The Determiners Model: a Cognitive Model of Macro Development and U-shaped Learning in a Micro Domain

Karin Zondervan (karinz@ai.rug.nl)
Niels A. Taatgen (niels@ai.rug.nl)
Department of Artificial Intelligence, Grote Kruisstraat 2/1
9712 TS Groningen, Netherlands

Abstract

In this paper we present a cognitive model that simulates children's acquisition of the functions of determiners. Karmiloff-Smith (1979a) shows in an experiment that the performance-curve on the indefinite article through the years has an U-shape. She uses the representational redescription (RR) theory to explain this (Karmiloff-Smith, 1992). We have taken this theory as a starting point for our ACT-R model. The RR theory describes cognitive development as a process in which knowledge becomes more and more explicit. Our model adds two things to the RR-theory: it explicates what the exact representations are at each representational level and it specifies the exact transition mechanisms. We have used the model to develop a more general scheme for designing developmental cognitive models.

Introduction

Computational modeling research on learning and development has focused mainly on the learning of very specific tasks or micro-developmental changes. Developmental psychology research also describes macro-developmental changes. Developing cognitive models simulating that sort of changes is a relative new challenge. In this paper, it is argued that the *representational redescription* (RR) theory of Karmiloff-Smith (1992) and the ACT-R cognitive architecture (Anderson & Lebiere, 1998) can act as a template for creating macro-developmental computational models. The RR theory distinguishes three levels of representation. These levels can be mapped on components and mechanisms of ACT-R. The RR theory does not explicate what the exact representations are at each level, nor the exact mechanisms that produce transitions between levels. Hopefully, the mapping onto ACT-R can provide more information about these interesting topics.

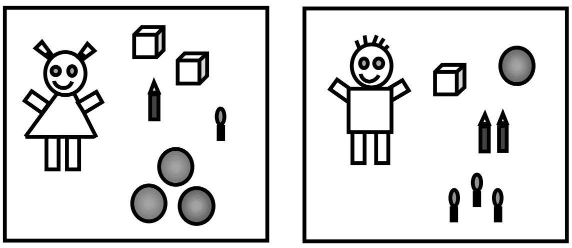

Figure 1: The playrooms

We have used the above principles to implement a model that learns the functions of determiners in French. It simulates the results found by Karmiloff-Smith in her playrooms experiment (Karmiloff-Smith 1979a, experiment 12). In this experiment French children between three and eleven years old were tested on their ability to distinguish the use of the definite and the indefinite article. The experimental setup consisted of two dolls – a boy and a girl – each placed in their own playroom, surrounded by a set of objects (see Figure 1). The sets are carefully composed. If the girl doll owns one object of a certain type, i.e. a book, then the boy owns several items of that type and vice versa. The experimenter asks the child to which doll the sentences "Lend me the X" and "Lend me a X" was addressed. All objects in the playroom are referred to with feminine French words, so the experiment concentrated on the contrast between *une* (French for *a*) and *la* (French for *the*). The results of the experiment show that children of all ages performed almost perfect when judging a *la*-sentence. For the *une*-sentences a U-shape in performance emerged. (see Figure 2). In the model we developed, we focused on this U-shaped learning and tried to find an explanation for it. The results of the experiment show that the confusion is caused by the fact that in French, the word *une* has two meanings. It is an indefinite article (English: a), but it is also a numeral (English: one).

We will now summarize the RR theory and describe the explanation this theory gives for the U-shaped learning described above. Later on, the RR explanation and an explanation based on the cognitive model will be compared.

The Representational Redescription Theory

The basic idea of the representational redescription theory is that in each micro domain children go through several levels of representation. Going from one level to the next is accomplished by the process of redescription. The result of this redescription is an abstraction of the implicit knowledge into more explicit knowledge. This process can continue during development and even more explicit knowledge can become available to the cognitive system. Karmiloff-Smith distinguishes three levels: Implicit (I), Explicit-1 (E1) and Explicit-2/3 (E2/3). Every transition to a higher level causes task specific details to be lost, but it also creates more abstract and flexible knowledge, that becomes more and more task independent. The more explicit the knowledge gets, the better access the child has to it. For every microdomain, e.g. learning the meaning of numbers, the transitions from one level to the next have their own timing and speed. If the child is at the implicit level for a certain task, it can use knowledge that is incorporated in procedures to perform the task. It cannot verbalize its behavior. If the child has arrived at the most explicit level (E2/3) in a particular micro domain, it is able to verbalize the knowledge it uses. At the intermediate E1 level, verbalization is not yet possible and the redescribed representations are only visible in the behavior of the child. This changed behavior at the E1 level is sometimes redundant. It can cause a temporary decrease in performance.

Mapping these levels onto ACT-R can roughly be done as follows. At level I the knowledge is implicit and task specific. The learning behavior at this level corresponds to a certain extent to instance-based learning, as described by Logan (1988). Lebiere, Wallach and Taatgen (1998) successfully used the instance theory to build ACT-R models of implicit learning tasks. Instance theory is translated into ACT-R as a strategy of memorizing all earlier encountered items and, when confronted with a new item, recalling from declarative memory the best matching example. We will call this the analogy strategy. At level E1 the representations are more general. The cognitive system no longer uses specific instances, but uses more generally applicable rules instead. Our model tries to find regularities in instances and transforms them into general production rules. It uses the analogy strategy mentioned above in combination with the mechanism of production compilation[1] (Taatgen & Anderson, 2003). Hence, representations at the E1 level can be mapped onto new production rules that are made by the production compilation mechanism. Procedural knowledge cannot be verbalized, so in order to move on to level E2/3, the procedural knowledge has to be redescribed into a declarative representation.

How does the RR theory explain the U-shape in learning the function of *une* in the playrooms experiment? Around age five, children seem to prefer the meaning of the numeral when hearing *une*, also if the speaker intended the indefinite article. This misinterpretation causes them, when hearing "[Prête moi] une X", to choose for the playroom with exactly one X. Karmiloff-Smith (1992) explains the U-shaped behavior for *une* in three phases. In the first phase the child has two independently stored level-I representations for understanding and producing the indefinite article and the numeral. These representations are independently stored form-function pairs. The child is in this phase not aware of the functional relationship between these representations. It does not yet perceive of them as being part of a linguistic subsystem. After representational redescription the original level-I representations are still part of the cognitive system, but the system now also has this knowledge available in E1 format. These representations now explicitly mark the relationship between the identical forms of *une* as an indefinite article and *une* as a numeral. This explicitly represented link causes the sudden occurrence of errors of interpretation of the indefinite article in the playrooms experiment. Further representational redescription could then cause the performance to go up (the second part of the U-shape). Karmiloff-Smith does not explain in detail how this can happen. She focuses on the ability to verbalize. She concludes that around age 9 the children make explicit reference to the *linguistic* clue that – in fact – all children must have used in the experiment.

The ACT-R model we have developed, is inspired by the explanation above. But developing a simulation model forces the developer to formalize the process much more and to specify many details and rules. In the remainder of this paper we will present our formalization of the above process and compare it with the RR theory.

Discovering Relevant Dimensions

A problem in developing computational models of development is that they are focused on a particular aspect of behavior, while at the same time other aspects influence the execution of the task. The question is: Which aspects are outside the scope of the simulation task and which ones are part of it? In what ways do aspects that are outside the scope still influence the behavior of the subject? While we are growing up we learn all sorts of tasks in all sorts of domains. What we have learned in one task can sometimes be transferred to another task. This transferred knowledge helps to learn new strategies for the task at hand. Besides that, in learning strategies for a particular task, we can make use of more general strategies that we learned earlier. The literature about cognitive models of change (Simon & Halford, 1995; van Rijn et al., in press) suggests that in all cognitive problem solving people have a general strategy that can get them started: determine which properties of the environment are important for this task and look for regularities in the values of these properties. We call these properties *dimensions*. This regularity-seeking strategy evolves (probably through representational redescription) into new, more specialized rules for a specific domain of problem solving. Our mind conducts this process in a rational way, in line with the basic idea of ACT-R. For a certain task, we have more than one specialized strategy at our disposal. These strategies can be implemented as a set

[1] This particular mechanism is not part of the standard ACT-R distribution, see Taatgen (2002) for details.

of competing rules. In the case of searching for relevant aspects of the environment the competition is between strategies that are looking for yet another dimension to incorporate in the decision process and strategies that are a bit lazy and jump straight into the decision process with less information. This competition leads to rational behavior, because it can find a balance in the trade-off between speed and accuracy. If a certain dimension leads to far better answers, then trying to find its value in the environmental setting is worthwhile. If the cognitive system spends a lot of time finding a dimension that has not much effect on the correctness of the answer, then the system is not efficient and can be optimized. Finding the best balance in this trade-off is exactly what cognitive development is about. Throughout our childhood we gather more knowledge, which opens doors and enables us to find new possibly relevant dimensions, which we did not notice before. The simulation model of determiners is based on the same idea. It starts with a very limited set of relevant dimensions, but general cognitive development in other cognitive tasks paves the way for the discovery of new relevant dimensions. They in turn, cause the model to develop production rules, which capture the regularities in these dimensions. In the next section we will find out which dimensions are relevant in the determiners model and what production rules the model will learn.

Learning the Functions of Determiners

Around age eleven we become experts in the functions of determiners. What dimensions in the environment do these experts need to find the function of a determiner in a particular sentence? It actually comes down to two questions: *Is the noun, following the determiner already a topic of discourse?* and *Is the object referred to unique in this particular context?* We distinguish three dimensions you have to consider in order to be able to answer these two questions. Let us call them *number*, *focus-before* and *focus-afterwards*. *Number* has the value *one* if the object is unique in the present context. If the (sort) object is not unique, *number* has the value *more*. The sentences uttered before the present sentence plus the words in the present sentence before the determiner, provide information about which object is the focus of discourse up until now. We call this information *focus-before*. After the sentence with the determiner is heard, the topic of discourse shifts and the speaker and listener now have a new, common focus (*focus-afterwards*). The focus-before can be nothing (*empty*) or can be an object (we will call this value *symbol*). The *focus-afterwards* depends on what the speaker wishes to communicate. If he wants to introduce a new object to the discourse, the *focus-afterwards* is the object itself (we have generalized this to *symbol*). If the speaker wants to express the amount of the sort object under focus, this *amount* is the *focus-afterwards*. There are more things you can express with determiners, but we will limit our attention to the ones above. In the model, the relevant dimensions become available one at the time, depending on the experiences of the child. At the start of the simulation time (age three) the model can distinguish whether or not an object is unique in the present environment. It is not yet able to count or to use a numeral to express amounts of objects. As soon as it has achieved the skills of counting and expressing amounts, it recognizes that there are two possible functions for the determiner *une*: *introduction* and *expressing-the-amount*. At this stage, the model does not have enough knowledge to be able to predict the value of the focus-afterwards when hearing a sentence. It does not conceive of sentences as being part of a discourse. Henceforth, it does not (yet) take into account what information was given in the sentence before. In its daily life, a child encounters many situations in which it needs information of earlier sentences, e.g. when it hears: "He is an idiot!". The speaker does not point to someone but refers back to a person he introduced in the sentence before. In a situation like this, it is very clear that you need to recall some information from earlier sentences. At a certain moment in development the dimension seeking strategy of the child finds out about this *focus-before*. As soon as this dimension is clear enough for the child and has proven its relevance, the child can start to use it in the far less clear cut situation of determiners. In this way the dimension *focus-before* becomes part of the simulation model. The model is now able to predict the shift in focus, before it actually hears the determiner.

When hearing a construction with a determiner, the model uses an analogy strategy to find the function of the determiner. It starts with gathering information about the present situation. Depending on which dimensions are part of its declarative memory, it finds more or less information in the context. It then uses this information to search in the declarative memory for a similar example of the use of the heard determiner. It performs the actual analogy with this example as the source. The outcome is the function of the determiner. As a side-effect missing dimensions of the context also get a value. This analogy strategy takes much time. The production compilation mechanism of ACT-R compiles new production rules out of the analogy strategy to speed up the process. Every time a new dimension is added to the model, the model makes use of this new information through a more elaborate analogy strategy. Production compilation on this extended analogy strategy will lead to new, more advanced production rules. In the following subsections we will give a short description of a selection of the emerging production rules and explain how they give rise to the U-shaped performance with *une*-items.

Overview of the model

At a very abstract level, we can describe the developmental process of the model as follows. Throughout the simulation the model uses analogy to interpret the determiner, and searches actively for an example. In the first phase, situational context implicitly takes care of reminding the child of the right interpretations. The two functions of *une* are called for in different situations and are not confused. The production rules the model learns in this stage do not influence behavior. In the second phase, the analogy process provides more information. Proceduralisation leads to rules that skip searching for an example. Two rules for *une* evolve, which have the same form. These rules lead to generalization, but also to confusion. One *une*-rule leads to less harmful overgeneralization. It wins the competition. It does not resolve the confusion in the right way, so the model does not stop searching for new dimensions. Phase three starts when the dimension *focus-before* becomes available. Now the child can recognize two different situations, an *introduction*-situation and an *expressing-the-amount*-situation. Specialized rules for each situations emerge through the process of production compilation. The ongoing search for new dimensions is very valuable. The trade-off between speed and accuracy is optimized through this process. In the next subsection we will take a more detailed look at the model.

Details of the model

In the first phase, the model hears a lot of setences with determiners in it. In the process of deciding on the function of a determiner, the model can only take the dimension *number* into account. The production compilation mechanism discovers regularity in this dimension and builds the following new rules:

1. IF determiner=une AND number=more THEN function=introduction.

2. IF determiner=une AND number=one THEN function=unknown.

The value *unknown* is a consequence of the fact that the model has at this stage no idea of counting and expressing amounts of things. Hence, it can make an explicit link between *une*, *more* and the function *introduction*, but not between *une*, *one* and the function *expressing-the-amount*. The model does not always encounter situations in which the number of the objects is clear. For unclear situations it learns the following rules:

3. IF determiner=une THEN number=more AND function=introduction.

4. IF determiner=une THEN number=one AND function=unknown.

This type of rules are needed in the experiment, because in the experimental situation the number is always *unknown*. Rules that lead to an unknown function receive a penalty in the form of extra effort. In ACT-R this means that these rules will take more simulated time. The assumption is that this extra effort is needed to infer meaning in another way. When the model is in the experimental situation it can use both rule 3 and rule 4 in the case of *une*-items. Most of the time rule number 3 will have a higher expected gain, because it will not lead to an unknown function. As a consequence the rational model chooses rule 3 in approximately 90% of the cases (see Figure 2, age three).

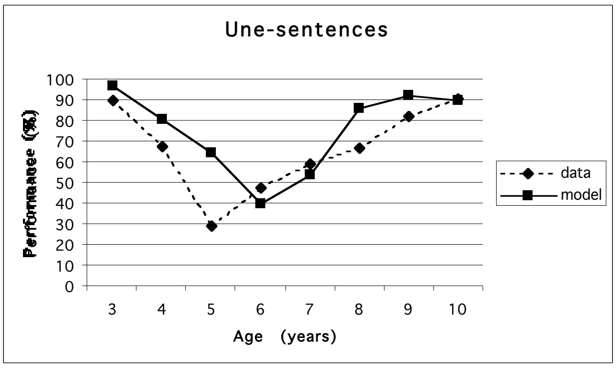

Figure 2: The experimental results of the subjects (cross-sectional) and the experimental results of a run of the model (longitudinal).

In the second phase things are getting a bit complicated. The model now recognizes that there are two different functions when hearing *une*. It also notices the different values of *focus-afterwards*. It processes examples in which *une* is used as a numeral and it still recognizes the cases in which *une* is used in its indefinite function (function=introduction). For situations where number cannot be found in the context, it finds these new regularities:

5. IF determiner=une THEN focus-afterwards=symbol AND number=more AND function=introduction.

6. IF determiner=une THEN focus-afterwards=amount AND number=one AND function=expressing-the-amount.

Notice that these rules provide more information, but are both too generally applicable. When the intended function of a heard determiner was *expressing-the-amount* the model can use the wrong rule and understand *introduction* and vice versa. The former misinterpretation is much more severe than the latter. In case of the second misinterpretation the model will still – as a sort of side-effect – accept the entrance of the object as the focus of conversation. If the model hears (in French): "Give me a plate", it can misinterpret this as a demand for exactly one plate, but it will still give the speaker what he wanted. Misunderstanding an intended *expressing-the-amount* function can give rise to more problems, because in that case the model misses important information. If someone exclaims to the model (in French): "I have one shoe!" and the model interprets the *une* as *a*, then it will be happy for this person. It will not understand the message that having one shoe is not okay, because you need *two* shoes to be able to walk on the streets! Therefore, if the model thinks *introduction* was the intended function while it actually was *expressing-the-amount*, it receives much extra effort (simulated time). If the model misinterprets an *expressing-the-amount* function for an *introduction* function only a small amount of extra effort is given. We conclude that the overgeneralization of rules 5 and 6 is not symmetrical. The rational model always searches for the fastest and best fitting rules. The consequence is that it can be satisfied with a certain amount of overgeneralization, if this leads to a fast answer that is right most of the time. The overgeneralization of rule 6 comes with less extra work afterwards, so this rule will get stronger and stronger, although it does not lead to the perfect solution. In the experimental situation it always causes the model to choose the wrong playroom. As rule 6 gains strength, the performance in the experiment will decrease (see Figure 2, age three to five).

The turning point in this trend emerges in the first part of the third phase (around age six). The dimension *focus-before* comes into play. The model has become sensitive to the shift in focus and how to predict it. More advanced analogy rules can now be applied. These lead again to the compilation of more sophisticated production rules:

7. IF determiner=une AND focus-before=empty AND focus-afterwards=symbol THEN number=more AND function=introduction.

8. IF determiner=une AND focus-before=symbol AND focus-afterwards=amount THEN number=one AND function=expressing-the-amount.

The most important aspect of these new set of rules is that every rule fires in a different situation. The IF-part is different for every rule. This means that these rules do not lead to overgeneralization. They instead use enough information to lead to the correct answer in every situation. In the experimental context, this means that the process of the growing strength of rules 7 and 8 causes an increase in performance on the playrooms task (see Figure 2, from seven years to ten years old). At the end of the simulation the performance has risen to the same level as at the start, approximately 90%.

Comparison between the RR Explanation and the Model

As we have seen, the RR-theory describes the development we have simulated as going from the implicit level to levels that are more and more explicit. In this terminology, probably contrary to the default use of the terms implicit and explicit, the use of instances and analogy is an implicit (level I) strategy. Production rules 5 and 6 lead to generalization and represent knowledge at the E1-level. Our model is in fact an iterated process of representational redescription. Every time a new dimension is added, the model tries to redescribe the new relations it has found. These new relations are based on a new, more elaborate analogy strategy. New available dimensions are a prerequisite for this new, more elaborate, analogy strategy. The remaining question is: Where do these dimensions come from? We postulate, in line with the RR-theory, that they are implicitly present in knowledge, used in other cognitive tasks the system performs. A process of representational redescription makes this knowledge more explicit and less task dependent. At the highest RR-level, this knowledge is part of the declarative memory in the form of abstract dimensions. Only at this level this knowledge can be used in our determiners task. There is, however, a missing link in the ACT-R model. The more specialized production rules the model developed in the last phase, possibly contain knowledge that can be of interest to other linguistic tasks. Unfortunately, the present version of ACT-R does not give us any clue of how these specialized production rules can be redescribed into abstract knowledge in declarative memory!

Conclusions

The determiners model is a simulation of a very specific task in a small micro domain. The simulated time is approximately nine years. In this regard it can be seen as a model at a macro-developmental level. The challenges of this type of modeling are finding the right boundaries and finding the right level of abstraction of the task. The RR-theory forced us to define very strictly which knowledge the model had available at a certain moment and in which form. After defining the knowledge in a very precise way, the boundaries of the model – which knowledge components have to emerge through the simulation and which ones are to be taken for granted? – were easier to specify. The RR paradigm and its translation into ACT-R terms pave the way for a general template for developing cognitive models at a macro-developmental level. This template looks like this:

Micro domain:
…………………………………………………………..
Task:
…………………………………………………………..
Available dimensions (or other declarative knowledge) at the start:
…………………………………………………………..
Available strategies (or productions) at the start:
…………………………………………………………..
Later available dimensions (in order of appearance):
…………………………………………………………..
New strategies (or productions) learned by the model:
…………………………………………………………..
New dimensions (or other declarative knowledge) learned by the model:
…………………………………………………………..

Properly filling in this form is half of the work of developing a macro-developmental cognitive model. This template, the production compilation mechanism and the general strategy of searching for dimensions that capture the regularities in a certain task can function as ready to use building blocks for cognitive models. Besides that, if we would have a whole set of cognitive models which are developed using the above template, these models could be seen as parts of a bigger picture and we would be able to incorporate them in a more general developmental model. This would be a step ahead on the way to an unified theory of development and learning.

Looking back, we can conclude that the determiners model shows how humans can make use of general, redescribed knowledge to specialize their general strategy of searching for regularities in their environment. For this specialization, the model uses the outcome of the representational redescription process: domain specific knowledge becomes more explicit and available for other tasks. The major contribution of the determiners model to the field of cognitive modeling is – in our opinion – that it shows how ACT-R and the RR theory can be combined in designing cognitive models of development. The translation of the RR framework into ACT-R gives real content to the representations and the redescriptions the RR theory is all about.

References

Anderson, J.R., & Lebiere, C. (1998). *The atomic components of thought.* Mahwah, NJ: Lawrence Erlbaum Associates.

Karmiloff-Smith, A. (1979). *A functional approach to child language: A study of determiners and reference.* Cambridge, MA: Cambridge University Press.

Karmiloff-Smith, A. (1992). *Beyond modularity: A developmental perspective on cognitive science.* Cambridge, MA: MIT / Bradford Books.

Lebiere, C., Wallach, D., & Taatgen, N.A. (1998). Implicit and explicit learning in ACT-R. In Frank Ritter & Richard Young (eds.), *Proceedings of the Second European Conference on Cognitive Modelling*. Nottingham, UK: Nottingham University Press.

Logan, G.D. (1988). Toward an instance theory of automization. *Psychological Review*, *95*, 492-527.

Simon, T.J. & Halford, G.S. (1995). *Developing Cognitive Competence.* Hillsdale, NJ: Lawrence Erlbaum Associates.

Taatgen, N.A. (2002). Production Compilation. Retrieved from the internet on 13-02-2003 from http://act-r.psy.cmu.edu/workshops/workshop-2002/talks/NielsTaatgen-ProdComp.pdf.

Taatgen, N.A., & Anderson, J.R. (2003). Why do children learn to say "Broke"? *Cognition*, *86(2)*, 123-155.

van Rijn, H., van Someren, M., & van der Maas, H. (in press). Modeling developmental transitions on the balance scale task. *Cognitive Science.*