CHREST Models of Implicit Learning and Board Game Interpretation

Peter Lane[1] and Fernand Gobet[2]

[1] School of Computer Science, University of Hertfordshire, Hatfield AL10 9AB, UK
peter.lane@bcs.org.uk
[2] School of Social Sciences, Brunel University, Uxbridge UX8 3BH, UK
fernand.gobet@brunel.ac.uk

Abstract. A general theory of intelligence must include learning, the process of converting experiences into retrievable memories. We present two CHREST models to illustrate the effects of learning across two different time scales (minutes and years, respectively). The first is an illustration of implicit learning, checking the validity of strings drawn from an artificial grammar. The second looks at the interpretation of board-game positions. The same learning and retrieval mechanisms are used in both cases, and we argue that CHREST can be used by an artificial general intelligence to construct and access declarative memory.

1 Introduction

The processes behind the acquisition and retrieval of patterns remain a major challenge for theories of artificial intelligence. Pattern recognition, and its role in categorising and interpreting perceived information for later cognition, is an important element of high performance in many domains, especially of a problem-solving nature. A dramatic example of the power of human memory is provided by the famous encounter in 1996 between IBM's Deep Blue computer and the then world chess champion, Gary Kasparov. The computer relied in part on an extensive process of search, eight magnitudes greater than what the human could achieve, and yet the matches ended in a tie. The human's advantage over the computer was his large declarative and procedural memory, built up over 20 years of dedicated chess experience.

Humans' reliance on prior experience is apparent in many situations, even as everyday as perceiving a string of letters when reading. For a native reader, a string of letters may provide detailed information; for someone who does not know the language, the same letters may well be meaningless. Similarly, an expert in any domain will rapidly interpret and categorise stimuli from that domain. A master-level chess player shown a chess position will frequently indicate the previous history of the game, the likely next few moves, and the key strategic features, all within a few seconds [4]. This ability is not restricted to chess, but is found in other domains of expertise [5, 10]. Studies have revealed the highly specialised nature of these memories [2], and any general theory of intelligence must account for, and model, their acquisition and use.

Developing a theory to cover learning in tasks that last perhaps a few minutes and learning over the years required to reach high levels of expertise means working at both a general and a specialised level. The general level is needed to ensure the theory is widely applicable. The specialised level is needed to ensure the theory can capture phenomena at the highest ability. This challenge reveals an apparent contradiction: how can we use highly specialised experiments to study general-purpose mechanisms? The solution which appears most promising at present is to use a *cognitive architecture*.

Taatgen and Anderson [25] describe a cognitive architecture as intended to "[supply] a general theory of cognition that is independent of particular phenomena" (p. 694). They also highlight one important question when building a model using an architecture, which is "to what extent is the intelligence in the architecture or in the model" (p. 694)? When using an architecture, it is important to ensure explanations of the behaviour are due to mechanisms within the architecture, and not any special processing added in to the model. An ideal way to achieve this is to develop multiple models which utilise and demonstrate the same core set of mechanisms provided within the architecture [18, 20].

In this paper, we present two models built using the CHREST (Chunk Hierarchy and REtrieval STructures) cognitive architecture, to illustrate its ability to acquire and retrieve patterns. The first is a model of implicit learning, identifying strings which fit a grammar after a short training period. The second is more specialised, looking at the interpretation of chess positions, using patterns which would be learnt by a human over several years. The two models rely on general-purpose learning mechanisms to develop a discrimination network, sorting perceived patterns to familiar chunks. We claim that the intelligence behind the models is within the architecture (the general-purpose learning mechanisms), and thus that CHREST is providing an explanation of learning suitable for general application. Beyond the models reported here, CHREST has been used to model performance in different board games [3, 4, 9, 16], a card game [23], in natural-language acquisition [7, 12] and diagrammatic reasoning [15].

2 Overview of CHREST

CHREST is a symbolic cognitive architecture explaining how experience affects our ability to remember, categorise and think about the world. A distinctive component of the architecture is its *discrimination network*, used to retrieve information from long-term memory (LTM). CHREST models typically begin by training the model with data, from which this network, and associated long-term memories, are constructed.

The four main components of the architecture and their connections are shown in Fig. 1. First, there is the input/output unit, with separate mechanisms for handling perception visually (with a simulated eye) and verbally (with a phonological loop). Second, there is a short-term (or working) memory, which is limited to holding four items of information at a time. Third, there is the long-term memory, which is a memory holding familiar patterns (known as "chunks")

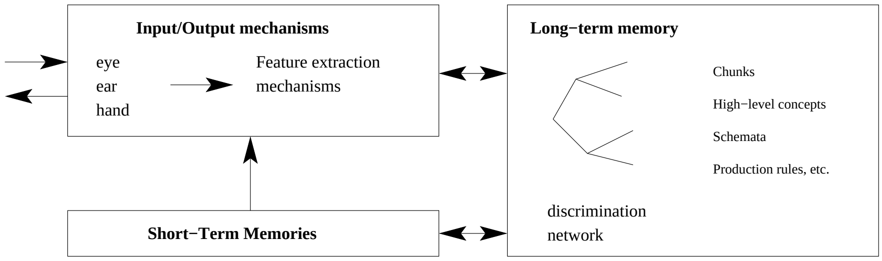


Fig. 1. An overview of the CHREST architecture.

and associations between them (including productions and "templates"). Fourth, there is an index into LTM, the discrimination network.

The role of the discrimination network is to sort an incoming pattern to the most relevant part of LTM. Although the main aspect of the network is its discrimination component, like a decision tree, sorting information from a root node to an appropriate chunk, the network also has an associative aspect, which links chunks to other chunks within LTM. The discrimination network acts as a retrieval device and a similarity function. Its role is analogous to the hidden layers of a connectionist network, or the RETE network of Soar [14].

The network is constructed incrementally, as the model perceives information. All information is in the form of a *pattern*, which is a list of primitive elements. For example, the string "VXPVXS" would be represented as a list of characters: `< V X P V X S >`. Patterns on a chess board would be represented as lists of items-on-squares: `< [P 2 5] [R 1 5] [K 1 7] >`.

The perceived pattern is sorted through the model's discrimination network by checking for the presence of elements on the network's test links. After sorting, the chunk reached is compared with the perceived pattern to determine if further learning should occur. If the perceived pattern contains everything in the chunk and some more, then *familiarisation* adds information to the chunk. If the perceived pattern contains different information to the chunk, then *discrimination* adds a further test and node to the network. Thus, discrimination increases the number of distinct chunks that the model can identify, whereas familiarisation increases the amount of information that the model can retrieve from that chunk. Fig. 2 illustrates the two learning mechanisms.

When presented with a chess board as input, CHREST uses its perceptual mechanisms to scan the chess board, extracting patterns of pieces, sorting them through the discrimination network, and so retrieving chunks to place into working memory. The eye movements are guided by heuristics. One heuristic guides the next fixation, in a top-down manner, to a position expected to help the model sort deeper into memory. Other heuristics guide the model to follow lines of attack/defence, or look at different sections of the board; see [4] for details.

CHREST is available at `http://chrest.info`. The description and models within this paper refer to version 4 of the implementation. More information on how CHREST works can be found in [3, 11, 17].

(a) After learning < V X P S >   (b) Discrimination after seeing < V X P V X S >

(c) Familiarising the pattern < V X P V X S >

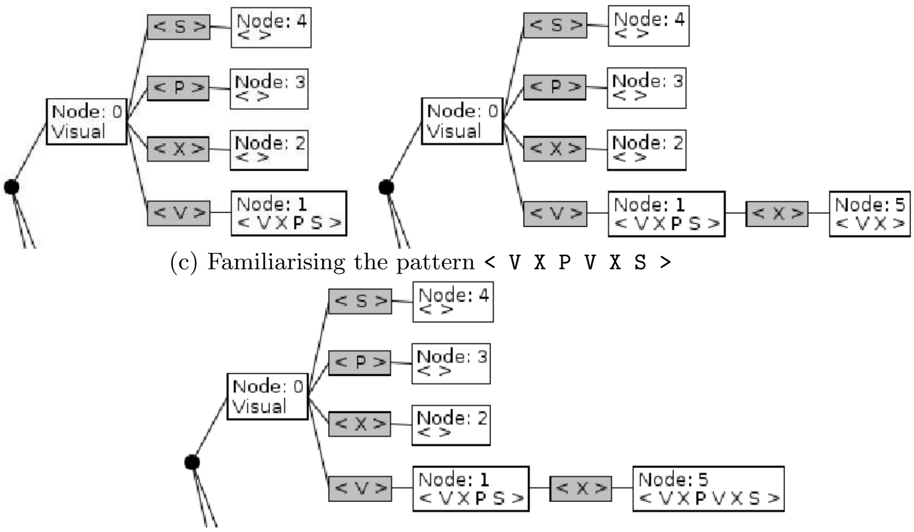


Fig. 2. Illustration of familiarisation/discrimination learning process

3 Model of Implicit Learning

This experiment reproduces that reported by [13]. The aim is to demonstrate the implicit learning of rules about valid and invalid strings constructed from a Reber grammar [21]. The CHREST model is based on a technique used in EPAM-VOC [12] for the non-word repetition task. Essentially, it is assumed a limited number of chunks may be stored within a short duration phonological loop. The model separates the input string into chunks, and rejects any string which will not fit into the loop (having more than 4 chunks), or that has more than one single element chunk; this last is a measure of unfamiliarity.

Valid strings were constructed from the grammar given in Figure 2 of [13]. 18 of these were randomly selected for training, and a different 22 for testing. A further 22 random strings were constructed from the letters of the grammar, each string of length 6, 7, or 8 letters. The random strings all ended in an 'S' and were checked that they were not accidentally a valid string. Examples of valid and invalid strings are shown in Table 1.

Table 2 presents the results from averaging 100 runs of CHREST – each run used a different set of strings for training and testing, constructed as above. The model was trained using one pass of the training data. Each test string was presented twice, making a total of 44 strings in each condition. As in [13] we show the 'hits', the number of correctly identified valid strings (true positives); the 'correct rejections', the number of correctly identified invalid strings (true negatives); the 'misses', the number of incorrectly identified valid strings (false negatives); and the 'false alarms', the number of incorrectly identified invalid strings (false positives).

Valid	**Invalid**
TTS	TXTVPS
VXPS	VXVPXS
TPPTS	TTVXPVS
VXPVXS	XPXVVXS
TPPPPTS	PXPVVTVS
VXPVXXPS	PVVVVVTS

Table 1. Examples of strings used for implicit-learning model.

	Human	ACT-R	**CHREST**
Hits:	33.00	34.00	32.84
Correct rejections:	36.00	39.00	38.52
Misses:	11.00	10.00	11.16
False alarms:	8.00	5.00	5.48

Table 2. Results from implicit-learning model. Human and ACT-R results from [13]. All results are out of 44 trials.

The results demonstrate a strong correlation between those of CHREST, of ACT-R and the humans. In particular, the model does an excellent job of identifying the hits and correct rejections. There is also a slight tendency to be better at rejecting the invalid strings, as found in both humans and ACT-R. A significant advantage over the ACT-R model of [13] is that the CHREST model constructs its own declarative memory of chunks to determine familiarity, and hence whether a given string fits the learnt pattern.

4 Model of Board Game Interpretation

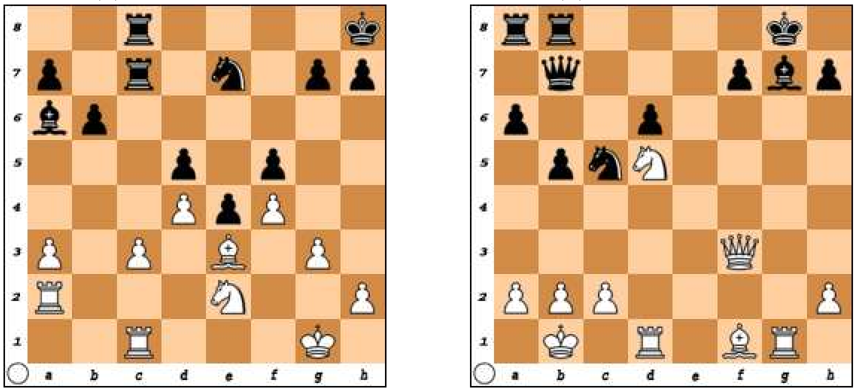
We next develop a model to reflect learning on human terms of several years, and attempt a more subtle interpretation task. In earlier work [17], we tested CHREST's ability to categorise chess positions by opening using perceptual chunks. Fig. 3 illustrates two typical positions with their openings; note the model only has access to the position, not the preceding moves, and that the game has progressed beyond the opening stages. CHREST was shown to categorise positions as well as a state-of-the-art statistical learning algorithm. This ability to categorise a chess position by opening is an important interpretation step, enabling a master player to retrieve memory cues about strategies, previous games, and likely tactics.

In our second model we go beyond simple classification, and explore CHREST's ability to retrieve multiple interpretative cues from a position, comparing these interpretations with those given by a master-level chess player. Fig. 3 gives some examples of the interpretations used below the diagrams; some of these require a knowledge of chess. The word 'outpost' appears twice, and, loosely speaking, an outpost for one side is a square in the opponent's territory which is hard for the opponent to attack. A 'bad bishop' is one which is hampered by its own pawns.

Each interpretation was treated as a simple verbal pattern, giving its name. During training, the verbal patterns were learnt alongside the positions to which they apply, and cross-modal associative links formed from the visual chunks recognised in the position to the verbal chunks. Associative links were restricted so that an interpretation referring to a white knight could only be linked with

(a) French Position

(b) Sicilian Position



black-control-semiopen-cfile,
black-controls-white-squares,
weakpawn-for-white-on-c3,
badbishop-for-white,
knight-has-outpost-on-c4, hanging-pawns

knight-outpost-on-d5, backward-pawn-d6,
black-king-side-underattack, open-gfile

Fig. 3. Example chess positions, their openings, and interpretative cues.

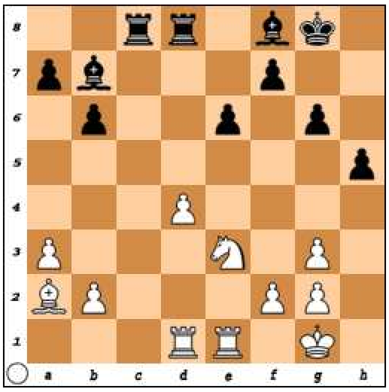
chunks containing a white knight, etc. During testing, the position provides the visual input. As CHREST retrieves chunks when perceiving the test position, it retrieves the associated verbal information. This information is then output as the model's interpretation.

500 chess positions were collected, at the 20th move in the game, and annotated with interpretations.[3] The data were randomly divided into two, a training (70%) and a test set (30%). Interpretations were used only if they appeared in more than 20 of the positions (to ensure sufficient numbers for reliable training and testing). This left a total of 36 target interpretations, with an average of 4.1 interpretations per position. A model was created training 2 times on the training dataset, with 100 fixations (approx. 30 simulated seconds) on each position during training and testing. The trained model had 37,198 chunks in its visual long-term memory.

Figure 4 illustrates a typical interpretation output by the model; the three parts of the interpretation on which the model agrees with the target interpretation are in italics. Table 3 shows the number of positions which received an accurate interpretation, and how many accurate interpretations were made per position. Table 4 shows a sample of interpretations and the frequency of correct, missed and false alarms made by the model on the test set.

The results demonstrate that in more than half of the positions, the model outputs at least one interpretation in agreement with that of the human. We regard this as a promising result, demonstrating the validity of creating a CHREST model of human-level scene interpretation. Looking at the intepretations which

[3] Prathiba Yuvarajan provided the interpretations, under funding from the ESRC.



Target interpretation
black-control-open-cfile
black-control-semiopen-dfile
doubled-pawn-for-white
uncoordinated-pieces-for-white
weak-pawn-structure-for-white
Model's interpretation
black-control-semiopen-dfile
black-controls-black-squares
uncoordinated-pieces-for-white
weak-pawn-structure-for-white

Fig. 4. Example interpretation of a position.

CHREST gets right and wrong reveals a number of interesting features. Evaluation can be difficult in some cases, for example an interpretation incorrect by commission (a false alarm) may be arguably correct, or at least useful. In the example interpretation, black has his dark-square bishop but white does not, and so control of dark-squares by black is a useful theme to be aware of.

Some of the interpretations wrong by omission are also due to limitations in the current representation used by CHREST. For example, recognising an open file requires noticing the *absence* of pieces, and our model currently does not represent empty squares. Another issue is the position-independence of some features, such as 'doubled-pawns'. A chunk linking the 'doubled pawns' interpretation to pawns on d3/d4 will not be matched by pawns on g2/g3, accounting for the 30 misses.

These issues of representation have been noted and defended previously [16] based on the focus on modelling whole-board retrieval. By extending modelling to consider local regions of the board, CHREST's visual pattern-coding is likely to require modification to use position-independent representations and to include empty squares. We believe this change will be straight-forward to make.

Correct interpretations	Frequency
0	64
1	75
2	12
3	1

Table 3. Frequency of positions with given number of correct interpretations.

Feature	Correct	Misses	False Alarms
weak-pawn-structure-for-white	58	25	30
uncoordinated-pieces-for-white	13	25	35
knight-outpost-on-d5	3	12	1
doubled-pawn-for-white	9	30	0

Table 4. Selected interpretations

5 Discussion and Conclusion

The two CHREST models presented above exemplify the effects on pattern recognition of learning across two different time scales. The implicit-learning model captures the effects of learning that occurs over a short time scale, with relatively few stimuli. The model of board-game interpretation illustrates the effects of learning on long time scales, with large numbers of stimuli. As an architecture, we can claim that CHREST has been successful in providing general-purpose mechanisms applicable to multiple models. Within the implicit-learning model, CHREST relies on a phonological loop, as also used in EPAM-VOC [12], along with its discrimination network, which provides the 'units' of memory to use in recognising grammatical strings. Within the board-game interpretation model, CHREST has constructed a large discrimination network to aid the recall of chunks from LTM. The board-game model also relies on its perceptual mechanisms, which have been tested in other tasks [3, 4].

There are several architectures besides CHREST, popular ones including ACT-R [1] and Soar [14, 20], and a natural question is how CHREST fits within the spectrum of other architectures. We suggest that CHREST can provide an explanation of how declarative memory is constructed and indexed, especially in its links with perception, whether visual or verbal. The challenge of constructing declarative memory has been described for ACT-R by [13], where their model of the implicit-learning task above required the modeller to construct the bigrams to form their model's declarative memory. Also Laird, after describing the requirements of Soar, states 'we will still fall short of creating human-level agents until we encode, or *until the systems learn on their own*, the content required for higher-level knowledge-intensive capabilities' [14, p.40 (emphasis added)]. With CHREST, the construction of the discrimination network and associated learning of chunks and their relations is a natural way to explain the origins of (some aspects of) declarative memory in knowledge-intensive tasks.

Apart from learning, Langley et al. [19] suggested that many cognitive architectures are overly focussed on problem-solving tasks, and that attention should be given to categorisation and understanding. The same authors suggest that architectures need to consider 'visual, auditory, diagrammatic and other specialised representation schemes' used by humans, and should better reflect the limited resources available for perceiving and affecting the world. As the models in this paper make clear, CHREST currently has this focus on categorisation and understanding. Previous work has already demonstrated models using auditory [12] and diagrammatic [15] representations, and the current paper illustrates the visual representation used for chess positions [4, 16]. CHREST's working-memory parameters and time constraints mean that perception and time for recall are limited by available (simulated) resources.

A way to move forward would be to combine architectures which focus on different aspects of cognition to form a more comprehensive architecture that might capture learning, categorisation, understanding and problem-solving tasks. The choice of architecture and the way to make the combination is not, however, clear. Two distinct approaches may initially be identified, depending upon how closely recognition processes are thought to be involved with problem solving. One approach is more modular, and the other more integrative.

A modular combination might take support from Dual Process theories [6, 24], in which an intuitive, pattern-matching process (System 1) is hypothesised to be replaced when necessary by a distinct, analytical, problem-solving process (System 2). A natural analogue would be to combine CHREST with ACT-R or Soar to simulate more complex implicit-learning tasks: using CHREST as the intuitive component and ACT-R/Soar as the analytical component. CHREST would determine and recognise the entries in declarative memory, which ACT-R/Soar could then use in problem solving.[4]

In contrast, a more integrative approach would consider pattern recognition tied in more closely with problem-solving, almost intertwined. A previous proposal along these lines using CHREST was made in the SEARCH model [8]. Thus, except in very artificial tasks, System 2 nearly always operates with System 1, shedding serious doubts about the independent use of the two systems as proposed by Dual System theories. For example, when a chess expert tries to find the best move in a position, the variations that are being consciously searched are nearly always supplemented by unconscious pattern-recognition mechanisms [4].

Whether the future holds a modular or integrated combination of architectures, it is apparent that a theory of artificial general intelligence will reflect the contributions of several current architectures. This combination would, we suggest, present a better explanation of general intelligence, covering a wider range of phenomena than either alone, whilst combining their strengths. In particular, we argue that CHREST is a suitable architecture for studying tasks involving categorisation and understanding based on prior expertise, and have demonstrated some new results in these areas. An important area of ongoing research is to consider how these recognition processes interact, and may be combined, with current theories of problem solving.

References

1. Anderson, J.R., Bothell, D., Byrne, M.D., Douglass, S., Lebière, C., Qin, Y.L.: An integrated theory of the mind. Psychological Review **111**(4) (2004) 1036–1060
2. Bilalić, M., McLeod, P., Gobet, F.: Specialization effect and its influence on memory and problem solving in expert chess players. Cognitive Science **33** (2009) 1117–1143
3. Bossomaier, T., Traish, J., Gobet, F., Lane, P.: Neuro-cognitive model of move location in the game of Go. In: Proceedings of the 2012 International Joint Conference on Neural Networks. (2012)
4. de Groot, A.D., Gobet, F.: Perception and Memory in Chess: Heuristics of the Professional Eye. Assen: Van Gorcum (1996)
5. Ericsson, K.A., Charness, N., Feltovich, P.J., Hoffman, R.R.: The Cambridge Handbook of Expertise. New York, NY: CUP (2006)

[4] A previous proposal [22] argued for a combination of EPAM (the predecessor of CHREST), GPS (a predecessor of ACT-R/Soar) and UNDERSTAND (for language).

6. Evans, J.: Dual-processing accounts of reasoning, judgment and social cognition. Annual Review of Psychology **59** (2008) 255–278
7. Freudenthal, D., Pine, J.M., Gobet, F.: Simulating the referential properties of Dutch, German and English root infinitives in MOSAIC. Language Learning and Development **15** (2009) 1–29
8. Gobet, F.: A pattern-recognition theory of search in expert problem solving. Thinking and Reasoning **3** (1997) 291–313
9. Gobet, F.: Using a cognitive architecture for addressing the question of cognitive universals in cross-cultural psychology: The example of awalé. Journal of Cross-Cultural Psychology **40** (2009) 627–648
10. Gobet, F.: Psychologie du talent et de l'expertise. Bruxelles: De Boeck (2011)
11. Gobet, F., Lane, P.C.R., Croker, S.J., Cheng, P.C.H., Jones, G., Oliver, I., Pine, J.M.: Chunking mechanisms in human learning. Trends in Cognitive Sciences **5** (2001) 236–243
12. Jones, G.A., Gobet, F., Pine, J.M.: Computer simulations of developmental change: The contributions of working memory capacity and long-term knowledge. Cognitive Science **32** (2008) 1148–1176
13. Kennedy, W., Patterson, R.: Modeling intuitive decision making in ACT-R. In: Proceedings of the Eleventh International Conference on Computational Modeling. (2012) 1–6
14. Laird, J.E.: The Soar Cognitive Architecture. MIT Press (2012)
15. Lane, P.C.R., Cheng, P.C.H., Gobet, F.: CHREST+: Investigating how humans learn to solve problems with diagrams. AISB Quarterly **103** (2000) 24–30
16. Lane, P.C.R., Gobet, F.: Perception in chess and beyond: Commentary on Linhares and Freitas (2010). New Ideas in Psychology **29** (2011) 156–61
17. Lane, P.C.R., Gobet, F.: Using chunks to categorise chess positions. In: Research and Development in Intelligent Systems XXIX: Proceedings of the Thirty-Second SGAI International Conference on Artificial Intelligence, Springer-Verlag (2012)
18. Lane, P.C.R., Gobet, F.: A theory-driven testing methodology for developing scientific software. Journal of Experimental and Theoretical Artificial Intelligence (accepted)
19. Langley, P., Laird, J.E., Rogers, S.: Cognitive architectures: Research issues and challenges. Cognitive Systems Research **10** (2009) 141–160
20. Newell, A.: Unified Theories of Cognition. Cambridge, MA: Harvard University Press (1990)
21. Reber, A.S.: Implicit learning of artificial grammars. Journal of Verbal Learning and Verbal Behaviour **6** (1967) 855–863
22. Richman, H.B., Simon, H.A.: Context effects in letter perception: Comparison of two theories. Psychological Review **3** (1989) 417–432
23. Schiller, M., Gobet, F.: A comparison between cognitive and AI models of blackjack strategy learning. In: Proceedings of the 35th German Conference on Artificial Intelligence, New York: Springer (2012)
24. Sloman, S.: The empirical case for two systems of reasoning. Psychological Bulletin **119** (1996) 3–22
25. Taatgen, N., Anderson, J.R.: The past, present, and future of cognitive architectures. Topics in Cognitive Science **2** (2010) 693–704