# Language Acquisition:
# Coping with Lexical Gaps

Uri ZERNIK

Artificial Intelligence Program
GE, Research and Development Center
PO Box 8
Schenectady, NY 12301
USA

**Abstract**

Computer programs so far have not fared well in modeling language acquisition. For one thing, learning methodology applicable in general domains does not readily lend itself in the linguistic domain. For another, linguistic representation used by language processing systems is not geared to learning. We introduced a new linguistic representation, the Dynamic Hierarchical Phrasal Lexicon (DHPL) [Zernik88], to facilitate language acquisition. From this, a language learning model was implemented in the program RINA, which enhances its own lexical hierarchy by processing examples in context. We identified two tasks: First, how linguistic concepts are acquired from training examples and organized in a hierarchy; this task was discussed in previous papers [Zernik87]. Second, we show in this paper how a lexical hierarchy is used in predicting new linguistic concepts. Thus, a program does not stall even in the presence of a lexical unknown, and a hypothesis can be produced for covering that lexical gap.

## 1. INTRODUCTION

Coping with unkowns is an integral part of human communication which has been ingnored by previous linguistic models [Chomsky81, Bresnan82, Gazdar85]. Consider the following sentence produced by a second language speaker:

John suggested her to go out, but she refused.

This incorrect use of suggest could be viewed as a communication failure, since by text-book grammar suggest does not take this form of the infinitive. Alternatively, this can be viewed as a surprising success. In spite of missing lexical information, a person managed to convey a concept, rather than give up the communication task altogether. Our aim here is to explain such robust human performance in computational terms, and consequently to describe the principles underlying the program RINA [Zernik85, Zernik86] which models language acquisition.

### 1.1 The Modelled Behavior

The problems arising from incomplete lexical knowledge are illustrated through the following scenario. In this scenario RINA encounters two unknown words plend*, and doove, and uses the word suggest whose lexical definition is incomplete.

**Input text:**

**User:** Corinne needed help with her homework.
Her friend Frank called and **plended** her to come over.
But she **dooved** to stay home.

Paraphrased text:

**RINA:** Frank **suggested** her to come over.
Corinne turned down the suggestion.

RINA reads a paragraph provided by a user, and then generates text which conveys the state of her knowledge to the user. The first problem is presented by the word plend which does not exist in RINA's lexicon. RINA is able to extract partial information: *Frank communicated a concept to Corinne regarding coming over*. It is not clear, however, who comes over. Did Frank *promise* Corinne to come over to **her**, or did Frank *ask* Corinne to come over to **him**?

The word doove is also unknown. Here too, RINA can guess the main concept: Corinne *decided* not to come over. This hypothesis is not necessarily correct. However, it fits well the context and the structure of the sentence.

At this point, RINA must respond to the input text by generating a paraphrase which conveys its current hypothesis. Also in generation, RINA faces the problem of incomplete lexical knowledge. In absence of specific knowledge regarding the use of suggest, RINA produced an incorrect sentence: he suggested her to come over, which albeit incorrect, is well understood by a human listener.

### 1.2 The Issues

The basic problem is this: how can any program parse a sentence when a lexical entry such as doove or plend is missing? And equivalently, how can a program use a lexical entry--suggest--which is not precisely specified? Three knowledge sources must be negotiated in resolving this problem.

---

* The dummy words plend and doove are used here to bring home, even to native English speakers, the problems faced by language learners.

**Syntax and Control:** In Frank asked Corinne to come over, the word ask actually *controls* the analysis of the entire sentence [Bresnan82a], and determines the answer to the elementary question,

*who comes to whom?*

The embedded phrase to come over, which does not have an explicit subject obtains its subject from the *control matrix* [Bresnan82a] of ask. Accordingly, Corinne is the subject of "coming over". On the other hand, in he plended her to come over, the controlling word plend, is yet unknown. In absence of a control matrix it is not clear how to interpret to come over. How can a program then, extract even partial information from text in such circumstances?

**Lexical Clues:** Although plend itself is unknown, The form of the sentence X plended Y to come over, suggests that "X communicated to Y a concept regarding coming over". Three assumptions are implied: (a) plend is a communication act, (b) Y is the actor of "coming over", (c) "coming over" is only a hypothetical, future act (and not an act which took place in the past). How is this intuition, which facilitates the initial hypothesis for plend, encoded in the lexicon?

**Contextual Clues:** The hypothesis selected for doove above is a direct consequence of the context, which brings in a structure of plans and goals: (1) Corrine has an outstanding goal; (2) Frank suggests help. Given this background, the selected hypothesis is: (3) Corinne rejects the offer. This selection is problematic since doove could stand for other acts, e.g., she wanted to stay, she tried to stay, and she forgot to stay, etc. Thus, how does the context impact the selection of a hypothesis?

Some of the issues above can be handled by specific heuristic rules, custom tailored for each case. However, the challenge of this entire enterprise is to show how a unified model can employ its "normal" parsing mechanism in handling "exceptions".

### 1.3 The Hierarchical Lexicon

Humans perceive objects in conceptual hierarchies [Rosch78, Fahlman79, Shapiro79, Schank82]. This is best illustrated by an example from peoples's communication. Consider the question: what is Jack? The answer Jack is a cat is satisfactory, provided the listener knows that a cat is a mammal and a mammal is an animate. The listener need not be provided with more general facts about Jack (e.g., Jack has four legs and a tail), since such information can be accessed by *inheritance* from the general classes subsuming a cat. In fact, for a person who does not know that cats are mammals, an adequate description of Jack should be more extensive.

Hierarchical organization is essential in dynamic representation systems for three reasons:

o **Economy:** A feature shared by multiple instances should not be repetitively acquired per each instance. Such redundancy should be avoided by inheriting shared features from general classes.

o **Learnability:** As shown by [Winston72, Mitchell82, Kolodner84], through a hierarchy learning can be reduced to a search process. When one acquires a new zoological term, for example *feline*, one can traverse the conceptual hierarchy, by *generalizing* and *specializing*, until the appropriate location is found for *feline* in the hierarchy--above a number of specific species, and below the general *mammal*.

o **Prediction:** Hierarchy accounts for *predictive power*, which allows learning models to form intelligent hypotheses. When first observing a leopard and by assuming it is a feline, a learner, who has not been exposed to prior information about leopards, may hypothesize that this new animal feeds, breeds, and hunts in certain ways, based on his/her knowledge of *felines* in general.

While it is clear how hierarchy should be used in representing zoological concepts, it is not clear how it applies in representing linguistic concepts. Can linguistic systems too benefit from a hierarchical organization? Following [Langacker86] and [Jacobs85] we have shown in DHPL (Dynamic Hierarchical Phrasal Lexicon) [Zernik88] how elements in a lexicon can be organized in a hierarchy and thus facilitate a dynamic linguistic behavior.

## 2. THE LEXICAL HIERARCHY FOR COMMUNICATION ACTS

Consider DHPL's lexical hierarchy for *communication acts* [Kiparsky71]. This is a hierarchy by generality where specific instances reside at the bottom, and general grammar rules reside at the top. Given this hierarchy, which turns out to be incomplete, RINA is capable of coping with a missing specific phrases by inheriting form general categories.

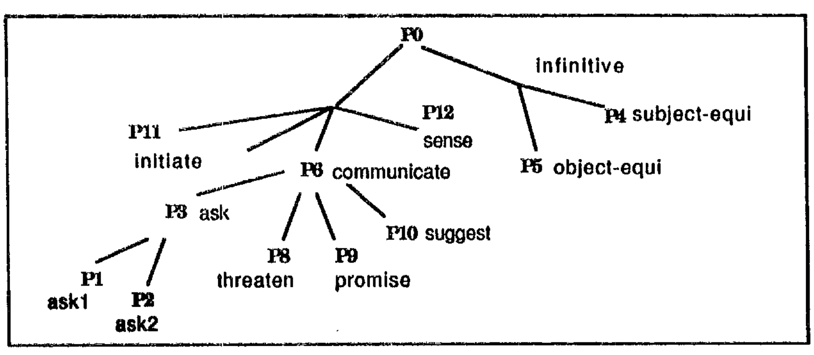


Figure 1: The Hierarchy for Complement-Taking Verbs

Each node in this hierarchy, denoted for reference purposes by a mnemonic word, is actually a full-fledged *lexical phrase*–an association of a *syntactic pattern* with its *conceptual meaning*.

### 2.1 Specific Phrasal Entries: Two entries for ASK (P1 and P2)

Consider the representation of the word ask as it appears in the sentence below:

(1) The meeting was long and tedious.
So Frank **asked** to leave early.

**pattern:** X:person ask:verb Z:act
**concept:**X communicated that act Z **by X** can achieve a goal G of X.

The word ask is given in the lexicon as an entire phrase, or a *pattern-concept* pair [Wilensky81]. The abbreviated notation for P1 above stands for a full-fledged frame [Mueller84] as shown below:

```
(pattern (subject (instance X))
         (verb    (root ask)
         (comp    (concept Z))
(concept (mtrans  (actor X)
                  (object (plan Z)
                          (achieve (goal-of X)))))
```

The pattern of the phrase has three constituents: a *subject* X (Frank), the *verb* itself, and a *complement* Z (to leave early). In particular, the semantics of the phrase specify that X is the subject of the embedded act Z, a fact which is not explicit in the text. However, this specification fails in capturing further sentences, such as the following one.

(2) Frank asked **the chairman** to adjourn the meeting.

There are two discrepancies: (a) this sentence includes a direct object (the chairman), and (b) Frank is not the subject of the complement as prescribed in phrase P1. Thus, a second phrase P2 is added on to account for sentences of this kind.

**pattern:** X:person ask:verb **Y:person** Z:act
**concept:**X communicated **to Y** that act Z **by Y** can achieve goal G of X

However, in order to cope with lexical unknowns, common properties shared by such phrases must be extracted and generalized.

### 2.2 Generalized Features

The phrases P1 and P2 above can be abstracted in three ways: (a) along semantics of general equi rules, (b) along the semantics of the word ask, and (c), along semantics of general communication verbs. When an unknown word is encountered, its behavior is derived from these general categories.

**(a) The general entry for ASK (P3):** The semantic properties of ask itself can be generalized through the follwing phrase:

**pattern:** X:person ask:verb Z:act
**concept:** X communicate that act Z can achieve a goal G of X

This generalized phrase simply states the meaning of ask, namely "X communicates that act Z can achieve a goal of X", regardless of (a) who is the object of the communication act, and (b) who executes the act Z.

**(b) The general EQUI-rule (P4 and P5):** Semantic properties can be generalized across complement-taking verbs:

**pattern:** X:person V:verb Z:act
**concept:** X is the subject of the embedded act Z

**pattern:** X:person V:verb Y:person Z:act
**concept:** Y is the subject of the embedded act Z

These phrases dictate the default identity of the implicit subject in complement-taking verbs: it is either the object, or the subject (if an object does not exist) of the embedding phrase.

**(c) The general COMMUNICATION act (P6):** Semantic features of communication acts can be further abstracted.

**pattern:** X:person V:verb Y:person Z:infinitive-act
**concept:** Y communicated Z to Y

Phrase P6 implies that (1) X communicated an act Z to Y, and (2) Z is a hypothetical act. When a new word is encountered, for which no specific phrase can be indexed in the lexicon, a hypothesis is constructed by inheriting general features from these general phrases.

## 3. PHRASE INTERACTION

How does the lexicon become operational in processing text? Consider the following three sentences, ordered according to their complexity.

(1) Frank came over.
(2) Frank asked Corinne to come over.
(3) Frank plended Corinne to come over.

(1) Sentence (1) is analyzed by simple table lookup. A phrase (P7-come over) is found in the lexicon, and its concept is instantiated.

(2) No single lexical phrase matches sentence (2). Therefore, the analysis of (2) involves interaction of two lexical phrases (P2-ask and P7-come over).

(3) No specific lexical phrase matches (3), since it includes an unknown word. Therefore the analysis of (3) requires the use of generalized phrases, as elaborated below.

### 3.1 Unification with a General Phrase

No specific phrase in the lexicon matches the word plend, but a hypothesis regarding the new word can be inherited from general phrases. What general phrase should be used? In our algorithm [Zernik88], properties are inherited from the *most specific* phrase which matches the input clause. In the case of plend above, properties are inherited from two generalized phrases P5-communicate and P6-object-equi, as shown in the figure below:

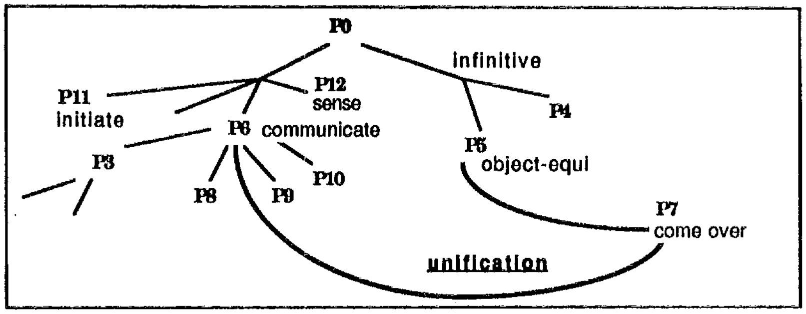


Figure 2: Unification with a Generalized Phrase

While, a single concept was constructed for the word ask in the previous example, for plend there are multiple possiblities to consider. Steps (2) and (3) are carried out for each.

(1) **Select** in the hierarchy all possible categories (general phrases) which match the unknown word. The communication act (P6) is one possible category for plend.

(2) **Unify** the appropriate phrases. The general phrase P6-communicate leaves some parameters unspecified. In particular, the identity of the subject of the embedded phrase is yet unknown–who is supposed to come over to whom? This missing argument is derived by unification with phrase P5, which dictates the default *object-equi*: the listener is supposed to come over to the speaker.

(3) **Instantiate** the constructed hypotheses:

F.13 communicated to C.17 that C.17 will come over to F.13, where coming over achieves a goal of C.17.

Several such hypotheses are instantiated.

(4) **Discriminate** among the multiple hypotheses by their semantic ingredients. For example the preceding context suggests that Corinne's goal (and not Frank's goal) is active. This feature discriminates between two acts such as promise and plead.

## 5. Conclusions

While parsing in general presents problems of ambiguity, in the presence of a lexical gap a situation becomes even further *under-constrained.* So in the case above there are many legitimate hypotheses. In our method we pick one hypothesis which matches the context, and present it to user who may continue the ineraction by providing additional examples.

Our model explains a range of generation and comprehension errors made by language learners who are forced to utilize general approximations. Although the resulting hypotheses sounds awkward (to a native speaker), they certainly convey the main concepts, and a user becomes acknowledged of the model's state of knowledge. The general principle of operation is summarized below:

*Specific phrases are preferred to general phrases. However, in absence of a precise specific phrase, inherit properties of general phrases.*

References


[Bresnan82] Bresnan, J. and R. Kaplan, "Lexical-Functional Grammar," in *The Mental Representation of Grammatical Relations*, ed. J. Bresnan, MIT Press, MA (1982).

[Bresnan82a] Bresnan, J., "Control and Complementation," in *The Mental Representation of Grammatical Relations*, ed. J. Bresnan, The MIT Press, Cambridge MA (1982).

[Chomsky81] Chomsky, N., *Lectures on Government and Binding*, Foris, Dordrecht (1981).

[Fahlman79] Fahlman, S. E., *NETL: A System for Representing and Using Real-World Knowledge,* MIT Press, Cambridge, MA (1979).

[Gazdar85] Gazdar, G., E. Klein, G. Pullum, and I. Sag, *Generalized Phrase Structure Grammar,* Harvard University Press, Cambridge MA (1985).

[Jacobs85] Jacobs, Paul S., "PHRED: A Generator for Natural Language Interfaces," UCB/CSD 85/198, Computer Science Division, University of California Berkeley, Berkeley, California (January 1985).

[Kiparsky71] Kiparsky, P. and C. Kiparsky, "Fact," in *Semantics, an Interdisciplinary Reader*, ed. D. Steinberg L. Jakobovits, Cambridge University Press, Cambridge, England (1971).

[Kolodner84] Kolodner, J. L., *Retrieval and Organizational Strategies in Conceptual Memory: A Computer Model,* Lawrence Erlbaum Associates, Hillsdale NJ (1984).

[Langacker86] Langacker, R. W., "An Introduction to Cognitive Grammar," *Cognitive Science* **10**(1) (1986).

[Mitchell82] Mitchell, T. M., "Generalization as Search," *Artificial Intelligence* **18**, pp.203-226 (1982).

[Mueller84] Mueller, E. and U. Zernik, "GATE Reference Manual," UCLA-AI-84-5, Computer Science, AI Lab (1984).

[Rosch78] Rosch, E., "Principles of Categorization," in *Cognition and Categorization*, ed. B. Lloyd, Lawrence Erlbaum Associates (1978).

[Schank82] Schank, R. C., *Dynamic Memory,* Cambridge University Press, Cambridge Britain (1982).

[Shapiro79] Shapiro, S.C., "The SNePS Semantic Network Processing System," in *Associative Networks*, ed. N.V. Findler, Academic Press, New York (1979).

[Wilensky81] Wilensky, R., "A Knowledge-Based Approach to Natural Language Processing: A progress Report," in *Proceedings Seventh International Joint Conference on Artificial Intelligence*, Vancouver, Canada (1981).

[Winston72] Winston, P. H., "Learning Structural Descriptions from Examples," in *The Psychology of Computer Vision*, ed. P. H. Winston, McGraw-Hill, New York, NY (1972).

[Zernik85] Zernik, U. and M. G. Dyer, "Towards a Self-Extending Phrasal Lexicon," in *Proceedings 23rd Annual Meeting of the Association for Computational Linguistics*, Chicago IL (July 1985).

[Zernik86] Zernik, U. and M. G. Dyer, "Disambiguation and Acquisition through the Phrasal Lexicon," in *Proceedings 11th International Conference on Computational Linguistics*, Bonn Germany (1986).

[Zernik87] Zernik, U., "Learning Phrases in a Hierarchy," in *Proceedings 10th International Joint Conference on Artificial Intelligence*, Milano Italy (August 1987).

[Zernik88] Zernik, U. and M. G. Dyer, "The Self-Extending Phrasal Lexicon," *The Journal of Computational Linguistics: Special Issue on the Lexicon* (1988). to appear.