A Formal Basis for Performance Evaluation of Natural Language Understanding Systems

Giovanni Guida[1] and Giancarlo Mauri[2]

Istituto di Matematica, Informatica e Sistemistica
Università di Udine
Udine, Italy

The task of evaluating the performance of a natural language understanding system, despite its largely recognized relevance, is still poorly defined. It mostly relies on intuitive reasoning and lacks a sound theoretical foundation. This paper sets a formal and quantitative proposal for this task. In particular, a measure of performance that allows the basic input-output characteristics of a system to be evaluated is introduced first at an abstract level. The definition of concrete measures is then obtained by assigning actual values to the functional parameters of the abstract definition; some particular cases are shown and discussed in detail. Finally, the task of measuring performance in practice is considered, and a model for experimental performance evaluation is presented. Comparison with related works is also briefly discussed; open problems and promising directions for future research are outlined. A limited case study experimentation with the model proposed is presented in the appendix.

1. Introduction

Research on natural language processing has recently been featured by the design and implementation of a number of experimental systems. Recent survey reports (Waltz 1977, Kaplan 1982) mention more than one hundred items among the most successful and relevant systems in the classical application fields of data base inquiry, machine translation, question answering, and man-machine interfacing.

This trend is not surprising in the context of research whose specific aim is that of providing automated tools for the understanding or translating of natural languages; but it is also evident even in natural language research with a more theoretical flavour. The successful construction of a good performing system is in fact often considered as the most evident proof of the validity of a theory, and, therefore, designing running systems is routine, and even sometimes the specific goal of several researchers.

The task of evaluating the performance of a given system and that of comparing the behaviour of different systems appears, therefore, to be a fundamental issue. Despite its large recognized relevance (Woods 1977, Tennant 1980), measuring the performance of a system for natural language processing is still poorly defined. It mostly relies on intuitive reasoning and lacks a sound theoretical foundation. As Tennant clearly points out (1980), there is a nearly complete absence of meaningful evaluation in current natural language processing research. This leaves several crucial questions unanswered:

- What is the relevance and value of obtained results?
- How general are the proposed solutions?
- How do they compare with other proposals?
- What problems are still open?
- What directions have to be followed?
- What issues are to be faced in the progress of the research?

[1] Address:
Prof. Giovanni Guida
Dipartimento di Elettronica
Politecnico di Milano
P.zza Leonardo da Vinci, 32
I-20133 MILANO, Italy
Also with Milan Polytechnic Artificial Intelligence Project, Milano, Italy.

[2] Also with Istituto di Cibernetica, Università di Milano, Milano, Italy.

0362-613X/84/030015--16$03.00

The lack of evaluation constitutes a serious obstacle to the development of a sound technology in natural language processing.

The purpose of this paper is to provide a formal and quantitative model for the performance evaluation task. In particular, we give a formal definition of "understanding power", and we propose some techniques for measuring this feature in practice. Our proposal is based on several assumptions we discuss below.

First, we assume as object of our attention only that module of a natural language system that is devoted to understanding natural language, that is, to mapping input expressions into formal internal representations. This can clearly include several kinds of processing activities, such as linguistic analysis, reasoning, inferencing, etc.; but must have as ultimate goal the construction of a correct internal representation, not the production of any type of service to the end user of the natural language system. Thus, for example, a question answering system (Tennant 1979) does not belong to the class of natural language understanding systems that concern us; instead, it is the natural language interface it contains that meets exactly our requirements.

Second, we assume the following naïve notion of performance: the extent to which a system is able to correctly understand natural language expressions in a given application domain. The resources needed by the system to accomplish its task are irrelevant in this case. In other words, we want to capture and measure the "power" of the system, in terms of how much and how well it is capable of understanding, not its "efficiency", that is, how much does it cost (for example, in terms of time and memory requirements) to understand what it is capable of understanding.

Third, we want to define a measure of performance that allows the evaluation of the input-output characteristics of a particular system in a given domain. This kind of measure is clearly inappropriate to reveal and test features, such as the power of a model as opposed to that of a particular implementation of it, the applicability of the model to other domains, its extensibility, etc., which are more closely related to the internal structure and mode of operation of a system, rather than to its input-output behaviour. The goal of evaluating such more general properties, worked on by Tennant (1980) through the method of *abstract analysis* (mainly based on taxonomies of conceptual, linguistic, and implementational issues), is not considered in this work.

This paper is organized in the following way. In section 2 we discuss in an intuitive, yet precise, way the basic concepts involved in the performance evaluation problem, in order to have a sufficiently clear specification of what we want to formalize. Then, in section 3, we give an abstract definition of the formal model, and in section 4 we discuss some actual cases of particular interest. Section 5 presents some techniques that could be used to measure in practice the performance of a natural language understanding system. In section 6 we discuss some concluding remarks, and present open problems and promising topics for future research. A limited case study experimentation with the model proposed is presented in the appendix.

2. Basic Definitions and Statement of the Problem

Let us introduce some background definitions needed to clearly state the problem of performance evaluation, as discussed in this work. The model of natural language understanding we are going to define is so conceived as to include only those very few features that are relevant for the purpose of performance evaluation and is strictly tailored to this particular goal.

Let an *expression* of a natural language be any finite sequence of legal words and punctuation marks from the given language. Let A be the set of all expressions of a natural language.

Note that the above definition is very loose and does not take into account the structure of the expressions. So an expression can be a sentence, a dialogue, a meaningless sequence of words, the whole content of a book, or just a single word. Introducing a more definite notion of expression is not necessary at this point for our purpose of stating the problem of performance evaluation.

Although the above definition includes expressions of arbitrarily (finite) length, so that A contains infinitely many expressions, in a more pragmatic approach the length of existing expressions of a natural language at a given moment of its history has an upper bound. Therefore, it makes sense to restrict our attention to a finite subset E of A, containing all expressions of length less than or equal to an appropriately fixed integer n.

Let L be the set of all *meaningful expressions* of a natural language, that is, of all expressions to which humans attach a meaning. Note that L is defined on a purely semantic basis, so that expressions of L do not have to be syntactically correct with respect to any fixed syntax, and that, generally, more than one mean-

ing may be attached to the same expression, that is, expressions are not required to be univocal.

Let S be the set of all possible *meanings* that can be attached to expressions of E.

We do not face here the problems of what S actually contains or of how S could be represented explicitly (which mostly pertain to cognitive psychology); let us assume S merely as that basic datum, shared by all humans speaking a given language, which allows effective interpersonal communication.

We call the *semantics* of a natural language the total function $f: E \to 2^S$ (into 2^S), which associates to each expression of E the set of all its possible meanings.

Clearly the function f can be computed by any person who can understand perfectly the natural language to which the expressions of E belong (theoretical problems concerning subjective interpretation and disagreement between different people are not considered here).

Moreover, $f(e) = \phi$ denotes that no meaning is associated to the expression e, and hence $e \in L$ iff $f(e) \neq \phi$.

Each expression $e \in E$ such that $|f(e)| \leq 1$ is called an *univocal* expression.

Let now D be a nonempty subset of S that contains meanings all related to a unique subject ("what we are speaking of", "the topic of the discourse", "the conceptual competence of a natural language understanding system"); we call D a *domain*.

Let f_D be the restriction of f to D defined as: $f_D(e) = f(e) \cap D$, for any $e \in E$.

Let $L_D = E - f_D^{-1}(\phi)$ be the restriction of L to D.

It is obvious that $L_D \subseteq L \subseteq E$.

Let us now try to formalize the concept of natural language understanding system.

The main problem is that of giving a formal representation to the informally defined domain D. To this purpose, we take a finite set of symbols B, called *alphabet,* and then we construct a set R of sequences of arbitrary finite length over B (that is, $R \subseteq B^*$), in such a way that to every element $d \in D$ an element of R, $r = h_D(d)$, is associated by a bi-univocal function h_D. The sequence $r = h_D(d)$ is called the *representation* of d, while the set R is called a *representation language* for D.

Obviously, the map h_D^{-1} is a total function $h_D^{-1}: R \to D$, which associates to every sequence of R its informal meaning in D. Both h_D and h_D^{-1} are known to man, in the sense that he is able to compute them.

We are now able to formalize the naïve notion of natural language understanding system in the following way.

Let $D \subseteq S$ be a domain and R a representation language for D. A *natural language understanding system* $U_{R/D}$ in R on D is an algorithm that computes a total function $g^U_{R/D}: E \to 2^R \cup \{\perp\}$ (into $2^R \cup \{\perp\}$), where $\perp$ is called the *undefined symbol.* $g^U_{R/D}(e) = \perp$ denotes that U is unable to assign a meaning to the expression e, that is, that it fails in computing $g_{R/D}(e)$ (not that e has no meaning in the domain D!).

Note that in the above definition we have assumed that a system $U_{R/D}$ should accept as input not only expression of L_D but, generally, all expressions of E. The reason for this choice is that a basic feature of natural language understanding is also to recognize that some expressions are meaningless (they belong to E–L) or are in no way related to a given domain D (they are in $L - L_D$). Clearly, this feature is often less important than the capability of correctly understanding expressions of L_D, but this can be appropriately taken into account when defining a measure of performance.

Measuring the performance of a natural language understanding system $U_{R/D}$ may now be defined as evaluating how well $U_{R/D}$ is capable of explicitly representing in R the meaning of expressions of E.

To define such a notion in quantitative terms we can first extend the bi-univocal function $h_D: D \to R$ to the function (bi-univocal if $\perp$ is not considered)

$$\tilde{h}_D: 2^D \to 2^R \cup \{\perp\},$$

defined by:

$$\tilde{h}_D(x) = \bigcup_{d \in x} \{h_D(d)\},$$

for $x \in 2^D$.

Figure 1 illustrates the definitions of the functions f, f_D, h_D, $\tilde{h}_D$, and $g^U_{R/D}$ presented above.

Considering now the three functions f_D, $\tilde{h}_D$, and $g^U_{R/D}$ defined above, if we denote $\tilde{h}_D \circ f_D = \bar{g}_D$, the performance of $U_{R/D}$ can then be expressed as the degree of precision to which $g^U_{R/D}$ approaches $\bar{g}_D$ over E.

This task raises, however, some difficult problems. Two basic questions are:

(i) how to define the "difference" between $g^U_{R/D}$ and $\bar{g}_D$ over E in such a way to match the intuitive notion of performance;

(ii) how to measure such a "difference" in practice, that is, through an effective experimental procedure.

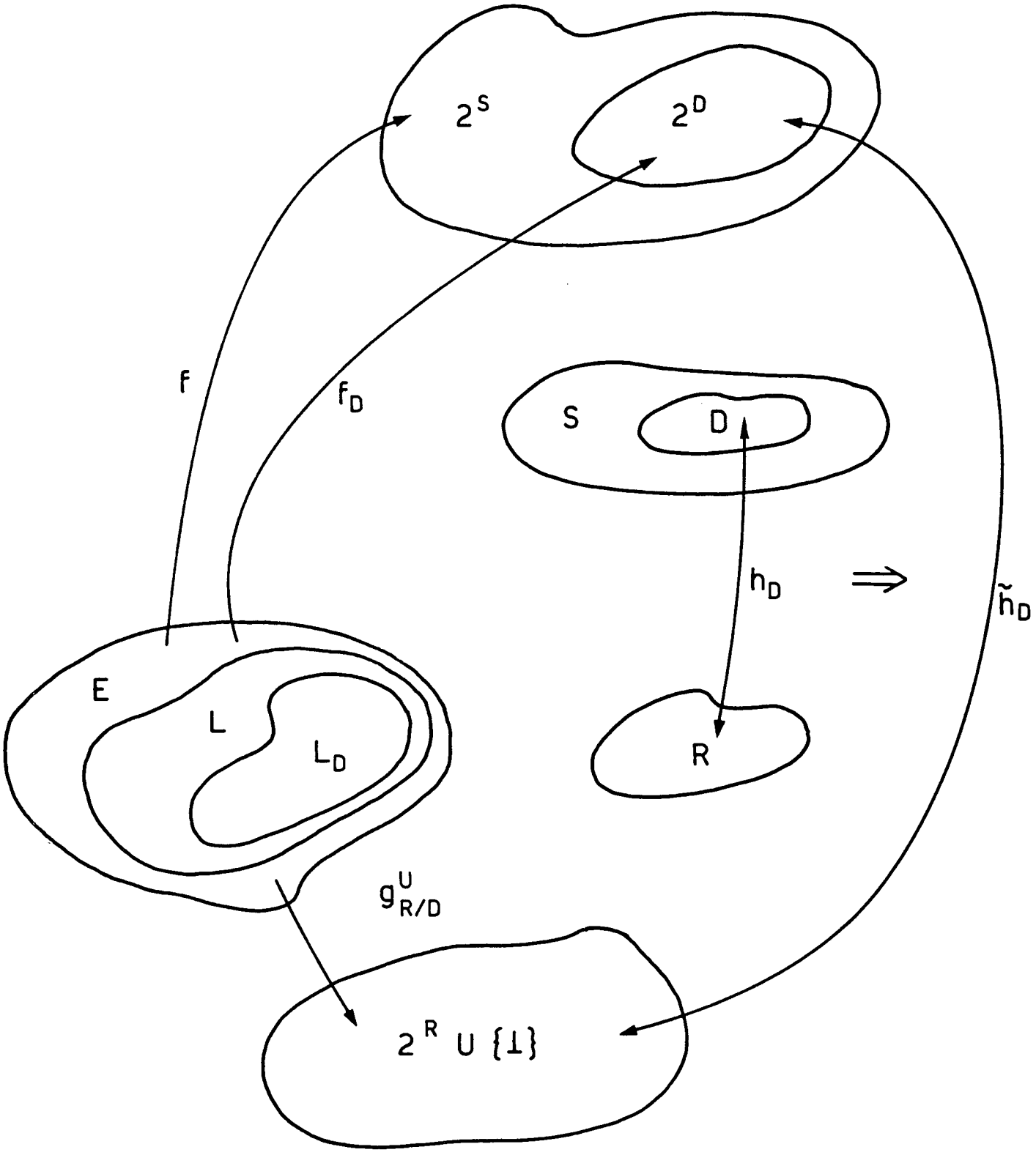

Figure 1. Relationships between the functions f, f_D, h_D, $\tilde{h}_D$, and $g^U_{R/D}$.

Both of these problems are discussed in the following sections (the former in sections 3 and 4, and the latter in section 5).

3. A Theoretical Framework

Before tackling the core topic of this section in a formal way, let us examine from an intuitive point of view the basic requirements for a measure of performance π to be reasonably acceptable. The primary goal is that it should allow consistent comparison among different systems, in the sense that if $\pi(U_1) = \pi(U_2)$ the behaviour of the two systems U_1 and U_2 should be sufficiently similar, and that if $\pi(U_1) > \pi(U_2)$, U_1 should perform better than U_2.

Furthermore, this comparison should be as fine and precise as possible, in such a way to capture all the essential features of the behaviour of a system U in a given domain.

Finally, comparison might be between two different systems, between two versions of the same system, between a system and a given set of issues, or between a system and an independent scale (Tennant 1980).

To capture the intuitive notion of performance according to the above requirements, at least two points of view seem worth considering. First, a measure of performance should give a numerical value for the "distance" between the two functions $g^U_{R/D}$ and $\bar{g}_D$, that is, the measure should allow us to formalize how near $g^U_{R/D}(e)$ approaches $\bar{g}_D(e)$ for any $e \epsilon E$, or, more explicitly, how well each expression $e \epsilon E$ is understood by the system U. Second, it should weight this notion of "distance" in such a way as to take into account the fact that, generally, it is not equally important to understand well any expression in E; for example, it could be reasonable to suppose that correct understanding of expressions in L_D is far more relevant than in $E - L_D$, or that correct understanding is more important for frequently used expressions than for unusual and rare ones.

According to the above remarks, an appropriate notion of *performance* π will depend on two basic parameters:

(i) the *shifting* μ

between $g^U_{R/D}(e)$ and $\bar{g}_D(e)$ for any $e \epsilon E$

(ii) the *importance* ρ

for any expression $e \epsilon E$ to be correctly understood.

Different choices of μ and ρ clearly provide different notions of performance, $\pi[\mu,\rho]$, that fit different needs for capturing particular classes of features in a natural language understanding system.

Let us now go further in defining an appropriate formal framework embedding the above ideas. In the following, we shall omit in f_D, $g^U_{R/D}$, and $\bar{g}_D$ the superscript U and the subscripts R/D and D, whenever this will not cause ambiguities.

Let R be a representation language for a domain $D \subseteq S$ a *shifting function* μ on R is a function

$$\mu:(2^R \cup \{\perp\}) \times 2^R \rightarrow [0,1],$$

such that:

- for each pair (r,r'), $\mu(r,r') = 0$ iff $r = r'$;
- there exists a pair (r,r') such that $\mu(r,r') = 1$.

From an intuitive point of view, $\mu(g(e),\bar{g}(e))$ represents the "difference" between the (set of) meaning(s) of e computed by a natural language understanding system U, which is expressed by g(e), and its correct (set of) meaning(s) $\bar{g}(e)$. Hence, the value $\mu(g(e),\bar{g}(e)) = 0$ denotes perfect understanding of e, while $\mu(g(e),\bar{g}(e)) = 1$ denotes the worst case of misunderstanding of e.

Given the set E of all expressions of a natural language of length less or equal than an appropriately fixed integer n, an *importance function* ρ on E is a function

$$\rho:E \rightarrow [0,1].$$

Intuitively, $\rho(e)$ represents the importance that the meaning of e is correctly understood by the system U. The value $\rho(e) = 0$ denotes that it is not at all important that e be understood correctly or incorrectly; values of $\rho(e)$ greater than 0 denote the greater importance for e to be understood correctly. Given a shifting function μ on R and an importance function ρ on E, a *performance measure* π for natural language understanding systems $U_{R/D}$ is the function

$$\pi[\mu,\rho]:\{U_{R/D}\} \rightarrow [0,1],$$

defined by:

$$\pi[\mu,\rho](U_{R/D}) = \frac{\sum_{e \epsilon E} \mu(g^U_{R/D}(e),\bar{g}_D(e)) \cdot \rho(e)}{\sum_{e \epsilon E} \rho(e)}.$$

Clearly, π ranges from the value 0, in the case where all expressions of E are correctly understood, to the value 1, in the case where all expressions are completely (that is, in the worst manner) misunderstood,

independently of the choice of ρ (of course, $\rho \equiv 0$ is not allowed, being meaningless).

$\pi[\mu,\rho]$ provides a very synthetic representation of the performance of U that can be useful in several cases of evaluation and comparison. A richer and more informed picture of the performance of a system U fully coherent with the above definitions can be obtained in the following way, for the cases where the ranges of μ and ρ are finite. For given shifting μ and importance ρ, let range(μ) = $\{\delta_1,\ldots,\delta_n\}$ and range(ρ) = $\{\omega_1,\ldots,\omega_m\}$. Then we pose:

$$E_{i,j} = \{e \mid \mu(g(e),\bar{g}(e)) = \delta_i \text{and} \rho(e) = \omega_j\},$$

for $i \in \{1,\ldots,n\}$ and $j \in \{1,\ldots,n\}$.

Clearly, $\bigcup E_{i,j} = E$ and all $E_{i,j}$ are pairwise disjoint. Therefore, $\{E_{i,j}\}$ is a partitioning of E.

Now let:

$$p_{i,j} = \frac{|E_{i,j}|}{|E|},$$

for $i \in \{1,\ldots,n\}$ and $j \in \{1,\ldots,m\}$. (We remember that E has been assumed to be finite, and hence so is $E_{i,j} \subseteq E$).

The $n \times m$ matrix $[p_{i,j}]$ is called the *μ-ρ-profile* of U and and provides a far more informed representation of the performance of U than the value $\pi[\mu,\rho]$. In fact, $[p_{i,j}]$ allows one to discover and analyse the specific features of the system, going beyond the global value $\pi[\mu,\rho]$.

The relation between $[p_{i,j}]$ and $\pi[\mu,\rho]$ is straightforward:

$$\pi[\mu,\rho] = \sum_{i=1}^{n} \sum_{j=1}^{m} p_{i,j} \cdot \delta_i \cdot \omega_j \cdot \frac{|E|}{\sum_{e \in E} \rho(e)}.$$

Note that $[p_{i,j}]$ depends on μ and ρ only through the partitioning $\{E_{i,j}\}$ they induce on E, but it is independent of the actual values of δ_i and ω_j.

Different choices of μ and ρ clearly provide different measures of performance that can be compared, in general, only on a qualitative and intuitive basis. Therefore, evaluating the performance of a system U requires first the definition of μ and ρ, and then the computation of $\pi[\mu,\rho]$. Clearly, the most critical of these two steps is, from a conceptual point of view, the first as it completely determines the "goodness" of the measure and its actual matching with desired intuitive requirements. The second is only difficult from the computational point of view since E is usually very large and, hence, it is not possible to evaluate the sum in the definition of $\pi[\mu,\rho]$ in a direct, exhaustive way.

In the next section we discuss in detail the problem of appropriately defining μ and ρ, while section 5 is devoted to the topic of actually computing $\pi[\mu,\rho]$.

4. Some Significant Choices of Shifting and Importance Parameters

Having discussed in the previous section an abstract theory of performance evaluation, we now deal with some implementations of it that may be of practical interest. Clearly, an implementation is obtained by assigning actual functions as values for the (functional) parameters μ and ρ in the definition of π. Different choices of μ and ρ will yield different models for performance evaluation and will allow one to analyse different features of the systems to be evaluated.

Since μ and ρ are fully independent parameters, we shall deal with each separately.

Let us begin with the shifting function μ; in order that only the effect of μ be relevant to π, we shall suppose throughout the following discussion that ρ has the constant value $\rho(e)=1$ for any $e \in E$.

The simplest case is that where μ may assume only two (boolean) values 0 and 1, denoting a correct and a wrong understanding, respectively. Such a boolean shifting function is denoted by μ_1 and formally defined by:

$$\mu_1(r',r'') = \begin{cases} 0 & \text{if } r' = r'' \\ 1 & \text{if } r' \neq r'' \end{cases}$$

for any pair $(r',r'') \in (2^R \cup \{\perp\}) \times 2^R$.

The intuitive meaning of μ_1, when used to evaluate a natural language understanding system U, is straightforward: $\pi[\mu_1,1](U) = x$ denotes the percentage of expressions of E that U is unable to understand correctly (clearly, $1-x$ is the percentage of expressions correctly understood by U).

The above definition of μ is very crude; in fact, systems with the same $\pi[\mu_1,1]$ can show a very different behaviour, and, furthermore, $\pi[\mu_1,1](U_1) > \pi[\mu_1,1](U_2)$ does not generally ensure that U_1 performs better than U_2.

A slight improvement can be obtained by splitting the case $r' \neq r''$ into two subcases that cover, when evaluating U, the following situations:

(i) U is unable to assign a meaning to an expression e (that is, it fails); hence, $g(e) = r' = \perp \neq r'' = \bar{g}(e)$

(ii) U assigns to an expression e a meaning that is not the correct one; hence, $g(e) = r' \neq r'' = \bar{g}(e)$, with $g(e) \neq \perp$.

It seems quite reasonable that generally case (i) is less serious than case (ii), so that we can propose a new definition of shifting μ_2:

$$\mu_2(r',r'') = \begin{cases} 0 & \text{if } r' = r'' \\ \delta & \text{if } r' = \perp \\ 1 & \text{if } r' \neq \perp \text{ and } r' \neq r'' \end{cases}$$

where $\delta \in (0,1)$.

Clearly, the choice of δ strongly affects the values of $\pi[\mu,1](U)$ and will depend on how much we want to distinguish between cases (i) and (ii) mentioned above.

Going further to propose more fitting definitions of μ, we may want to analyze in more detail the case $r' \neq \perp$ and $r' \neq r''$. Recalling that r' and r'' are sets of strings in R, we can distinguish the following cases:

(i) U assigns to an expression e the value ϕ (that is, no meaning), while it has a well-defined meaning;

(ii) U assigns to an expression e a proper nonempty subset of its meanings;

(iii) U assigns to an expression e all its correct meanings and, in addition, other incorrect ones;

(iv) U assigns to an expression e a proper nonempty subset of its meanings and, in addition, other incorrect ones;

(v) U assigns to an expression e a nonempty set of meanings that is fully different from the correct one.

Formally, we can define the shifting μ_3 that covers all such situations by:

$$\mu_3(r',r'') = \begin{cases} 0 & \text{if } r' = r'' \\ \delta_1 & \text{if } r' = \perp \\ \delta_2 & \text{if } r' = \phi \text{ and } r'' \neq \phi \\ \delta_3 & \text{if } r' \neq \phi \text{ and } r' \subset r'' \\ \delta_4 & \text{if } r' \supset r'' \text{ and } r'' \neq \phi \\ \delta_5 & \text{if } r' \cap r'' \neq \phi \text{ and } r' - r'' \neq \phi \\ & \quad \text{and } r'' - r' \neq \phi \\ 1 & \text{if } r' \neq \phi \text{ and } r' \cap r'' = \phi \end{cases}$$

where $\delta_i \in (0,1)$, for $i = 1, 2, 3, 4, 5$.

It could be reasonably assumed $\delta_1 < \delta_2 < \delta_3 < \delta_4 < \delta_5$, since the situations to which they are attached are generally considered as denoting increasing degrees of misunderstanding (note that μ_3 deals in great detail with the case of ambiguous understanding, where at least one of r' or r'' is not a singleton).

Along the line of reasoning shown in the above definitions, several other improvements are possible. For example, we can further refine the above case (v), $r' \neq \phi$ and $r' \cap r'' = \phi$, by taking into account the actual structure of the elements of r' and r''. R being a well-defined formal language, we can first define an appropriate notion of "distance" $\tilde{\mu}$ between elements of R, and then extend it to nonempty disjoint elements of 2^R.

This kind of refinement is particularly significant when both r' and r'' are singletons, that is, understanding is not ambiguous, as is often the case. Also, it generally allows far more meaningful definitions of shifting, thus further approaching the intuitive notion of "distance" as "degree of understanding".

Let us turn our attention now to the importance function ρ.

Also for this function, a first simple proposal can be a boolean definition: no importance at all is assigned to expressions in $E - L_D$ and the same (not null) importance to every expression in L_D. So we can define ρ_1 as:

$$\rho_1(e) = \begin{cases} 0 & \text{if } e \notin L_D \\ 1 & \text{if } e \in L_D \end{cases}$$

for each $e \in E$.

A refinement of ρ_1 can be obtained by analyzing the case $e \in L_D$ and taking into account the frequency of use of expressions in L_D. This will give more importance to the correct understanding of more frequently used expressions and less importance to that of rare or unusual ones. From the human point of view, it is obvious that texts with a greater frequency are used, and hence understood, by a larger number of people.

Therefore, it seems meaningful to consider a system that can understand quite well the relatively small number of the most common texts and fails on the most unusual ones, to be better than a system that understands a lot of very rare texts but often fails in understanding the most common ones.

Formally, we can define the frequency of expressions of e as a map $z : E \to [0,1]$, with the constraint that $\sum_{e \in E} z(e) = 1$. Then, we can define a new importance function ρ_2 such that:

$$\rho_2(e) = \begin{cases} z(e) & \text{if } e \in L_D \\ 0 & \text{otherwise} \end{cases}$$

The frequency function z(e) can be effectively determined by collecting, through an appropriate experimental activity, a meaningful bag of texts T, in which each $e \in E$ appears with a given integer multiplicity m(e), and then by computing

$$z(e) = \frac{m(e)}{\sum_{e \in E} m(e)}.$$

A totally different criterion that could be used to refine the definition of importance functions is structural complexity of the expressions of E (or of L_D).

A very crude notion of structural complexity is simply given by the length of an expression e. In this case, given a chain $0 = \ell_0 < \ell_1 < \ldots < \ell_{m-1}$ of m non-negative integers, we can partition E into m classes:

$$E_1 = \{e \mid \ell_0 < |e| \leq \ell_1\}$$
$$E_2 = \{e \mid \ell_1 < |e| \leq \ell_2\}$$
$$\vdots$$
$$E_m = \{e \mid \ell_{m-1} < |e|\}.$$

Then, a new importance function ρ_3 is defined by:

$$\rho_3(e) = \omega_i \text{ iff } e \in E_i,$$

where $\omega_i \in [0,1]$, for $i = 1,\ldots,m$.

It is worth noting that the length of a text is not independent of its frequency of use; we feel that in several application domains (such as, for example, man-machine interaction) short texts are much more frequent than long ones and that texts exceeding a given length are not used at all.

A more refined notion of structural complexity of an expression may be given by taking into account its syntactic structure, defined on the basis of an appropriate set of characteristic features – see, for example, the classification proposed in Tennant (1980). E can be partitioned into different and disjoint classes E_i, according to the set of syntactical features they match, and an importance function ρ_4 can be defined as above:

$$\rho_4(e) = \omega_i \text{ iff } e \in E_i,$$

where $\omega_i \in [0,1]$, for $i = 1,\ldots,m$.

Let us note that, contrary to the above illustrated relation between the length of a text and its frequency, it seems reasonable to consider syntactical complexity as fully independent of frequency; in fact, quite complex syntactical features (such as ellipsis, anaphora, broken text, etc.) are frequently found in several application domains.

Finally, a couple of other possible choices for assigning the importance function ρ are worth mentioning: one based on the notions of "information content" or "structural complexity" according to Kolmogorov (1965, 1968), and the other based on the concept of "semantic complexity" of an expression, which could be formally defined, for example, in the represented domain R. However, some more theoretical work on these notions is necessary before we can use them for our needs; hence we will not further develop these notions here.

5. Measuring Performance in Practice

In the preceding sections, some theoretical tools for measuring the performance of a natural language understanding system have been illustrated. At this point we have to put them to work: that is, we must discuss how the performance of a system can be actually evaluated and how the comparison between two different systems can be carried out.

We distinguish two steps in the process of performance evaluation:

(i) to assign the functions μ and ρ;

(ii) to compute $\pi[\mu,\rho]$.

Let us examine in detail each of the two points.

The choice of the shifting function μ depends only on the degree to which we want to refine the notion of error in understanding and on the varying importance we want to assign to each type of error. Hence it is often only a matter of subjective feeling choosing appropriate values for μ in order to analyse particular features of the system to be evaluated. Also, the definition of μ is strongly dependent on the representation language R for the domain D: the richer and more structured R is, the more refined and subtle are the possible definitions of μ.

On the contrary, however, the choice of the importance function ρ can generally be based on more objective arguments, once an appropriate ranking among the desired understanding capabilities of the system to be evaluated has been defined. For example, in the case where the frequency of texts is taken into account, an appropriate experimental activity can provide reliable statistical estimations for the frequency z(e) of each expression $e \in E$, thus allowing the effective computation of $\rho(e)$. (Problems connected with the choice of a meaningful sample to estimate z(e) – which could freely include millions of millions of expressions – are not dealt with here, since they are more related to statistics than to computational linguistics.)

Clearly, the choice of μ and ρ fully determines the numerical value of $\pi[\mu,\rho]$ (or of the matrix $[p_{i,j}]$) in correspondence to a given system U. How a change in μ or ρ can affect $\pi[\mu,\rho]$ is generally impossible to pre-

dict, since this strongly depends on the particular features of U. Therefore, evaluating a system with different choices of μ or ρ can indeed provide a clearer image of its performance. Although the comparison between different values of π obtained with different pairs (μ,ρ) is often only a matter of intuitive reasoning, an interesting particular case that can be conveniently dealt with formally is briefly sketched below.

A shifting function μ' is a *refinement* of a shifting function μ ($\mu' \sqsupseteq \mu$) iff:

- $\text{range}(\mu) = \{\delta_1,\ldots,\delta_n\}$ with $\delta_1<\delta_2<\ldots<\delta_n$;
- $\text{range}(\mu') = \{\delta'_1,\ldots,\delta'_{n'}\}$, with $\delta'_1< \delta'_2<\ldots< \delta'_{n'}$ and $n' \geq n$;
- the partitioning $\{E'_i\}$ of E induced by μ' is a refinement of the partitioning $\{E_i\}$ of E induced by μ;
- for each class $E_i = \bigcup_{t\in T} E'_{t'}$, where $T = \{t_1,\ldots,t_i\} \subseteq \{1,\ldots,n'\}$, with $t_1< t_2<\ldots< t_i$:

$$\delta_{i-1}<\delta'_{t_1}<\delta'_{t_2}<\ldots<\delta'_{t_i} = \delta_i.$$

In an analogous way we can define the *refinement* ρ' of an importance function ρ ($\rho' \sqsupseteq \rho$).

A pair (μ',ρ') is a *refinement* of a pair (μ,ρ) (we write $(\mu',\rho') \sqsupseteq (\mu,\rho)$) iff $\mu' \sqsupseteq \mu$ and $\rho' \sqsupseteq \rho$.

It is straightforward to prove that:

For any system U and any two pairs (μ,ρ) and (μ',ρ'), $(\mu',\rho') \sqsupseteq (\mu,\rho)$ implies $\pi[\mu',\rho'](U) \leq \pi[\mu,\rho](U)$.

For example, the shifting function μ_3 in section 4 refines μ_2, which in turn refines μ_1, that is, $\mu_1 \sqsubseteq \mu_3$. For the importance function ρ, on the other hand, not one of the functions ρ_1, ρ_2, ρ_3, ρ_4, in section 4 is a refinement of any other one.

It is worth noting that, when defining appropriate pairs (μ,ρ) to evaluate a system, there are basically two ways of reasoning for comparing different choices: the first one is to start from a first basic proposal and to proceed through successive refinements until the desired degree of precision and detail is reached; the second one consists in proposing functions corresponding to several different points of view and then integrating them together in a well-balanced synthesis. Generally, the first approach is appropriate for the definition of μ, while the second one can be utilized for the choice of ρ.

Let us turn now to the problem of computing $\pi[\mu,\rho]$, once μ and ρ have been assigned.

Obviously, it is unrealistic to compute the *exact* value of π by considering the behaviour of the system with respect to every expression $e\in E$. Hence, a sequence of test cases has to be considered (Gold 1967).

Figure 2 shows a model for experimental performance evaluation. A GENERATOR provides at each time instant i (i=1,2,...) an expression $e_i\in E$. Then, the system U to be evaluated computes the meaning $g(e_i)$, which is compared by μ with the correct meaning $\bar{g}(e_i)$ supplied by an EVALUATOR (a man supposed to be able to compute $\bar{g}$, that is, both f and $\tilde{n}$). Finally, the value $\rho(e_i)$ is computed, and the current value of

$$\pi_i = \frac{\sum_{j=1}^{i} \mu(g(e_j),\bar{g}(e_j)) \cdot \rho(e_j)}{\sum_{j=1}^{i} \rho(e_j)}$$

is determined.

The major problem with the computation of π is the design of the GENERATOR, that is, the choice of the sample of E to be used for the evaluation of the system U.

The mathematically simplest case is the one where a subset $B \subseteq E$ is randomly generated on the basis of a given probability distribution in E (for example, equiprobability); then,

$$\pi_B = \frac{\sum_{e\in B} \mu(g(e),\bar{g}(e)) \cdot \rho(e)}{\sum_{e\in B} \rho(e)}$$

is a random variable such that $E(\pi_B) = \pi$ for reasonable distributions, where $E(\pi_B)$ denotes the expectation of π_B. The value of $E(\pi_B)$ may be estimated by means of statistical techniques such as, for example, the maximum likelihood function. Here, we will not give a detailed account of such techniques. They can be easily found in classical works of statistics and sampling theory (Kobayashi 1978; Cox, Hinkley 1977; Mood, Graybill 1980), when needed.

A different technique would be that of fixing a confidence interval, and then establishing the number n of tests to be generated in order to obtain the value of $\pi(\mu,\rho)$ within the given confidence level, by means, for example, of χ^2 techniques.

In addition to these elementary statistical methods, more sophisticated sampling techniques can be used. This requires us first to choose a partitioning of E into meaningful classes, and then to define a sample stratified according to the considered partitioning. In this

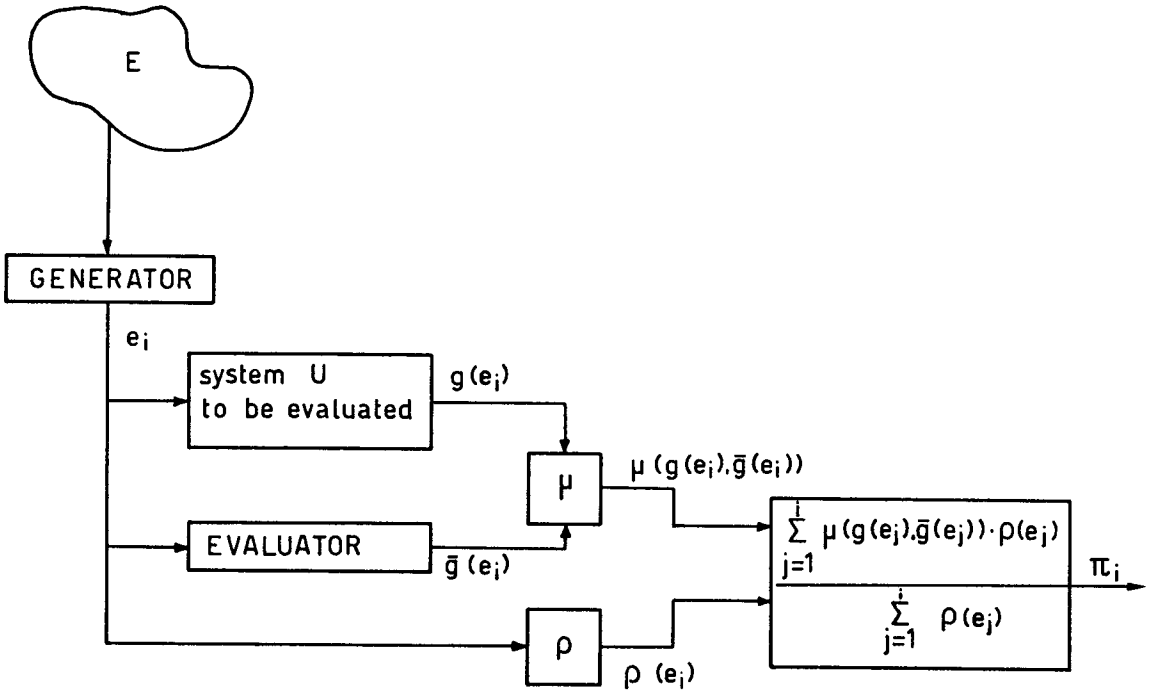

Figure 2. A model for experimental performance evaluation.

case, the GENERATOR might not work on a purely random basis.

All the above-mentioned techniques are independent of the choice of ρ, and do not take into account specific goals that could be assigned to performance evaluation (for example, syntactic capabilities, linguistic or conceptual competence, etc.). Such *general purpose* methods can sometimes provide a too much global and too less meaningful evaluation. Moreover, the sample to be used for the computation of π is generally very large and hard to collect.

Special purpose evaluation, centered on the analysis of some specific features of U, can often be more interesting and easier to implement. In this case, the specific *goal* of the measurement should be carefully taken into account in the definition of ρ, and both the goal and ρ should direct the choice of the appropriate sample of E to be used for the experimental computation of π. More precisely, an experimental (special purpose) evaluation session could be organized as follows:

1. precisely individuating the system U, the domain D, and the representation language R;
2. defining the goals of the evaluations;
3. deciding which samples of E to collect and how to collect them;
4. defining μ;
5. defining ρ (and how to compute it for the chosen samples);
6. computing π (and/or $[p_{i,j}]$).

Note that several μ and ρ could be generally considered for a careful experimentation. Moreover, steps 3, 4, and 5 might require, in critical cases, specific pre-experimentation and some refinement loops for appropriate tuning.

In the appendix, a limited case study experimentation is briefly discussed.

6. Discussion and Future Research Directions

In this paper we have presented a model for performance evaluation of natural language understanding systems. The main task of this model is that of providing a basis for a quantitative measure of how well a system can understand natural language, thus allowing an objective and experimental comparison of the performance of different systems.

Before discussing some open problems and illustrating the main lines of future research, let us briefly discuss some further features of our approach by comparing it to the classical work by Tennant (1979, 1980) and by Finin, Goodman, and Tennant (1979). Tennant's proposal is based on the three main concepts of habitability, completeness, and abstract analysis. This last point is not considered here, as explained in section 1 (see further in this section for its

possible relevance to future work); we therefore focus on the first two. From a naïve point of view, habitability is used to test whether or not the system does what it was designed to do; completeness is introduced to test whether or not the system meets users' requirements. More precisely, Tennant introduces the two notions of *coverage* and *completeness* to denote, respectively, the capabilities (both conceptual and linguistic) that the designer has put within a system, and (similarly to Woods, Kaplan, Nash-Webber 1972 though differing from Woods 1977) the degree to which the capabilities expected by a set of users can actually be found in the system coverage. Furthermore, *habitability* denotes (quite differently from Watt 1968) the degree to which a system can actually exhibit the capabilities that it was designed to have.

Our approach is based on a slightly different model and provides in some sense a refinement of the above concepts.

We denote by the term *competence* the capabilities that a system is actually able to show, while by the term *coverage* we refer, according to Tennant, to the theoretical capabilities that a system should have as a consequence of its design specifications.

More precisely, the conceptual coverage of a system $U_{R/D}$ is formalized in our model by the domain D, which represents, in fact, the range of concepts that are within the domain of discourse of a given application.

The linguistic coverage clearly includes L_D but, generally, is not limited to L_D since understanding a language in a given domain also implies the capability of recognizing that some expressions are not meaningful in that domain.

In general, for a given importance function ρ, we can assume that the linguistic coverage is defined by:

$$L'_D = \{e \mid e \epsilon E \text{ and } \rho(e) > \Delta\},$$

where $\Delta (0 \leq \Delta < 1)$ is a fixed bound.

The linguistic competence can then be defined as:

$$\tilde{L}'_D = \{e \mid e \epsilon L'_D \text{ and } g(e) = \bar{g}(e)\},$$

and the conceptual competence as:

$$\tilde{D} = \bigcup_{e \epsilon \tilde{L}'_D} f_D(e).$$

The above concepts are summarized in Figure 3.

Our definition of performance $\pi[\mu,\rho]$ tries to give a global idea of how well the competence of a system (without distinction between conceptual and linguistic aspects) approaches its coverage. This measure is quite similar to, and provides a refinement of, the concept of habitability, involving also to some extent the notion of completeness. In fact, both the choice of D as an adequate domain and the definition of ρ as a suitable importance function (and, therefore, of L'_D) implicitly refer to a set of users and then to completeness.

It is apparent that the proposal introduced in this paper demands further work, both theoretical and experimental, in order to have fully adequate tools for performance evaluation.

First of all, some of the concepts presented here have to be further discussed and expanded. For example, in the definition of π, we have normalized it with respect to ρ by setting:

$$\pi = \frac{\sum \mu \bullet \rho}{\sum \rho}.$$

A different choice could be:

$$\pi = \frac{\sum \mu \bullet \rho}{|E|},$$

where μ and ρ are given the same importance (in this case the value $\pi = 1$ would be reached only when all expressions of E are fully misunderstood, that is, $\mu \equiv 1$, *and* when it is important at the highest degree that each of them is correctly understood, that is, $\rho \equiv 1$). While we have preferred here the first definition, arguments could be given in favour of the second.

A second critical point is the definition of the μ–ρ-profile $[p_{i,j}]$. This could be further extended so as to provide a picture of several dimensions *(features,* for example: frequency, syntactic complexity, information content, etc.). Third, it is worthwhile considering and improving the notion of refinement: in fact, the present definition is not stable with respect to the choice of μ and ρ. That is, it could be that, given two systems U and U':

$$\pi[\mu,\rho](U) < \pi[\mu,\rho](U')$$

and, for some refinement (μ',ρ') of (μ,ρ):

$$\pi[\mu',\rho'](U) > \pi[\mu',\rho'](U'),$$

so that the refinement of the evaluation criteria may give an inversion of the first evaluation. A formal development of the three points mentioned above will

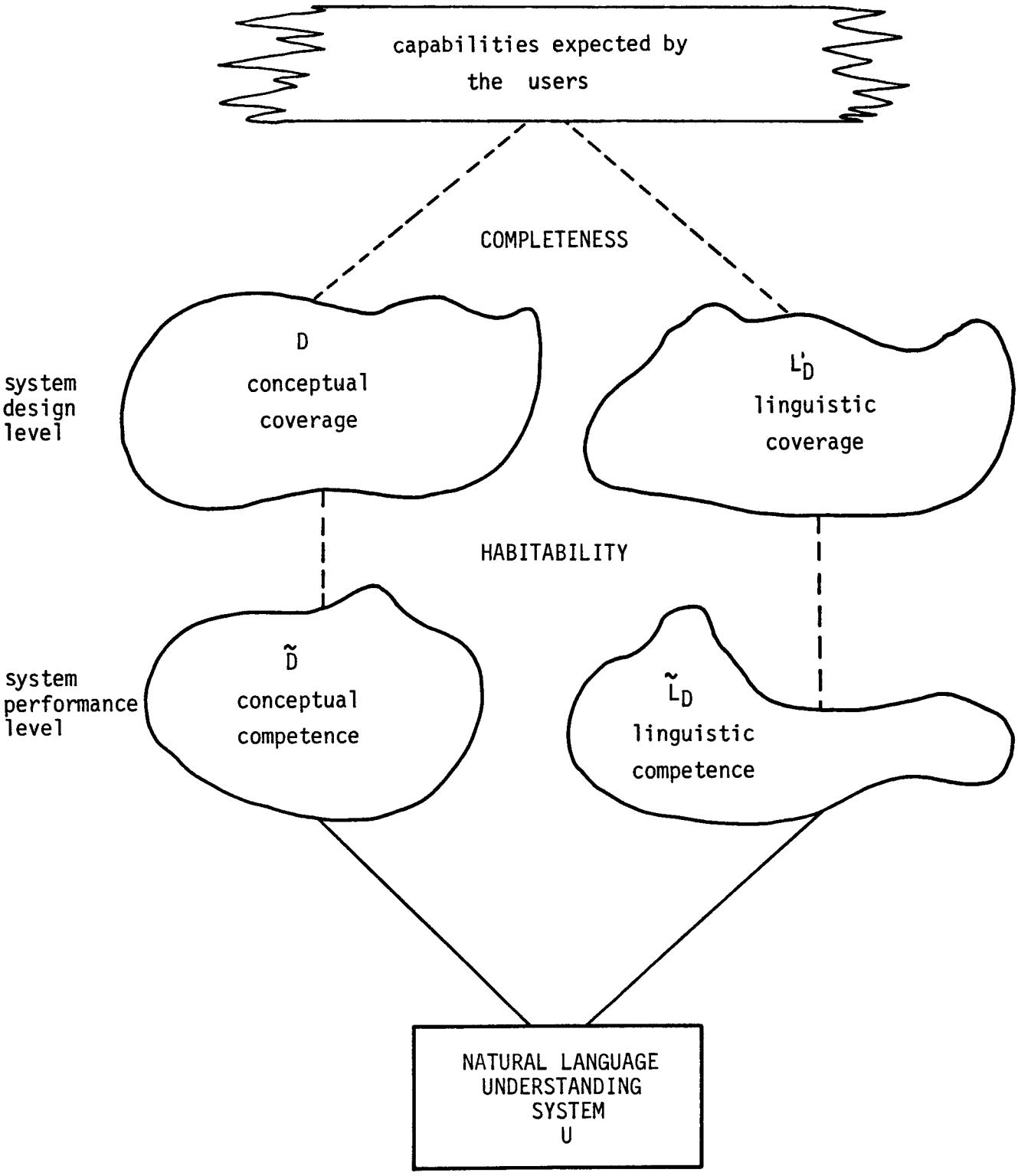

Figure 3. Coverage and competence of a natural language understanding system.

be part of a future paper.

For what concerns the main directions in the development of the current research activity, we mention:

- experimentation with the model proposed in the evaluation of large systems;
- development of appropriate sampling techniques for the experimental evaluation of π;
- experimentation with several different choices of μ and ρ;
- design of techniques for special purpose evaluation (choice of the goal, definition of μ and ρ, sampling, etc.);
- analysis of the adequacy of the notion of μ-ρ-profile for representing all interesting details of the performance of a system.

Beyond these issues we also point out two more ambitious and promising problems; they will be faced in future work. The approach to performance evaluation presented in this paper has two major limitations: first, it is only concerned with input-output behaviour and does not take into account the internal model on which a system is based; second, it does not deal with the efficiency of the natural language understanding process. As far as the former topic is concerned, it is clear that, except in the case where commercial applications are considered, one is primarily interested in models rather than in particular implementations. It is far more significant that a model, a knowledge representation method, and a parsing algorithm have been designed to build natural language understanding systems rather than that a specific system has been constructed in a particular domain for a particular use. Tennant (1980) (see also Woods 1977) proposes a method, called *abstract analysis,* to organize in an informal but disciplined way the evaluation, through taxonomies of conceptual, linguistic, and implementational issues, of the internal behaviour of a natural language system (including analysis of failure causes, domain dependent features, knowledge base completeness and closure, algorithm deficiencies, extensibility, etc.). A very demanding research issue that could substantially contribute to the development of the research on natural language processing is the definition of more formal methods that, starting from the above proposal, allow a "deep" evaluation and comparison of systems on the basis of their internal structure and mode of operation, opposed to the "surface" measure of their input-output behaviour, as considered in the present paper.

Concerning the latter topic, efficiency, two aspects seem worth considering: the experimental measure of the efficiency of a specific system in understanding natural language that could appropriately complete the concept of performance defined in the present work; and the theoretical evaluation of the complexity of the general model underlying the construction of a particular system, which could possibly complete the notion of "deep" evaluation mentioned above.

Acknowledgements

We are grateful to the anonymous referees for their useful criticism and suggestions.

We would also like to acknowledge the appreciated support provided by CSELT Laboratories (Torino, Italy) with the experimentation of the PARNAX system.

References

Comino, R.; Gemello, R.; Guida, G.; Rullent, C.; Sisto, L.; and Somalvico, M. 1983 Understanding Natural Language Through Parallel Processing of Syntactic and Semantic Knowledge: An Application to Data Base Query. In *Proc. 8th Int. Joint Conference on Artificial Intelligence.* Karlsruhe, West Germany: 663-667.

Cox, D.R. and Hinkley, D.V. 1974 *Theoretical Statistics.* Chapman and Hall, London.

Finin, T.; Goodman, B.; and Tennant, H. 1979 JETS: Achieving Completeness Through Coverage and Closure. In *Proc. 6th Int. Joint Conference on Artificial Intelligence.* Tokyo, Japan: 275-281.

Gold, E.M. 1967 Language Identification in the Limit. *Information and Control* 10: 447-474.

Kaplan, J. 1982 Special Section: Natural Language Processing. *ACM SIGART Newsletter* 79: 27-109 and 80: 59-61.

Kolmogorov, A.N. 1965 Three Approaches to the Concept of "The Amount of Information". *Probl. of Information Transmission* 1(1): 3-11.

Mood, R.S. and Graybill, H.J. 1980 *Introduction to Statistics.* McGraw-Hill, Englewood Cliffs, New Jersey.

Tennant, H. 1979 Experience with the Evaluation of Natural Language Question Answerers. In *Proc. 6th Int. Joint Conference on Artificial Intelligence.* Tokyo, Japan: 874-876.

Tennant, H. 1980 Evaluation of Natural Language Processes. Report T-103. Coordinated Science Laboratory, University of Illinois, Urbana, Illinois.

Waltz, D. 1977 Natural Language Interfaces. *ACM SIGART Newsletter* 61: 16-64.

Watt, W.C. 1968 Habitability. *American Documentation* 338-351.

Woods, W.A. 1977 A Personal View of Natural Language Understanding. *ACM SIGART Newsletter* 61: 17-20.

Woods, W.A.; Kaplan, R.M.; and Nash-Webber, B. 1972 The Lunar Sciences Natural Language Information System: Final Report. Report 2378. Bolt Beranek and Newman, Cambridge, Massachusetts.

Appendix

In this appendix we present a limited case study experimentation with the model proposed, which should help in concretely conveying the ideas on how an evaluation session could be carried out in practice. Wider experiments will be the subject of a future paper.

For this limited experimentation we have chosen the PARNAX system (Comino et al. 1983): a natural language interface for querying in Italian ADABAS data bases. The toy data base utilized concerns the employees of a company, and contains just the EMPLOYEE file with the following record structure:

NAME
DATE-OF-BIRTH
PLACE-OF-BIRTH
PLACE-OF-RESIDENCE
HIRING-DATE
DEPARTMENT
JOB-LEVEL
DEGREE

The PARNAX system, that is, the natural language understanding system U to be evaluated, maps natural language queries (more generally, query dialogues) into expressions of the formal query language used to access the ADABAS data base, namely the NATURAL language.

Owing to the very simple data base chosen, the domain D is reasonably limited, and so are L_D (the set of all possible queries in Italian to the EMPLOYEE file) and R (the set of all possible NATURAL queries).

We consider two goals for this experiment: namely, evaluating some aspects of the conceptual competence and of the linguistic competence.

According to these goals, two samples of queries have been collected from $L'_D \supseteq L_D$ (recall that the linguistic coverage $L'_D \subseteq E$ should generally be larger than L_D – see section 6):

A: A sample of casual queries to the data base. Nine hundred fifty queries have been collected from 90 people chosen from several different classes of possible users of the data base.

B: A sample of linguistic variations for expressing a specific request. The sentence to be rephrased has been chosen of medium-level complexity with respect to the sample data base so as to allow meaningful linguistic variations to be formed. The query utilized is: "Tell me the birth-date of all employees who have a master degree in mathematics". Five hundred queries have been collected from 35 people.

The sample A was then slightly preprocessed before being used for the evaluation. A new sample A' was obtained from A through the following operations:

(i) eliminating all queries expressing the same request (not having the same answer!), except one;

(ii) eliminating all queries differing only for the values of the attributes (for example, "Tell me the birth-date of John" and "Tell me the birth-date of Robert"), except one.

Note that this preprocessing constitutes a kind of very naïve stratification, consisting in the choice of only some elements as representative for a class of cases. A' contains about 800 queries.

The shifting function μ has been chosen to be the boolean function:

$$\mu(g(e),\bar{g}(e)) = \begin{cases} 0 & \text{if } g(e) = \bar{g}(e) \\ 1 & \text{if } g(e) \neq \bar{g}(e) \end{cases}$$

(see section 4, μ_1) for the analysis both of A' and B.

Several different choices have been taken for the importance function ρ. For what concerns the sample A', two functions have been considered:

$- \rho_1 \equiv 1$

$$- \rho_2(e) = \begin{cases} 0 & \text{if } e \notin L_D \\ 1 & \text{if } e \in L_D \end{cases}$$

(see section 4, ρ_2).

For the sample B the most natural choice for ρ seemed to be the frequency of queries. Unfortunately, only very few (3!) queries in sample B turned out to be repeated (one repetition for two queries, two repetitions for another query; note that this result is highly surprising even if the set from which B has been extracted is enormously large). Therefore, frequency was abandoned. Two other criteria were considered: namely length and structural features.

For what concerns the former, the length (number of words) of each sentence was computed first and Table 1 obtained. The length interval [5,26] was then partitioned into three parts:

I_1 = [12,16], into which about 70% of the sentences of B fall;

I_2 = [7,11]U[17,18], includes about 25% of the expressions of B;

I_3 = [5,6]U[19,26], to which less than 5% of the sentences of B belong.

The following importance function ρ_3 has been defined:

$$\rho_3(e) = \begin{cases} 0.70 & \text{if } e \in I_1 \\ 0.25 & \text{if } e \in I_2 \\ 0.05 & \text{if } e \in I_3 \end{cases}$$

Here, the importance of recognizing an expression is assumed to be proportional to the "weight" (cardinality) of the length class to which it belongs. For what concerns the latter criterion, we first defined a taxonomy of linguistic elements suitable for analysing the structural features of the sentences of Sample B. The following attributes were considered (Tennant 1980):

D declarative structure
G interrogative structure
E imperative structure
T telegraphic sentences
V cleft and discontinuous sentences, parenthetic clauses, inversions
M multiple sentences
R relative clauses
I interrogative clauses
N non-finite clauses (-ing, -ed participle)
P prepositional phrases
Q quantifiers, predeterminers
S possessive and demonstrative clauses, personal pronouns

The sentences of B were then classified according to the above taxonomy by assigning to each of them all the relevant attributes. Fifty-two classes were obtained:

44 containing 1 to 12 sentences (total 315)
5 containing 13 to 24 sentences (total 92)
3 containing 25 to 36 sentences (total 93)

LENGTH	NUMBER OF SENTENCES
5	2
6	7
7	11
8	17
9	13
10	25
11	40
12	53
13	130
14	62
15	45
16	50
17	24
18	12
19	5
20	2
21	1
26	1

Table 1.

This result suggested restricting the analysis to a smaller number of classes to be obtained through a less refined taxonomy of linguistic elements. The following attributes, which characterize the most crude features of the sentence structure, were chosen: D, G, E, T, V, M. Ten classes have now been obtained as shown in Table 2 (each class is denoted by the string of attributes that characterizes the structure of the sentences belonging to it).

STRUCTURE	NUMBER OF SENTENCES
T	22
D	91
DV	6
DM	27
G	79
GM	5
E	174
EV	44
EM	33
EVM	19

Table 2.

The importance function ρ_4 has therefore been defined to be exactly the frequency of the class to which every expression belongs, that is:

$$\rho_4(e) = \begin{cases} .044 & \text{if } e \epsilon T \\ .182 & \text{if } e \epsilon D \\ .012 & \text{if } e \epsilon DV \\ .054 & \text{if } e \epsilon DM \\ .158 & \text{if } e \epsilon G \\ .010 & \text{if } e \epsilon GM \\ .348 & \text{if } e \epsilon E \\ .088 & \text{if } e \epsilon EV \\ .066 & \text{if } e \epsilon EM \\ .038 & \text{if } e \epsilon EVM \end{cases}$$

Using the samples A' and B and the μ and ρ functions defined above, the performance of the PARNAX system was evaluated. The following results were obtained:

Case 1: sample A' with μ and ρ_1

$$\pi_1 = \frac{472}{800} = 0.59 \qquad [p_{i,j}] = \begin{bmatrix} 0.41 \\ 0.59 \end{bmatrix}$$

Case 2: sample A' with μ and ρ_2

$$\pi_2 = \frac{371}{701} = 0.53 \qquad [p_{i,j}] = \begin{bmatrix} 0.00 & 0.41 \\ 0.12 & 0.47 \end{bmatrix}$$

Case 3: sample B with μ and ρ_3

$$\pi_3 = \frac{81{,}4}{274{,}4} = 0.30 \qquad [p_{i,j}] = \begin{bmatrix} 0.49 & 0.19 & 0.01 \\ 0.19 & 0.09 & 0.03 \end{bmatrix}$$

Case 4: sample B with μ and ρ_4

$$\pi_4 = \frac{21.778}{98.916} = 0.22 \qquad [p_{i,j}] = \begin{bmatrix} 0.03 & 0.15 & 0.00 & 0.01 & 0.11 & 0.00 & 0.29 & 0.07 & 0.02 & 0.01 \\ 0.01 & 0.04 & 0.01 & 0.04 & 0.05 & 0.01 & 0.06 & 0.02 & 0.04 & 0.03 \end{bmatrix}$$

A few comments on the above results can be added. A global analysis of π shows that the linguistic capabilities of PARNAX are generally higher than the conceptual ones ($\pi_1,\pi_2 >> \pi_3,\pi_4$). In particular, the value of π_4, which relies on a fine analysis of syntactic features, seems very good.

Examining further the μ-ρ-profiles obtained (except case 1, which is not meaningful), several interesting details of the system performance may be pointed out. Cases 2 and 3 show that the system performs better in correspondence to sentences with higher importance, and, hence, it is reasonably well tailored. On the contrary, it generally lacks robustness. Finally, case 4 (especially when the analysis with 52 classes, not reported here, is considered) provides to the system designer a lot of useful suggestions for corrections and improvements.