Nonrestrictive Concept-Acquisition by Representational Redescription

WEI Hui†, YUAN Liang†

†Department of Computer Science, Lab of Algorithm for Cognitive Model, Fudan University, Shanghai 200433,China

Summary

A rich concept system, having fine-granularity and abundant semantic relationships, is absolutely necessary for any artificial intelligence system, especially when this system is confronted with a domain-open task, or a task beyond the range of classical expert system, such as let an AI system comprehend sentences in our daily life. These tasks exhibit their characteristics more from cognitive view instead of from engineering view, because problem-solving in these situations involves much more common sense and knowledge from diverse domains, or in other words this kind of domain-open problem-solving requires a comparatively dense concept system to broaden the base of semantic. However, most of existing expert systems and other knowledge-based systems can not meet such requirements because they are designed for those domain-restricted applications. In Cognitive Psychology and Developmental Psychology, the study of concept acquisition offered many cases and evidences in the processes of cognitive skills' maturation in behavior level. And this revealed that the development is one of the most important attributes of man's concept system. And a main lack of the classic expert systems is their knowledge bases are not constructed by development. Inspired by the concept acquisition of Developmental and Cognitive Psychology, we can improve the knowledge base design along the same way. Whereas what those concept development theories can provide are only coarse procedures or very abstract frames from the point of view of algorithm, because some crucial issues like the representation, evolution, storage, and learning process of concept etc. are not described. Maybe the realization details of constructing a concept system in a computer are not the aim of Psychology, but they are the core problems in artificial intelligence. In an operable and practicable level, we propose a concept acquisition and development method, which was inspired by Karmiloff-Smith's Representation Redescription (RR) supposition. They divide the cognition development into three phases and four levels of representation: Implicit (I), Explicit 1 (E1), Explicit 2 (E2) and Explicit 3 (E3). But in primitive RR how it realizes the representation and redescription was not mentioned, that is to say the RR only has basic and abstract sprit and no implementation consideration. Thus we should develop a systematic method to avoid its ambiguity, formalize its architecture, represent its learning results, arrange its learning processes, and match the concept's inner and static structure with its outer and dynamic applicability, and we propose using Object-Oriented (OO) theory to explain RR's basic sprit and use this new RR-OO model in the growth of concept system. The aim of it is to construct a knowledge system through a new mode and make it adept at task-diversity. In detail we firstly use RR to analyze the concept developing from low level to high level, and find that its representation should become more and more general and flexible. Thus we need a well-formed representation to solidify this gradual change of learning a concept both in its inner structure and its outer appearances. The OO theory has accurate definitions in class and data abstraction, encapsulation, visibility of object's attributes and behaviors, polymorphism, overloading, modularity and inheritance. This makes it to be a perfect tool to meet the requirements of Implicit and Explicit representation of RR. We apply an OO-similar frame to formalize and represent the developmental process of a concept, i.e. a concept system in one's mind can be simulated by an object system, because an object system is convenient to describe a real world, and its objects can be seen as element blocks of a knowledge system. In this paper an elementary cognition skill, counting, is taken as an example to demonstrate the development of a concept, and its evolution is materialized by the forms of objects in their attributes and behaviors. This new method is much more computable than classical RR, and more suited for us to build a base-wider semantic system for those domain-open applications of AI.

Key words:

Concept acquisition, Representational redescription, Knowledge representation

1. Development of concept system

People have skillful cognitive behavior, such as problem solving, reasoning, decision-making, programming, nature language understanding and scene understanding, all this depends on a comparatively complete concept system. This behavior is different from those artificial intelligence systems which aim for special application. Those systems usually have strict restriction in problem content, representation formats, application background and predetermined conditions, while People's cognitive behavior processes various uncertain requirements. So People's knowledge system must be general rather than special. What is more, people use just a single system to solve all kinds of intelligence tasks. This requires high in the amount of knowledge system, and the representing, storing and processing of knowledge. Knowledge

Manuscript received December 8 2006.
Manuscript revised December 25 2006.

representation of today's expert systems and knowledge-based systems can't satisfy the requirements of cognition. As we all know, concepts are the core of knowledge. After conceptualizing, the perceptive experience becomes knowledge, which is easier for stating. To be the nucleus of intelligence behavior, there is no difference in principle between concepts and knowledge. So we use concept system in the rest of paper.

The processing of unrestrictive knowledge requires a complete concept system [1]. The most obvious example is that people make use of knowledge not stick to one pattern in nature language understanding, content-based understanding and common sense problem solving.

The construction of concept system can not accomplish in an action. Conversely, it forms gradually by interrupted learning. So investigations at algorithm level on this problem helps to construct better knowledge structure. This is a nuclear problem in artificial intelligence [2].

Knowledge representation has been considered as one conventional part of a whole system of artificial intelligence. However, little work has been done on the new knowledge representation and other subjects such as knowledge representation adequacy, knowledge structure, concept system and conceptualization. As a common problem, the starting point of these investigations relies on the practical need for knowledge engineering rather than the theoretical need for the interpretation of knowledge, which is feasible to realize but hard to reach the depth of psychology theory.

Many groups have done investigations on the subject of Developmental Psychology and Cognitive Psychology [3]. Research on the process mostly concentrated on the qualitative investigation of the mechanism rather than that of algorithm level. Existed models of interpretation for Knowledge Representation seem too crude to easily operate due to the facts that such models lack detailed investigation of the joint between representation and processing, and include too much knowledge skill whose mechanism remain ambiguous. This paper is to propose one practical method of representation of concept development which gives computer system sufficient language support for fulfillment of knowledge-based appointments. From the view point of theoretical background, research on knowledge structure constitutes the interdisciplinary core problem comprising several fields [4, 5].

2. Representational redescripion

A number of investigations have been reported in the field of Developmental Psychology. Two representative views are Piaget's cognitive constructivism and Fodor's nativism. Integrating some aspects of the two theories, Karmiloff-Smith proposes the Representational Redescription (RR) model [6]. The model claims that people acquire knowledge through a proper Representational Redescription progress. In the RR model, knowledge can be stored and represented in several levels. There are three phases and four representation level.

Phase 1: People extract information from environment and transform the information into representation accessories. Representation accessories are just added to the system, without changing the existing representations or relating with them. The representation at this stage is called Implicit representations (I- level). At this level, environmental stimulus is coded with a procedural format. I-level representations are stored dependently. It can generate successful behavior, but this kind of behavior is inflexible.

Phase 2: During phase 2, Endogenously provoked change becomes the most important. Representation at this phase is Explicit 1 representations (E1-level). E1-level is a reduced redescription of I-level representations. E1-level representations deal with explicit representations, they can be manipulated and related to other knowledge. They are not accessible to consciousness, but are apparently beyond procedural level.

Phase 3: During this phase, external materials and internal representations are integrated. The control between outside and inside can reach a balance. Representation during this phase is Explicit 2/3 representations (E2/3-level). The two levels are accessible to consciousness. The difference between them is that E3-level is available for verbal report but E2-level isn't.

The result of each phase should be behavioral mastery, which can generate coherent and successful behavior. Redescription happens between two neighboring phases. During redescription process, the procedural, independently-stored and implicit representations gradually become flexible, explicit and can be manipulated. Redescription takes place in all domains and all the moments during the development of cognition, and is restricted by the existing knowledge and the level of explicitness.

The RR model applies a new method to study the development of cognition for construction of concept system resulting from the development of representation. However, as the model belongs to the field of Psychology, it isn't accurate enough for calculating angle. Karmiloff-Smith proposes four level of representations without describing the representation formats and how redescription to carry out in detail. In other words, the RR model deals with the content of representations rather than the format [7]. In order to apply RR model for construction of concept system, all these problems should be solved.

3. Representation fulfillment

There are four levels of representations in the RR model. The standards used to distinguish different levels, such as "Can be accessible to consciousness or not", "Can be available by verbal report or not", are hard to be calculated and represented in concept system. Formal definition of object-oriented is a method to depict the behavior and relations of individuals. Here we use this method for the development and construction of our concept system. The object-oriented method is a well-structured and well-known method to describe physical world. It is both theoretically and practically well identified. It's useable especially due to its encapsulation and inheritance. A cognitive technique of counting will be taken as an example to explain the development of cognition by the evolution of objects' attributes and behavior.

3.1 I-level representations

According to the behavioral characteristics of I-level representations in Developmental Psychology, we assume objects that represent isolated events are the basic of representations at this level. The objects have few attributes, or variables. Their behavior is ambiguous and can't be interrupted. There is no modularity and lack for large-scale decomposition and small-scale combination, both of which depend on the course, roles and place of events. The only visible part of an object is the whole behavior, which has few entering parameters. The behavior can hardly be regulated and aims for a unitary goal.

At this level, a certain event of counting is represented by an actual object CountApple. The components such as APPLE1-3, the objects to be counted and MARY, the counter are constants. A table describes the changes after every behavior if necessary. The whole process is inseparate.

```
CountApple
{
const Apple APPLE1, APPLE2, APPLE3;
        const Person MARY;
        const Position HERE, THERE;
        Process()
        {
            //The Initial State
```

	Initial State
APPLE1	Position(APPLE1) = HERE
APPLE2	Position(APPLE2) = HERE
APPLE3	Position(APPLE3) = HERE

```
        MARY.Move(APPLE1);
        MARY.Say(ONE);
        //The State after MARY.Move(APPLE1)
```

	Initial State
APPLE1	Position(APPLE1) = HERE
APPLE2	Position(APPLE2) = HERE
APPLE3	Position(APPLE3) = HERE

```
        MARY.Move(APPLE2);
        MARY.Say(TWO);
        //The State after MARY.Move(APPLE2)
```

	Initial State
APPLE1	Position(APPLE1) = THERE
APPLE2	Position(APPLE2) = HERE
APPLE3	Position(APPLE3) = HERE

```
        MARY.Move(APPLE3);
        MARY.Say(THREE);
        //The State after MARY.Move(APPLE3)
```

	Initial State
APPLE1	Position(APPLE1) = THERE
APPLE2	Position(APPLE2) = THERE
APPLE3	Position(APPLE3) = HERE

```
    }
}
```

	Initial State
APPLE1	Position(APPLE1) = THERE
APPLE2	Position(APPLE2) = THERE
APPLE3	Position(APPLE3) = THERE

3.2 E1-level representations

Several I-level objects induce a super-level object. The latter diverges, to some extent, on the attributes and behavior. Objects at E1-level improve on modularity and parameter level, which means that many unitary functional modules can be extracted. Another considerable change is to enhance the visibility of attributes and behavior of objects.

Again, take the study of counting for example. A few times of practice (in other word, there have been some I-level objects of counting events) induce an advanced representation. The counter, the set to be counted and the result of counting have been explicit at this level, while the principles of counting remain implicit. An important point is that representations at this level can be simplified by using other existing concepts.

```
enum Number{ONE, TWO, THREE, ..., TWENTY};
//declare an enum type Number
const Number CountList[20]={ONE, TWO, THREE, …,
TWENTY };
//define an array of Number to keep 1 to 20 orderly
class CountEvent
//a class has been induced and it's the remarkable progress of
concepts acquisition
{
    Public:
     //the counter, the set to be counted and the result of
     counting is explicit at this level
    Set<Object> s1;
```

```
    //The Set here and the List below are existing concepts in
    the system, and the Object is the super-class of all the
    objects class such as Apple
    Person Operator;
    Number result;
    void Count()
        {
            int i;
            Number num;
            for(i=0; !s1.empty(); i++)
            {
                Object a1 = s1. Select ();
                //select an element from the set randomly
                num = CountList[i];
                Operator.Say(num);
                  s1.erase(a1);
                  //eliminate the element from the set
            }
            result = num;
           }
    //describe the whole process of counting and keep the result
    into variable result
}
```

3.3 E2-level representations

Attributes and behavior have been sufficiently extracted at this level. Some of them become friendly so that they can relate to other objects. That is to say those attributes and behavior can use existing concepts defined in other objects. Therefore the abstract level of objects and the potential probability of referring by others increase.

A NumberList class appears at this level. The information of numbers, such as the sequence of numbers, get a number's successor, is defined in that class. The CountEvent class use operations defined in the NumberList class without accessing numbers directly.

```
class NumberList: public List<Number>
{
   Public:
   NumberList()
   {
       assign(20);
       push(ONE, TWO, THREE,…,TWENTY);
   }//initiate the list with number 1 to 20
}
//NumberList define the list of numbers, it is a sub-class of List,
so it inhere some behavior of List class, such as GetFirst(),
GetNext

class CountEvent
{
   Public:
   Set<Object> s1;
   Person Operator;
   Number result;
   void Count()
   {
       NumberList c1;
       Number n1;
       Object a1 = s1. Select();
       n1 = c1.GetFirst();
       //Get the first number of the list
       Operator.Say(n1);
       s1. erase(a1);
       while (!s1.empty())
       {
          a1 = s1.Select();
          //Select an element from the set randomly
          n1 = c1. GetNext(n1);
          Operator. Say(n1);
          s1.erase(a1);
       }
       result = n1;
    }
}
```

3.4 E3-level representations

Further divergence and specialization take place within objects. Some objects split so that their components become new objects. This enhances the possibility of successful matching in other applications. Later the references of these objects can be extended to other domains. The more unitary an object's function and semantics are, the easier it can be expressed to outside.

The principle of one to one matching (we should attach one and only one element orderly to a single number) has been externalized to a single class at E3-level. One to one matching is a general method. After externalization it can be used in other domains.

```
class OneToOne
{
      Public:
      List<Bio>
      OneToOne(Set<Object> set, List<NUMBER> list)
       //match the elements of the set and the list one to one, and
         return a list of a binary group of result
}

Struct Bio{
      Number number;
      Object object;
}
//define the structure of the binary group which expresses the
result of one to one matching
class CountEvent
{
       Public:
       Set<Object> s1;
       Person Operator;
       Number result;
       NumberList c1;
       OneToOne o1;
       Number Count()
       {
```

```
        Bio b1;
        b1 = o1.OneToOne (s1, c1);
        result = b1.GetLast().number;
    }
}
```

The development of representations just shows the development of concept system. One thing should be emphasized is that at any time the representation's won't be the final representations in system. At one time there may be several levels of representations for a same concept. Different levels of representations are used for different problems.

4. Discussion

The essential goal of the research reported here is to use reference and evidence to construct a semantic symbol system which can be implemented at algorithm level. More complicated and domain unrestrictive applications require reasoning with more complete semantic information. It can't carry out on the basis of a hypothesis that the structure in low level can be no continuous, which is the traditional hypothesis in artificial intelligence. This paper proposes a cognitive model rather than a knowledge acquisition model. It helps us to understand the representations and processing principles of concept system at the level of detail.

Acknowledgment

This research is supported by the NSFC under grant no 60303007 and China Basic Research Project (973) under grant no. 2001CCA03000, and open project of Key Laboratory of Intelligent Information Processing, ICT of CAS under grant no. IIP2002-3.

References

[1] Wei Hui, The Correlation in Cognition and Systematic View to The Building of Intelligent Model (in Chinese), Postdoctoral Dissertation, Zhejiang University, Hangzhou, 2000

[2] Mark H. Bickhard & Loren Terveen, Foundational issues in artificial intelligence and cognitive science, Amsterdam, Elsevier, 1995, 11~18

[3] Liu Ailun, Shui Rende edited, Psychology of thinking (in Chinese), Shanghai: Shanghai education press, 2002

[4] Kurt Vanlehn edited, Architectures for Intelligence, New Jersey: Lawrence Erlbaum Associates Publishers, 1991

[5] Robert L. Solso edited, Mind and Brain Sciences in 21st Century, Cambridge: The MIT Press, 1997

[6] Karmiloff-Smith, A. Beyond modularity: A developmental perspective on cognitive science. MIT Press, 1992

[7] Carassa, A., Tirassa, M. Representational redescription and cognitive architectures. Behavioral and Brain Sciences, 1994