AI Working Paper 195

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

ARTIFICIAL INTELLIGENCE LABORATORY

THE USE OF THREAD MEMORY IN AMNESIC APHASIA AND CONCEPT LEARNING.(note 0)

Lucia M.Vaina and Richard D. Greenblatt

ABSTRACT. We propose a new type of semantic memory, called thread memory. The primitives of this memory are threads, defined as keyed multilink, loop-free chains, which link semantic nodes. All links run from superordinate categories to subordinate categories. This is the opposite direction to those in the usual tree structure in that brother nodes in the tree share the structure above their common ancestors. The most valuable feature of the thread memory is its capacity to learn. A program which can learn concepts using as data children's primer books, was written by R. Greenblatt and runs on the LISP-MACHINE at the MIT-AI Laboratory. We have considered the thread memory as working hypothesis for exploring the mechanisms of naming deficits in aphasia and the ways of rehabilitation.

INTRODUCTION

Memory is concerned with how knowledge is stored, cross-referenced, indexed, retrieved, accumulated and modified. The elements of memory are concepts. Concepts form hierarchies leading to more complicated concepts. The way knowledge is arranged determines how we understand, solve problems, remember, and learn.

In this paper we are interested in implementing a semantic memory, so we propose a conceptual structure and a set of mechanisms which operate on it. We provide a computer simulation in the form of a program written in LISP. We introduce a novel "memory" structure that will be called "thread memory" for reasons that will became clear; its structure was motivated partly by empirical results from psychological experiments, testing, and observations made in less structured situations. We also used ideas suggested by clinical information about how damaged memory mechanisms breaks down in cases of aphasia and agnosia.

We begin with a very condensed exposition of our position with respect to "fundamental principles". In our view, the initial sources of concepts are the perceptual systems, such as language, vision, touch, etc. Different kinds of concepts are initially treated in their own ways, but eventually -- at some level of representation -- all information conveyed by the perceptual systems is supposed to become compatible; this level is called the conceptual structure and it constitutes the "working space" of memory. In addition the conceptual structure memory has a set of mechanisms which process the information represented in the concepts.

We do not want to pretend to solve all problems in this one paper. So we will accept, as a working position, the view proposed by many natural language processing researchers that semantic structure -- the representation of information conveyed by language perception -- is embedded in such a conceptual structure, which determines much of how we organize our human experience. This hypothesis says that how we learn to represent and to deal with the world has very little to do with how we learn the language per se. Still, language learning is important in world-learning since, besides the "universal" linguistic aspects of syntax, phonology, and projection rules, etc., there are large non-universal parts, e.g., in the lexicon and in non-universal parts of the syntax.[10]

There are several general criteria which are applied in the evaluation of any theoretical model: for example, the criteria of economy, efficiency, universality, reliability, etc. We claim that the application of these criteria should be purpose-related: for example, our proposed thread memory might be considered too redundant and wasteful of storage space. However, this may make it more resistant to damage and decay. A memory that satisfies perfectly some criteria of universality might turn out to be unrealistic and uninteresting if it told us only generalities and little about how to process the particular experiences of the individual.

In the view we take, the most important aspect of semantic memory organization is how effectively it can be made to "learn". We believe that 'thread memory' provides a better base for a learning system than others like those of (Fahlman [8], Brachman [7]). We intend to show the evidence for this claim in several ways.

We take human memory as our model for a learning system. However the usual tools of experiment and introspection applied to the human memory system are quite untrustworthy. We believe that the study of how brain damage disrupts the use of language provides a more reliable tool. Indeed, the most direct motivation for the details of our 'thread memory' comes from Warrington's analysis of three cases of visual agnosia[25].

Perhaps the most important feature required of a memory good for sophisticated learning is the avoidance of local conventions with global consequences. That is, one wants to be able to add information on the basis of considerations whose scope is very local to the specific information item to be added. For example, one should have to be concerned only with the goal context at the moment the addition is made; one should not have to worry about all sorts of possible future unrelated tasks. We shall illustrate this principle, and show how our proposed memory organization supports it, both in structure and in application, The best evidence for such a proposal is a working computer program; this is the only convincing way to show that a theory of learning is effective, complete, and applicable -- not to mention its practical utility. Even more important -- during the construction of the theory -- is the way that the practicalities of a computer program continually direct one's attention to points which otherwise might well be glossed over. Such points frequently form the basis for new advances in conceptualization.

More metaphysically, it is our view that because both computer programs and humans are computational elements governed by universal laws, one would expect them to develop in somewhat similar directions -- even without any explicit attempt to make this happen. Although there are surely profound structural differences between the brain and the present-day computer, the computer system designer, working within his paradigms of practicality, efficiency, and elegance, may be driven to similar solutions as was nature's human cognitive development. In any case, there is little to lose in believing this, until we are flooded with too many, too adequate theories!

Thread Memory, and the Semantic Interpreter SEMI (note 1)

Threads are a new generic data organization [9] which seems to have great potential for computer concept construction. In this section, we describe the basic idea and point out some of its basic computational properties.

A thread is a keyed multi-link loop-free chain, which links semantic nodes. A typical thread might be

mallard -> living -> animal -> bird -> duck -> species-of-duck.

In this notation, the first token (called the key) is not actually stored as part of the thread. Instead, it represents the stimulus by which the rest of the thread may be accessed. The links running from node to node are entirely unique to the particular key; the same nodes could be linked by other threads in either the same or different fashion since those other threads would have different keys. (We actually provide a mechanism whereby a single thread can link the same nodes in more that one fashion simultaneously. However, we defer discussion of this latter mechanism for the moment.)

Note that all links run from superordinate categories to subordinate categories [25]. In other words, all access paths run from more general categories to more specific categories. This is the opposite direction to those in the usual tree structure, where brother nodes in the tree share the structure above their common ancestor.

Representing Threads on the Computer

We introduce a single node, called the SEMI-ROOT. All threads originate at the SEMI-ROOT. Thus, given a key (mallard, for example), we reference (mallard, SEMI-ROOT), to obtain 'living'. If we next reference (mallard, living) we obtain animal, and so on. For reasons which will become evident, we require all normal threads to end back on their keys. Thus the example becomes

mallard -> SEMI-ROOT -> living -> animal -> bird ->
--> duck -> species-of-duck -> mallard.

Remarks

Thread memory seems to make information "come out" in the "right order". What could be more sensible, when first thinking of mallard, than to retrieve living, its most general classification? If that doesn't give us the cue we want, we next try animal, etc. The memory seems suited to certain recognition tasks. Suppose we see a duck we think might be a mallard. By retrieving on mallard we can not only obtain information to help verify this guess, but also obtain a large amount of correct information in case the duck we saw turns out to be a red crested saddleback. Even if it turns out we saw a sea gull, we may well get large amounts of useful data by retreving on mallard. Furthermore, if we do decide that it's a sea gull, it is easy to determine if we have made use of any incorrect information about mallards. (Namely any information beyond the fork point). (The same kind of property was used by Marr and Nishihara [14] for object recognition).

The problems of dynamically adding to this sort of data base are considerably simplified. We see that simply adding a new item to the data base involves no modification at all of any previously existing links.

This greatly reduces the necessity to "understand" the database when updating it, a very desirable property. Additionally, this property may simplify the implementation of "contexts". At least, the behavior of the data base prior to some addition can be simulated by a simple operation performed on the results returned by the data base after the addition.

The data base is ideally suited to answering "compare and contrast" questions. Given mallard and elephant, one immediately has (for example) "they are both living, and both animals, but a mallard is a bird while an elephant is a mammal". We have come to call this operation on two threads "finding the fork point", and we believe it has great importance.

Deduction in a thread memory system

Suppose we have

- > ELEPHANT -> ANIMAL -> ELEPHANT

-> ANIMAL -> LIVING-THING -> ANIMAL.

We are asked "Is ELEPHANT a LIVING-THING?" The necessary one step deduction is performed by searching each thread whose semantic node is on the ELEPHANT thread, for the node LIVING-THING. After the appropriate intermediate node ANIMAL is found, one can perform an interesting operation called "assimilating" the deduction. This consists of inserting the goal node on the original thread in the position immediately before the intermediate node, obtaining, for example,

ELEPHANT -> LIVING-THING -> ANIMAL -> ELEPHANT.

Note that the general to specific nature of the thread is preserved. If the same inquiry is repeated, the answer will now be immediately available. In other words, the assimilating process turns what was a one step deduction into a zero step deduction, and by induction, can turn an N step deduction into a zero step deduction, provided the system is asked the appropriate "leading" questions. However, it would be quite tedious if one were required go through this "leading" process each time. This can be avoided in a natural fashion by "building up" on an intermediate node. Suppose we also have

LIVING-THING -> PHYSICAL-OBJECT -> LIVING-THING.

If we ask once "Is an ANIMAL a PHYSICAL-OBJECT"?, obtaining - > ANIMAL -> PHYSICAL-OBJECT -> LIVING-THING -> ANIMAL, then henceforth we can show that any ANIMAL is a PHYSICAL-OBJECT using only a one step deduction.

In other words, with the same assimilation mechanism we can easily chain together intermediate deduction steps so that the entire chain is available as a single deduction step. Thus the depth of deduction required is no longer the "distance" between the two nodes, but instead the number of "jumps" between "corridors" (or deduction chains) that are required

[23]. Motion "along" a corridor comes for free.

To look at it in another way, such corridors in effect form a "chart", which can greatly reduce the danger of combinatorial explosion. Visualize a search space as originating at a point and consisting of a wedge shaped space of possibilites expanding off to the distance. Then the corridors correspond to particular line segments within the wedge, some touching the origin and some not. Doing a depth N search can be visualized as searching the rectangle N wide centered on each corridor which touches the origin, plus a rectangle N-D wide centered on any corridor which passes through one of those rectangles with a distance D of closest approach to the generating corridor, and so on, until N is reduced to 0. With appropriate corridors in place, very modest search depths such as one or two can find solutions to non-trivial problems.

As such a memory structure is filled out, one can see that there is quite a bit of redundancy. This would seem to be a very desirable property when building a system out of unreliable elements. In addition, such redundancy can lead to a very simple model of forgetting. One can simply delete nodes at random at a low rate. If the relevant thread is in use, the deleted node will be rederived, otherwise, it will remain "forgotten".

From the point of view of economy of storage elements, the general-to-specific method is not as economical as the specific-to-general method, where brothers in the tree share links above the point in the tree where the brothers merge. However, this loss of economy is minimized if, as we believe, the semantic tree is shallow and bushy.

One problem that becomes evident is that it is frequently desired to have a node participate in several independent hierarchies, such as

BOY -> PERSON -> MALE -> BOY and

BOY -> PERSON -> CHILD -> BOY

We simply allow this, and term the collection the general thread of BOY. An individual path is referred to as a simple thread. Most operations are performed upon simple threads, so, the issue of selecting simple threads arises. We postulate a mechanism called "general context" to deal with this problem, which is briefly discussed in the section of aplications (for more detail see [9]). We also retain, as a medium cost operation, the ability to "map" over all simple threads of a general thread, if desired.

There are some facts we wish to store about nodes which may not lend themselves to a hierarchical representation. To help deal with this, we postulate property lists of semantical nodes which act in a fashion similar to LISP property lists. The property lists associate attributes with symbols. Property lists have an even number of elements, and in each pair of elements the first element is a symbol called indicator, and the second is the value. The indicator serves as the name of the property and the value as the value of the property. Thus, a thread can be associated with a property of a node. The retrevial operation operates on an entire simple thread, not only on a single node. Thus, for example, we might retrieve the property COLOR-OF from the thread

CLYDE -> LIVING-THING -> ANIMAL -> ELEPHANT -> CLYDE. Thus, a single property stored on ELEPHANT, would be sufficient to say all elephants are GREY. We notate such a thread by

(ELEPHANT COLOR-OF) -> GREY

where the property's indicator is COLOR-OF, and its value is the key of the desired thread. (In the implementation, the system generates this key internally from a sequence such as G0001, G0002, etc.)

Performing property retrieval operations threadwise helps "chunk" information in many cases. For example, suppose we encounter BERTHA, the blue elephant. We then retrieve COLOR-OF from the thread

BERTHA -> LIVING-THING -> ANIMAL -> ELEPHANT -> BERTHA.

The retrieval operation would return ((ELEPHANT GREY) (BERTHA BLUE)) which would give a tipoff that BERTHA is a special elephant. By retrieving both facts in a single step, we insure that both are taken into consideration and avoid bad logic based on only one of them in ignorance of the other.

The existence of certain property threads can be specially built into the system. An example is the IS-NOT-A thread. Storing ANIMAL on the IS-NOT-A thread of PLANT, for example, will enable us to distinguish all PLANTs from all animals and ANIMALs from all plants by means of a single thread operation.

Distinguishing an arbitrary PLANT from an arbitrary ANIMAL is somewhat more interesting. Without heuristics, four nested loops would be required, but by using the "finding the fork point" heuristic, only two nested loops are required. The outer two loops are the same in either case, namely FOR-EACH simple thread of CLYDE, FOR-EACH simple thread of DAISY. The straightforward approach would then have use loop over each node in the simple thread of CLYDE and for each of them loop over each node in the simple thread of DAISY, looking for a contradiction. In many cases, this computation can be greatly shortened by the finding the fork point of the two threads. It is clear that any nodes before the fork point are the same on both threads and so cannot cause a contradiction. Once the fork point is reached, there is an excellent possibility that the two forking nodes are the desired contradiction. If we have, for example:

DAISY -> LIVING-THING -> PLANT -> DAISY

CLYDE -> LIVING-THING -> ANIMAL -> ELEPHANT -> CLYDE.

the fork point is exactly the desired contradiction. In current practice, the computer program examines althogether four possibilites to allow for possible "mismatch" in the degree to which the threads are "built up". That is, in addition to looking for a contradiction on the thread T1(n) vs T2(n) (which is the fork point itself), we also test T1(n+1) with T2(n), T1(n+1) with T2(n+1), and T1(n) with T2(n+1). Additionally, we test if T1(n+1) is equal to T2(n) or vice-versa. If so, a false fork point has been found, the odd node is discarded, and the search for the true fork point is resumed. When these procedures are exhausted, the threads are assumed non-contradictory without examining further subordinate categories for possible contradictions.

The individually keyed nature of threads helps control the damage which may occur in case that inconsistent data should be stored in the memory. Suppose, for example, we have said that (IS-NOT-A PLANT ANIMAL) then encounter an EUGLENA which is both a PLANT and an ANIMAL. We observe that there is no way the system's reasoning about other plants and animals can become confused, since the contradictory information is confined to the EUGLENA thread, which is completely inactive unless EUGLENAs are under consideration. Even with reference to EUGLENAs, reasonable things will happen. The EUGLENA will inherit the properties of both PLANTS and ANIMALS. Nothing "gross" can happen in areas of knowledge remote from the contradiction. If the system is quizzed on the contradiction itself, by (IS-? ANIMAL EUGLENA) it will say yes, and it will also say yes to (IS-? PLANT EUGLENA). This is because it looks first in the IS-A heirarchy (for a possible yes answer) before it looks for IS-NOT-A property threads (for a possible no answer).

Another sense in which thread memory deals with the internal difficulties caused by inconsistent data concerns controlling the depth of deduction. Deep deduction is undesirable because it is likely to expose inconsistencies, and it has potential to spread confusion to areas of knowledge distant from the original contradiction. Thread memory provides an environment where one can limit the depth of deduction, while still not compromising ultimate system capabilities. (However one may require, "helping questions" instead).

In many cases, the same or similar information can be represented either by a property thread, or by a specialized-nominal. Thus we might have (IS-A GREY-THING ELEPHANT) or (COLOR-OF ELEPHANT GREY). Although these represent similar information, they may well have different computational properties. However, the system should clearly be capable of accepting information in either form, and, possibly with the aid of prompting questions, replying to inquiries phrased in either way.

Another important relation in the memory is the HAS relation. The HAS properties interact with the main IS-A hierarchy in special ways. We consider HAS as a two place operator, HAS (QUANTIFIER, NAME). ONE is a very common case of QUANTIFIER, and that combination is frequently referred to by just HAS-A (NAME). A node can have simultaneously many HAS properties active; at present we simply combine them all into an unordered list. However, the problem of providing structure to this list is very interesting and may be essential. Each HAS property results in the creation of a thread. This thread cannot be referenced directly, but only by means of a NAME-CHAIN. Thus (DOG HEAD) would reference the thread stored on the HAS list of DOG under the name HEAD. In this way, relations can be built up between the various threads that a single node HAS. These relations are inherited, in a parallel fashion, by any node subordinate to the given node. For example, if (IS-A DOG SPOT), then (HAS-A HEAD SPOT) and SPOT's HEAD is related to SPOT's other parts in a manner parallel to how a DOG's HEAD is related to a DOG's other parts.

The QUANTIFIER in a HAS relation actually serves a purpose somewhat more general than its name would suggest. Namely, in addition to simply specifying how many of a given item there are, it can also contain association lists to HAS keys individually enumerating them (or some of them). Thus we might have (HAS (FOUR ((TWO FRONT) (TWO REAR)) ((TWO LEFT) (TWO RIGHT)) ((A LEFT-FRONT) (A LEFT-REAR) (A RIGHT-FRONT) (A RIGHT-REAR))) LEGS). In general such HAS quantifier lists are not inherited en masse. Instead, each individual term is built up as required.

In addition to forward directed reasoning, (ie, SPOT IS-A DOG, therefore SPOT HAS-A HEAD), one would also like the HAS database to serve a recognition function (e.g. What has a head, four legs, and is in the garden?). HAS threads, as described, are not suitable for this purpose, since they offer no improvement on a linear search over all known items. Therefore we fall back on specialized nominals to perform this function. For example we also have (IS-A FOUR-LEGGED-THING DOG), (IS-A THING-WITH-HEAD DOG), etc. With this representation, it is much easier to intersect the various classes efficiently.

Theoretical background and suggestions for future developement.

Our concern in this section is to review critically some of the alternatives that have been proposed to represent the conceptual organization of an individual as he learns about his environment, as a part of our effort to answer the question of how one (computer or human) learns to represent knowledge about the environment.

There is an extensive literature about concepts, but most of it does not seem relevant for our purpose. We are interested not in a history of concept representation, but rather in an attempt to offer a working model (a program suported by theory and applications). Our account of thread memory was suggested by E. Warrington's results in a study [25] of patients selected on the basis of failure to recognize or identify common objects (visual object agnosia). Although they had no deficit in perception, or in intellectual function as measured by I.Q. tests, their knowledge of subordinate categories was less reliable than the knowledge of superordinate categories. It was noticed that objects from taxonomic categories comprising many exemplars that were differentiated only by details presented special difficulty. For example the patients could recognize a flower but not which particular flower, some couldn't differentiate between fruit and vegetable etc..

Two kind of mistakes were noted: 1) the object was described in terms of a very general category and the superordinate class was used appropriately. (hammer - some kind of a tool). 2) Semantic errors, where the response was an alternative item from the same category. (cat-dog).

The patients could differentiate animals from plants quite correctly, birds from insects, but the ability to differentiate among animals on the basis of attributes and asociations was very poor. One of the patients with a milder deficit only made errors differentiating objects on the basis of their attributes and not by their associations.

Warrington's motivation for suggesting that links point from the more general to the more specific, rather then the more usual specific to general, in a nutshell, was that the observed syndrome could be modelled by postulating a simple break in the thread. That is, if pointers are stored from specific to general, a break in a pointer would result in mallard remaining connected, for example to duck and species-of-duck, but becomming disconnected from living, animal, and bird. A memory failure of this sort seems highly implausible and in fact was not observed. Instead, mallard remained connected to living, animal, and bird, for example, but became disconnected from duck and species-of-duck. Such a failure mode seems much more plausible.

Another attempt to deal with conceptual structures in a way which is close to our view is E. Rosch's early work [20]. Rosch's fundamental hypothesis is that our perception of the environment is organized around some central foci which become prototypes for the learned categories. When someone hears a category name like bird for example, what sort of mental representation occurs in the memory? Is it a list of defining features of that category, an image, a code for the category's prototype? From the experiments that Rosch conduced in the early 70's it seems that what one generates when one hears a category name is not a list of relevant features, but rather the best example of a member of that category. How do we get the prototypes? In general by ostension, or by definition, that is built up from more simple and already known concepts. How does one classify objects? Of the many possible levels of abstraction on which an object can be classified there is one level which is more basic psychologically, namely, - as the experiments show- the level at which one can obtain the most information with the least cognitive effort. We know that young children or people beginning to learn a new language very frequently use the generic concept name 'thing' instead of more specific names. The basic level gets more specialized as the individual's kowledge gets to be more specific and detailed. If in a mini world one can get along by using very general concepts, in a more sophisticated world, a large universe of discourse, too much generality can lead to ambiguity and misunderstanding.

For reasons of economy of cognitive computation the human memory chooses as a basic level the most inclusive level at which it is possible to represent an image of the "best example" of the class. The " best example" is considered to be the average member of the class. Rosch states that natural categories have an internal structure: on the one hand, they have a core meaning, which is the prototype, or the best example, and on the other hand, they have a distance dimension which is defined by decreasing similarity of other instances to the prototype.

It follows from a number of experiments that within a given group of related concepts three levels of abstraction are chosen: a superordinate category, a basic level, and a subordinate category. For example the superordinate category of furniture has two basic-level categories: chair and lamp, and subordinate to them are the types of chairs and lamps.

In a recent paper [15], G.Miller, following Lyons [13], introduces the technical term "hyponymy" to represent hierarchically a group of related concepts, such as TABLE and FURNITURE. A word B is a hyponym of a word A if for any "x", the sentence "x is a B" entails that "x is an A". To be able to characterize the lexical taxonomy more fully, Miller states that the direct hyponyms of a superordinate term constitute a contrastive set of terms whose extensions are "mutually exclusive and whose combined extension exaust the extension of the superordinate term". There are hierarchies other than hyponyms that relate nominal concepts: the part-whole relation, locative inclusions,etc.. These hierarchies can also be characterized by transitive asymmetric relations, as is the IS-A relation of hyponymy. We call the set of these other relations the "complexity" type of relations, as opposed to the "abstraction" type of relation.

Miller points out the importance of redundancy rules for concept learning. For example when a child learns TABLE he learns whatever TABLE shares with FURNITURE and when he then learns CHAIR, he learns whatever CHAIR shares with FURNITURE, but now it will be easier to learn. He doesn't have to know explicitly by a given rule that there is a common part between TABLE and FURNITURE, or CHAIR and FURNITURE. Redundancy is going to play a key role in the system that we propose, and we will show in an explicit way what its effective role is in learning as well as in forgetting and remmebering. Lexical knowledge that conforms to the redundancy rules is not isolated from the general conceptual system, so usually we will find TABLE toghether with CHAIR and BED and FURNITURE, etc.. So, for example, TABLE gets its meaning from its place in the conceptual system. As more concepts from a related area are introduced, learning effectiveness increases. This occurs by means of inheritance. That is, superordinate nodes are created from which new threads can inherit properties, thus obviating the necessity of relearning them.

We claim, with Miller, that this way of computing meaning maximizes learning effectiveness. We will show also its importance for the process of memory rehabilitation.

In [61] Miller makes an interesting remark, saying that in addition to lexical knowledge about TABLE an individual has practical knowledge, about the function of TABLE. In other words, we can say that the memory includes two related but distinct meanings of TABLE, one is the lexical meaning, and the other is a more general meaning of "anything serving the function of a TABLE". [16]. Neither account has explanatory power, but we believe that we can explain Miller's otherwise correct intuition by representing the conceptual structures in thread memory. If we base our representation only on the IS-A thread, then we will not be able to take into account the function that different instances of the same object are expected to have.

What is the functional information and how is it represented? How does one relate perceptual (lexical) information to functional information? Miller proposes that an identification device for an object has both perceptual (P) and functional criteria (F), that constitute somewhat "fuzzy" thresholds for inclusion in categories. Let us take HAMMER as an example. Objects that satisfy both kinds of criteria are literal HAMMERS. There are also 'figurative' HAMMERS, which satisfy only one set of criteria. For example, an ICECREAM HAMMER looks like a HAMMER but it cannot be used as such. It thus constitutes a "fake" HAMMER that satisfies only the perceptual criteria. A ROCK can be used as a HAMMER, so it satisfies the functional criteria, but it obviously does not satisfy the perceptual criteria. If someone can identify a HAMMER but doesn't know what it is used for, then he will never identify a ROCK as a HAMMER. The knowledge that is invoked to determine whether an object performs a given function, results from the systematization of conclusions that an individual has reached by practice or inferred from some theoretical knowledge (laws of mechanics, electricity, etc.). Miller proposes that the set (F) of functional criteria be described in modal terms of possible, and he argues that the relevant judgments of possibility depend on the system of practical knowledge. This is the same intuition that we had in developing our system. To obtain a thread we make use of our common sense knowledge. To represent the functional criteria we use the LEADS-TO and HAS relations. We believe that what has to be given to talk about the functionality of an instance is the concept and its uses.

In the applications we will see that in nearly all cases of aphasia (exceptions are extremly rare) the patient is not able to name the object, but can recognize its use (given by the functional description). For example, in some aphasias, when the doctor shows the patient a SAW, he recognizes that it is used TO CUT WOOD, but he cannot show how. The patient recognizes what the use of the SAW is but he doesn't remember how to use it. In the same time when he is asked to pick up the object that is used to CUT WOOD, he picks up the SAW. The same is true in the case of children learning a new concept; they learn what it is for, what purpose it serves, and then later on, they learn how to use it. It seems to us that the relation between a concept and its use is a fundamental one, but it has not been the focus of our research yet. We will outline anyway some preliminary ideas that remain to be worked out in the future. We make the hypothesis that if nominal concepts were not linked to their USE, then it would be very unlikely that an individual could cope with the environment. The hierarchy of USES is structured by LEADS-TO, and the top-down orientation of the pointer requires that USES inferred from the input be connected to the USES already existing in the individual's memory system.

The way in which we envision this association of USES and CONCEPTS is as: (LEADS-TO (USE A TO-B) R), which gives the thread (R LEADS-TO) USE -> A -> TO-B. If we input (USE C TO-B), we will obtain in the same way (R LEADS-TO) USE (A & C) TO-B etc.. Eventually we will get a set of concepts that have the same use. If we express USES as RULES in LISP for example, then they consist of two parts: a condition and an action. If the condition of a rule is fulfilled then the action is executed. The conditions play the role of minicontexts, that is, restrictive or sinequanon conditions, for the performance of actions.

The question arises, what is it that we associate with the USES? Thread memory represents concepts by a general thread which is a set of simple threads that have the same key- the concept in question. Obviously we don't associate with the goals every single thread of the general thread. What we do instead, is to have a 'stereotype thread', which represents the 'stereotype meaning' of the concept in question. The notion of 'stereotype meaning' was introduced by Putnam [17]. A stereotype meaning is a set of beliefs associated with terms. The need for stereotypes is not primarily to fix the extension of a term, but for discussion, for communication. As Putnam puts it [18], " the language is not only used to verify and falsify and classify; it is also used to discuss". The amount of information contained in 'meanings ' varies with the nature of the information, the kind of concept, the speaker's experience, etc..

The fact that a feature is included in the stereotype associated with a concept doesn't necessarily mean that all the instances of that concept have that feature, nor that all the normal instances have the feature.

Most stereotypes capture the information relevant to the paradigmatic members of a class, but that may not always be the case. The information contained in the stereotype is not necessarily correct, since it may happen that a concept has been acquired incorrectly. Putnam gives as an example the stereotype of GOLD which contains the feature 'yellow', because the gold that we see has the color yellow even though pure gold is nearly white. A stereotype, in other words, is built from the frequency of a feature in instances rather than from analytic truths about it. Stereotype are used to communicate information and to understand and convey meaning. We believe Putnam's intuition is correct, but he offers little beyond this intuition. He gives no mechanism for effectively obtaining the 'stereotype meaning' of a term, and he doesn't show how a stereotype can be compatible with the cases it contradicts. For example,how the stereotype of a tiger which is a feline, of certain size, has black stripes, etc.., can get along with an instance of tiger which is unstriped?. What Putnam tells us is that it is possible, to see tigers without stripes and still to accept them as tigers.. He also tells us that if we discover that the stereotype was based on incorrect informtion we don't get a logical contradiction. If tigers ceased to have stripes, they wouldn't be tigers any less than before. How does one changes the stereotype meaning of a term? Putnam doesn't provide us with any procedure, but certainly it wasn't his intention to do so. (The aim of the philosophers is different from ours. They don't SOLVE problems, they POINT OUT problems. They make observations and hypotheses and link them in coherent systems, or theories).

We propose to implement stereotypes by means of a bundling mechanism applied to entire concepts, rather than by an explicit stereotype data structure. In other words, when the stereotype of a concept is desired, it is produced by starting at the "top" of the concept thread structure, and following the "thickest" bundle down to some instance of the concept, which then serves as the desired stereotype. The concept itself thus serves as a base from which to generate the stereotype, and there is no need for a separate data structure [9],[22].

Proceeding in this way has important advantages. First, it is in accord with the general principle that it is best to avoid local conventions with global consequences. If we had a data structure of some sort specifically for sterotypes, that data structure would of necessity have to be updated, at least in some cases, when a new instance of the concept was encountered. That, in turn, would constitute an undesirable global consequence, since it would involve diverting attention to the general issue of stereotypes, and away from the issue at hand, to which stereotypes might or might not be relevant. We claim, moreover that it is exceedingly difficult to learn this kind of behavior. In order to insure correct operation, there would have to be a set of conventions which say that when certain kinds of instances are encountered, the sterotype is updated in such and such a way, etc. These conventions must be formulated and debugged, and it is very hard to see exactly how to do this. Second, one would very likely be driven to introduce additional data structure for statistical purposes, so as to keep track, for example, of whether this is the first "stripless tiger" we have seen or whether maybe we really had the wrong impression and the sterotypical tiger is really stripless. Any such data structure would merely compound the difficulties mentioned above.

Using the bundling idea, we get a entirely different and much brighter picture. When encountering a new instance of a concept, we merely add it in the usual way. There is no non-local computation, and in particular, none having to do with sterotypes. Moreover, the thread memory itself serves the statistical function. After we see enough stripless tigers as compared to striped ones, the bundle leading to stripless tigers will become thicker than the one leading to striped tigers, and the sterotypic tiger will change. Note that the mechanism will in no way be "aware" at the time it sees the critical stripless tiger that a change in its stereotypic concepts is occurring. Instead, the next time a sterotypic tiger is called for, it will simply turn out to be stripless instead of striped.

Applications of thread memory.

We consider the most important feature of thread memory to be its ability to learn. What are the necessary properties of a memory that permit learning? As Quine points out [19] one thing that is basic for the activity of learning is the ability of an individual to recognize perceptual similarities. By recognizing perceptual similarity one can relate new episodes to past episodes. In order to do that, Quine argues, episodes leave traces, which preserve enough information to show perceptual similarity between a current episode and a later one.

Perceptual similarity is charactrized by degree and strength. By degree of perceptual similarity we understand the degree in which an episode is similar to an other episode. In other words, we say that "A is more similar to B than A is similar to C", where A and B are already perceived episodes. The trace of a past episode fluctuates in strength, where strength is related to the possibility to reactivate a trace. Traces tend to wear out with time, but they can be strenghtened by repetition, we are reminded of past episodes by similarities in the present.

Differences in degree of similarity must be explicit in the individual's learning pattern. Perceptual similarity varies with the individual but the same time, as Quine claims, it has a degree of objective validity because of its innatnes. An individual's inductive expectations are reached by extrapolating along lines of perceptual similarity: similar experiences are expected to lead to similar results.

We have focused our attention, first, on the learning of nominal concepts from children's reading books and second, on the rehabilitation of patients with anomia.

We restrict our applications to the semantic level of the language, accepting the hypothesis that the various levels of language are autonomous. As Jakobson puts it, this autonomy

> doesn't mean isolationism; all levels are interrelated. Autonomy doesn't exclude integration, and even more -- autonomy and integration are closely linked phenomena. But in all linguistic questions and especially in the case of aphasia, it is important to approach the language and its disruption in the framework of a given level, while remembering at the same time that any level is what the Germans call das Teilganze and that the totality and the interrelation between the different parts of the totality have to be taken into account. Here very often linguists commit a dangerous error, namely, they approach certain levels of language with the attitude of heteronomy (colonialism), rather than of autonomy. They treat one level only from the point of view of another level." [12]

At the present we don't deal with the aquisition of grammar, or with its analog, syntactic aphasia, nor do we take the phonological level into consideration.

Learning nominal concepts.

It has been shown by several researchers that children first learn one word sentences, then phrases such as "blue sky" and "little boy", and finally learn subject and predicate construction. As Jakobson points out [11], the acquisition of such a construction is agenuine mental and verbal revolution. Only when the child is able to use the subject and the predicate in relation, the spontaneous use of language begins. With the first nouns that a child uses, there is supposed to be assiociated a "psychological predicate", such as "see", or "give", etc. When a child sees a cat and says "cat" or wants an apple and says "apple", the "psychological predicates" "see " and "give" are assumed to be implicit in his utterance. Only after the child learns some nominal concepts, from his everyday environment, does he begin to learn verbal categories.

The corpus of examples we have used consists of children's primers. The reasons for this choice are the following:

(1) the concepts that primers contain are very simple; one doesn't need much previous knowledge to be able to learn them.

(2) The domain is open-ended. That is, we have extensive materials available which lead to grade school readers, etc.

(3) The input stimuli are mainly simple words, which are easily input to the computer. However, the books also have pictures whose inputting presents technical problems. Usually, examples can be chosen so that the pictures are relatively unimportant. In many cases the older sort of primer, which tends to be fairly self contained, may be more suitable for this purpose than the more modern ones which attempt to build upon the child's life experience to a greater extent.

How does the computer proceed to learn nominal concepts? It observes that a certain class of words, DICK, JANE, and SPOT, for example, may occur in certain positions relative to other words. Moreover, observes that words in this class have certain common features. If an unknown word appears in a context in which a "noun" is expected, the system can guess that this word is a "noun", and assign it properties which are common to "nouns". This sort of concept is not limited to parts of speech. If the computer observes that something which IS-A person appears in a certain context, a concept may be formed exactly as before.

How does the computer go about noticing these regularities? We start by inputting some sentences from small children's speech, or from their very first books, as LEADS-TO threads on S (for sentence). For example: (LEADS-TO (SEE DICK RUN) S) is typed in as text. This results in the thread (S LEADS-TO) SEE -> DICK -> RUN being formed in the computer.

We then input (SEE JANE RUN) in the same manner, resulting in a similar simple thread being added. We obtain

(S LEADS-TO) SEE -> DICK -> RUN

(S LEADS-TO) SEE -> JANE -> RUN

At this point, a generalized thread optimization method, called "collapsing the bubble", can come into play. As it stands, the fork point, the point where the two simple threads start to differ, is immediately below the root. Both simple threads, however, have the same first token, namely SEE. In such a case, collapsing the bubble means rearranging things so that both simple threads share a single pointer to SEE, and the fork occurrs below that node. We will have then

(S LEADS-TO) SEE -> DICK -> RUN

SEE -> JANE -> RUN

In a similar fashion, the combining fork can be moved up resulting in

```
        (S LEADS-TO) SEE -> DICK -> RUN
                          -> JANE ->
```

The structure is now a "one token bubble", which triggers micro-concept formation. We obtain then:

(S LEADS-TO) SEE -> C1 -> RUN

(C1 EXAMPLE-OF) -> DICK

(C1 EXAMPLE-OF) -> JANE and following additions to other threads,

DICK -> C1 -> DICK

JANE -> C1 -> JANE

In sum, this says that in the particular context of SEE xxx RUN, the tokens DICK and JANE may be used interchangably. Later, if other words are seen in the context of SEE xxx RUN, they may be added to the microconcept. Of course the context doesn't have to be verbalized in the case of a small child, but can occur as a "pshychological predicate".

As the system processes other sentences, it may well notice other contexts in which DICK and JANE are used interchangably, this will result in other microconcepts similar to C1. What we need now is a mechanism by which similar microconcepts can be recognized and grouped. It would be most undesirable, however, to forcibly identify two microconcepts as identical at a single moment in time. Doing so might well prove to be erroneous, and recovery from such an error might be very difficult. Instead, we seek a mechanism whereby then can gradually become more and more closely associated. The first step in providing such a mechanism is called micro-generalization. Roughly, this process consists of locating shared superordinate classes of the available exemplars of the micro-concept. In our example of SEE xxx RUN, the main micro-generalization we are aiming for turns out to be something like ANIMATE-OBJECT. However, it is quite acceptable and in fact desirable to bring over other shared superordinate nodes such as PHYSICAL-OBJECT, HUMAN-BEING, etc. Each superordinate node "brought over" is placed on the (<concept> EXAMPLE-OF) thread in the same relative position it had in the IS-A thread of the exemplar. The usual thread memory operations of bundling, and collapsing the bubble are then allowed to operate. Note that it is not necessary that all exemplars of a microconcept share a superordinate category for it to be "brought over". Micro-generalization does not necessarily occur "synchronously" with anthing else, in particular, it is not necessarily synchronous with "conscious" activity. Instead, certain rates and policies are defined, and nodes "migrate" in accordance with these regardless of what storage or retrieval operations are taking place in the thread memory. Given the three examples (SEE DICK RUN), (SEE JANE RUN), and (SEE SPOT RUN), the microconcept exemplar thread, after some time, might look like this.

(C1 EXAMPLE-OF)

-> PHYSICAL-OBJECT[3] -> ANIMATE-OBJECT[3] ->

- > HUMAN [2] -> BOY[1] -> DICK

- > GIRL[1] -> JANE

-> DOG [1] -> SPOT

(The numbers in brackets are the thickness of the relavent strands).

By treating every context of every word as a separate micro-concept, we clearly achieve great generality. The actual learning process would be very painful if it were necessary to rederive from scratch all knowledge about a word each time it was seen in a new surface context. To avoid this, we define a measure of closeness between concepts called neighborness. The idea is that if the desired information can not be found from the micro-concept at hand, it can be "borrowed" if necessary from a closely neighboring concept. If the result proves acceptable, the neighbor-ness of the two concepts can be further reinforced; if it leads to a gaffe, it can be inhibited. In all cases we retain the ultimate capability to rebuild the concept completely from scratch, if necessary. Note that neighborness need not be an absolute measure; we need only decide which of two micro-concepts is nearer to a third.

The suggestion for a neighbor-ness measure is motovated by the bundling analogy. We visualize the simple threads of an EXAMPLE-OF thread layed out as if they were strands of a frayed string. A large bundle leaves the SEMI-ROOT, dividing into smaller and smaller sub-bundles until the end consists of the individual strands, fully separated. Taking three such EXAMPLE-OF threads, we proceed from SEMI-ROOT, considering each segment by segment. We consider the existance of mutual segments and the thickness of those mutual segements are the primary factors contributing to neighborness. For example, suppose we are now presented with the sentences (DICK IS A BOY), (JANE IS A GIRL) and (SPOT IS A DOG). Suppose further that we manage to microconceptualize BOY, GIRL and DOG, such that we are left with DICK, JANE, and SPOT forming single-token bubbles, and thus they get microconceptualized into some microconcept C2, which would subject to the process of micro-generalization, and might very well become a copy of the micro-concept C1 presented above.

Rehabilitation of patients with traumatic aphasia.

Our interest is focused exclussively on the aphasia resulting from brain damage in people who previously have used language normaly. The good recoveries in severe cases of aphasia are rare.

It is very likely that aphasics have a reduced set of words available for communication, or perhaps a reduced access to a preserved set.[1] Thus difficulties in naming as well as in word-finding are common characteristics of aphasic syndromes. This selective impairement is generally called anomia, and the syndrome that is characterized by it, amnesic aphasia though various authors also give it different names. For example Head calls it "nominal aphasia" and Wepman to it as "semantic aphasia". Anomia presents some of the most difficult problems faced by doctors and therapists attempting to restore speech in patients with traumatic aphasia [24]. An important part of this entreprise is to try to restore the stability of memory. No other task in the treatment of sensory aphasia is so difficult as restoration of the ability to rememeber words.

Patients show severe impairment in the ability to recall words even long after they have regained the ability to understand others people speech. Their active speech continues to be restricted by an amnesic type of disorder.

In the following, using thread memory, we provide a simple model of anomic aphasia and give some suggestions about the way in which damaged memory recovers. We believe that loss of memory, in the case of anomic aphasia, is in large part due to the inhability of the patient to perform certain operations such as: accessing a thread, when the key is given. To recover from this loss he has to perform other operations that lead eventually, when combined, to the same result: retrieving the meaning of the word.

What happens in the aphasic's memory? We use the thread memory as an explanatory model and as corpus of data patients that were presented at the Aphasia Rounds at the Boston Veterans Administration Hospital (VAH) in June 1979, with some examples from the literature as well. From the cases presented at the VAH we can see that there is some residual verbal material that is left in the aphasic's memory, so, the patient's ability to speak can be unlocked through the use of certain paradigms. This remark is consistent with our suggestion that in a thread memory concepts are represented by general threads rather than by simple threads. The information about each concept is represented by a set of threads each of them supposed to end in the key, which is the concept represented. We postulate that an individual recognizes a concept if he can access its general thread, or at least the needed simple complete thread. By a complete thread we mean a thread that ends in a node identical to its key. In a normal memory all the simple threads constituting a general thread are complete. In the case of aphasia the threads can be are broken, they don't end in the key [24]. To get the patient to access the right complete thread is a hard task for a therapist. In an experiment done at the VAH, the patient J.J. with a global aphasia, was shown a set of objects: a ball, a tooth brush, a wallet, a lock. When asked which object was the ball he pointed at the lock, when asked what children use to play with he mummbled that he didn't know, when asked what jumps on the floor, he pointed at the ball. Thread memory can explain these results, if we assume that the general thread "ball" was damaged, but some simple threads in it were left intact. The patient was unable to recognize all the characteristics of the object, but if the therapist happened to ask "the right question" that is, one that input an intact thread, then the patient answered correctly. This provides support for the hypothesis that concepts are stored as general threads and not as simple threads. We have defined a general thread as being a set of threads that are pointed to by the same key.

Other data collected at the VAH or represented in the literature [4],[24] show that patients may memorize a word, but then forget it very quickly, even repeating it several times.

A way which we think that could lead to a fixation of generalized verbal image is by using a given key on a general thread. The advantage of using a general thread instead of a simple thread is that the patient accesses at one time a variety of association with a particular key, and on the basis of these associations can recall words much more dependably in context than by their simple memorization. In this use of our proposed model of memory the advantage of having the pointers oriented from the more general to the more specific becomes clear. In bottom up models of memory it would be difficult to access the more specific information.

In thread memory we express a general thread in the following way:

BOY -> LIVING THING -> PERSON -> MALE -> BOY
BOY -> MALE -> STUDENT -> BOY
BOY -> STUDENT -> BLOND HAIR -> BOY
BOY -> PERSON -> SON -> BOY
BOY -> LIVING THING -> CHILD -> BOY
etc..

Through the common key one can access information from any simple thread, by making selections. In the case of brain damage if cognitive functions, such as making selections, are impaired, the only way to perceive cognitively the symbol accessed by a key is to get one of the intact simple threads activated. An aphasic may very well not recognize that 'a boy is a son', but at the same time recognize that 'a boy is a child'. We have evidence that after a thread is accessed, the patient is able to use the knowledge associated with that thread, or with the part of it that is left intact.

For example, at the VAH, the patient F.B. with a Wernicke aphasia, was asked to show parts of his body, as pointing at his nose, etc... He pointed instead at his knee. We would model this in thread memory as follows:

NOSE -> PART- OF- BODY -> PART- OF- THE FACE -> NOSE

NOSE -> HUMAN -> PART-OF BODY -> NOSE etc..

The hint that the question refers to "part of body" was given to the patient, so he activated the threads containing it, but leading to a wrong semantic node: "knee", instead of "nose" . An other example is the following: the therapist asked the patient J.K. to point at the window, or at the door. He didn't understand the question. When the same question was repeated more in detail: "I will ask you to show me some objects in this room", and then "show me the window", or "show me the door", the answers were correct. In the case of J.K. he could answer correctly only at one qustion at the time. For example in response to the question "Show me the window and then show me the door", he pointed at the first object correctly and then totaly forgot about the second. How can we explain that with thread memory? First by knowing a more general node (e.g. part of body), and the key, it is sure that the patient if he is capable at all to understand the question, will access the right thread or a neighbor thread (e.g. instead of "nose", "knee"). By neighbor threads [24] we mean threads that have in common beside the general nodes (as thing, living thing, animal, etc..), more particularizing nodes (as part-of-body, or object-in-room, object-to-use a kitchen),etc.. We can talk about degrees of neigborhood between threads depending upon the number of nodes that they have in common, or in otherwords depending upon the depth of the fork-point. This gives us evidence that there is rather a top-down orientation of the pointers in a thread .

Another example in the support of the hypothesis of the top down orientation of the pointers, is the following: at the VAH, the patient SM very aphasic, was shown a list of words: LAKE, POND, STREAM, WATER. He was asked to point at POND and STREAM, but in both cases pointed at WATER. The question was whether he understood that both elements were composed of water ? Then he was given the list: SOW, POND, AXE, DESK and he was asked again to point at POND. In this case his answer was corect. He gave the correct answer, to the questions: WHICH ONE IS A BODY OF WATER?, and WHICH ONE IS A FAT PIG? It is pretty clear that the patient MS understood the meaning of the words written on the list, but that again, he didn't have the complete thread. He had something like

POND -> ... -> WATER

STREAM -> .. -> WATER

instead of

POND -> .. -> WATER -> .. -> POND

where POND and STREAM in the first place, at the beggining of the thread are the verbal stimuli. Each time the patient accessed the right key, because he got the same meaning. The threads were broken after a certain level, so the patient could not get the specific name that normaly is stored as the last element of the thread, and coincides with the key by which the thread is accessed. The patient was asked to look again at the list SOW, POND, AXE, DESK, and to point at the word that is the most similar to LAKE. Correctly he pointed at POND, an act which suggests that he knew the meaning of words and was able to recognize similarities. This capacity is essential in learning or relearning (which is the case in memory rehabilitation in aphasia).

A number of writers [4] on aphasia have noted that a patient who is unable to recognize isolated words is often able to recognize their meaning if they occur in the context of other words. So for example a patient who is unable to name an object can sometimes recall the name by producing a sentence in which the name appears. So, if he cannot name the word BOY, he may be able to recall it by saying " see girl play ..see boy play", "girls run .. boy run". This compensatory mechanism is made possible by the fact that while the aphasic loses the ability to produce isolated words, which is to associate a key with a thread , he still retains larger, familiar speech patterns which are organized by the LEADS-TO link. The idea is that the patient conserves some patterns, that could be typed in as

```
(LEADS-TO (SEE GIRL PLAY) S)
```

resulting in the following thread in the computer

```
(S LEADS-TO) SEE -> GIRL -> PLAY
```

Knowing that BOY and GIRL belong to the same paradigm, the patient will then say correctly: SEE BOY PLAY, which is obtained from the remembered pattern and the known paradigm obtained initially by 'collapsing the bubble'. (We have seen that by this operation one obtains micro-concepts, which are sets of tokens that can be used interchangeably in the same context). Usually, in some cases of aphasia, the patients can trigger the rest of the paradigm by having an example activated. .

Another case where thread memory organization can be proved useful, is in explaining those types of aphasia in which the patients manifest difficulties in memorizing the meaning of individual words, but they have fewer difficulties in memorizing lists of related words. We believe that in the case of a memory organized as thread memory is, a charting path through a search space can help to restore the meanings of words. The search space is a whole chunk of threads that satisfy certain properties. The charting path results from experience and, in general, represents some sterotyped activity. The aphasic recalls the experience, which has been repeated many times, and then he names a single word. In an experiment done at the VAH, the patient BF, who had a very severe Wernicke aphasia , was shown a set of objects: a COMB, a BELL, a RAZOR, a SAW, a HAMMER. He was asked to name them but he was completely unable to do so; asked what the use was of some of this objects his answers were reasonably correct. He could remember the functions of the objects but not their name.

Concluding Remarks

There is no doubt that the nature of a theory depends on the questions it is designed to answer. The thrust of our research is consistent with the recent trends in cognitive science which focus on the modelling of cerebral mechanisms by which concepts are assimilated, stored, retrieved and combined. Research to verbal learning, lexical semantics, natural language processing has brought out a great deal about the development and mental structure of concepts, but without contributing materially how storage, utterance, retrieval, etc..come about. By modelling the acquisition of concepts in children, we can learn about the source of their language difficulties and how this difficulties can be met. The study of cognitive processes in brain damaged patients is an opportunity both for important theoretical work and for direct application of significant theoretical results to practical problems of rehabilitation. Extensive experimentation and theoretical ananlysis is called for in terms of both children's concepts learning and brain damaged patients regaining the ability of language comprehension and language production. A great deal has already been done (see [24] for a survey), but it is fair to say that the most important work lies ahead of us.

Notes

0) We wish to thank W. A. Martin, Norman Geshwind, Marvin Minsky, Harold Goodglass, David McDonald, Henry Lieberman and Edgar Zurif for their very meaningful comments of the first version of this paper and for encouraging us to pursue our research.

1) The actual program, SEMI, developed by Richard Greenblatt runs on the LISP MACHINE at the Artificial Intelligence Laboratory at MIT. At this stage of its existence the program is able to conceptualize nominals, to answer questions about them, to make deductions and to remember them.

BIBLIOGRAPHY

General references.

[0] Geshwind,N. (1972) " LANGUAGE AND THE BRAIN" in "Scientific American", vol.22/4

[1] Goodglass,H. and Geshwind,N (1976) "LANGUAGE DISORDERS (APHASIA)", in E.C.Carterette and M.P.Friedman (eds) HANDBOOK OF PERCEPTION, Vol.7 Speech and Language. New York: Ac. Press.

[2]H.Hecaen, M.L.Albert (1978) "HUMAN NEUROPSYCHOLOGY",John Wiley&Sons, Inc.

[3]Lesser,R. (1979) "LINGUISTIC INVESTIGATIONS OF APHASIA" Elsevier, New York.

[4]Luria,A.R. (1947) "TRAUMATIC APHASIA", Academy of Medical Sciences Press, Moscow.

[5]Luria,A.R. (1976) "THE NEUROPSCHYCOLOGY OF MEMORY",V.H.Winston&Sons, Washington D.C.

[6]Weinreb,D. and Moon,D. (1979) LISP MACHINE MANUAL,M.I.T.-Artifucial Intelligence Lab.

References.

[7]Brachman,R.J. (1977) "A STRUCTURAL PARADIGM FOR REPRESENTING KNOWLEDGE", Ph.D.dissertation, Division of Engineering and Applied Phisics, Harvard University.

[8]Fahlmann,S.E. (1977) "A SYSTEM FOR REPRESENTING AND USING REAL-WORLD KNOWLEDGE", Ph.D. dissertation, Dep. of Electrical Engineering and Computer Science,M.I.T.

[9] Greenblatt,R.D (in preparation) "INTRODUCTION TO THREAD MEMORY"

[10]Jackendoff,R. (1978) "Grammar as evidence for conceptual structure" in "LINGUISTIC THEORY AND PSYCHOLOGICAL REALITY" eds. M.Halle, J.Bresnan, G.Miller M.I.T.-Press, Cambridge.

[11]Jakobson,R. (1977) "Aphasic disorders from a linguistic angle" in SCANDINAVIAN JOURNAL FOR LOGOPEDICS AND PHONIATRICS ,vol:2, nr:2,Special Paedagogisk Forlag, Danmark.

[12]Jakobson,R. (1971) "STUDIES ON CHILD LANGUAGE AND APHASIA", Mouton Publ.House, Holland.

[13] Lyons, J. (1968) "INTRODUCTION TO THEORETICAL LINGUISTICS", Cambridge Univ.Press.

[14]Marr,D and Nishihara,K. (1977) "REPRESENATION AND RECOGNITION OF THE SPATIAL ORGANIZATION OF THREE DIMENSIONAL SHAPES", M.I.T.- A.I. lab. Memo. 416.

[15]Miller,G.A., Johnson-Laird,P.N. (1976) "LANGUAGE AND PERCEPTION" Harvard Univ.Press, Cambridge, Mass.

[16]Miller,G.A.(1978) "Lexical meaning" in "SPEECH AND LANGUAGE IN THE LABORATORY SCHOOL AND CLINIC",edited by James F.Koronogh, M.I.T.Press, Cambricge,Mass.

[17]Putnam,H. (1975), "The meaning of meaning" in " MIND, LANGUAGE, REALITY" Phil.Papers. vol2. Cambridge Univ. Press

[18]Putnam,H. (1978), " Meaning and the moral sciences", Routledge&Kegan Paul.

[19]Quine,W.N.(1974), "THE ROOTS OF REFERENCE", Open Court Publ.Comp.

[20]Rosch,E.H.(1973), "On the internal structure of perceptual and semantic categories" in "COGNITIVE DEVELOPMENT AND AQUISITION OF LANGUAGE",Academic Press,New-York.

[21]Schank,R. C.and Abelson,R.P.(1977) "SCRIPTS, PLANS, GOALS, AND UNDERSTANDING", Hillsdale: New Jersey: Lawrence Earlbaum Associates.

[22] Vaina,L.M. (in preparation) "APLICATION OF POSSIBILITY THEORY TO ANOMIC APHASIA"

[23] Vaina,L.M. and Greenblatt R.G. (in preparation) "MEMORY AND REASONING"

[24] Vaina,L.M. (in preparation) "NAMING PROCESS AND ITS DISRUPTION IN APHASIA"

[25]Warrington,E.K. (1975) "The selective impairement of semantic memory" in "QUARTERLY JOURNAL OF EXPERIMENTAL PSYCHOLOGY", vol:27,nr:3.