# PHRAN AND PHRED: ANALYSIS AND PRODUCTION USING A COMMON KNOWLEDGE BASE

Robert Wilensky

Computer Science Division, Department of EECS, University
of California, Berkeley, California 94720, USA

## 1.0 Introduction

We propose a model of language use that is derived from viewing language processing systems as knowledge-based systems. The knowledge that needs to be represented and organized here is the large amount of knowledge about what the utterances of a language mean. In this paper, I describe some of the theoretical underpinnings of the model, and then describe two programs, PHRAN and PHRED, that are based on these ideas. We have conducted a number of experiments with these systems that have some bearing on the utility of the model's presumptions, including testing these systems on other languages (Spanish and Chinese), and implementing one of them in a relational data base system.

## 2.0. The assumptions of the model

### 2.1. The Importance of Non-generative Language

Language user knows a great number of fact about what utterances of their language mean. That is, in addition to knowing the meanings of a large number of words, they know the significance of a set of meaningful linguistic units that are not necessarily understood in terms of their components. Our conjecture is that such units constitute a very considerable fraction of the language knowledge needed by an intelligent language processor.

## 2.2. Sharable Knowledge Base

In our model, it is assumed that the knowledge used for analysis and for production is by and large the same. That is, there is only one data base of knowledge about the meanings of a language's forms. By having the knowledge of the two components be a shared data base, only one form of representation is needed. Moreover, the addition of new knowledge to this data base extends the capabilities of both systems simultaneously.

As this requirement forces knowledge to be represented declaratively, the other benefits of such representations are enjoyed as well. For example, in this format, knowledge about the language is kept separate from the processing strategies that apply this knowledge to the understanding and production tasks. Thus adding new knowledge requires only adding new assertions to the data base, not writing and debugging new code. In addition, other knowledge besides the meaning of a phrase can be easily associated with such declarative representations.

## 3.0. PHRAN and PHRED

We have been developing this model of language use in two related programs, PHRAN (PHRasal ANalyzer) and PHRED (PHRasal English Diction). PHRAN is a language understanding program written by Yigal Arens. It reads English sentences and produces representations from them that encode their meaning. PHRED is a natural language production mechanism developed by Steven Upstill. PHRED takes meaning representations as input and expresses them in English sentences.

Both PHRAN and PHRED share a common data base of language knowledge. This data base contains declarative representations about what the phrase of the English language mean. This knowledge is stored in the form of pattern-concept pairs. A pattern is a phrasal construct of varying degrees of specificity. The concept part of a pattern-concept pair is a conceptual template that represents the meaning of the associated phrase. Together, these pairs associate different forms of utterances with their meanings.

PHRAN understands by reading the input text and trying to find the phrasal patterns that apply to it. Eventually, the conceptual template associated with the desired pattern is used to generate the structure denoting the meaning of the utterance. PHRED produces sentences that encode an idea by examining the same knowledge base.

## 4.0 Spanish and Chinese PHRAN

We have build both a Spanish and a Chinese version of PHRAN simply by changing the pattern-concept data base. These programs lend support to some of the claims we make for our model. We found that it was possible to rewrite most of the patterns into phrases of another language without having the knowledge encoder learn anything about the inner workings of the program. This suggests that a system like PHRAN could be designed to allow fairly easy construction of a language processor for a new language, or to allow for the addition of special purpose phrases or jargon by some user who was not an expert AI programmer.

## 5.0 AI and Relation Data Bases

We implemented a version of PHRAN in a conventional data base system. PHRAN was re-written in EQUEL, a query language for the INGRES relational data base system developed at Berkeley. Tests were run to compare the relative performance of the systems on various size data bases.

The results can be summarized as follows: The LISP version is considerably faster when the data base of pattern--concept pairs is small. However, when the data base is large (2000 words and 500 patterns), the EQUEL version is about 3 times faster than the LISP version. Thus performance problems in natural language may be solved by importing developments in data base technology as the size of our knowledge bases grow.