a contextual preference for operators: in a given context, the operator will propose legal steps in a given order.

Reversibility
Operators are reversible in two ways. Given a resulting state, an operator can propose legal steps leading to this state and the associated starting state. Given a step, an operator can propose a starting state in which this step would be legal.

Logical Capabilities
The role of the logical part of the model is to evaluate situations and design goals. Its specific form is motivated by independent studies, particularly conversation modelling (Dessalles, 1993).

Logical Knowledge Representation
Logical knowledge is represented by first-order logical rules, in an extension of the negative conjunctive normal-form:

List of terms ⇒ Mod

Each term in the list is in conjunction with the rest of the list, and **Modality** (noted '**Mod**') is either **Undesirable** or **False**. Facts are stored in memory with no specific order, in the following basic form:

(Fact, Truth Value)

where **Truth Value** can be either '**true**' or '**false**'. Facts with an unknown truth value are not stored in memory.

Saturation Detection
The first capability that we put forward for the logical part of our model is the systematic detection of rule saturation. A rule is said to be saturated when all the terms of the rule are known to have the truth-value with which they appear in the rule. Depending on the modality, an undesirable or paradoxical situation will be detected in this case. We call such a situation a **problematic situation**.

Counter-Factual Production
To get out of a problematic situation, the subject has to change the truth-value of one term of the saturated rule. This is done by producing a counterfactual. A counterfactual is a term with a truth-value that is known to be false but that cancels the problematic aspect of the current situation. This counter-factual generation can be done repeatedly until the situation is no longer problematic.

Coupling logical and calculation capabilities

Problem representation
The problem representation is split into two parts. In the logical part, the situation is represented by facts. In the calculation part, the situation is represented by states. Goals are represented by undesirability rules in the logical part, and are not represented in the calculation part.

Goal-Oriented, Preference-Oriented Exploration

The strategy used to solve the problem is to explore the search space until reaching a state where the current undesirability is no longer saturated. It can be written in the following form:

```
OPERATORS: EXPLORE PROBLEM SPACE WITHIN SEARCH
           DEPTH
IF CURRENT UNDESIRABILITY SATURATED
      CONTINUE EXPLORATION
ELSE
      PLAY PROPOSED MOVE(S)
      IF NEW UNDESIRABILITY NEW_UND
            CURRENT UNDESIRABILITY = NEW_UND
      ELSE
            STOP
```

With a restricted search depth, the set of reachable states is limited. It often happens that all of them are uninteresting. In this case, the preferred move of the operator will be played, and the search process will start again from the new state reached. After a few steps made along according to mere preference, and if no interesting state is reached, the search stops: this is a dead end.

Getting Out of Dead End: 'Counter-factual' and Operator Reversibility

A dead-end situation is characterised by the fact that the current undesirability is out of reach of the operator. The strategy used to get out of dead ends can be sketched this way:

```
SELECT A TERM OF THE CURRENT SATURATED RULE
INVERT THE TRUTH VALUE OF THIS TERM
IF A NEW RULE BECOMES SATURATED
      REPEAT THE PROCESS
ELSE
      CALL OPERATOR WITH
            CURRENT STATE AS STARTING STATE
            DESIRED STATE AS ENDING STATE
TURN SITUATION RETURNED BY OPERATOR
      INTO UNDESIRABILITY RULE
RE-START SEARCH PROCESS
```

EXPERIMENTS

Our experiment is based on the comparison between solutions given by our model and by human subjects. We performed a step by step comparison between both solutions. In order to be able to compare the solutions after the first difference in move, our solution is bound to follow the subject's solution. At each step, our model computes its next move, which we compare to the human move. The human step is always the one played. Differences are counted, and whenever the erroneous move was chosen due to operator preferences, the involved preference is inverted.

We tested the system on 40 protocols, produced by seven novice subjects, and totalling 1462 steps. We also tried different others models. Besides random strategies (pure random, random without moving the same disk twice, preferences replaced by random[1]), we experimented with a model inspired by (VanLehn, 1991).

[1] That is, our model where the operator preferences were replaced by random choice.

In this latter model, three steps out of four are forced steps. Each time the model moves Disk 1, the next allowed move is to take the only other legally moveable disk and to put it on the only legal peg. Each time the model moves Disk 2, the next allowed step is to put Disk 1 back on it. The model chooses the optimal move for each unresolved move. Without correction, this algorithm always gives the optimal solution.

RESULTS AND DISCUSSION

For each model, we computed the percentage of correctly predicted moves out of the total number of moves. The results obtained after these trials are:

Random:	33.68%
Random without backtrack:	68.88%
Our model without preferences:	73.76%
Inspired by VanLehn:	78.66%
Our model:	80.71%

The results given by models involving random may vary by 1%. Results given by our model also vary by 0.7% around the value we give, because initial preferences are fixed at random. The results of the VanLehn inspired model do not vary.

The differences between the three first models and ours are significant ($chi^2 = 15.49$, $p < 0.0005$, for the comparison between our model and the random and logic model). The difference between our model and the VanLehn inspired model is not significant ($chi^2 = 1.51$, $p < 0.25$). The VanLehn inspired model gives good result principaly because it takes avantage of task specific constraints. Yet, the VanLehn inspired model generates by itself only the optimum solution, and cannot be, as such, a good model of human behaviour.

The comparison with the three random models is interesting. Our model without preferences is much better that random alone and is significantly better than random without backtrack. This confers an independent validation to the logical part of our model. Also, the results given by the complete model are better than those obtained by replacing preference by random, which indicates that, on the calculation side, preferences better account for human behaviour than random choices do.

ACKNOWLEDGEMENTS

We thank Dr. Bernard Auriol for his help.

REFERENCES

Dessalles, J-L. (1993). *Modèle cognitif de la communication spontanée, appliqué à l'apprentissage des concepts - Thèse de doctorat.* Paris: ENST - 93E022.

Johnson-Laird, P. N. (1983). *Mental Models.* London: Cambridge University Press.

Johnson-Laird, P. N. & Byrne, R. M. J. (1991). *Deduction.* Lawrence Erlbaum Associates.

VanLehn, K. (1991). "Rule acquisition events in the discovery of problem-solving strategies". *Cognitive Science, 15.*

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

Simulating chess players' recall: How many chunks and what kind they can be?

Heikki Hyötyniemi
Control Engineering Laboratory
Helsinki University of Technology
Otakaari 5 A, FIN-02150 Espoo, Finland
+358 9 451 3327

Pertti Saariluoma
Cognitive Science
P.O. Box 13 (Meritullinkatu 1 B)
FIN-00014 University of Helsinki, Finland
+358 9 19123458

ABSTRACT

This paper presents a numeric rather than symbolic approach to the chunking problem. The application area is the expert recall of chess board configurations. It is shown that a relatively low number of 'skilled' chunks is enough to explain the chess players recall of chess positions.

Keywords

Chunking, mental images; neural networks

INTRODUCTION

Chess players' recall of chess positions has been one of the major experimental paradigms in basic cognitive skills research (Chase and Simon 1973, Djakov, Petrovski and Rudik, 1926, de Groot 1965, 1966). In this research it was shown that expert chess players are superior to novices in recalling real game positions but not essentially better in recalling their randomized versions. The finding has been generalized over a large number of cognitive skills and it has proven to be very stable.

Perhaps the only issue of real concern has been the number of chunks experts have to learn to achieve their skill. Simon and Gilmartin (1973) argued that they must have learned, at least, 50,000 to 100,000 chunks. The evidence was based on simulation. However, Holding (1985) noticed that in these models the locations of the pieces were absolutely coded. Consequently, it was possible to assume that much lesser a number of chunks could explain the performance of the subjects. Saariluoma (1994) and Gobet and Simon (1996) have met the criticism by showing that chess players recall is impaired by transposition of the chunks on a chess board, which is critical to Holding's (1985) argumentation.

Another, theoretical presupposition in the original Simon and Gilmartin (1973) argumentation is the reliance on symbolic modeling. It might be possible that the whole philosophy of symbolic modeling is not adequate approach to the problems of human memory. As is well known various types of neural networks have challenged very deeply the idea of symbolic modeling. The evidence is today vast and it should be discussed in the context of chess players' memory recall as well.

In this paper one specific type of neural network model is used to simulate chess players' recall. The outcome of simulation shows that if neural networks are used the number of chunks could be reduced substantially. Thinking of the large support neural network models have in modeling human memory processes, the neural simulation makes it necessary to rethink the explanatory validity of Simon and Gilmartin (1973) argumentation and all models of the same type.

In this experiment, the framework differs very much from the traditional symbolic setting. For example, the chunks are now numeric and real-valued; and rather than expanding, they become more and more specialized as the training goes on. This view of chunks is in contrast with the original chunk idea.

ADAPTATION ALGORITHM

There are various neural network algorithms for pattern classification and feature extraction tasks available (see Bishop, 1995). The following approach[1] is specially tailored for self-organizing search of correlation structures. In statistical terms, it is a special combination of *cluster analysis* and *principal component analysis*; the resulting set of features can also be interpreted as sparsely coded, non-orthogonal *factors*.

The memory structure is a derivation of the Kohonen self-organizing map (Kohonen, 1984). There are *N nodes*, each of which is characterized by a *prototype vector* θ_i, where $1 \le i \le N$. The dimension of the vectors is *n*. The prototype vectors should represent the observed input vectors as accurately as possible – to reach this goal, the standard self-organization algorithm has been modified: rather than constructing only a set of *N* cluster centers characterized by the prototype vectors, the prototype vectors are interpreted now as 'coordinate axes' in the input data space, spanning a rather low-dimensional subspace. The algorithm can be implemented as follows.

1. Take the next input vector sample f.
2. Select the node with the best correlation with the input vector *f*, that is, determine the 'winner' index *c* such that the absolute value $|\varphi_c|$, where $\varphi_c = \theta_i^T f$, reaches its maximum value.
3. Calculate the 'neighborhood' parameter $h_{c,i}$ between the network nodes *i* and the winning node *c*. This parameter has value near 1 if the nodes are 'near' each other in the net, and lower value otherwise, as presented in (Kohonen, 1984).

[1] The analysis and other applications of the algorithm are presented in Hyötyniemi (1997) and (1998).

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

4. Apply the Kohonen type adaptation (Kohonen, 1984) of the network using the vector $\phi_c \cdot f$ as input. That means, for each network node i update the vector θ_i as $\theta_i \leftarrow \theta_i + \gamma h_{c,i} \cdot (\phi_c f - \theta_i)$. The parameter γ is a decaying function of time to assure that the network finally converges.
5. Normalize the feature vectors: $\theta_i \leftarrow \theta_i / \sqrt{\theta_i^T \theta_i}$ for all $1 \le i \le N$.
6. Eliminate the contribution of the feature number c by setting $f \leftarrow f - \phi_c \cdot \theta_c$.
7. If m features have not yet been extracted, go back to Step 2, otherwise, go to Step 1.

After the network has converged, the prototype vectors represent *features* that can be used to construct the input patterns. That means, given an input vector f, find the sequence of φ_i values as presented in Steps 2 – 7 above (ignoring the updating steps 3 – 5), so that the estimate for f can be constructed as a weighted sum of the features

$$\hat{f} = \theta_1 \varphi_1(f) + \cdots + \theta_N \varphi_N(f).$$

In this context, it is assumed that the extracted features are the *chunks*, conveying the dependency relations between the input elements. The number N stands for the capacity of the long-term memory, while the parameter m is the size of the short-term memory. It is also assumed that at any instant only the references to the static memory structures and the respective weights are operated on.

SIMULATION EXPERIMENTS

To apply the presented algorithm, the input data is first coded appropriately. This means that one must present the chess piece configuration as a vector of real numbers. The coding is now location-based and rather trivial.

It is assumed that the lower-level processing has produced the *component level* constructs, that means, the visual image has been analyzed and atomic information about the board has been extracted – these *information atoms* are now something like 'white king in g1', etc. For simplicity and for generality, it is assumed that each of these information atoms spans a dimension of its own in the input data space – this means that the input vector is 768 dimensional (six pieces of two colors, together 12 alternatives, for each of the 64 board locations). Naturally, this coding is far from optimal - the complexity of different modalities is changed to the high-dimensionality of the input vector space.

In the experiments, 5000 samples were iteratively used for training the network model. These samples were successive piece configurations during real chess games, given in random order. The simulation was implemented in a Matlab environment. The huge size of the data structures made the simulations rather capacity-demanding.

Three chunk models were extracted: the first with only 9, the second with 25, and the third with 100 chunk prototypes available, so that $N = 9$, $N = 25$, and $N = 100$, respectively. Five chunks were used to reconstruct the observed configuration, that means, $m = 5$. There were 500 additional game positions for testing purposes. To visualize the high-dimensional vectors representing the board and the chunks, the numerical values of the vector elements were thresholded – that means, if the value of the element exceeded 0.5, it was assumed that the corresponding piece was there; otherwise its contribution was ignored. No rules of chess were incorporated – in principle, it is possible that, say, two white kings will be displayed simultaneously, but because of the 'skilled' chunk prototypes, this seldom happens[2].

CONCLUSIONS

In the presented approach, the chunks are not 'crisp' - rather, their constituents have continuous (or fuzzy) values. This is one reason why *scalability* seems to apply, so that allocating more resources results in better reconstruction of the piece locations. For the 868 chunks, the average recall rate was about 75%.

Because of the numerical nature of the chunks, they are flexible and they can be added together in a natural way. Due to the possibility of combining chunk prototypes, a rather low number of 'skilled' chunks seems to be enough to reach relatively high level of accuracy.

REFERENCES

Bishop, C.M. (1995). *Neural Networks for Pattern Recognition.* Clarendon Press, Oxford.

Chase, W. G. and Simon H. A. (1973). The mind's eye in chess. In W. Chase (ed.), *Visual information processing.* Academic Press, New York.

de Groot, A. D. (1965). *Thought and Choice in Chess.* Mouton, The Hague.

de Groot, A. D. (1966). Perception and memory versus thought: Some old ideas and recent findings. In B. Kleinmuntz (ed.), *Problem Solving.* New York.

Djakov, I. N, Petrovsky, N. B., and Rudik, P. A. (1926). *Psihologia Shahmatnoi Igry (Chess psychology).* Avtorov, Moskow.

Gobet, F. and Simon, H.A. (1996): Templates in Chess Memory: A Mechanism for Recalling Several Boards. *Cognitive Psychology,* 31, pp. 1 – 40.

Holding, D. H. (1985). *The Psychology of Chess Skill.* Erlbaum, Hillsdale, N.J.

Hyötyniemi, H. (1997). On the Statistical Nature of Complex Data. In *SCAI'97 – Sixth Scandinavian Conference on Artificial Intelligence: Research Announcements* (ed. Grahne, G.), Helsinki University, Department of Computer Science, Report C-1997-49, Helsinki, Finland, pp. 13 – 27. Available from http://Saato014.hut.fi/Hyotyniemi/publications/.

Hyötyniemi, H. (1998). Automatic Structuring of Unknown Dynamic Systems. In *Soft Computing in Engineering Design and Manufacturing* (eds. Chawdhry, P.K., Roy, R., and Pant, R.K.), Springer-Verlag, London, pp. 410 – 419. Available from http://Saato014.hut.fi/Hyotyniemi/publications/.

Kohonen, T. (1984). *Self-Organization and Associative Memory.* Springer-Verlag, Berlin.

Saariluoma, P. and Hohlfeld, M. (1994). Chess players' long range planning. *European Journal of Cognitive Psychology,* 6, pp. 1–12.

Saariluoma, P. (1994). Location Coding in Chess. *The Quarterly Journal of Experimental Psychology,* 47A (3), pp. 607–630.

Simon, H. A. and Gilmartin, K. (1973). A simulation of memory for chess positions. *Cognitive Psychology,* 5, pp. 29–46.

[2] A simulation environment, implemented in Java, can be found at http://Saato014.hut.fi/Hyotyniemi/publications/97_scai.htm

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

Is Mental Imagery Symbolic? Exploratory Simulations in an Interactive Activation Model

Rita Kovordányi
Dept. of Computer and Information Science
Linköpings Universitet
S-581 83 Linköping, Sweden
+46 13 28 14 30
ritko@ida.liu.se

ABSTRACT

In this article we present an interactive activation simulation framework for mental image reinterpretation. By varying central parameters in this framework, two qualitatively different models have been emulated: One in which reinterpretation is obtained via a series of symbolic inference steps, and one in which reinterpretetation is driven by parallel operations on a depictive mental image. The simulations are run with the following objectives: 1. To minimally verify that the models can produce reinterpretations. 2. To verify that the parametric relationships predicted by the models hold in the face of empirical constraints on the simulation outcome. 3. To expose unforeseen parametric constraints which are entailed by the two models.

INTRODUCTION

When we close our eyes and mentally image a capital 'X', superimpose a capital 'H' on it, and recognize a "bow tie" in the resulting image, we generate, manipulate, and reinterpret mental images. Psychological experiments on human performance reveal interesting anomalies in how easily mental images are reinterpreted.

Are mental images *reinterpretable* because they supplement symbolic structures with new affordances? These and related matters lie at the heart of 'the imagery debate' (e.g., Kosslyn 1994; Pylyshyn, 1981). This article investigates the role of visual versus symbolic representations as a mediating factor in mental reinterpretation tasks.

Two models of mental image reinterpretation have been

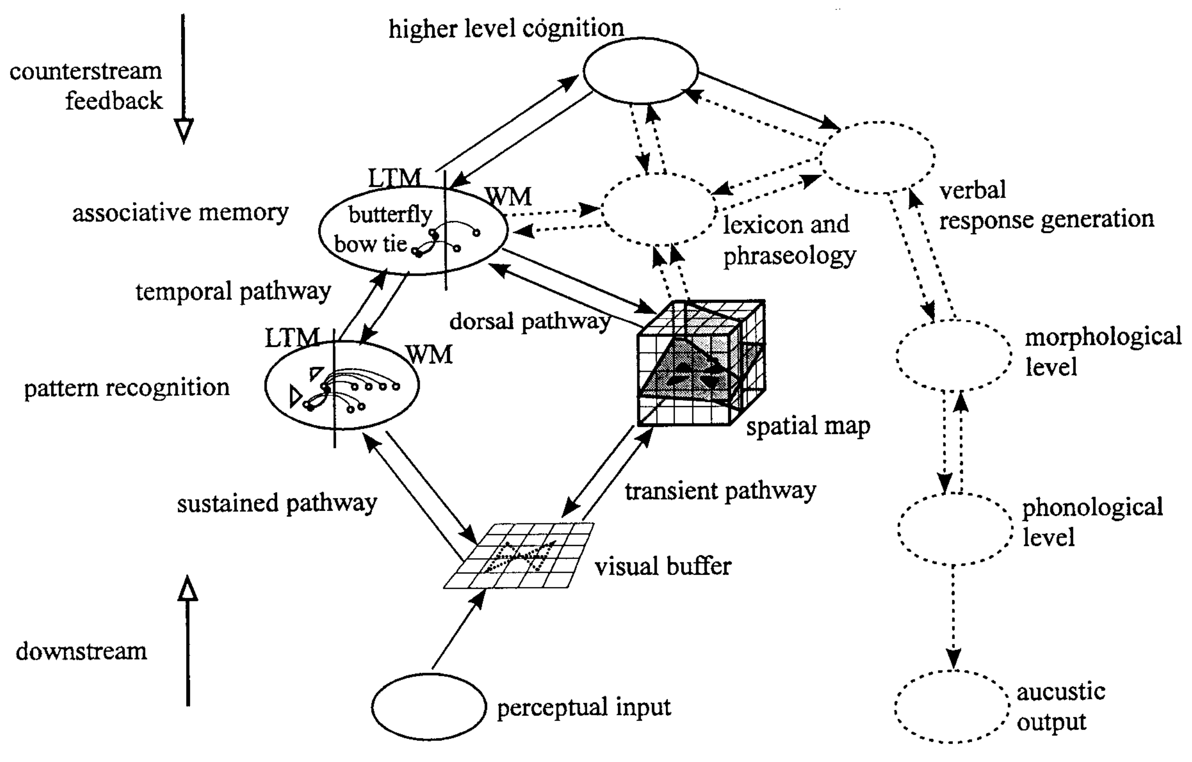


Figure 1. The visual system with its major subsystems at different levels of processing.

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

compared using McClelland & Rumelhart's interactive activation model (1981; 1994/1988) as a parametric framework. Our ambition has been to keep the set of working hypothesis concerning the neural architecture and processing style employed to a minimum, and instead explore the gaps which are left unspecified by empirical data.

Both models are discrete and deterministic, and can be conceived of as a set of constraints on inter-parameter dependencies within the envelope of the simulation framework. The two models stipulate that different parametric relations should hold in order for the simulation outcome to conform to the empirical constraints. *Simulation outcome* is measured as the relative conformance with empirically based constraints on how reinterpretation performance should change *when simulation is switched from perceptual to mental mode.*

Experimental data on mental image reinterpretation

Contrary to the classical findings on mental image reinterpretation difficulties, Finke, Pinker and Farah (1989) demonstrated that mental images can be as easy to reinterpret as perceptual images when the interpretations generated comprise verbal descriptions of *geometric patterns* contained in the image. Two types of reinterpretations seem to be involved: *Geometric reinterpretations*, when the composite image is described in simple geometric terms, for example, "two adjacent triangles pointing towards each other", and *symbolic reinterpretations*, in which the image is freely associated with an object or concept, for example, "a bow tie". In experiment 1 (Finke et al. 1989) relative performance rate for symbolic reinterpretations was on the average 30-50% of the possible total produced during imagery *and* perception. As opposed to this, up to 80-90% of the geometric reinterpretations were detected in the mental images proper.

VISUAL VERSUS SYMBOLIC REPRESENTATIONS

In a very general sense, qualitatively different styles of computation is afforded by symbolic representations as opposed to visual representations, with the main difference being that of accessibility in a linked versus a directly addressable data structure.

We operationalize these different assumptions, and would like to examine whether visual representations are needed as a *mediating link* between old and new interpretations in an interactive activation model. We have two possible hypothesis:

1. The subjective experience of "seeing mental images" is a non-functional side-effect of symbolic knowledge being activated in associative long term memory. No "*real*" image is formed in the visual buffer during imagery, so mental reinterpretations have to be based on inferences using "lateral" associations between symbolic representations.
2. A mental image is recreated in the visual buffer, and this image plays a pivotal role in mental reinterpretation. In this case, it is the image that drives processing towards a new interpretation, while the image's symbolic content acts as a source for indexing and sustaining, and thereby locking, the current interpretation.

Methodology

We evaluate the two representational hypothesis by freely exploring parametric variants of a simulation framework (Fig. 1) and by evaluating these variants against the empirical constraints of Finke, Pinker and Farah (1989). Simulation outcomes depend at the outset on the parameter constraints imposed by the individual models *plus* the following minimal assumptions about the neural architecture and processing style of the human visual system:

- Visual subsystems are hierarchically organized into processing levels.
- Adjacent processing levels in the visual system communicate with each other reciprocally.
- Visual processes operate in cascade.
- Mental imagery reuses parts of the visual system. In particular, images formed during mental imagery are assumed to reside in the visual buffer.

What is measured?

Keeping exploration of the parameter space within the envelope of the simulation framework and within the parametric constraints imposed by a particular model, simulation of the models should substantiate that whenever the system's transition behavior between perceptual and mental modes conforms to the empirical constraints, the parametric relations predicted by the models hold. Based on the interdependencies which can be detected when simulation results are systematically plotted against parameter combinations, the soundness of the two models can be evaluated and additional properties which necessarily follow from the two models can be exposed.

For a preliminary analysis of our simulation results, see www.ida.liu.se/~ritko

REFERENCES

Finke, R. A., Pinker, S. & Farah, M. J. (1989). Reinterpreting visual patterns in mental imagery. *Cognitive Science, 13*, 51-78

Kosslyn, S. M. (1994). *Image and Brain: The resolution of the imagery debate*. Cambridge, MA: MIT Press

McClelland, J. L. & Rumelhart. D. E. (1994/1988). *Explorations in parallel distributed processing: A handbook of models, programs and exercises.* Cambridge, MA: MIT Press

Pylyshyn, Z. W. (1981). The imagery debate: Analogue media versus tacit knowledge. *Psychological Review, 87*, 16-45

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

Modelling Individual Differences in Reasoning

Padraic Monaghan
Centre for Cognitive Science
University of Edinburgh
2 Buccleuch Place
Edinburgh, EH8 9LW, UK
+44 131 650 4415
pmon@cogsci.ed.ac.uk

ABSTRACT

The serialist-holist learning style distinction has received renewed interest due to its predictive power with regard to students' responses to new learning situations. In particular, individual differences in students' computer use indicate an area where knowing about style differences is of theoretical interest and practical import. This study concerns the differing responses of students to a computer-based logic program – Hyperproof – where serialist-holist style differences emerge spontaneously in the proofs produced by students. Proof style and strategy change are found to relate to independent measures of reasoning ability. These different strategies are analysed in terms of working memory load, and this points towards potential methods of modelling the serialist-holist learning style.

Keywords

Serialist-holist, reasoning, working memory, learning.

INTRODUCTION

Students use different strategies when they solve problems. Certain patterns of behaviour in new learning situations have been expressed in terms of the serialist-holist distinction. However, the environments where these differences have been diagnosed and observed have been complex, subjective, and lengthy, hence assessing contributing factors that influence performance has been difficult.

Currently, we have been studying a computer-based environment for problem-solving called Hyperproof (Barwise & Etchemendy, 1994) where serialist and holist strategies emerge spontaneously. This environment has the advantage over previous studies of learning strategies in that it is a constrained context within which variables can be manipulated, and detailed data on performance can be collected as students' interactions with the problem are logged by the computer.

This paper presents the background necessary for modelling serialist-holist learning styles, and offers a preliminary model of the interaction between changing problem requirements and strategy selection. Modelling differences in strategies within the restricted domain of Hyperproof will help to define what the serialist-holist distinction means from a cognitive perspective.

THE SERIALIST-HOLIST DISTINCTION

Pask (1976) used the serialist-holist distinction to describe the different strategies used by students in new learning situations. Serialists concentrate on concrete instances within the learning framework, building up an overall understanding of the situation by forming links between low-level features. In contrast, holists prefer to focus on the global structure of the learning situation, filling out the details once the structure has been explored. Roughly speaking, the serialist is a 'bottom-up' learner, whereas the holist's approach is 'top-down'.

Versatile students will select the strategy that is most appropriate to the task, and this requires a combination of awareness of the task constraints and of the individual's own resource limitations and aptitudes. Pask has found that most students are inflexible in their approach to problems – a student that always uses one particular strategy when solving problems is said to have a learning 'pathology'.

These differences have proved to be ubiquitous and pervasive in a variety of different learning situations. In research on human-computer interaction, for example, the distinction does much to classify and predict the different responses of students to alternate interfaces (for a review see Helander, 1990, pp.541-580). Though important to learning, little computational or cognitive research has been directed towards defining or describing the different processes that underly each learning strategy.

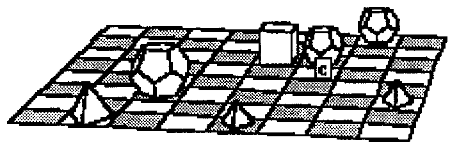
HYPERPROOF

Hyperproof is a multimodal computer-based tool designed to teach first order logic through the dual presentation of a graphical situation and sentential descriptions of elements of the situation. The graphical situation is made up of objects of varying size and shape taking up positions on a chess-board. One particular type of problem requires the student to concretise an abstract situation: in order to solve the problem, the student must express graphically information that is given in a sentential (propositional calculus) form. A simple example of this type of task is illustrated in Figure 1.

In this problem the student is required to display all situations that are consistent with the given information. In short, the several ways that the labels 'a' and 'b' and the predicate information 'object a is a dodecahedron' and 'a and b are in the same row' have to be illustrated one after another in the graphical part of the window. There are two distinct strategies by which all the situations can be constructed. One method will apply all pieces of sentential information simultaneously in each situation, thus the strategy is a

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

1. ▪ Given



2. ▪ Dodec(a) Given
3. ▪ SameRow(a, b) Given

Figure 1: The Hyperproof problem.

case-by-case method and therefore serialist. The alternative method applies one piece of information at a time, with the second piece of information being superimposed onto the situation formed from applying the first sentential expression. As this proof is less concrete, it is interpreted as reflecting a holist strategy. This latter method is akin to constructing nested assumptions in a logical proof.

TOWARDS A COMPUTATIONAL MODEL

Serialist and holist strategies, as described with respect to the problem in Figure 1, have been observed in the proofs of students on a Hyperproof course (Cox, Stenning & Oberlander, 1994; Monaghan, 1998). These different uses of strategy have been related to an independent measure of reasoning ability (derived from the analytic reasoning section of the USA graduate recruitment exam (GRE)). Two Hyperproof problems solved under exam conditions were analysed. These questions contained as a main subtask the above type of problem, one question requiring the construction of three situations, the other requiring nine situations to be indicated. Students using a serialist strategy on the simpler problem and a holist strategy on the complex problem were better GRE reasoners than other groups, including the 'pathological' students who rigidly used only one strategy on the Hyperproof problems ($F(3, 18) = 5.69$, $p<0.01$). This suggests that there are general strategic approaches to complex problem solving situations that are more successful than others.

A preliminary model of the Hyperproof problem assessed the working memory load at each step in the proofs as a result of applying the different strategies. The holist strategy minimises working memory load, but more steps in the proof are required: seven to the serialist's five for the Figure 1 example. For students that are good at solving problems, strategy choice seems to be a pay-off between working memory load and the effort required to structure the solution. For simple problems, like the one illustrated, a serialist method may be more efficient. For more complex problems, a holist proof will reduce the working memory load.

The Hyperproof environment provides a suitable domain for studying serialist-holist strategies from a computational perspective. It also allows for a study of learning pathologies and strategy change under different conditions. A cognitive model of serialist-holist strategy use will have implications for several areas of cognitive science research. Principally, it will provide a formalism of what the different strategies mean from a computational perspective allowing better provision of resources in areas such as human-computer interaction. Also, insight into the cognitive properties of substeps in problem-solving procedures would result (Catrambone, 1996). Finally, the cognitive properties of external representations during problem-solving can be assessed (Scaife & Rogers, 1997).

ACKNOWLEDGMENTS

This work was supported by ESRC research studentship R00429634206.

REFERENCES

Barwise, J. & Etchemendy, J. (1994). *Hyperproof.* CSLI Lecture Notes. Chicago: Chicago University Press.

Catrambone, R. (1996). Generalizing solution procedures learned from examples. *Journal of Experimental Psychology: Learning, Memory, and Cognition* 22, 1020-1031.

Cox, R., Stenning, K., and Oberlander, J. (1994). Graphical effects in learning logic: reasoning, representation and individual differences. *Proceedings of the Sixteenth Annual Conference of the Cognitive Science Society*. Pittsburgh: Lawrence Erlbaum Associates, 237-242.

Helander, M. (Ed.) (1990). *Handbook of Human Computer Interaction* (2nd edition). North Holland: Elsevier Science Publishers.

Monaghan, P. (1998). *Holist and Serialist Strategies in Complex Reasoning Tasks: Cognitive Style and Strategy Change.* (Research Paper EUCCS/RP-73). University of Edinburgh, Centre for Cognitive Science.

Pask, G. (1976). Styles and strategies of learning. *British Journal of Educational Psychology*, 46, 128-148.

Scaife, M. & Rogers, Y. (1997). *External Cognition: How do Graphical Representations Work?* (Cognitive Science Research Papers 335). University of Sussex, School of Cognitive and Computing Sciences.

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

A Feedforward Connectionist Account of Causal Discounting and Augmentation

Frank Van Overwalle & Dirk Van Rooy
Department of Psychology
Vrije Universiteit Brussel
Pleinlaan 2, B–1050 Brussel, Belgium
+32 2 629 25 18
Frank.VanOverwalle@vub.ac.be

ABSTRACT
We investigated the degree of discounting and augmentation of a target cause given varying frequencies of a competing cause. Several experiments showed that greater frequencies by which the competing cause covaried with the effect resulted in greater discounting or augmentation of a target cause. These competition size effects cannot be explained by current attribution theories in social psychology, but can be accounted for by a feedforward connectionist framework (Van Overwalle, 1998).

Keywords
Connectionism, Causal Judgments, Blocking.

INTRODUCTION
According to Kelley (1971), perceivers take into account not only how a possible factor <u>covaries</u> with the event, but also how this factor <u>competes</u> with rival factors that serve as alternative explanations. Despite the central place accorded to the covariation principle in attribution theory, Kelley (1971) argued that this principle in itself is insufficient to explain how perceivers select between competing causes. To account for such competition, Kelley (1971) proposed two complementary principles of discounting and augmentation.

The <u>discounting</u> principle specifies that if the influence of a cause is clearly established, perceivers will disregard other possible causes as irrelevant. The opposite tendency is described in the <u>augmentation</u> principle which specifies that if the inhibitory influence of a cause is firmly established, perceivers will overestimate the strength of a facilitatory cause to compensate for the inhibitory effect.

Our major question was whether discounting and augmentation of a target cause would be influenced by the frequency (or size) by which the competing cause covaried with the outcome. Based on a novel feedforward connectionist approach of causality (Van Overwalle, 1998), we predicted that greater frequencies would result in greater discounting or augmentation. Such competition size effect is not anticipated by current attribution theories in social psychology.

METHOD
In three experiments, the strength of competition was manipulated by varying how often the competing cause covaried alone with its outcome : Either one time (small size) or five times (large size). In contrast, the frequency of the target cause remained constant throughout all conditions. Type of competition was manipulated by pairing the competing cause with an outcome that was either similar to the target outcome (discounting) or opposite (augmentation). In addition, we manipulated the order in which the target information was presented (backwards or forwards) and the format of presentation (sequential trial-after-trial or summarized in short sentences).

RESULTS
Our results confirmed the feedforward connectionist account. First, in all experiments, we found that a higher frequency of covariation of a competing cause reliably increased the amount of discounting and augmentation of a target cause. These results are problematic for statistical models based on the notion of probability (e.g., Cheng & Holyoak, 1995) or of constraint satisfaction (Read & Marcus-Newhall, 1993). Second, the size effects were stronger when the information was presented in a sequential format, which is consistent with the feedforward connectionist view that the most natural way of processing causal information occurs on a trial-by-trial incremental basis. Third, there were no differences between forward and backward competition, supporting the notion that missing factors must be coded as absent as proposed by Van Hamme and Wasserman (1994).

REFERENCES
Cheng, P. W., & Novick, L. R. (1990). A probabilistic contrast model of causal induction. *Journal of Personality and Social Psychology, 58,* 545—567.

Kelley, H. H. (1971). Attribution in social interaction. In E. E. Jones, D. E. Kanouse, H. H. Kelley, R. E. Nisbett, S. Valins & B. Weiner (Eds.) *Attribution : Perceiving the causes of behavior* (pp. 1—26). Morristown, NJ : General Learning Press.

Read, S. J. & Marcus-Newhall, A. (1993). Explanatory coherence in social explanations : A parallel distributed processing account. *Journal of Personality and Social Psychology, 65,* 429—447.

Van Hamme, L. J. & Wasserman, E. A. (1994). Cue competition in causality judgments : The role of nonpresentation of compound stimulus elements. *Learning and Motivation, 25,* 127—151.

Van Overwalle, F. (1998) Causal Explanation as Constraint Satisfaction : A Critique and a Feedforward Connectionist Alternative. *Journal of Personality and Social Psychology,* in press.

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

Models of Two-person Games in ACT-R and SOAR

Frank E. Ritter
University of Nottingham
Nottingham NG7 2RD, UK
frank.ritter@nottingham.ac.uk

Dieter P. Wallach
Saarland University
66041 Saarbrücken, Germany
dwallach@cops.uni-sb.de

We were interested in understanding and comparing how ACT-R (Anderson & Lebière, in prep.) and SOAR (Newell, 1990) could each model a given dataset. We analyze and compare two models in their ability to account for a classical 2 person game, including the effort necessary to create and run them. In comparing the models and their results we provide two sample models and start to explore the potential role of abstract models and different types of data.

Game description. In two player, 2x2 games each player can choose one of two alternatives in each round. The players are rewarded according to a payoff matrix. The prisoner's dilemma is an example of such a 2 person game.

We used data from a classical experiment (Suppes & Atkinson, 1960) of how people learn when they play a normal form, two player 2x2 game with a nontrivial unique mixed strategy equilibrium. Table 1 shows the payoff matrix used in the experiment that we model here. This matrix has a unique mixed strategy equilibrium point, that is, a stable set of strategies, when Player 1 chooses option A1 with probability 1/3 and player 2 chooses option A2 with probability 5/6. Figure 1 shows the empirical choice frequencies of option A for player 1 (A1) and player 2 (A2) aggregated in 5 blocks with 40 rounds each, of 20 pairs of participants playing the game for 200 rounds (Erev & Roth, 1998).

ACT-R model. Figure 2 shows the structure of the ACT-R model used to account for this data. For a full description of the ACT-R model see Bracht, Wallach and Lebière (1998). The model consists of two simple productions for each player representing the options available:

Rule1: If Player 1 chooses => choose Option A.

Rule2: If Player 1 chooses => choose Option B.

In every round, both of these productions are applicable for each player modeled. ACT-R's subsymbolic cost learning mechanism learns the relative payoff of each production rule and updates their expected gain based on the outcome of the round. In general, ACT-R selects the production rule with the

		Player 2 Option A	Option B
Player 1	Option A	2, 4	6, 0
	Option B	3, 3	1, 5

Table 1. Payoff matrix used by the models here.

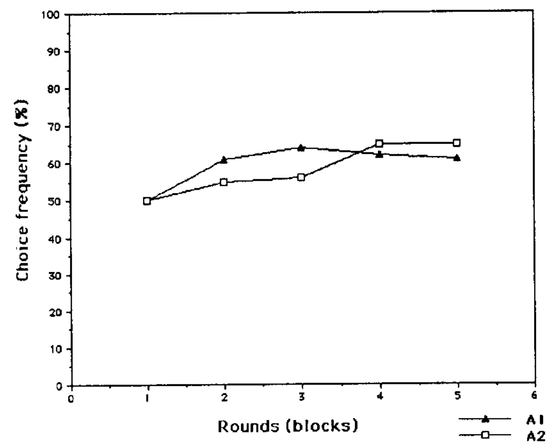


Figure 1. The evolution of strategies in the subjects on the Table 1 payoff matrix

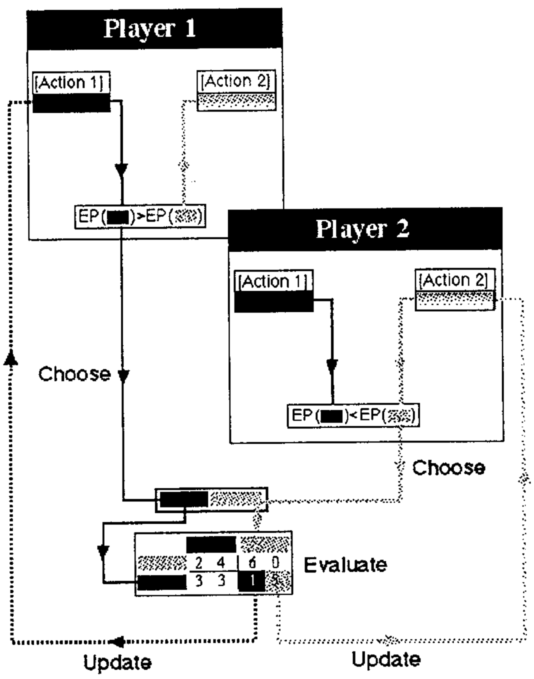


Figure 2. Description of the ACT-R model.

highest expected gain. Two architectural parameters were used to fit the model to the data (*expected gain noise* and *number of previous production applications*). The model with the same parameter settings has also been applied successfully to data from three other experiments taken from Erev and Roth (1998).

SOAR model. The easiest way to explore a SOAR model of this task is to create an abstract model. An abstract model is based on an information process-

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

ing model or architecture. It predicts what a running model would do, without implementing the internal behaviors (e.g. Langley, 1996; Ohlsson & Jewett, 1994).

An abstract model of the simplest SOAR model could start with a single operator representing each choice. Each round, an operator is randomly chosen to apply. After each round, the expected values of each of the four payoffs occurring can be computed for each player. Operators that do better than the average payoff can be duplicated through a reflection-like process (not specified, but similar to the process in Bass et al., 1995). Various other ways of duplicating operators are possible (e.g. duplicate operators as many times as their payoff). In SOAR these processes are determined not by the architecture but by knowledge. It is fairly straightforward to implemented a program to compute the expected population of operators on each round. The results of this program are shown in Figure 3. While this model is not currently based on a running Soar model, creating such a model should be straightforward. Deriving its predictions is much simpler as an abstract model, for programming an interface to record multiple rounds and games would be less straightforward.

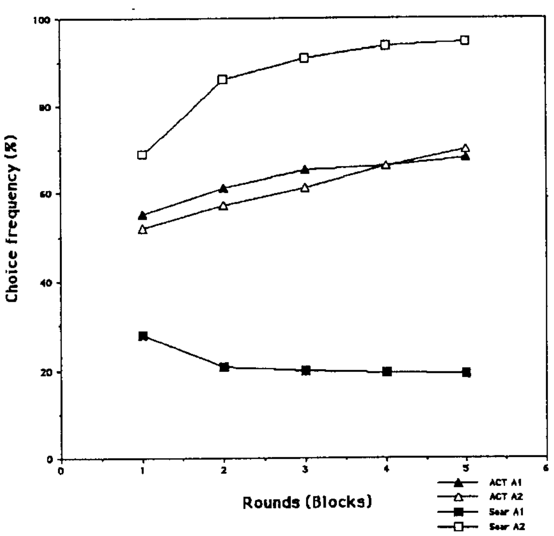


Figure 3. The evolution of strategies in the two models on the Table 1 matrix.

Comparisons

Model fit. As Figure 1 shows, the ACT-R model captures the general tendencies in the empirical data quite nicely. In addition to this *short term* prediction, the model converges asymptotically to the equilibrium of classical game theory in the *long term* (after >1500 rounds). The initial Soar model, on the other hand, does not match the subject data (short term) nearly as well, but instead appears to quickly converge to near the equilibrium.

Effort. Both models took about the same time to implement (4-5 hours), including the ability to automatically run and trace the model. Both models can run 200 rounds of 20 subject pairs in under 30s.

Abstract models. The Soar model would not be as easy to run if it was implemented in Soar productions. It would not be straightforward to implement an abstract version of the ACT-R model based on its current mechanism, but it is easy to create an abstract model of the operator population model in ACT-R (as a rule population), or an ACT-R model directly based on this principle. The difficulty of creating abstract models within each architecture varies by task, but appears to be generally easier in SOAR. Creating full models appears, however, to be more difficult. In this task, the SOAR architecture appears to have less to say than ACT-R because it lacks architectural mechanisms to account for the learning observed here. While the Soar model does not match nearly as well (yet), it allows the space of possible models to be explored quite quickly (about 5 min. per model).

Conclusions

These results are very interesting, for they start to suggest possible trade-offs in modeling; between abstract and information processing models, and between architectures. This work also emphasizes the role of usability as a necessary precondition for explorations of this kind.

REFERENCES

Anderson, J. R. & Lebière, C. (in prep.). *The atomic components of thought.* Hillsdale, NJ: Lawrence Erlbaum Associates.

Bass, E. J., Baxter, G. D., & Ritter, F. E. (1995). Using cognitive models to control simulations of complex systems. *AISB Quarterly*, 93, 18-25.

Bracht, J., Wallach, D. & Lebière, C. (1998). On the need and performance of cognitive game theory: ACT-R in experimental games with unique mixed strategy equilibria. To appear in *The Economic Science Association (ESA) Annual Conference.*

Erev, I. & Roth, A. (1998, in press). Predicting how people play games: Reinforcement learning in experimental games with unique, mixed strategy equilibria. *American Economic Review.*

Langley, P. (1996). An abstract computational model of learning selective sensing skills. In *Proceedings of the 18th Annual Conference of the Cognitive Science Society.* 385-390. Hillsdale, NJ: Lawrence Earlbaum Associates.

Newell, A. (1990). *Unified theories of cognition.* Cambridge, MA: Harvard University Press.

Ohlsson, S. (1995). Abstract computer models: Towards a new method for theorizing about adaptive agents. In *Machine Learning: ECML-95.* Berlin: Springer-Verlag.

Suppes, P. & Atkinson, R. C. (1960). *Markov learning models for multiperson interactions.* Stanford, CA: Stanford University Press.

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

A connectionist account of Illusory Correlation

Dirk Van Rooy
Department of Psychology
Vrije Universiteit Brussel
Pleinlaan 2, B–1050 Brussel, Belgium
+32 2 629 26 03
dvrooy@vub.ac.be

ABSTRACT
Illusory correlation occurs when perceivers make an erroneous judgment of a relation between two or more unrelated categories. In this study, subjects read information about members of 4 groups, which differed in size : Group A contained twice as much behaviors as group B, group B twice as much as C and so on. The behavioral information about these groups was identical, in that 33% of the behaviors engaged in by the members were undesirable and 67% desirable. Preliminary results show that a greater amount of members in each category leads to a decrease of the illusory correlation effect. These results can be readily accounted for by a feedforward connectionist framework (Van Overwalle, 1998).

Keywords
Illusory Correlation, connectionism.

INTRODUCTION

Illusory correlation occurs when perceivers make an erroneous judgment of a relation between two or more unrelated categories. The original demonstration by Chapman (1967) showed how subjects overestimated the co-occurrence of long words in the context of a list of relatively short words. Presumably, the distinctiveness of the long word pairs led to a more thorough processing, which led to the illusory correlation effect.

Hamilton and Gifford (1976) applied this mechanism to the formation of group stereotypes. In their study subjects read statements about members of a majority group, labeled A, and a minority group, labeled B. Both groups revealed the same ratio of desirable to undesirable behaviors. After reading the statements, subjects overestimated the frequency of negative behaviors by group B members and also had a more negative impression of group B. According to Hamilton and Gifford, the less frequent and therefore more distinct undesirable group behaviors apparently received more extensive encoding. This probably led to greater accessibility in memory, leading to errors in frequency estimation and impression formation.

Recently several studies challenged the distinctiveness-paradigm (Smith, 1991, Fiedler, 1991). These studies claim that the phenomenon is not so much the consequence of mere distinctiveness of the stimuli, but simply reflects the general working of the human memory. Although this critique is well elaborated, it leaves certain question unanswered. The aim of the present research is to answer these questions by approaching the illusory correlation phenomenon form a connectionist angle.

The aim of the present research is to approach the illusory correlation phenomenon from a connectionist angle. Our connectionist approach depicts learning as a gradual process, during which associations between group membership and desirability are formed instantaneously. Every time a member of a certain group performs a (un)desirable behavior, the association between that group and (un)desirable behavior in general becomes stronger. As more learning takes place, these associations become stronger and are easy to discriminate, so the perceiver can form a relatively correct impression of a group based on these associations. However when these associations between group membership and desirability are weak, they are hard to discriminate and judgments will be prone to illusory correlation effects. Therefore, the main prediction of our connectionist model is that an increase in the amount of behaviors will lead to a decrease in the illusory correlation effect. Although apparently trivial, this effect is not a straightforward prediction of the distinctiveness hypotheses or any other recent model.

METHOD

Methodology and instructions followed the Hamilton and Gifford (1976) paradigm. Table 1 summarizes the distribution of the behavioral information for the 4 groups.

Table 1

Number of desirable and undesirable behaviors assigned to each group

Group :	A	B	C	D
Desirable behaviors	16	8	4	2
Undesirable behaviors	8	4	2	1

Subjects sat at individual computers and were told that the experiment concerned "the way people process and retain information". Furthermore they were told that they would receive information concerning four groups (A, B, C and D), these groups represented groups in the real world and that group A was bigger than group B, group B bigger than group C and so on. Finally they were told to read each statement carefully. Each statement remained

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

on the screen until the subject pushed the space bar. After reading all statements, subjects completed a filler task, a free recall task, a group assignment task, a frequency estimation task and a group evaluation task.

RESULTS

Overall, the results confirmed our hypotheses. We expected that groups with more members (or behaviors) would be less subject to illusory correlation. Specifically this means that as the groups became smaller, group evaluations would become less favorable and relatively more undesirable behaviors would be attributed to these groups.

Likability Ratings. The main effect of group was significant, $F(3, 72) = 4,62$, $p < ,005$, revealing as expected that groups were rated less favorable as they became smaller.

Frequency Estimation. There was no significant main effect of group ($p > ,1$). However, contrast analyses show that subjects tended to attribute less undesirable behaviors to Group A than to other groups, $F(1, 24) = 3.93$, $p < .06$. This might indicate that only for group A the association between group membership and desirability was well established, enabling subjects to make a fairly accurate judgment.

Group Assignment. Analyses showed that subjects were more likely to assign desirable as opposed to undesirable behaviors to group A, $F(1,24) = 4.419$, $p < .05$. This confirms our prediction that for group A the associations between group membership and desirability are strong and therefore easy to discriminate. As subjects experienced more desirable group A behaviors then undesirable, the association between group A and desirable behavior is stronger than the association with undesirable behavior, leading to a tendency to assign more desirable behaviors to group A. The contrast analyses show the reverse effect for group D, in that more undesirable as opposed to desirable behaviors were assigned to this group, although this was only marginally significant, $F(1,24) = 3.841$, $p = .06$. This is probably due to the fact that there was only 1 undesirable behavior in group D, which would have made it very distinctive.

Free Recall. Two separate proportions were used : General free recall reflects the recalled behaviors regardless of whether they were correctly associated with a group. Correct free recall reflects only those behaviors correctly assigned to a group.

With respect to general free recall we see as predicted that relatively more undesirable behaviors are attributed to groups B,C and D in comparison with group A, $F(1, 57) = 5.11$, $p < .03$. This can be due to the overall response bias to attribute negative behaviors to smaller groups in parallel with the likability of those groups. As stated before, this confirms our prediction that the associations between group membership and desirability are weak for the smaller groups, leading to illusory correlations.

There is however another possible explanation. According to our connectionist model, during learning strong associations tend to suppress weaker associations (competition effect). For instance, that would mean that the strong association between group membership and desirability for group A would suppress the associations of the unique behaviors with that group. This would be less the case for the smaller groups, where the associations between group membership and desirability are weaker. As a result, more unique behaviors should be recalled by the subjects as the groups become smaller. In fact this is partly confirmed by the data for correct free recall : undesirable behaviors were recalled better than desirable for group D, $F(1, 57) = 5.82$, $p < .03$. However, the data for the other groups show no sign of this competition effect as recall is weak in all these cells. This is nonetheless an important aspect of connectionist learning models, as it easily explains distinctiveness effects. Hence further research into this matter is required.

REFERENCES

Chapman, L.J. (1967). Illusory correlation in observational report. *Journal of Verbal Learning and Verbal Behavior, 6,* 151—155.

Fiedler, K. (1991). The tricky nature of skewed frequency tables : An information loss account of distinctiveness-based illusory correlations. *Journal of Personality and Social Psychology, 60,* 24—36.

Hamilton, D. L., & Gifford, R.K. (1976). Illusory correlation in interpersonal perception : A cognitive basis of stereotypic judgment. *Journal of Experimental Social Psychology, 12,* 392—407.

Smith, E. R. (1991). Illusory correlation in an simulated exemplar-based memory. *Journal of Experimental Social Psychology, 27,* 107—123.

Van Overwalle, F. (1998) Causal Explanation as Constraint Satisfaction : A Critique and a Feedforward Connectionist Alternative. *Journal of Personality and Social Psychology,* in press.

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

Analogical Problem Solving by Adaptation of Schemes

Ute Schmid

Department of Applied Computer Science, Technical University Berlin
FR 5-8, Franklinstrasse 28, 10587 Berlin, Germany
+49 30/314-23938, schmid@cs.tu-berlin.de

Abstract. We present a computational approach to the acquisition of problem schemes by learning by doing and to their application in analogical problem solving. Our work has its background in automatic program construction and relies on the concept of recursive program schemes. In contrast to the usual approach to cognitive modelling where computational models are designed to fit specific data we propose a framework to describe certain empirically established characteristics of human problem solving and learning in a uniform and formally sound way.

1 Introduction

The use of analogies is a powerful and ubiquitous strategy in human reasoning and problem solving. A lot of (symbolic, connectionist & hybrid) computational models have been proposed with the aim of getting more precise insights in the underlying processes (Anderson & Thompson, 1989; Falkenhainer, Forbus, & Gentner, 1989; Hummel & Holyoak, 1997) and with the aim of exploiting this strategy in AI applications (cf. case based reasoning).

Most of the computational models are focusing on analogical access and mapping thereby neglecting two crucial aspects of analogical problem solving: (1) generation of problem representations which are suitable for analogical problem solving (i.e. problem schemes), and (2) solving a target problem by adapting a - not necessarily isomorphical - source problem.

The model proposed by Anderson and Thompson (1989), for example, relies on schemes for representing the structure of problems and solutions which are available to the system from the beginning. Thereby the authors suppose that the system has already knowledge about the structure of the problem domain. But the crucial deficit of novices is that they have *no* knowledge about the structural characteristics relevant for problem solving (Novick, 1988; Schmid & Kaup, 1995). Otherwise, there would be no need for analogical problem solving. The problem could be solved by applying already acquired automatisms (production rules) or abstract schemes.

The examples Anderson and Thompson (1989) give for analogical transfer are restricted to generalized problem isomorphs, i.e. identical structures where predicate and operation symbols can be substituted in a unique way. There is no statement whether the model could be extended to adaptation of non-isomorphical structures. In everyday reasoning, availability of isomorphical source problems is the exception. Empirical studies demonstrate that people also *can* use partially isomorphical source problems (Pirolli & Anderson, 1985; Schmid & Kaup, 1995).

We are proposing a framework for analogical problem solving which overcomes the limitations described above: First we present our concept of problem schemes and a method for inferring such schemes from problem solving experiences. Than we describe our approach to analogical transfer which works for both isomorphical and non-isomorphical source problems.

2 Induction of Problem Schemes

The central concept of our approach is the notion of recursive program schemes (RPSs; see Schmid & Wysotzki, 1998 for the formal definitions). An RPS represents the structure of a problem as (recursive) equation. On the left side the name of the RPS and its parameters are given. The right side represents a operations together with their conditions for application. An RPS representing the knowledge of clearing a block is

clear-one-block(x, s) = if cleartop(topof(x)) then puttable(topof(x)) else s.

The variable s ("situation variable") represents the current problem state (for example *on(A, B), on(B, C), cleartop(A)*). This RPS can only be applied if *one* block is lying on block x. For the problem state given above it can be applied to block B only. An RPS representing the knowledge of clearing an arbitrary block in a tower is

clearblock(x, s) = if cleartop(x) then s else puttable(topof(x), clearblock(topof(x, s))).

The representation format of an RPS simultaneously catches the *structure* of a problem and its executable *solution strategy* (cf. Rumelhart & Norman, 1981).

In our program IPAL (Schmid & Wysotzki, 1998) we are modelling the acquisition of RPSs by a two-step process: In a first step some initial states of a problem are solved by applying predefined production rules using heuristic search. That is, without experience in a problem domain the system has to use a general purpose strategy which can be inefficient because search may lead to dead ends and there is need for backtracking. The solution sequences found for the initial states are composed into a so called initial program generalizing over the application con-

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

ditions of the predefined production rules and over the constants occuring in the initial states.

An initial program which represents the experience with towers up to three blocks is

```
if cleartop(x) then s else
 if cleartop(topof(x)) then puttable(topof(x),s)
  else if cleartop(topof(topof(x)))
   then puttable(topof(x),
         puttable(topof(topof(x)),s)).
```

Building initial program corresponds roughly to chunking and refinement of production rules.

By using a method for inductive program synthesis initial programs can be generalized to RPSs (Schmid & Wysotzki, 1998). The general idea of our algorithm is to identify a pattern and a substitution in the initial program which makes it possible to reproduce the whole structure. For the initial program given above the pattern is *if cleartop(x) then s else puttable(topof(x), m)* with the substitution $x \leftarrow topof(x)$. If found, the pattern and substitution are extrapolated to an RPS. This process describes a fundamental aspect of human intelligence: the ability of induction as for example described by (Holland, Holyoak, Nisbett, & Thagard, 1986).

3 Transformation Based Adaptation

RPSs formally are elements of a term algebra. That means, they represent syntactical structures only. The semantics of an RPS is gained by interpretation of the symbols in accordance to some domain model. Thereby an RPS represents the class of all structurally identical problems. This is a characteristic extremely suitable for analogical reasoning.

In IPAL already inferred RPSs are stored in memory. An RPS can be "unfolded" to an initial program again. If a new initial program is gained by investigating some problem states, the memory is checked for an RPS whose corresponding initial program is similar to the new one. In that case inference of a general solution strategy for the problem (represented by an RPS) can be omitted and the known RPS can be adapted to the new problem instead.

In our approach mapping and adaptation is performed by means of tree transformation. An initial program of an already known RPS is transformed to a new initial program by substitution, insertion and deletion of symbols. The set of transformations can than be applied to adapt the known RPS. Two initial programs are isomorphical if one can be transformed into the other by a set of unique substitutions only. We give an example of adaptation in the non-isomorphic case (see fig. 1). To transform "clearblock" into "factorial" we have to perform the unique substitutions *cleartop/equal0*, *s/1*, *puttable/mult*. Additionally we have two transformations for the "topof" symbol: substitute *topof/pred* and delete *topof*. By using contextual information, we can decide at which position in the RPS *topof* has to be deleted (in the first argument of *puttable* rsp. *mult*).

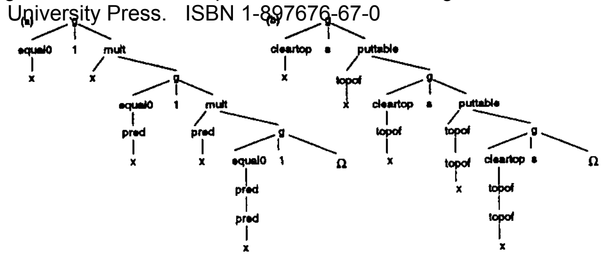


Fig. 1. Initial programs for (a) factorial and (b) clearblock ("g" represents the conditional "if-then-else"; Ω stands for "undefined")

4 Discussion and Further Work

We believe that our framework can be of use for the cognitive modeling community for two reasons: (1) it addresses the problem of scheme acquisition often neglected in computational approaches, (2) the notion of RPSs as representation format makes it possible to describe the structural characteristics of problems in a way which makes it possible to perform precise analyses of the structural similarity of problems. This can be useful to construct source-target relations for empirical studies of the conditions of successful adaptation in analogical problem solving.

References

Anderson, J., & Thompson, R. (1989). Use of analogy in a production system architecture. In S. Vosniadou & A. Ortony (Eds.), *Similarity and analogical reasoning* (p. 267-297). Cambridge University Press.

Falkenhainer, B., Forbus, K., & Gentner, D. (1989). The structure mapping engine: Algorithm and example. *Artificial Intelligence, 41*, 1-63.

Holland, J., Holyoak, K., Nisbett, R., & Thagard, P. (1986). *Induction - processes of inference, learning, and discovery.* Cambridge, MA: MIT Press.

Hummel, J., & Holyoak, K. (1997). Distributed representation of structure: A theory of analogical access and mapping. *Psychological Review, 104*(3), 427-466.

Novick, L. R. (1988). Analogical transfer, problem similarity, and expertise. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 14*, 510-520.

Pirolli, P., & Anderson, J. (1985). The role of learning from examples in the acquisition of recursive programming skills. *Canadian Journal of Psychology, 39*, 240-272.

Rumelhart, D. E., & Norman, D. A. (1981). Analogical processes in learning. In J. R. Anderson (Ed.), *Cognitive skills and their acquisition* (p. 335-360). Hillsdale, NJ: Lawrence Erlbaum.

Schmid, U., & Kaup, B. (1995). Analoges Lernen beim rekursiven Programmieren (Analogical learning in recursive programming). *Kognitionswissenschaft, 5*, 31-41.

Schmid, U., & Wysotzki, F. (1998). Induction of recursive program schemes. In *Proceedings of the 10th European Conference on Machine Learning (ECML-98).* Springer.

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

Toward a theory of the control of dynamic systems

Wolfgang Schoppek
Department of Psychology
University of Bayreuth
D-95440 Bayreuth,Germany
+49 921 555003
wolfgang.schoppek@uni-bayreuth.de

ABSTRACT

A categorization of three types of knowledge which can be relevant for the control of dynamic systems is suggested. These are (1) input-output knowledge, (2) structural knowledge which is subdivided in knowledge about effects and knowledge about dependencies, and (3) strategic knowledge. The assumptions are embedded in the theoretical framework of the ACT-R theory. An ACT-R model of the early stages of knowledge acquisition, and its implications for future research are described.

Keywords

knowledge acquisition, causal relations, ACT-R, dynamic system

INTRODUCTION

This contribution deals with the control of dynamic systems of the following type: There are about 2-4 input-variables which are exclusively controlled by the problem solver, and about the same number of output-variables whose values depend on the values of input- and output-variables. The systems are modelled by simultaneous linear equations. In order to minimize the variability of domain specific knowledge, the variables have phantasy names. As a consequence, only general prior knowledge, e.g. knowledge about causal relations, can be brought to bear in the problem solving process. Fig. 1 shows a simple example of such a system.

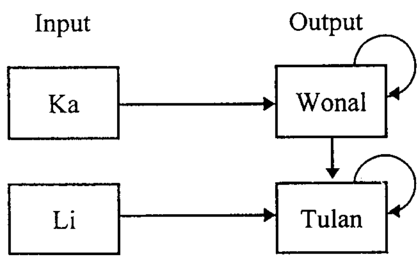


Fig. 1: A simple dynamic system

The control of dynamic systems is a form of complex problem solving. Unlike many other problem solving tasks, the effects of the operators are not explained in the instructions. The problem solver has to induce them by analyzing self generated state-action-state sequences.

Many authors assume, that controlling systems effectively requires structural knowledge. The notion of structural knowledge comprises knowledge about the variables and their causal relationships. But the results concerning the relation between structural knowledge and control performance are inconsistent. In some studies subjects report considerable structural knowledge, but fail to attain the goals for system control (Schoppek, in prep.). In other studies subjects are successful in controlling the system but can hardly report anything about its structure (Berry & Broadbent, 1984). There is, however, also evidence for a convergence of structural knowledge and control performance (Funke, 1992). It is obvious, that the construct of structural knowledge is too undifferentiated to account for the diversity of the results.

TYPES OF KNOWLEDGE FOR SYSTEM CONTROL

As a step towards an integrative explanation of these results I want to suggest a theoretical distinction of three different types of knowledge which can be relevant for the control of dynamic systems.

(1) Input-output knowledge (I-O-knowledge) represents interventions and their effects. These may be stored either external or in declarative memory. In early exploration phases I-O-knowledge is the material from which structural knowledge is induced. With extended practice, successful I-O-sequences can be recalled directly from declarative memory. A third possibility of using I-O-knowledge is the successive adjustment of an input-pattern without any induction of general rules.

(2) Structural knowledge is subdivided in two types: knowledge about effects (E-knowledge) and knowledge about dependencies (D-knowledge). E-knowledge is supposed to be acquired from an early stage of practice with the system. It can be induced quite easily from state-action-state sequences, provided that an appropriate input-strategy is applied. E-knowledge can be represented by solitary chunks. It is sufficient to answer most of the questionnaires that have been used to assess structural knowledge.

But the exact control of a dynamic variable requires knowledge about its dependencies. It is possible to search memory for all E-chunks containing the goal-variable in its output slot, but this is an error-prone procedure. In this situation an output-centerd integration of E-knowledge would be more effective. This is the hypothetical D-knowledge. Successful problem solvers seem to have access to this type of knowledge since they have no difficulties in quickly considering all dependencies of an output variable. D-knowledge can be deducted from E-knowledge, but this is an additional process. Thus deduction and use of D-knowledge takes more effort than induction of E-knowledge.

(3) Strategic knowledge comprises knowledge about how to acquire structural knowledge, (e.g. the strategy of isolated variation of conditions), and knowledge about certain input-strategies (e.g. the compensation of side-effects).

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

The three types of knowledge are differing in their generalizability. I-O-knowledge is only applicable for a single system and is goal specific. Structural knowledge refers to a single system, too, but is unspecific with respect to the goal states. Finally, strategic knowledge can be applied in the exploration and the control of many different systems.

THEORETICAL INTEGRATION

The assumptions are embedded in the theoretical framework of the ACT-R theory (Anderson, 1993). All the types of knowledge are supposed to consist of both declarative and procedural elements, whose parameters change with use according to ACT-R. Thus the theoretical distinction could serve as a link between the content-independent assumptions of the ACT-R theory and more specified models of system control.

EMPIRICAL SUPPORT

The assumptions are largely consistent with the data. Dissociations between verbalizable knowledge and control performance can be explained by the notion that most tasks for assessing structural knowledge can be solved with E-knowledge whereas successful system control requires more than access to single E-chunks. Findings that initial dissociations disappear with extended practice (Sanderson, 1989) are also in line with this explanation. Seemingly inconsistent results of tutoring structural knowledge, which were found in experiments of our workgroup are interpretable in terms of different focuses of the training procedures. A training which focused on D-knowledge (Preußler, 1997) lead to improved control performance whereas a training which focused on E-knowledge did not (Schoppek, in prep.).

ACT-R MODEL

I started to put these deliberations into practice in form of an ACT-R model which handles the static system depicted in fig. 2. At present the model is able to explore the system. It induces positive effects on the base of self generated data and creates single E-chunks for every detected effect. With this knowledge the model can produce judgements about effects in a fact-retrieval paradigm. Finally the model can use its E-knowledge to obtain simple goal states.

The main problem in this early stage of model construction is to find an appropriate representation of new causal knowledge. As indicated above, the model creates a new chunk for every detected effect. The chunk-type has three slots: „input", „output", and „factor". This takes into account that judgements about causal relations cannot be explained by the assumption of simple associations between cause and effect (Waldmann, 1996).

In the fact-retrieval task the model exhibits no effect of the number of outputs that are affected by an input (e.g. judgements of „Eltan-Ordal" and „Bulmin-Fontil" take the same time, although Bulmin affects only one output whereas Eltan affects three).

In a preliminary experiment five subjects explored the static system shown in fig. 2 and then processed the fact-retrieval task with pairs of variable-names. In contradiction to the model, there seems to be a fan effect: The judgements for the effects of input „Eltan" (fan 3) take longer than the judgments for the effects of „Bulmin" and „Dulan" (fan 1). This might, however, be due to the fact, that the judgements were based on a secondary verbal representation of a rather sensorimotor primary representation of the effect. Indeed, four of the five subjects reported that they memorized the effects in terms of locations and that memorizing the names was an additional demand. In the main experiment it will be tested if there are different effects depending on the presentation of spatial cues.

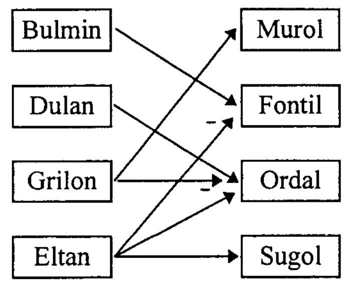


Fig.2: Static system controlled by the model

OPEN QUESTIONS

Thus even the initial representation of single causal relations can be regarded as an open question. A more serious problem is posed by the question, how the hypothetical D-knowledge is transformed into productions. Experienced problem solvers obviously dispose of such fairly complex productions.

Despite all those open questions I hope to have pointed out that there is a long way between the acquisition of single effect-chunks, including their application in fact-retrieval tasks, and the integrated use of this knowledge for the determination of input-values in order to obtain specific goal states.

REFERENCES

Anderson, J. R. (1993). *Rules of the mind.* Hillsdale, NJ: Lawrence Erlbaum Associates

Berry, D. C., & Broadbent, D. E. (1984). On the relationship between task performance and associated verbalisable knowledge. *The Quarterly Journal of Psychology, 34A*, 209 -231.

Funke, J. (1992). Dealing with dynamic systems: research strategy, diagnostic approach and experimental results. *The German Journal of Psychology, 16*, 24-43.

Preußler, W. (1997). Zur Rolle expliziten und impliziten Wissens bei der Steuerung dynamischer Systeme. *Zeitschrift für Experimentelle und Angewandte Psychologie, 43*, 399-434.

Sanderson, P.M. (1989). Verbalizable knowledge and skilled task performance: association, dissociation, and mental models. *Journal of Experimental Psychology: Learning, Memory, and Cognition,. 15*, 729-747.

Waldmann, M. R. (1996). Knowledge-based causal induction. *The Psychology of Learning and Motivation, 34*, 47-88.

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

A semi-symbolic cognitive model of usage polysemy

Sylvain Surcin
E.C.Art. / L.R.I.A. – University of Paris VIII
21, rue Baudelique
75018 Paris, France
tel.: +33 1 44 92 83 24
surcin@winimage.com

INTRODUCTION

The status of polysemy as a source of lexical ambiguities is still not clear neither among linguists, nor among computational semanticists. Here, we take a step outside of the classical debate among homonymy and vagueness and we postulate that polysemy is at the origin of most of the cases of ambiguity.

In this paper, we focus on a particular kind of polysemy that we call *usage polysemy*. Unlike other kinds of polysemy, usage polysemy cannot be reduced to operations of sense composition and selection. Usage polysemy takes place when a polysemous lexical unit has several closely related interpretations, corresponding to different uses and none can be said to be 'the right one'. It will be further defined in section THEORETICAL AND EMPIRICAL APPROACHES. Our aim is to design a cognitive model for the computation of usage polysemy, and to implement it as an expert agent cooperating with other agents in a Natural Language Processing (NLP) architecture.

We present our model in the PELEAS MODEL section. We designed it on the main postulate that interpreting usage polysemy is a process similar to translating an ambiguous expression. We also present the set of software pieces we developed around our model, along with a qualitative evaluation we have conducted at the time being. At last, in the CONCLUSION section, we give our temporary conclusion about our model.

THEORETICAL AND EMPIRICAL APPROACHES

A widely spread opinion among computational linguists is that polysemy is a false problem, and the ambiguities it generates are but artefacts produced by our models. The argument is that we, human beings, never fail when interpreting polysemy. But what to think about sentences like "The mother cell splits into two new identical cells"? Which is the right interpretation for "mother": generating, antecedent, prior, ruling or causal source? As a matter of fact, a human reader does not feel annoyed when reading such a sentence, because he/she unconsciously handles all the different interpretations simultaneously. We will show thereafter that this example falls in a particular category of polysemy we call *usage polysemy*, which, indeed, is not a problem once we do not require *the* right interpretation for a polysemous word.

Different kinds of polysemy

Lexical ambiguity has been abundantly studied and modelled by computational linguists. But what is usually referred to as 'polysemy' is described as *functional polysemy* by Prince and Bally-Ipsas (1991). It involves semantic features as much as syntactic ones in order to resolve the lexical ambiguities it generates, by restricting the selection of the concept which is compatible with the context. An other category of polysemy is described by Rastier (1996) as *sense polysemy*[1] and involves linguistic devices known as *isotopy* and *isosemy* in differential semantics. The last kind of polysemy we can distinguish involves also sociolinguistics data, as conventional uses of words, tropes and topoï. This is precisely this category we study here.

Usage polysemy

Our framework is composed of polysemous word occurrences for which there are no syntactic / semantic necessary and sufficient conditions, nor intralinguistic isotopy relationships allowing us to discriminate between the different possible interpretations. This means that all interpretations are closely related conceptual *points of view* on the word's meaning. They differ only by slight shades of meaning for the word's *usage*. These shades can be established in discourse, on a cultural basis.

Such phenomena have been observed by Tanaka and Umemura (1994) to occur frequently for *common words* (representing approximatively 30% of the lexicon for any given language). Common words are not terms: they are not used as items of a nomenclature but rather in the everyday discourse.

Usage polysemy of common words may arise in a various set of situations: (i) *usage transfers*: when a word is used outside of its most usual application field, mostly in order to illustrate a technical concept; (ii) *deliberate sense overlapping*: when an author play with the lexical ambiguity due to polysemy in order to describe a complex situation in a limited textual space; (iii) *joker words*: when a common word is so much used inside a linguistic community that its semantic contents becomes too generic; and (iv) *plays on words* referring to cultural references shared by the locutors.

Interpretation and translation

The most adapted linguistic theory for studying usage polysemy seems to be the differential semantics theory. However, it is too fuzzy to be implemented straight away, and does not account for the influence of sociolinguistic data on the behaviour of lexical units. Sticking to the interpretation paradigm of the differential theory, we

[1] The original 'polysémie d'acception' could be better translated into 'polysemy of linguistic aspects of the senses'

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0.

postulated that interpretation is similar to translation in a certain way. That is why we observed a team of technical translators resolving problems raised by cases of usage polysemy. Their procedure seems to be incremental and hierarchical: (i) to find a general semantic direction by probing the global context, (ii) to restrict the set of possible interpretations by finding textual markers in the local context, (iii) to list valid and plausible interpretations by using inhibition and reinforcement, and (iv) to produce a synthesised translation.

THE PELEAS MODEL

The model we designed is called PELEAS (Pyramids and Ellipses as Lexical Entries in Ambiguous Sentences). It is a lexicon driven by lexical entries, but each entry owns semasiological substructures.

Description of the model

The model was designed as a *dynamic lexicon*: it does not contain all possible interpretations of a word, but rather computes them from a minimalist static representation of well acknowledged uses. That is why it is constituted of a static part (this representation) and a dynamic part, which handles the salience attribution process. This model is in the same trend of representations as the Generative Lexicon of Pustekovsky (1991) and Edgar of Prince (1994) Our model differs from the Generative Lexicon because it does not try to specify the relationships between a word and its description further than "the descriptors of a word lexically co-occur in the close context of its occurrences". It also differs from Edgar by taking the sociolinguistics context into account, and by allowing a kind of variable depth reasoning.

Each entry is stored as a hierarchical graph where each level corresponds to a particular kind of description: (i) *notions* are 'general semantic directions', (ii) *domains* mark the influence of the activity fields on the discourse, (iii) *conceptual views* are partial concepts, and (iv) *features* are pertinent properties of these concepts. Included in the static representation are *contextual conditions* and *semantic constraints*. Contextual conditions are a set of rules for initial salience attribution corresponding to very particular and well-known influences of some morpho-syntactic markers for *this* entry interpretation.

The edges between a parent node and its children nodes correspond to an *is-described-by* or *is-specialised-by* relationship. The semantic constraints are the edges between sibling nodes. They can be either neutral (co-validity of descriptions), reinforcement connectors (implication / increase of salience between two nodes), or inhibition connectors (opposition / decrease of salience).

Interpreting a polysemous word becomes, in our model, attributing salience rates to each node of the lexical structure. We use four symbolic rates: (i) *ignored*, meaning "not pertinent in this context", (ii) *valid*, meaning "possible but not very important", (iii) *salient*, meaning "important" and (iv) *negated*, which means "important but in a negative way". We use a salience propagation algorithm, initiated by the triggering of the contextual conditions. This algorithm is similar to the resolution of a system of non-linear recurring equations of k variables, which converges in k steps, if k is the number of descriptors for nodes in the descriptive structure of a lexical entry. It terminates, in the worst case, in as many steps as there are nodes in the descriptive structure.

Implementation of the model

We have implemented this model in a pack of three software pieces: first, an engine, LightPeleas, managing the descriptive structures of a lexicon and applying the propagation algorithm on request. Then, a graphical editor, Melisande, to build and modify descriptive structures. And finally, Bard, a corpus parser that helps up to gather raw material for building the descriptive structures.

We implemented the engine LightPeleas as an ActiveX control. It publishes in the operating system 29 classes allowing the manipulation of any item from an entry to a single node or edge. Entries are stored on disk in a format we called PDL (Peleas Description Language). In order to help us use the interpretation given by LightPeleas for an entry, the output is a set of salient or negated conceptual views pondered by a "hint" between 0 and 1. It evaluates the plausibility of each interpretation (0 means 'perhaps', and 1 means 'rather sure')

So far, we used our system to build five descriptive structures (for 'mother', 'father', 'to devour', 'life' and 'little') and conducted a test with twenty-two sentences. The results we obtained were all sets of propositions with meaningful interpretations for the first or two first 'guesses'.

CONCLUSION

So far, we have delimited a kind of lexical ambiguities and their sociolinguistic cause: usage polysemy. We designed a model for its processing based on the observation of some translators' behaviour. This model is implemented and presents encouraging results so far.

REFERENCES

Prince, V., and Bally-Ipsas, R. (1991). *Un algorithme pour le transfert de règles pragmatiques dans le processus complexe GLACE*. Rapport interne du LIMSI n° 91-17, Université Paris XI, Orsay.

Prince, V. (1994). A discrete approach based on logic simulating continuity in lexical semantics. In *Continuity in Linguistic Semantics*, Fuchs, C. and Victorri, B. (eds). *Linguisticae Investigationes Supplementa*, vol. 19. John Benjamins Publishing Company, Amsterdam/Philadelphia.

Pustejovsky, J. (1991). The Generative Lexicon. In *Computational Linguistics*, vol. 17(4).

Rastier, F. (1996). *Sémantique interprétative*, 2nd ed. Presses Universitaires de France, Paris.

Tanaka, K., and Umemura K. (1994). Construction of a Bilingual Dictionary Intermediated by a third Language. In *Proceedings of COLING-94*, Kyoto, Japan.

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

ESQIMO : Modeling Analogy with Topology

Erika Valencia, Jean-Louis Giavitto, Jean-Paul Sansonnet
LRI ura 410 CNRS, Bâtiment 490 Université Paris-Sud,
91405 Orsay Cedex, France
+33 (0)1 69 15 42 25
{erika,giavitto,jps}@lri.fr

ABSTRACT

ESQIMO is a computational model for analogy solving based on a *topological formalism of knowledge representation*. The source and the target analogs are represented as *simplicial complexes* and the analogy solving is modeled as a topological *deformation* of these complexes along a polygonal chain.

Key Words: Analogy solving, Algebraic topology, Simplicial complexes, IQ-tests.

TOPOLOGY FOR KNOWLEDGE REPRESENTATION

A representational formalism for analogy must allow the explicit expression of the features involved in similarity. M. Johnson (Johnson, 1987) argues that mental images are too close to perception and that logic approaches are too syntactic and arbitrary for representational purposes. He proposes to use a topological structure to represent and solve metaphors (which he considers to be the generalization of analogies (Lakoff and Johnson, 1980)).

Simplicial Complexes

Cognitive models use different models of space (Freska, 1997; Johnson, 1987) and the central question is in the choice of the basic spatial entities in a spatial representation of knowledge. We take here the elementary spatial entities to be *simplicial complexes*.

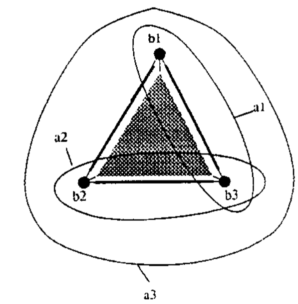
A simplicial complex is a couple (V, K) where V is a set of elements called vertices and K is a set of finite parts of V such that if $s \in K$, then all the parts $s' \subseteq s$ belongs also to K. The elements of K are called simplexes. The dimension of a simplex s is equal to $Card(s) - 1$. *All complexes with dimension* < 2 *are graphs.* Thus, simplicial complexes generalize semantic networks and allow the expression of hierarchies like in a relational graph.

The Q-Analysis

Atkin proposed the **Q-Analysis** (Atkin, 1981) to represent a binary relation λ between two sets with a simplicial complex. Let Λ be the incidence matrix of a binary relation $\lambda \subset A \times B$. Let $a \in A$, the set S_A of b_i such that $(a, b_i) \in \lambda$. All the elements b_i can be taken as vertices to represent the element a as a simplex. The whole matrix Λ can then be represented as a simplicial complex containing all the simplexes representing each element $a_i \in A$, we note it $K_A(B, \lambda)$ (see figure 1). Likewise, we can represent Λ^{-1} with the dual simplicial complex.

λ	a_1	a_2	a_3
b_1	1	0	1
b_2	0	1	1
b_3	1	1	1

(a) Incidence matrix of the binary relation λ



(b) Simplicial representation of λ

Figure 1: Representation of a binary relation

Extension of Q-Analysis

We extend the Q-Analysis to allow the representation of sets of predicates as a simplicial complex too. We can take a set of predicates $P = \{p_1, ..., p_n\}$ and represent the binary relation $\lambda \subset A \times P$ such that $(a_i, p_j) \in \lambda$ if $p_j(a_i)$ holds.

In this representational formalism, the same simplex is associated to elements of A that cannot be distinguished with the predicates of P available in the system. Moreover, two simplexes that have a smaller k-simplex in common are said to share a k-face. In terms of representation, it means that they have k features in common.

THE ESQIMO SYSTEM

A representational system is composed of a data structure and programs operating on it corresponding to reasoning tasks. We try now to model a simple analogy solving task using the representational structure proposed before : we chose the typical IQ-test problem. The system has to find an element D such that it completes a four-term analogy with three other given elements A, B and C.

The analogy is solved in 3 steps: find a relation R_{AB} between A and B, find the domain of C to apply R_{AB}, build $D = R_{AB}(C)$.

Representing the Problem

IQ-tests are given in terms of geometrical elements so that they can express many properties and stay simples. We

Ritter, F. E., & Young, R. M. (Eds.). (1998). Proceedings of the Second European Conference on Cognitive Modelling. Thrumpton (UK): Nottingham University Press. ISBN 1-897676-67-0

took the following properties of *shape*: round, square, triangular; *color*: white, black; and *size*: big, small. According to our formalism, we build the complex $\mathcal{C}(\Omega)$ representing all the properties. The figures of the test are seen as relations between the set of properties and the elements of the figures, so we represent them also as sub-complexes $\mathcal{C}(A), \mathcal{C}(B)$ and $\mathcal{C}(C)$ of $\mathcal{C}(\Omega)$. Note that this formalization does *not* depend on the geometrical nature of the elements.

ESQIMO's Algorithm

To find R_{AB} we look for a transformation T_{AB} between $\mathcal{C}(A)$ and $\mathcal{C}(B)$ along a polygonal chain from $\mathcal{C}(A)$ to $\mathcal{C}(B)$ in $\mathcal{C}(\Omega)$. A *Polygonal chain* is a sequence of simplexes belonging to the same complex and where two successive simplexes have a non empty intersection. An elementary step linking σ_i to σ_{i+1} in a chain is then viewed as an elementary transformation $T_{\sigma_i,\sigma_{i+1}}$.

If there are several such chains, then there are several possible relations between A and B. To minimize the number of solutions, we give a higher priority to chains that are short and of higher dimension. Indeed, they correspond to transformations with less steps, and with more properties conserved at each step.

When T_{AB} is found, we use the same algorithm to determine T_{AC}. This second transformation is used to determine the domain of $\mathcal{C}(C)$ on which we can apply T_{AB}. Several strategies have been implemented (Valencia, 1997) considering only the things that changed between $\mathcal{C}(A)$ and $\mathcal{C}(C)$, or considering only the invariants between them, or some other hybrid methods. Finally, we can apply T_{AB} to this domain and build $\mathcal{C}(D)$. The translation of $\mathcal{C}(D)$ into a geometrical element of the universe is then easy.

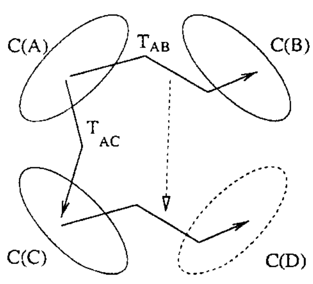


Figure 2: Analogy solving with ESQIMO

Remarks

The description of the properties of each figure in terms of predicates can be a problem for properties such as position. In that case, we can take only relative positions into account. Moreover, our transformations could be called 0−degree since they preserve the minimum of topological properties along a chain. The next step of this modelization would be to pair higher-order structures.

CONCLUSION

Different computational models have been developed to model analogy solving and are based on different representational structures. Among them, the ANALOGY system proposed by Evans (Evans, 1968) uses rules, the SME system proposed by Falkenhainer to illustrate Gentner's theory for analogy (Falkenhainer et al., 1989; Gentner, 1983) uses propositional structures, the ARCS system developed by Thagard and Holyoak to simultaneously satisfy the structural, semantic and pragmatic constraints uses neural networks and COPYCAT uses semantic networks with asynchronous parallelism. Like in SME, we focused on the structural constraint introduced by Gentner (Holyoak and Thagard, 1989) and we modeled the steps of analogy solving like in the ANALOGY system.

Our contribution lies in the search for a new representational structure (Valencia, 1997) that can be justified in terms of the naturality of a diagrammatic representation (Glasgow et al., 1995). Like in the COPYCAT project, we are concerned with the mechanisms of enrichment of a representation through analogy and our formalism can be seen as an intermediate structure between a symbolic and an analogical approach.

ESQIMO has been implemented in the ML programming language, the various strategies experimented and some solving examples are given in details at `http://www.lri.fr/~erika`.

REFERENCES

Atkin, R. H. (1981). *Multidimensional Man*. Penguin.

Evans, T. G. (1968). A program for the solution of a class of geometric analogy intelligence-test questions. In *Semantic Information Processing*, chapter 5, pages 271–353. The MIT Press.

Falkenhainer, B., Forbus, K. D., and Gentner, D. (1989). The structure-mapping engine. *Artificial Intelligence*, 41(1):1–63.

Freska, C. (1997). Spatial and temporal structures in cognitive processes. In Freska, C., Jantzen, M., and Valk, R., editors, *Foundations of Computer Science*, volume 1337 of *LNCS*. Springer-Verlag.

Gentner, D. (1983). Structure-mapping a theoretical framework for analogy. *Cognitive Science*, 7(2):155–170.

Glasgow, J., Narayanan, N. H., and Chandrasekaran, B. (1995). *Diagrammatic reasoning : Cognitive and Computational Perspectives*. AAAI Press/MIT Press.

Holyoak, K. J. and Thagard, P. (1989). Analogical mapping by constraint satisfaction. *Cognitive Science*, 13(3):295–355.

Johnson, M. (1987). *The Body in the Mind*. The University of Chicago Press.

Lakoff, G. and Johnson, M. (1980). *Metaphors we live by*. The University of Chicago Press.

Valencia, E. (1997). Un modèle topologique pour le raisonnement diagrammatique. Rapport pour le DEA Sciences Cognitives, LIMSI. See also `http://www.lri.fr/~erika`.