

Precisiated Natural Language (PNL)

Lotfi A. Zadeh

- This article is a sequel to an article titled “A New Direction in AI—Toward a Computational Theory of Perceptions,” which appeared in the Spring 2001 issue of *AI Magazine* (volume 22, No. 1, 73–84). The concept of precisiated natural language (PNL) was briefly introduced in that article, and PNL was employed as a basis for computation with perceptions. In what follows, the conceptual structure of PNL is described in greater detail, and PNL’s role in knowledge representation, deduction, and concept definition is outlined and illustrated by examples. What should be understood is that PNL is in its initial stages of development and that the exposition that follows is an outline of the basic ideas that underlie PNL rather than a definitive theory.

A natural language is basically a system for describing perceptions. Perceptions, such as perceptions of distance, height, weight, color, temperature, similarity, likelihood, relevance, and most other attributes of physical and mental objects are intrinsically imprecise, reflecting the bounded ability of sensory organs, and ultimately the brain, to resolve detail and store information. In this perspective, the imprecision of natural languages is a direct consequence of the imprecision of perceptions (Zadeh 1999, 2000).

How can a natural language be precisiated—precisiated in the sense of making it possible to treat propositions drawn from a natural language as objects of computation? This is what PNL attempts to do.

In PNL, precision is accomplished through translation into what is termed a *precision language*. In the case of PNL, the precision language is the generalized-constraint language (GCL), a language whose elements are so-called generalized constraints and their combinations. What distinguishes GCL from languages such as Prolog, LISP, SQL, and, more generally, languages associated with various logical systems, for example, predi-

cate logic, modal logic, and so on, is its much higher expressive power.

The conceptual structure of PNL mirrors two fundamental facets of human cognition: (a) partiality and (b) granularity (Zadeh 1997). Partiality relates to the fact that most human concepts are not bivalent, that is, are a matter of degree. Thus, we have partial understanding, partial truth, partial possibility, partial certainty, partial similarity, and partial relevance, to cite a few examples. Similarly, granularity and granulation relate to clumping of values of attributes, forming granules with words as labels, for example, young, middle-aged, and old as labels of granules of age.

Existing approaches to natural language processing are based on bivalent logic—a logic in which shading of truth is not allowed. PNL abandons bivalence. By so doing, PNL frees itself from limitations imposed by bivalence and categoricity, and opens the door to new approaches for dealing with long-standing problems in AI and related fields (Novak 1991).

At this juncture, PNL is in its initial stages of development. As it matures, PNL is likely to find a variety of applications, especially in the realms of world knowledge representation, concept definition, deduction, decision, search, and question answering.

Natural languages (NLs) have occupied, and continue to occupy, a position of centrality in AI. Over the years, impressive advances have been made in our understanding of how natural languages can be dealt with on processing, logical, and computational levels. A huge literature is in existence. Among the important contributions that relate to the ideas described in this article are those of

Biermann and Ballard (1980), Klein (1980), Barwise and Cooper (1981), Sowa (1991, 1999), McAllester and Givan (1992), Macias and Pullman (1995), Mani and Maybury (1999), Allan (2001), Fuchs and Schwertelm (2003), and Sukkarieh (2003).

When a language such as precisiated natural language (PNL) is introduced, a question that arises at the outset is: What can PNL do that cannot be done through the use of existing approaches? A simple and yet important example relates to the basic role of quantifiers such as *all*, *some*, *most*, *many*, and *few* in human cognition and natural languages.

In classical, bivalent logic the principal quantifiers are all and some. However, there is a literature on so-called generalized quantifiers exemplified by *most*, *many*, and *few* (Peterson 1979, Barwise and Cooper 1981). In this literature, such quantifiers are treated axiomatically, and logical rules are employed for deduction.

By contrast, in PNL quantifiers such as *many*, *most*, *few*, *about 5*, *close to 7*, *much larger than 10*, and so on are treated as fuzzy numbers and are manipulated through the use of fuzzy arithmetic (Zadeh 1983; Kaufmann and Gupta 1985; Hajek 1998). For the most part, inference is computational rather than logical. Following are a few simple examples.

First, let us consider the Brian example (Zadeh 1983):

Brian is much taller than most of his close friends. How tall is Brian?

At first glance it may appear that such questions are unreasonable. How can one say something about Brian's height if all that is known is that he is much taller than most of his close friends? Basically, what PNL provides is a system for precisiation of propositions expressed in a natural language through translation into the generalized-constraint language (GCL). Upon translation, the generalized constraints (GCs) are propagated through the use of rules governing generalized-constraint propagation, inducing a generalized constraint on the answer to the question. More specifically, in the Brian example, the answer is a generalized constraint on the height of Brian.

Now let us look at the balls-in-box problem:

A box contains balls of various sizes and weights. The premises are:

Most are large.

Many large balls are heavy.

What fraction of balls are large and heavy?

The PNL answer is: *most* \times *many*, where *most* and *many* are fuzzy numbers defined through their membership functions, and *most* \times *many* is their product in fuzzy arith-

metic (Kaufmann and Gupta 1985). This answer is a consequence of the general rule

$$\begin{array}{l} Q_1 \text{ As are Bs} \\ Q_2 \text{ (A and Bs) are Cs} \\ \hline (Q_1 \times Q_2) \text{ As are (B and C)s} \end{array}$$

Another simple example is the tall Swedes problem (version 1).

Swedes who are more than twenty years old range in height from 140 centimeters to 220 centimeters. Most are tall. What is the average height of Swedes over twenty?

A less simple version of the problem (version 2) is the following (*a** denotes "approximately *a*").

Swedes over twenty range in height from 140 centimeters to 220 centimeters. Over 70* percent are taller than 170* centimeters; less than 10* percent are shorter than 150* centimeters, and less than 15 percent are taller than 200* centimeters. What is the average height of Swedes over twenty?

A PNL-based answer is given in the sidebar.

There is a basic reason generalized quantifiers do not have an ability to deal with problems of this kind. The reason is that in the theory of generalized quantifiers there is no concept of the count of elements in a fuzzy set. How do you count the number of tall Swedes if tallness is a matter of degree? More generally, how do you define the probability measure of a fuzzy event (Zadeh 1968)?

What should be stressed is that the existing approaches and PNL are complementary rather than competitive. Thus, PNL is not intended to be used in applications such as text processing, summarization, syntactic analysis, discourse analysis, and related fields. The primary function of PNL is to provide a computational framework for precisiation of meaning rather than to serve as a means of meaning understanding and meaning representation. By its nature, PNL is maximally effective when the number of precisiated propositions is small rather than large and when the chains of reasoning are short rather than long. The following is intended to serve as a backdrop.

It is a deep-seated tradition in science to view the use of natural languages in scientific theories as a manifestation of mathematical immaturity. The rationale for this tradition is that natural languages are lacking in precision. However, what is not recognized to the extent that it should be is that adherence to this tradition carries a steep price. In particular, a direct consequence is that existing scientific theories do not have the capability to operate on perception-based information—information exemplified by "Most Swedes are tall," "Usually Robert returns from work at about 6 PM,"

What should be stressed is that the existing approaches and PNL are complementary rather than competitive.

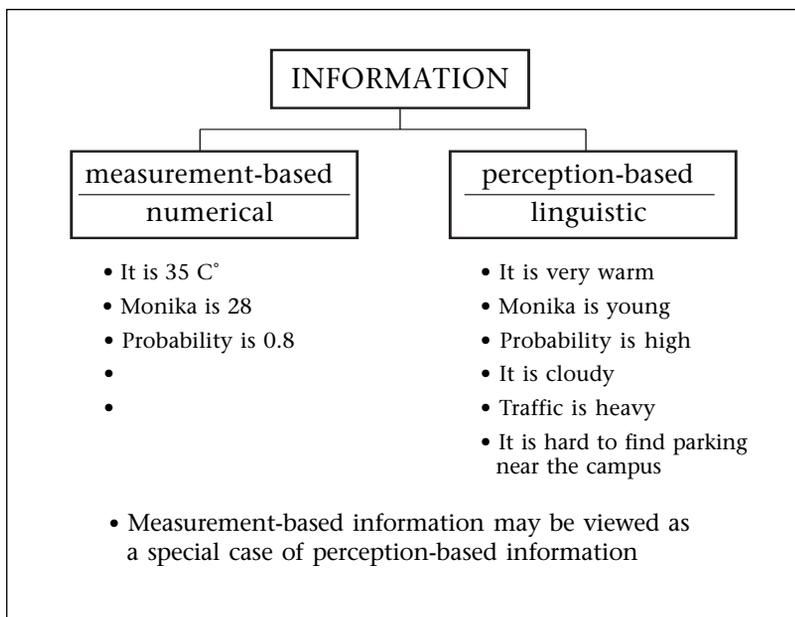


Figure 1. Modalities of Measurement-Based and Perception-Based Information.

“There is a strong correlation between diet and longevity,” and “It is very unlikely that there will be a significant increase in the price of oil in the near future” (figure 1).

Such information is usually described in a natural language and is intrinsically imprecise, reflecting a fundamental limitation on the cognitive ability of humans to resolve detail and store information. Due to their imprecision, perceptions do not lend themselves to meaning representation and inference through the use of methods based on bivalent logic. To illustrate the point, consider the following simple examples.

The balls-in-box example:

A box contains balls of various sizes. My perceptions of the contents of the box are:

- There are about twenty balls.
- Most are large.
- There are several times as many large balls as small balls.

The question is: What is the number of small balls?

The Robert example (a):

My perception is:

- Usually Robert returns from work at about 6 PM.

The question is: What is the probability that Robert is home at about 6:15 PM?

The Robert example (b):

- Most tall men wear large-sized shoes.
- Robert is tall.
- What is the probability that Robert wears large-sized shoes?

An immediate problem that arises is that of meaning precisiation. How can the meaning of the perception “There are several times as many large balls as small balls” or “Usually Robert returns from work at about 6 PM” be defined in a way that lends itself to computation and deduction? Furthermore, it is plausible, on intuitive grounds, that “Most Swedes are tall” conveys some information about the average height of Swedes. But what is the nature of this information, and what is its measure? Existing bivalent-logic-based methods of natural language processing provide no answers to such questions.

The incapability of existing methods to deal with perceptions is a direct consequence of the fact that the methods are based on bivalent logic—a logic that is intolerant of imprecision and partial truth. The existing methods are categorical in the sense that a proposition, *p*, in a natural language, NL, is either true or not true, with no shades of truth allowed. Similarly, *p* is either grammatical or ungrammatical, either ambiguous or unambiguous, either meaningful or not meaningful, either relevant or not relevant, and so on. Clearly, categoricity is in fundamental conflict with reality—a reality in which partiality is the norm rather than an exception. But, what is much more important is that bivalence is a major obstacle to the solution of such basic AI problems as common-sense reasoning and knowledge representation (McCarthy 1990, Davis 1990, Sowa 1991, 1999, Yager 1991, Sun 1994, Dubois and Prade 1996), nonstereotypical summarization (Mani and Mayberry 1999), unrestricted question answering, (Lehnert 1978), and natural language computation (Biermann and Ballard 1980).

PNL abandons bivalence. Thus, in PNL everything is, or is allowed to be, a matter of degree. It is somewhat paradoxical, and yet is true, that precisiation of a natural language cannot be achieved within the conceptual structure of bivalent logic.

By abandoning bivalence, PNL opens the door to a major revision of concepts and techniques for dealing with knowledge representation, concept definition, deduction, and question answering. A concept that plays a key role in this revision is that of a generalized constraint (Zadeh 1986). The basic ideas underlying this concept are discussed in the following section. It should be stressed that what follows is an outline rather than a detailed exposition.

The Concepts of Generalized Constraint and Generalized-Constraint Language

A conventional, hard constraint on a variable, X , is basically an inelastic restriction on the values that X can take. The problem is that in most realistic settings—and especially in the case of natural languages—constraints have some degree of elasticity or softness. For example, in the case of a sign in a hotel saying “Checkout time is 1 PM,” it is understood that 1 PM is not a hard constraint on checkout time. The same applies to “Speed limit is 65 miles per hour” and “Monika is young.” Furthermore, there are many different ways, call them modalities, in which a soft constraint restricts the values that a variable can take. These considerations suggest the following expression as the definition of generalized constraint (figure 2):

$$X \text{ isr } R,$$

where X is the constrained variable; R is the constraining relation; and r is a discrete-valued modal variable whose values identify the modality of the constraint (Zadeh 1999.) The constrained variable may be an n -ary variable, $X = (X_1, \dots, X_n)$; a conditional variable, $X|Y$; a structured variable, as in $\text{Location}(\text{Residence}(X))$; or a function of another variable, as in $f(X)$. The principal modalities are possibilistic ($r = \text{blank}$), probabilistic ($r = p$), veristic ($r = v$), usuality ($r = u$), random set ($r = rs$), fuzzy graph ($r = fg$), bimodal ($r = bm$), and Pawlak set ($r = ps$). More specifically, in a possibilistic constraint,

$$X \text{ is } R,$$

R is a fuzzy set that plays the role of the possibility distribution of X . Thus, if $U = \{u\}$ is the universe of discourse in which X takes its values, then R is a fuzzy subset of U and the grade of membership of u in R , $\mu_R(u)$, is the possibility that $X = u$:

$$\mu_R(u) = \text{Poss}\{X = u\}.$$

For example, the proposition p : X is a small number is a possibilistic constraint in which “small number” may be represented as, say, a trapezoidal fuzzy number (figure 3), that represents the possibility distribution of X . In general, the meaning of “small number” is context-dependent.

In a probabilistic constraint:

$$X \text{ isp } R,$$

X is a random variable and R is its probability distribution. For example,

$$X \text{ isp } N(m, \sigma^2)$$

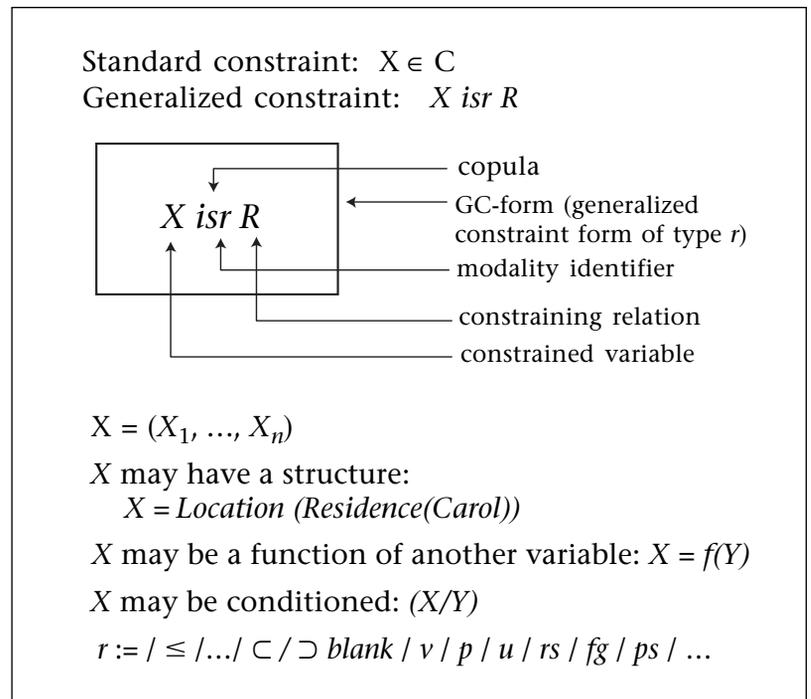


Figure 2. Generalized Constraint.

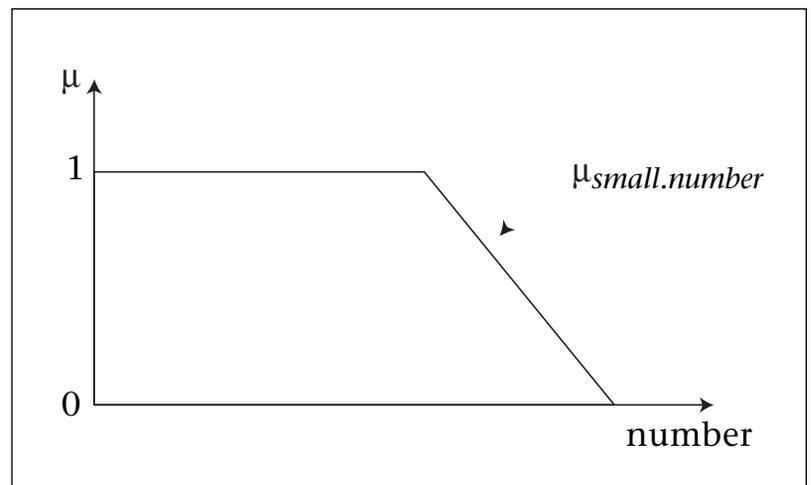


Figure 3. Trapezoidal Membership Function of “Small Number.” (“Small number” is context dependent.)

means that X is a normally distributed random variable with mean m and variance σ^2 .

In a veristic constraint, R is a fuzzy set that plays the role of the verity (truth) distribution of X . For example, the proposition “Alan is half German, a quarter French, and a quarter Italian,” would be represented as the fuzzy set

$$\text{Ethnicity}(\text{Alan}) \text{ isv } (0.5 \mid \text{German} + 0.25 \mid \text{French} + 0.25 \mid \text{Italian}),$$

in which $\text{Ethnicity}(\text{Alan})$ plays the role of the constrained variable; $0.5 \mid \text{German}$ means that

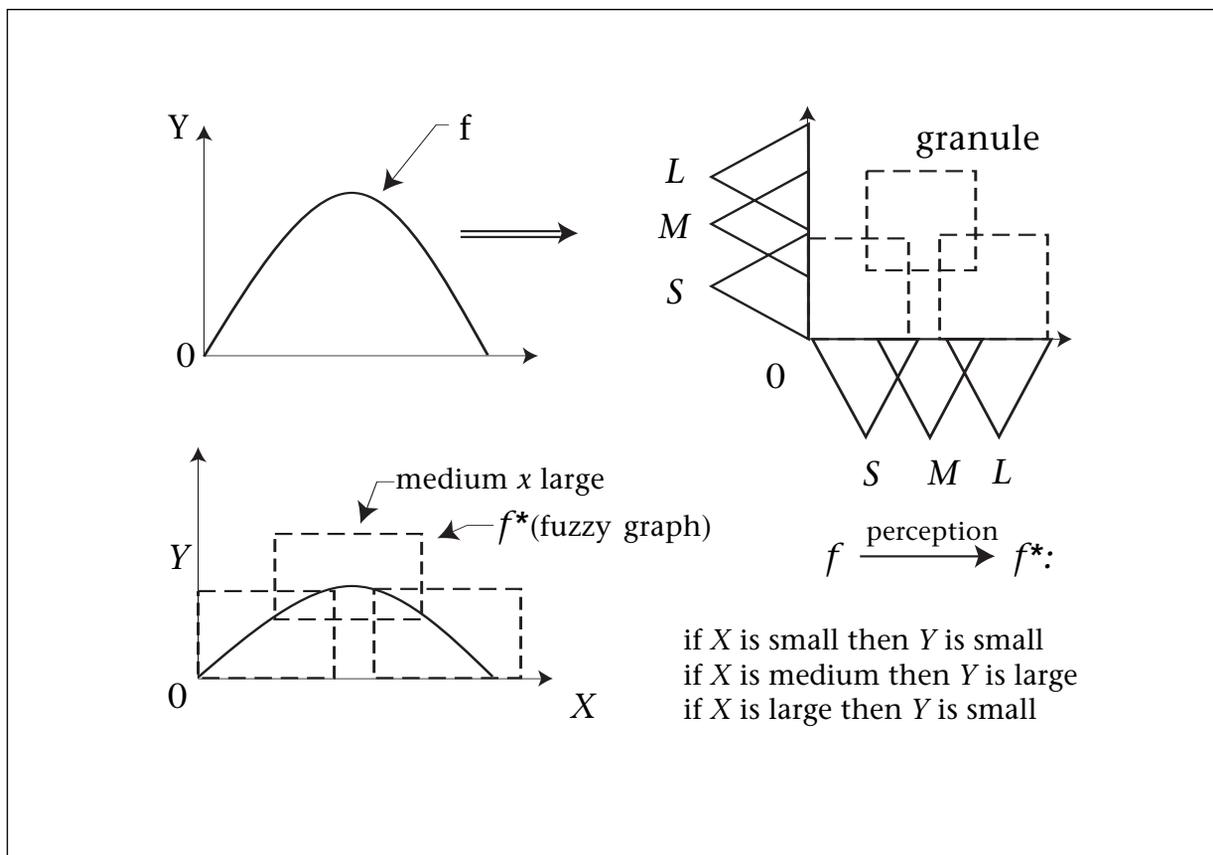


Figure 4. Fuzzy Graph of a Function.

the verity (truth) value of “Alan is German” is 0.5; and + plays the role of a separator.

In a usuality constraint, X is a random variable, and R plays the role of the usual value of X . For example, X is small means that usually X is small. Usuality constraints play a particularly important role in commonsense knowledge representation and perception-based reasoning.

In a random set constraint, X is a fuzzy-set-valued random variable and R is its probability distribution. For example,

$$X \text{ is } r \text{ (} 0.3 \setminus \text{small} + 0.5 \setminus \text{medium} + 0.2 \setminus \text{large} \text{),}$$

means that X is a random variable that takes the fuzzy sets *small*, *medium*, and *large* as its values with respective probabilities 0.3, 0.5, and 0.2. Random set constraints play a central role in the Dempster-Shafer theory of evidence and belief (Shafer 1976.)

In a fuzzy graph constraint, the constrained variable is a function, f , and R is its fuzzy graph (figure 4). A fuzzy graph constraint is represented as

$$F \text{ is } f \text{ (} \sum_i A_i \times B_{j(i)} \text{),}$$

in which the fuzzy sets A_i and $B_{j(i)}$, with j dependent on i , are the granules of X and Y , re-

spectively, and $A_i \times B_{j(i)}$ is the Cartesian product of A_i and $B_{j(i)}$. Equivalently, a fuzzy graph may be expressed as a collection of fuzzy if-then rules of the form

$$\text{if } X \text{ is } A_i \text{ then } Y \text{ is } B_{j(i)}, \quad i = 1, \dots, m; \quad j = 1, \dots, n$$

For example:

$$F \text{ is } f \text{ (} \text{small} \times \text{small} + \text{medium} \times \text{large} + \text{large} \times \text{small} \text{)}$$

may be expressed as the rule set:

- if X is *small* then Y is *small*
- if X is *medium* then Y is *large*
- if X is *large* then Y is *small*

Such a rule set may be interpreted as a description of a perception of f .

A bimodal constraint involves a combination of two modalities: probabilistic and possibilistic. More specifically, in the generalized constraint

$$X \text{ is } b \text{m } R,$$

X is a random variable, and R is what is referred to as a bimodal distribution, P , of X , with P expressed as

$$P: \sum_i P_{j(i)} \setminus A_i,$$

in which the A_i are granules of X , and the $P_{j(i)}$, with j dependent on i , are the granules of probability (figure 5). For example, if X is a real-val-

ued random variable with granules labeled *small*, *medium*, and *large* and probability granules labeled *low*, *medium*, and *high*, then

$$X \text{ isbm } (low \setminus small \setminus + high \setminus medium + low \setminus large)$$

which means that

- Prob {X is *small*} is *low*
- Prob {X is *medium*} is *high*
- Prob {X is *large*} is *low*

In effect, the bimodal distribution of X may be viewed as a description of a perception of the probability distribution of X. As a perception of likelihood, the concept of a bimodal distribution plays a key role in perception-based calculus of probabilistic reasoning (Zadeh 2002.)

The concept of a bimodal distribution is an instance of combination of different modalities. More generally, generalized constraints may be combined and propagated, generating generalized constraints that are composites of other generalized constraints. The set of all such constraints together with deduction rules—rules that are based on the rules governing generalized-constraint propagation—constitutes the generalized-constraint language (GCL). An example of a generalized constraint in GCL is

$$(X \text{ is } A) \text{ and } ((X, Y) \text{ is } B),$$

where A is the probability distribution of X and B is the possibility distribution of the binary variable (X,Y). Constraints of this form play an important role in the Dempster-Shafer theory of evidence (Shafer 1976.)

The Concepts of Precisiability and Precision Language

Informally, a proposition, p, in a natural language, NL, is precisiable if its meaning can be represented in a form that lends itself to computation and deduction. More specifically, p is precisiable if it can be translated into what may be called a precision language, PL, with the understanding that the elements of PL can serve as objects of computation and deduction. In this sense, mathematical languages and the languages associated with propositional logic, first-order and higher-order predicate logics, modal logic, LISP, Prolog, SQL, and related languages may be viewed as precision languages. The existing PL languages are based on bivalent logic. As a direct consequence, the languages in question do not have sufficient expressive power to represent the meaning of propositions that are descriptors of perceptions. For example, the proposition “All men are mortal” can be precisiated by translation into the language associated with first-order logic, but “Most Swedes are tall” cannot.

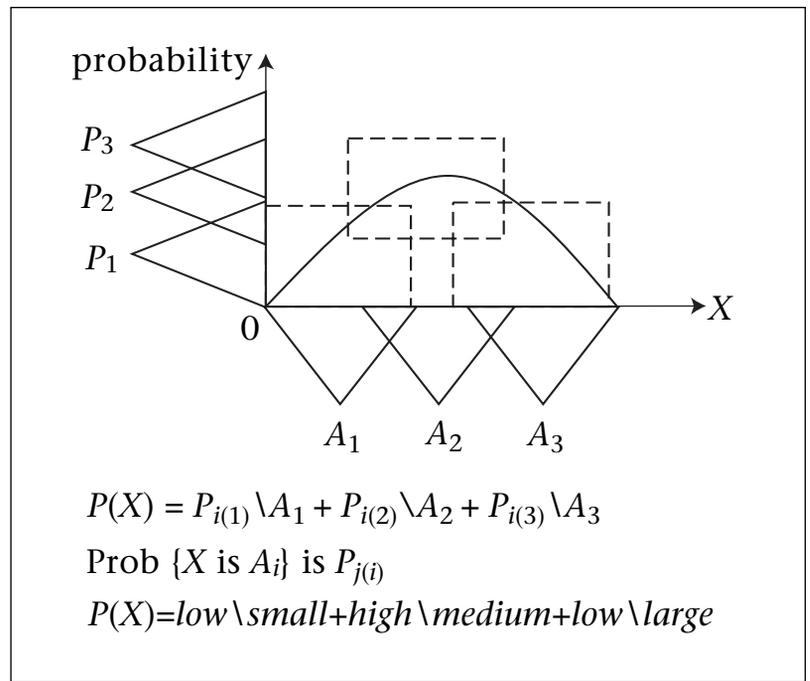


Figure 5. Bimodal Distribution: Perception-Based Probability Distribution.

The principal distinguishing feature of PNL is that the precision language with which it is associated is GCL. It is this feature of PNL that makes it possible to employ PNL as a meaning-precision language for perceptions. What should be understood, however, is that not all perceptions or, more precisely, propositions that describe perceptions, are precisiable through translation into GCL. Natural languages are basically systems for describing and reasoning with perceptions, and many perceptions are much too complex to lend themselves to precision.

The key idea in PNL is that the meaning of a precisiable proposition, p, in a natural language is a generalized constraint X isr R. In general, X, R, and r are implicit, rather than explicit, in p. Thus, translation of p into GCL may be viewed as explicitation of X, R, and r. The expression X isr R will be referred to as the GC form of p, written as GC(p).

In PNL, a proposition, p, is viewed as an answer to a question, q. To illustrate, the proposition p: Monika is young may be viewed as the answer to the question q: How old is Monika? More concretely:

- p: Monika is young → p*: Age (Monika) is young
- q: How old is Monika? → q*: Age (Monika) is ?R

where p* and q* are abbreviations for GC(p) and GC(q), respectively.

In general, the question to which p is an answer is not unique. For example, p: Monika is young could be viewed as an answer to the

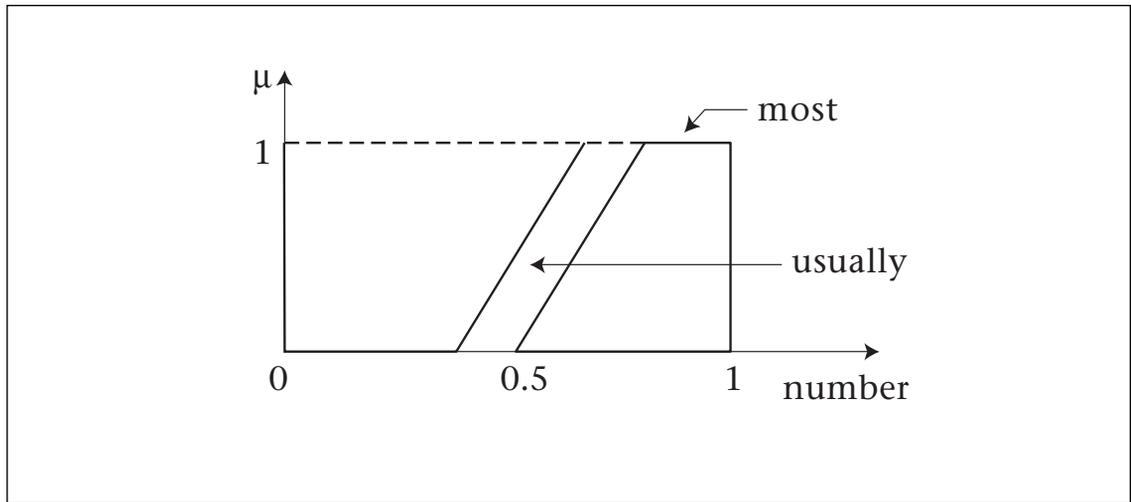


Figure 6. Calibration of Most and Usually Represented as Trapezoidal Fuzzy Numbers.

question q : Who is young? In most cases, however, among the possible questions there is one that is most likely. Such a question plays the role of a default question. The GC form of q is, in effect, the translation of the question to which p is an answer. The following simple examples are intended to clarify the process of translation from NL to GCL.

p : Tandy is much older than Dana \rightarrow
(Age(Tandy), Age(Dana)) is much.older,

where *much.older* is a binary fuzzy relation that has to be calibrated as a whole rather through composition of *much* and *older*.

p : Most Swedes are tall

To deal with the example, it is necessary to have a means of counting the number of elements in a fuzzy set. There are several ways in which this can be done, with the simplest way relating to the concept of Σ Count (sigma count). More specifically, if A and B are fuzzy sets in a space $U = \{u_1, \dots, u_n\}$, with respective membership functions μ_A and μ_B , respectively, then

$$\Sigma\text{Count}(A) = \sum_i \mu_A(u_i),$$

and the relative Σ Count, that is, the relative count of elements of A that are in B , is defined as

$$\Sigma\text{Count}(A/B) = \frac{\Sigma\text{Count}(A \cap B)}{\Sigma\text{Count}(B)},$$

in which the membership function of the intersection $A \cap B$ is defined as

$$\mu_{A \cap B}(u) = \mu_A(u) \wedge \mu_B(u),$$

where \wedge is min or, more generally, a t-norm (Pedrycz and Gomide 1998; Klement, Mesiar, and Pap 2000).

Using the concept of sigma count, the translation in question may be expressed as

Most Swedes are tall \rightarrow
 $\Sigma\text{Count}(\text{tall.Swedes}/\text{Swedes})$ is *most*,

where *most* is a fuzzy number that defines *most* as a fuzzy quantifier (Zadeh 1984, Mesiar and Thiele 2000) (figure 6).

p : Usually Robert returns from work at about 6 PM

q : When does Robert return from work?

X : Time of return of Robert from work,
Time(Return)

R : about 6 PM (6* PM)

r : u (usuality)

p^* : Prob {*Time(Return)* is 6* PM} is *usually*.

A less simple example is:

p : It is very unlikely that there will be a significant increase in the price of oil in the near future.

In this example, it is expedient to start with the semantic network representation (Sowa 1991) of p that is shown in figure 7. In this representation, E is the main event, and E^* is a subevent of E :

E : significant increase in the price of oil in the near future

E^* : significant increase in the price of oil

Thus, *near future* is the epoch of E^* .

The GC form of p may be expressed as

Prob(E) is R ,

where R is the fuzzy probability, *very unlikely*, whose membership function is related to that of *likely* by figure 8.

$$\mu_{\text{very.unlikely}}(u) = (1 - \mu_{\text{likely}})^2,$$

where it is assumed for simplicity that *very* acts as an intensifier that squares the membership function of its operand, and that the membership function of *unlikely* is the mirror image of that of *likely*.

Given the membership functions of *signifi-*

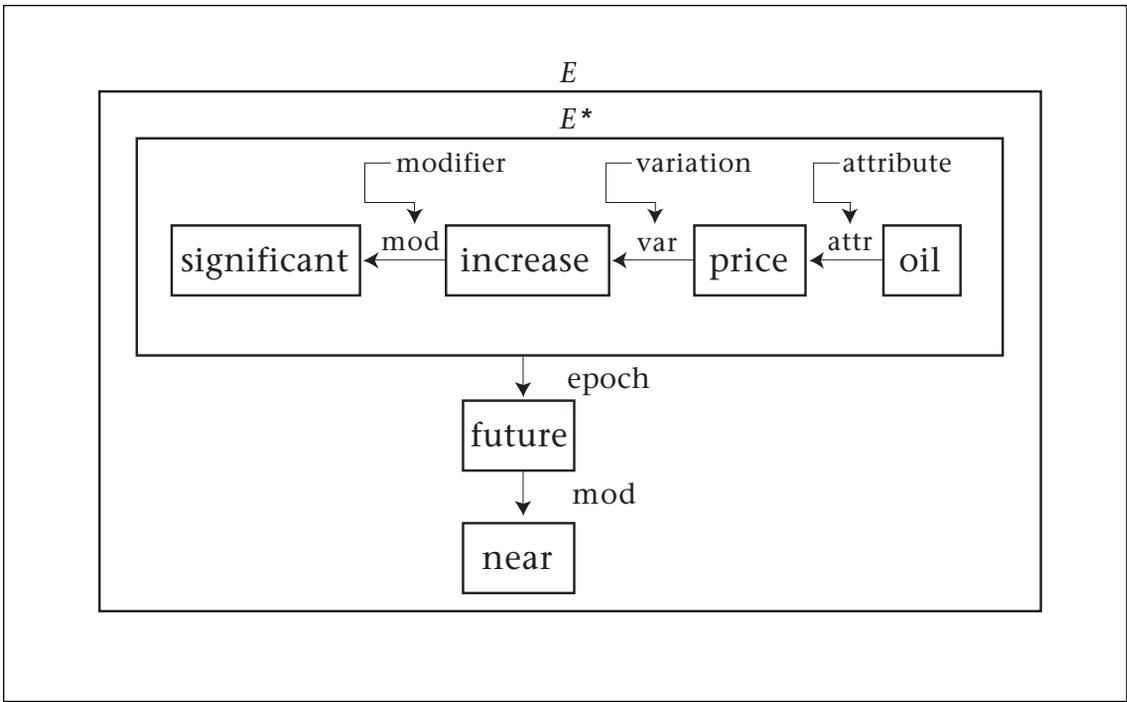


Figure 7. Semantic Network of *p*.

(It is very unlikely that there will be a significant increase in the price of oil in the near future.)

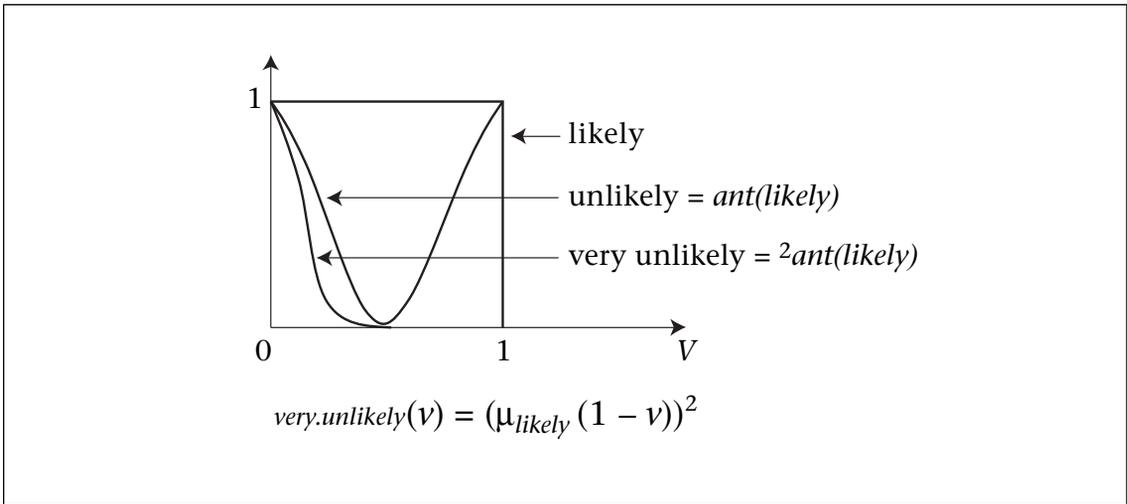


Figure 8. Precision of Very Unlikely.

cant increase and *near future* (figure 9), we can compute the degree to which a specified time function that represents a variation in the price of oil satisfies the conjunction of the constraints *significant increase* and *near future*. This degree may be employed to compute the truth value of *p* as a function of the probability distribution of the variation in the price of oil. In this instance, the use of PNL may be viewed as an extension of truth-conditional semantics (Cresswell 1973, Allan 2001.)

What should be noted is that precision

and meaning representation are not coextensive. More specifically, precision of a proposition, *p*, assumes that the meaning of *p* is understood and that what is involved is a precision of the meaning of *p*.

The Concept of a Protoform and the Structure of PNL

A concept that plays a key role in PNL is that of a protoform—an abbreviation of prototypical form. Informally, a protoform is an abstracted

The Tall Swedes Problem (Version 2)

In the following, a^* denotes “approximately a .” Swedes more than twenty years of age range in height from 140 centimeters to 220 centimeters. Over 70* percent are taller than 170* centimeters; less than 10* percent are shorter than 150* centimeters; and less than 15 percent are taller than 200* centimeters. What is the average height of Swedes over twenty?

Fuzzy Logic Solution

Consider a population of Swedes over twenty, $S = \{\text{Swede}_1, \text{Swede}_2, \dots, \text{Swede}_N\}$, with h_i , $i = 1, \dots, N$, being the height of S_i .

The datum “Over 70* percent of S are taller than 170* centimeters,” constrains the h_i in $h = (h_1, \dots, h_N)$. The constraint is precipitated through translation into GCL. More specifically, let X denote a variable taking values in S , and let $X|h(X) \text{ is } \geq 170^*$ denote a fuzzy subset of S induced by the constraint $h(X) \text{ is } \geq 170^*$. Then

Over 70* percent of S are taller than 170* \rightarrow

$$\text{(GCL): } \frac{1}{N} \sum \text{Count}(X|h(X) \text{ is } \geq 170^*) \text{ is } \geq 0.7^*$$

where $\sum \text{Count}$ is the sigma count of X s that satisfy the fuzzy constraint $h(X) \text{ is } \geq 170^*$.

Similarly,

Less than 10* percent of S are shorter than 150* \rightarrow

$$\text{(GCL): } \frac{1}{N} \sum \text{Count}(X|h(X) \text{ is } \leq 150^*) \text{ is } \leq 0.1$$

and

Less than 15* percent of S are taller than 200* \rightarrow

$$\text{(GCL): } \frac{1}{N} \sum \text{Count}(X|h(X) \text{ is } \geq 200^*) \text{ is } \leq 0.15$$

A general deduction rule in fuzzy logic is the following. In this rule, X is a variable that takes values in a finite set $U = \{u_1, u_2, \dots, u_N\}$, and $a(X)$ is a real-valued attribute of X , with $a_i = a(u_i)$ and $a = (a_1, \dots, a_N)$

$$\frac{\frac{1}{N} \sum \text{Count}(X|a(X) \text{ is } C) \text{ is } B}{\text{Av}(X) \text{ is } ?D}$$

where $\text{Av}(X)$ is the average value of X over U . Thus, computation of the average value, D , reduces to the solution of the nonlinear programming problem

$$\mu_D(v) = \max_{a_i} \mu_B\left(\frac{1}{N} \sum_i \mu_C(a_i)\right)$$

subject to

$$v = \frac{1}{N} \sum_i a_i \quad (\text{average value})$$

where μ_D , μ_B , and μ_C are the membership functions of D , B , and C , respectively. To apply this rule to the constraints in question, it is necessary to form their conjunction. Then, the fuzzy logic solution of the problem may be reduced to the solution of the nonlinear programming problem

$$\mu_D(v) = \max_{h_i} \left(\mu_{\geq 0.7^*} \left(\frac{1}{N} \sum_i \mu_{\geq 170^*}(h_i) \right) \wedge \mu_{\leq 0.1^*} \left(\frac{1}{N} \sum_i \mu_{\leq 150^*}(h_i) \right) \wedge \mu_{\leq 0.15^*} \left(\frac{1}{N} \sum_i \mu_{\geq 200^*}(h_i) \right) \right)$$

subject to

$$v = \frac{1}{N} \sum_i h_i$$

Note that computation of D requires calibration of the membership functions of $\leq 170^*$, $\leq 0.7^*$, $\leq 150^*$, $\leq 0.1^*$, $\geq 200^*$, and $\leq 0.15^*$. Note also that the fuzzy logic solution is a solution in the sense that

summary of an object that may be a proposition, command, question, scenario, concept, decision problem, or, more generally, a system of such objects. The importance of the concept of a protoform derives from the fact that it places in evidence the deep semantic structure of the object to which it applies. For example, the protoform of the proposition p : Monika is young is $PF(p)$: $A(B)$ is C , where A is abstraction of the attribute Age, B is abstraction of Monika, and C is abstraction of young. Conversely, Age is instantiation of A , Monika is instantiation of B , and young is instantiation of C . Abstraction may be annotated, for example, A /Attribute, B /Name, and C /Attribute.value. A few examples are shown in figure 10. Basically, abstraction is a means of generalization. Abstraction has levels, just as summarization does. For example, successive abstractions of p : Monika is young are $A(\text{Monika})$ is young, $A(B)$ is young, and $A(B)$ is C , with the last abstraction resulting in the terminal protoform, or simply the protoform. With this understanding, the protoform of p : Most Swedes are tall is QAs are Bs , or equivalently, $\text{Count}(B/A)$ is Q , and the protoform of p : Usually Robert returns from work at about 6 PM, is $\text{Prob}(X \text{ is } A)$ is B , where X , A , and B are abstractions of "Time (Robert returns from work)," "About 6 PM," and "Usually." For simplicity, the protoform of p may be written as p^{**} .

Abstraction is a familiar concept in programming languages and programming systems. As will be seen in the following, the role of abstraction in PNL is significantly different and more essential because PNL abandons bivalence. The concept of a protoform has some links to other basic concepts such as ontology (Sowa 1999; Smith and Welty 2002; Smith, Welty, and McGuinness 2003; Corcho, Fernandez-Lopez, Gomez-Perez 2003) conceptual graph (Sowa 1984) and Montague grammar (Partee 1976.) However, what should be stressed is that the concept of a protoform is not limited—as it is in the case of related concepts—to propositions whose meaning can be represented within the conceptual structure of bivalent logic.

As an illustration, consider a proposition, p , which was dealt with earlier:

p : It is very unlikely that there will be a significant increase in the price of oil in the near future.

With reference to the semantic network of p (figure 9), the protoform of p may be expressed as:

Prob(E) is A (A : very unlikely)
 E : $B(E^*)$ is C (B : epoch; C : near.future)
 E^* : $F(D)$ (F : significant increase;
 D : price of oil)
 D : $G(H)$ (G : price; H : oil)

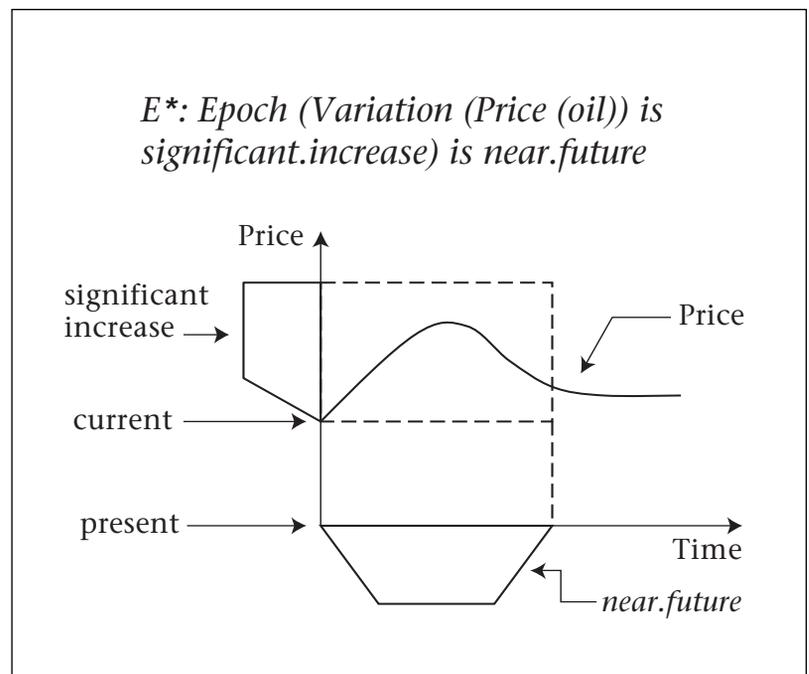


Figure 9. Computation of Degree of Compatibility.

Using the protoform of p and calibrations of *significant increase*, *near-future*, and *likely*, (figure 9), we can compute, in principle, the degree to which any given probability distribution of time functions representing the price of oil satisfies the generalized constraint, $\text{Prob}(E)$ is A . As was pointed out earlier, if the degree of compatibility is interpreted as the truth value of p , computation of the truth value of p may be viewed as a PNL-based extension of truth-conditional semantics.

By serving as a means of defining the deep semantic structure of an object, the concept of a protoform provides a platform for a fundamental mode of classification of knowledge based on protoform equivalence, or PF equivalence for short. More specifically, two objects are protoform equivalent at a specified level of summarization and abstraction if at that level they have identical protoforms. For example, the propositions p : Most Swedes are tall, and q : Few professors are rich, are PF equivalent since their common protoform is QAs are Bs or, equivalently, $\text{Count}(B/A)$ is Q . The same applies to propositions p : Oakland is near San Francisco, and q : Rome is much older than Boston. A simple example of PF equivalent concepts is: cluster and mountain.

A less simple example involving PF equivalence of scenarios of decision problems is the following. Consider the scenarios of two decision problems, A and B .

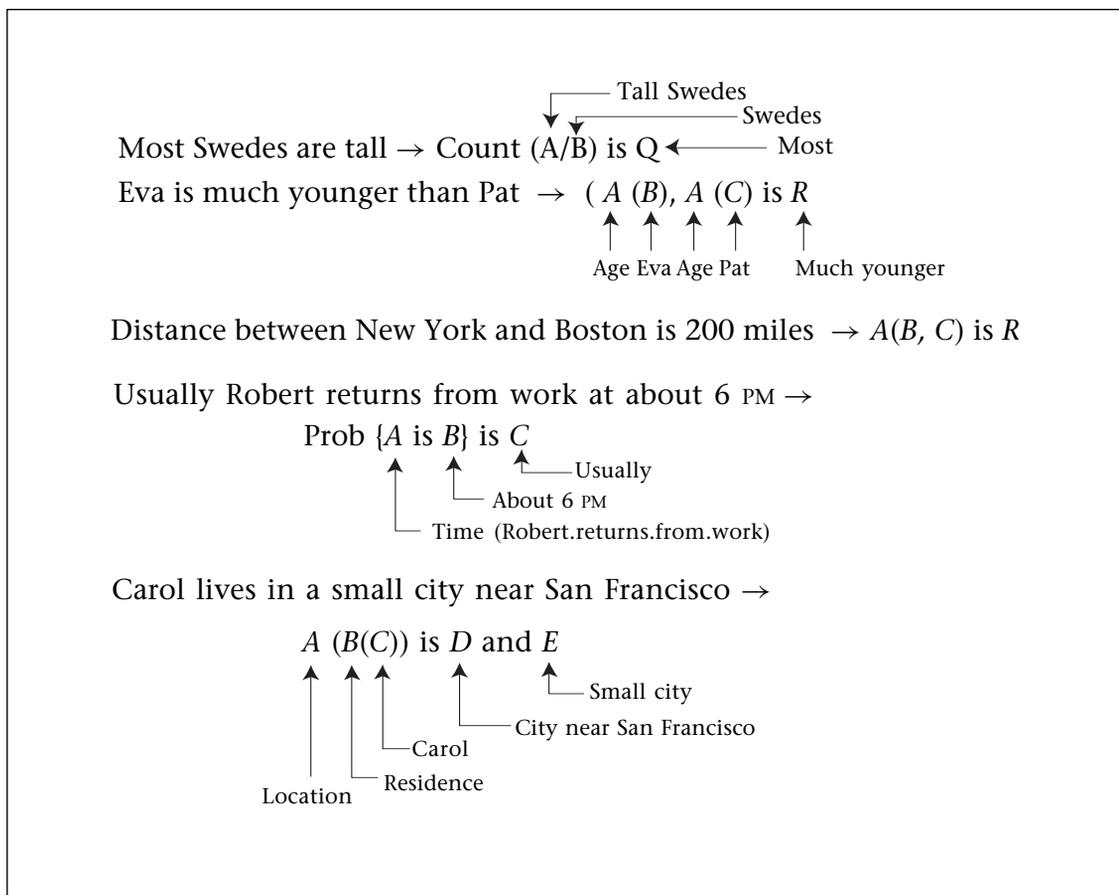


Figure 10. Examples of Translation from NL to PFL

Scenario A:

Alan has severe back pain. He goes to see a doctor. The doctor tells him that there are two options: (1) do nothing and (2) do surgery. In the case of surgery, there are two possibilities: (a) surgery is successful, in which case Alan will be pain-free; and (b) surgery is not successful, in which case Alan will be paralyzed from the neck down. Question: Should Alan elect surgery?

Scenario B:

Alan needs to fly from San Francisco to St. Louis and has to get there as soon as possible. One option is to fly to St. Louis via Chicago, and the other is to go through Denver. The flight via Denver is scheduled to arrive in St. Louis at time *a*. The flight via Chicago is scheduled to arrive in St. Louis at time *b*, with *a* < *b*. However, the connection time in Denver is short. If the connection flight is missed, then the time of arrival in St. Louis will be *c*, with *c* > *b*. Question: Which option is best?

The common protoform of *A* and *B* is shown in figure 11. What this protoform means is that there are two options, one that is associated with a certain gain or loss and another that has

two possible outcomes whose probabilities may not be known precisely.

The protoform language, PFL, is the set of protoforms of the elements of the generalized-constraint language, GCL. A consequence of the concept of PF equivalence is that cardinality of PFL is orders of magnitude lower than that of GCL or, equivalently, the set of precisable propositions in NL. As will be seen in the sequel, the low cardinality of PFL plays an essential role in deduction.

The principal components of the structure of PNL (figure 12) are (1) a dictionary from NL to GCL; (2) a dictionary from GCL to PFL (figure 13); (3) a multiagent, modular deduction database, DDB; and (4) a world knowledge database, WKDB. The constituents of DDB are modules, with a module consisting of a group of protoformal rules of deduction, expressed in PFL (figure 14), that are drawn from a particular domain, for example, probability, possibility, usability, fuzzy arithmetic (Kaufmann and Gupta 1985), fuzzy logic, search, and so on. For example, a rule drawn from fuzzy logic is the compositional rule of inference, expressed in figure 14 where *A*∘*B* is the composition of *A*

and B , defined in the computational part, in which μ_A , μ_B , and $\mu_{A \circ B}$ are the membership functions of A , B , and $A \circ B$, respectively. Similarly, a rule drawn from probability is shown in figure 15, where D is defined in the computational part.

The rules of deduction in DDB are, basically, the rules that govern propagation of generalized constraints. Each module is associated with an agent whose function is that of controlling execution of rules and performing embedded computations. The top-level agent controls the passing of results of computation from a module to other modules. The structure of protoformal, that is, protoform based, deduction is shown in figure 16. A simple example of protoformal deduction is shown in figure 17.

The world knowledge database, WKDB, consists of propositions that describe world knowledge, for example, Parking near the campus is hard to find on weekdays between 9 and 4; Big cars are safer than small cars; If A /person works in B /city then it is likely that A lives in or near B ; If A /person is at home at time t then A has returned from work at t or earlier, on the understanding that A stayed home after returning from work. Much, perhaps most, of the information in WKDB is perception based.

World knowledge—and especially world

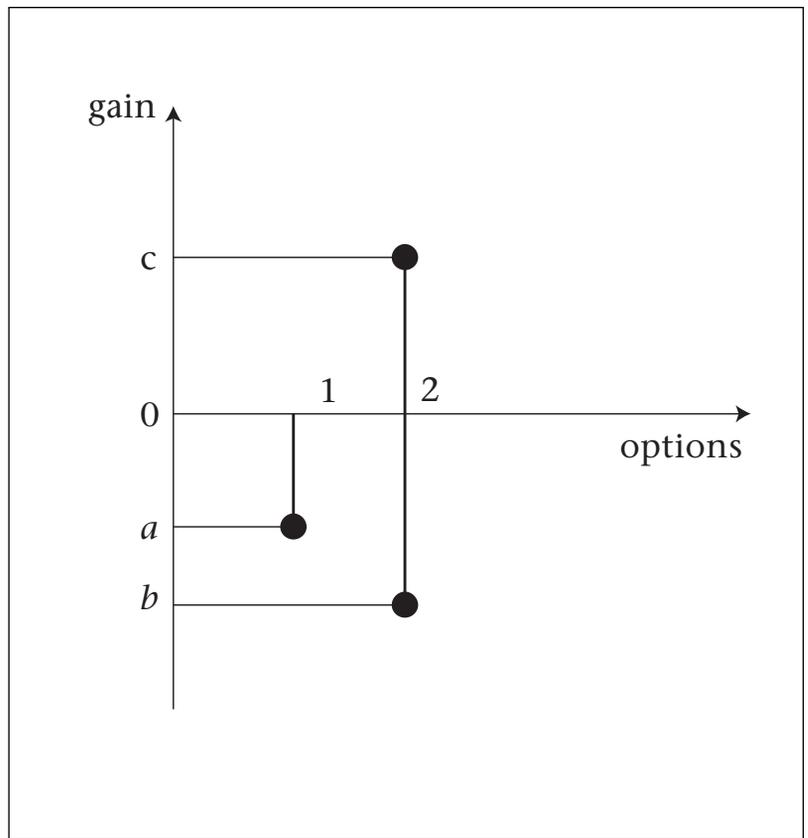


Figure 11. Protoform Equivalence of Scenarios A and B.

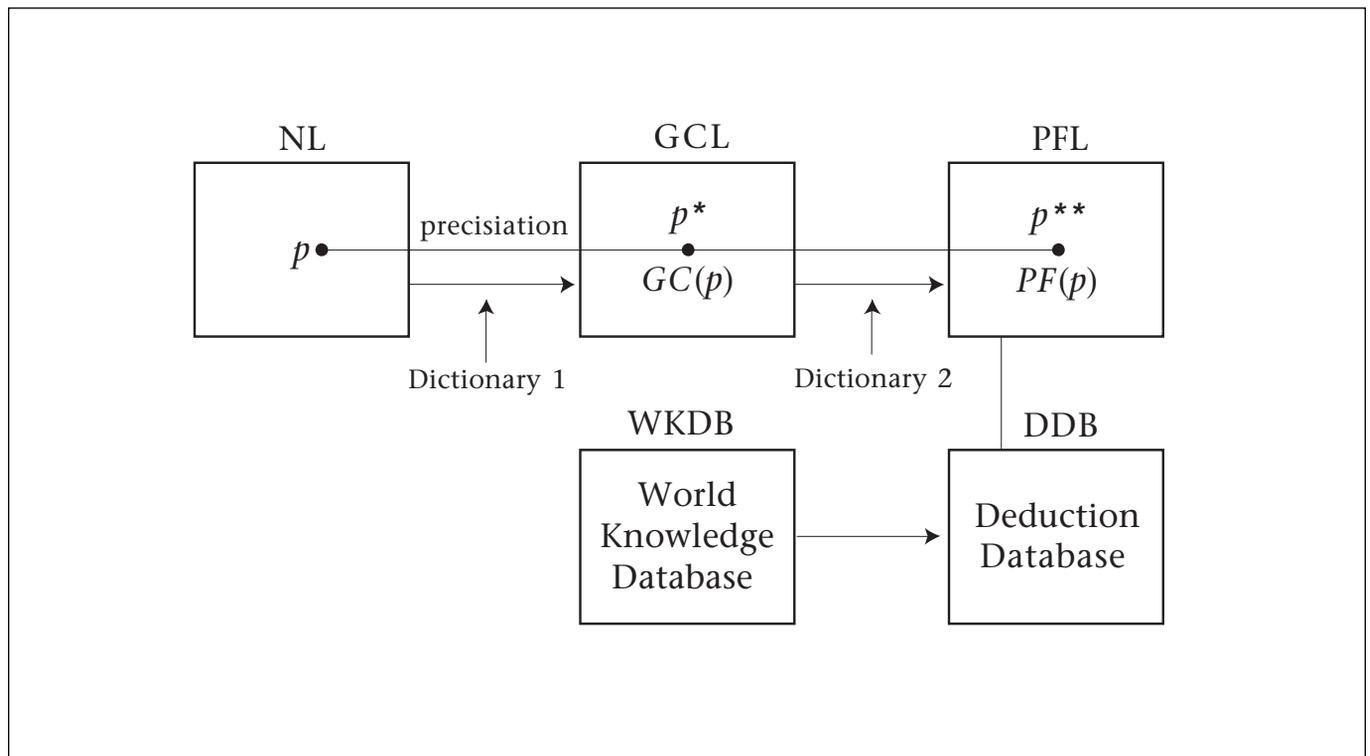


Figure 12. Basic Structure of PNL.

1:	
Proposition in NL	Precision
p	p^* (GC-form)
most Swedes are tall	$\Sigma Count (tall.Swedes/Swedes)$ is most
2:	
Precision	Protoform
p^* (GC-form)	$PF(p^*)$
$\Sigma Count (tall.Swedes/Swedes)$ is most	Q As are Bs

Figure 13. Structure of PNL Dictionaries.

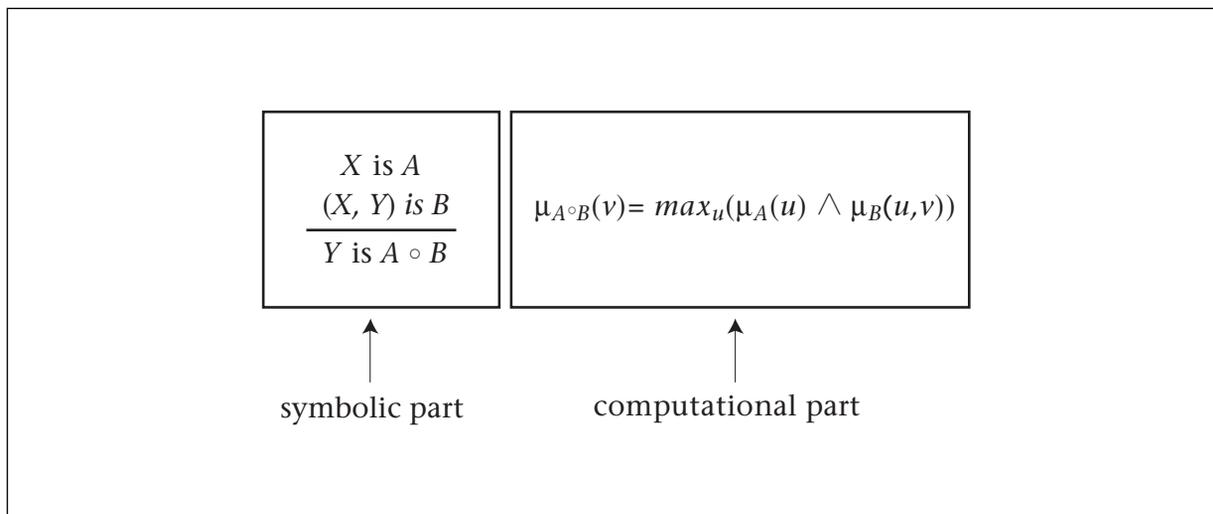


Figure 14. Compositional Rule of Inference.

knowledge about probabilities—plays an essential role in almost all search processes, including searching the Web. Semantic Web and related approaches have contributed to a significant improvement in performance of search engines. However, for further progress it may be necessary to add to existing search engines the capability to operate on perception-based information. It will be a real challenge to employ PNL to add this capability to sophisticated knowledge-management systems such as the Web Ontology Language (OWL) (Smith,

Welty, and McGuinness 2003), Cyc (Lenat 1995), WordNet (Feldbaum 1998), and ConceptNet (Lin and Singh 2004).

An example of PFL-based deduction in which world knowledge is used is the so-called Robert example. A simplified version of the example is the following.

The initial data set is the proposition (perception) p : Usually Robert returns from work at about 6 PM. The question is q : What is the probability that Robert is home at 6:15 PM?

The first step in the deduction process is to

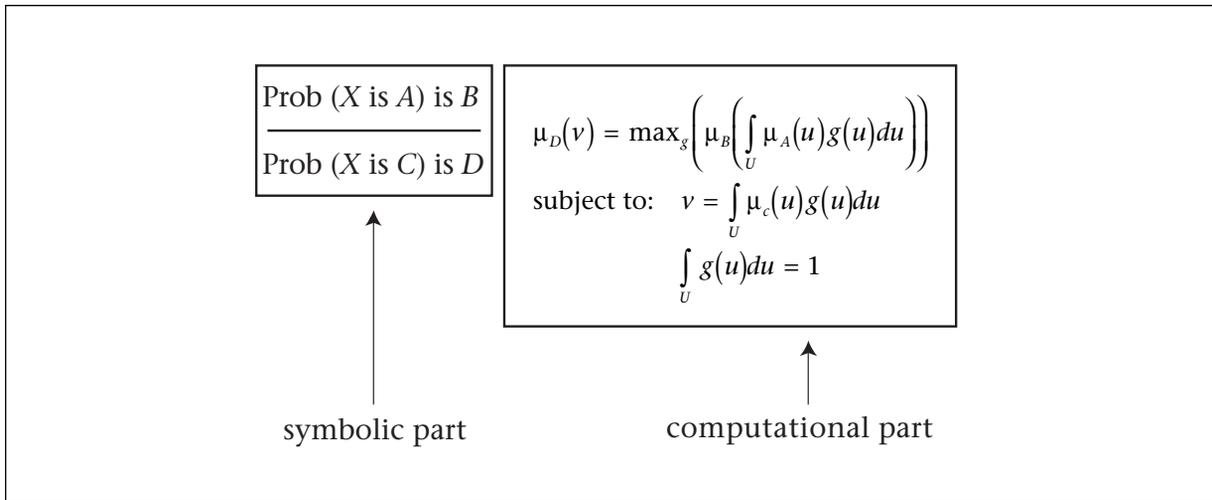


Figure 15. Rule Drawn from Probability.

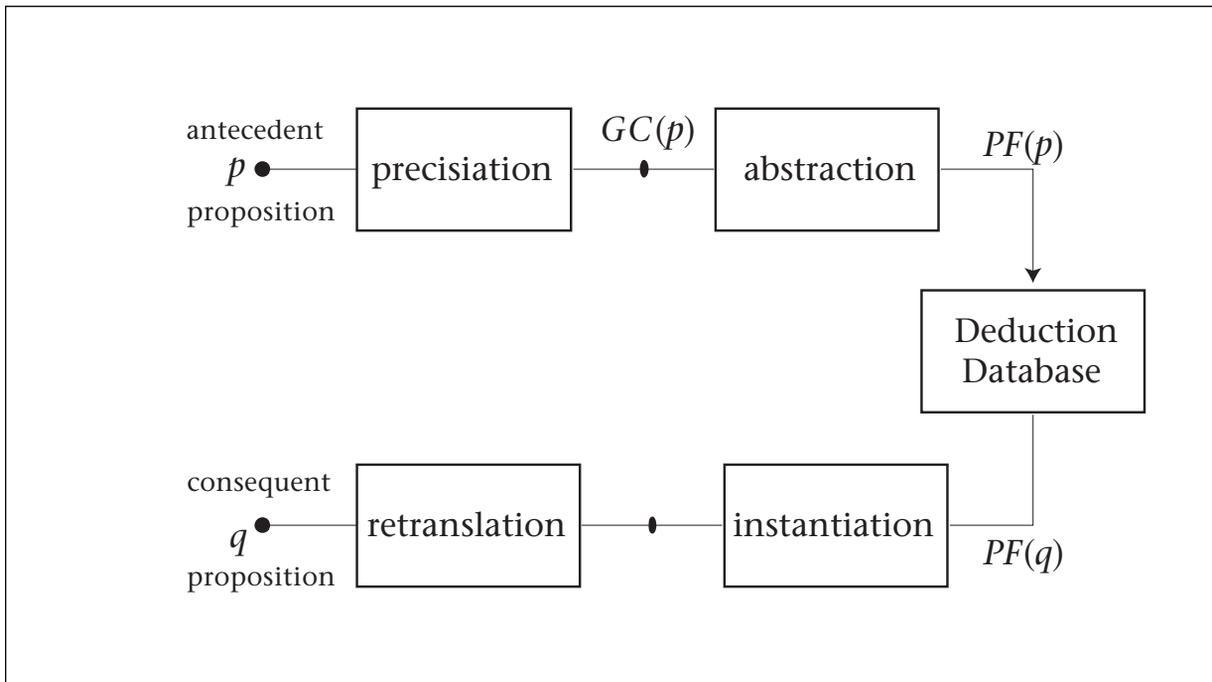


Figure 16. Structure of Protoform-Based Deduction.

use the NL to GCL dictionary for deriving the generalized-constraint forms, $GC(p)$ and $GC(q)$, of p and q , respectively. The second step is to use the GCL to PFL dictionary to derive the protoforms of p and q . The forms are:

p^* : Prob(Time(Robert.returns.from.work) is about 6 PM) is usually
 q^* : Prob(Time(Robert is home) is 6:15 PM) is ?E

and

p^{**} : Prob(X is A) is B
 q^{**} : Prob(Y is C) is ?D

The third step is to refer the problem to the top-level agent with the query, Is there a rule or a chain of rules in DDB that leads from p^{**} to q^{**} ? The top-level agent reports a failure to find such a chain but success in finding a proximate rule of the form

$\frac{\text{Prob}(X \text{ is } A) \text{ is } B}{\text{Prob}(X \text{ is } C) \text{ is } D}$

The fourth step is to search the world knowledge database, WKDB, for a proposition or a chain of propositions that allow Y to be re-

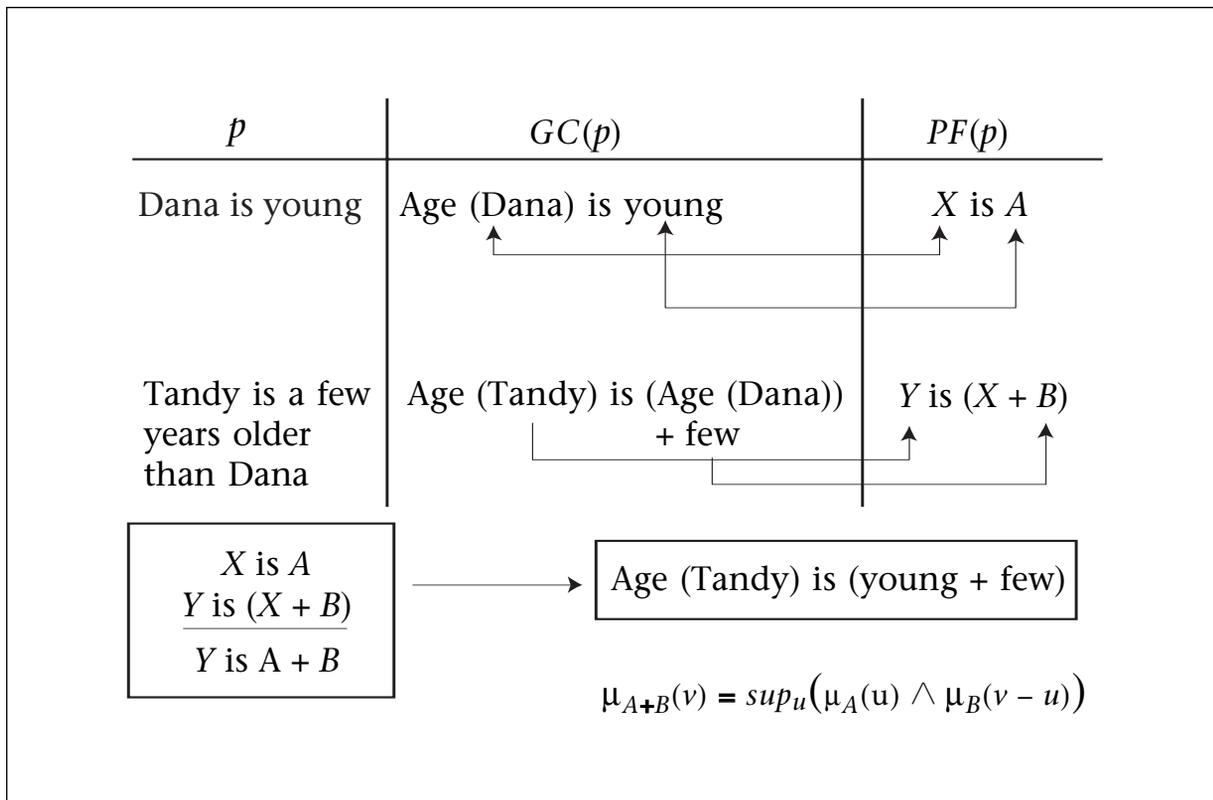


Figure 17. Example of Protoformal Reasoning.

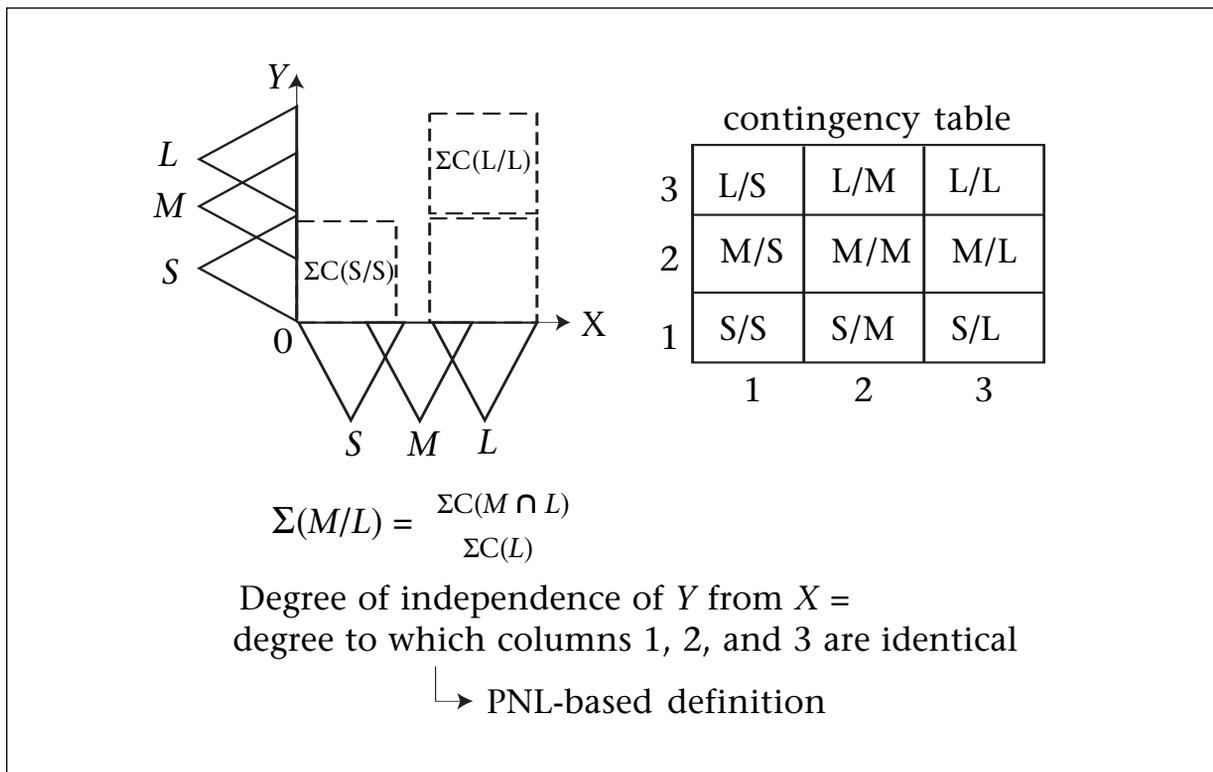


Figure 18. PNL-Based Definition of Statistical Independence.

placed by X . A proposition that makes this possible is (A /person is in B /location) at T /time if A arrives at B before T , with the understanding that A stays at B after arrival.

The last step involves the use of the modified form of q^{**} : $\text{Prob}(X \text{ is } E) \text{ is } ?D$, in which E is “before 6:15 PM.” The answer to the initial query is given by the solution of the variational problem associated with the rule that was described earlier (figure 15):

$$\frac{\text{Prob}(X \text{ is } A) \text{ is } B}{\text{Prob}(X \text{ is } C) \text{ is } D}$$

The value of D is the desired probability.

What is important to observe is that there is a tacit assumption that underlies the deduction process, namely, that the chains of deduction are short. This assumption is a consequence of the intrinsic imprecision of perception-based information. Its further implication is that PNL is likely to be effective, in the main, in the realm of domain-restricted systems associated with small universes of discourse.

PNL as a Definition Language

As we move further into the age of machine intelligence and automated reasoning, a problem that is certain to grow in visibility and importance is that of definability—that is, the problem of defining the meaning of a concept or a proposition in a way that can be understood by a machine.

It is a deeply entrenched tradition in science to define a concept in a language that is based on bivalent logic (Gamut 1996, Gerla 2000, Hájek 2000). Thus defined, a concept, C , is bivalent in the sense that every object, X , is either an instance of C or it is not, with no degrees of truth allowed. For example, a system is either stable or unstable, a time series is either stationary or nonstationary, a sentence is either grammatical or ungrammatical, and events A and B are either independent or not independent.

The problem is that bivalence of concepts is in conflict with reality. In most settings, stability, stationarity, grammaticality, independence, relevance, causality, and most other concepts are not bivalent. When a concept that is not bivalent is defined as if it were bivalent, the ancient Greek sorites (heap) paradox comes into play. As an illustration, consider the standard bivalent definition of independence of events, say A and B . Let $P(A)$, $P(B)$, and $P_A(B)$ be the probabilities of A , B , and B given A , respectively. Then A and B are independent if and only if $P_A(B) = P(B)$.

Now assume that the equality is not satisfied exactly, with the difference between the two

sides being ϵ . As ϵ increases, at which point will A and B cease to be independent?

Clearly, independence is a matter of degree, and furthermore the degree is context dependent. For this reason, we do not have a universally accepted definition of *degree of independence* (Klir 2000.)

One of the important functions of PNL is that of serving as a definition language. More specifically, PNL may be employed as a definition language for two different purposes: first, to define concepts for which no general definitions exist, for example, causality, summary, relevance, and smoothness; and second, to redefine concepts for which universally accepted definitions exist, for example, linearity, stability, independence, and so on. In what follows, the concept of independence of random variables will be used as an illustration.

For simplicity, assume that X and Y are random variables that take values in the interval $[a, b]$. The interval is granulated as shown in figure 18, with S , M , and L denoting the fuzzy intervals *small*, *medium*, and *large*.

Using the definition of relative Σ Count, we construct a contingency table, C , of the form shown in figure 18, in which an entry such as $\Sigma\text{Count}(S/L)$ is a granulated fuzzy number that represents the relative Σ Count of occurrences of Y , which are small, relative to occurrences of X , which are large.

Based on the contingency table, the degree of independence of Y from X may be equated to the degree to which the columns of the contingency table are identical. One way of computing this degree is, first, to compute the distance between two columns and then aggregate the distances between all pairs of columns. PNL would be used for this purpose.

An important point that this example illustrates is that, typically, a PNL-based definition involves a general framework with a flexible choice of details governed by the context or a particular application. In this sense, the use of PNL implies an abandonment of the quest for universality, or, to put it more graphically, of the one-size-fits-all modes of definition that are associated with the use of bivalent logic.

Another important point is that PNL suggests an unconventional approach to the definition of complex concepts. The basic idea is to define a complex concept in a natural language and then employ PNL to precisiate the definition.

More specifically, let U be a universe of discourse and let C be a concept that I wish to define, with C relating to elements of U . For example, U is a set of buildings, and C is the concept of tall building. Let $p(C)$ and $d(C)$ be, respectively, my perception and my definition of C . Let

$I(p(C))$ and $I(d(C))$ be the intensions of $p(C)$ and $d(C)$, respectively, with *intension* used in its logical sense (Cresswell 1973, Gamat 1996), that is, as a criterion or procedure that identifies those elements of U that fit $p(C)$ or $d(C)$. For example, in the case of tall buildings, the criterion may involve the height of a building.

Informally, a definition, $d(C)$, is a good fit or, more precisely, is cointensive, if its intension coincides with the intension of $p(C)$. A measure of goodness of fit is the degree to which the intension of $d(C)$ coincides with that of $p(C)$. In this sense, cointension is a fuzzy concept. As a high-level definition language, PNL makes it possible to formulate definitions whose degree of cointensiveness is higher than that of definitions formulated through the use of languages based on bivalent logic.

Concluding Remarks

Existing theories of natural languages are based, anachronistically, on Aristotelian logic—a logical system whose centerpiece is the principle of the excluded middle: Truth is bivalent, meaning that every proposition is either true or not true, with no shades of truth allowed.

The problem is that bivalence is in conflict with reality—the reality of pervasive imprecision of natural languages. The underlying facts are (a) a natural language, NL, is, in essence, a system for describing perceptions; and (b) perceptions are intrinsically imprecise, reflecting the bounded ability of sensory organs, and ultimately the brain, to resolve detail and store information.

PNL abandons bivalence. What this means is that PNL is based on fuzzy logic—a logical system in which everything is, or is allowed to be, a matter of degree.

Abandonment of bivalence opens the door to exploration of new directions in theories of natural languages. One such direction is that of precisiation. A key concept underlying precisiation is the concept of a generalized constraint. It is this concept that makes it possible to represent the meaning of a proposition drawn from a natural language as a generalized constraint. Conventional, bivalent constraints cannot be used for this purpose. The concept of a generalized constraint provides a basis for construction of GCL—a language whose elements are generalized constraints and their combinations. Within the structure of PNL, GCL serves as a precisiation language for NL. Thus, a proposition in NL is precisiated through translation into GCL. Not every proposition in NL is precisiable. In effect, the elements of PNL are precisiable propositions in NL.

What should be underscored is that in its role as a high-level definition language, PNL provides a basis for a significant enlargement of the role of natural languages in scientific theories.

Dedication

This article is dedicated to Noam Chomsky.

References

- Allan, K. 2001. *Natural Language Semantics*. Oxford: Oxford Blackwell Publishers.
- Barwise, J.; and Cooper, R.. 1981. Generalized Quantifiers and Natural Language. *Linguistics and Philosophy* 4(1): 159–209.
- Biermann, A. W.; and Ballard, B. W. . 1980. Toward Natural Language Computation. *American Journal of Computational Linguistics* (6)2: 71–86.
- Corcho, O., Fernandez-Lopez, M.; and Gomez-Perez, A. 2003. Methodologies, Tools and Languages for Building Ontologies. Where Is Their Meeting Point? *Data and Knowledge Engineering* 46(1): 41–64.
- Cresswell, M. J. 1973. *Logic and Languages*. London: Methuen.
- Davis, E. 1990. *Representations of Common-sense Knowledge*. San Francisco: Morgan Kaufmann.
- Dubois, D., and Prade, H. 1996. Approximate and Commonsense Reasoning: From Theory to Practice. In *Proceedings of the Foundations of Intelligent Systems. Ninth International Symposium*, 19–33. Berlin: Springer-Verlag.
- Fellbaum, C., ed. 1998. *WordNet: An Electronic Lexical Database*. Cambridge, Mass.: The MIT Press.
- Fuchs, N. E.; and Schwertel, U. 2003. Reasoning in Attempto Controlled English. In *Proceedings of the Workshop on Principles and Practice of Semantic Web Reasoning (PPSWR 2003)*, 174–188. Lecture Notes in Computer Science. Berlin: Springer-Verlag.
- Gamat, T. F. 1996. *Language, Logic and Linguistics*. Chicago: University of Chicago Press.
- Gerla, G. 2000. Fuzzy Metalogic for Crisp Logics. In *Discovering the World with Fuzzy Logic: Studies in Fuzziness and Soft Computing* (57), ed. V. Novak and I. Perfilieva, 175–187. Heidelberg: Physica-Verlag.
- Haack, S. 1996. *Deviant Logic, Fuzzy Logic: Beyond the Formalism*. Chicago: University of Chicago Press.
- Haack, S. 1974. *Deviant Logic*. Cambridge: Cambridge University Press.
- Hajek, P. 2000. Many. In *Discovering the World with Fuzzy Logic: Studies in Fuzziness and Soft Computing* (57), ed. V. Novak and I. Perfilieva, 302–304. Heidelberg: Physica-Verlag.
- Hajek, P. 1998. *Metamathematics of Fuzzy Logic: Trends in Logic* (4). Dordrecht, The Netherlands: Kluwer Academic Publishers.
- Kaufmann A.; and Gupta, M. M. 1985. *Introduction to Fuzzy Arithmetic: Theory and Applications*. New York: Van Nostrand.
- Klein, E. 1980. A Semantics for Positive and Comparative Adjectives. *Linguistics and Philosophy* 4(1): 1–45.

- Klement, P.; Mesiar, R.; and Pap, E. 2000. Triangular Norms—Basic Properties and Representation Theorems. In *Discovering the World with Fuzzy Logic: Studies in Fuzziness and Soft Computing* (57), ed. V. Novak and I. Perfilieva, 63–80. Heidelberg: Physica-Verlag.
- Klir, G. J. 2000. Uncertainty-Based Information: A Critical Review. In *Discovering the World with Fuzzy Logic: Studies in Fuzziness and Soft Computing* (57), ed. V. Novak and I. Perfilieva, 29–50. Heidelberg: Physica-Verlag.
- Lehmke, S. 2000. Degrees of Truth and Degrees of Validity. In *Discovering the World with Fuzzy Logic: Studies in Fuzziness and Soft Computing* (57), ed. V. Novak and I. Perfilieva, 192–232. Heidelberg: Physica-Verlag.
- Lehnert, W. G. 1978. *The Process of Question Answering—A Computer Simulation of Cognition*. Hillsdale, New Jersey: Lawrence Erlbaum Associates.
- Lenat, D. B. 1995. CYC: A Large-Scale Investment in Knowledge Infrastructure. *Communications of the ACM* 38(11): 32–38.
- Liu, H.; and Singh, P. 2004. Commonsense Reasoning in and Over Natural Language. In *Proceedings of the Eighth International Conference on Knowledge-Based Intelligent Information and Engineering Systems*. Brighton, U.K.: KES Secretariat, Knowledge Transfer Partnership Centre.
- Macias, B., and Stephen G. Pulman. 1995. A Method for Controlling the Production of Specifications in Natural Language. *The Computing Journal* 38(4): 310–318.
- Mani, I., and M. T. Maybury, eds. 1999. *Advances in Automatic Text Summarization*. Cambridge, Mass.: The MIT Press.
- McAllester, D. A., and R. Givan, R. 1992. Natural Language Syntax and First-Order Inference. *Artificial Intelligence*, 56(1): 1–20.
- McCarthy, J. 1990. *Formalizing Common Sense*, ed. V. Lifschitz and J. McCarthy. Norwood, New Jersey: Ablex Publishers.
- Mesiar, R., and H. Thiele. 2000. On T-Quantifiers and S-Quantifiers. In *Discovering the World with Fuzzy Logic: Studies in Fuzziness and Soft Computing* (57), ed. V. Novak and I. Perfilieva, 310–318. Heidelberg: Physica-Verlag.
- Novak, V., and I. Perfilieva, eds. 2000. *Discovering the World with Fuzzy Logic: Studies in Fuzziness and Soft Computing*. Heidelberg: Physica-Verlag.
- Novak, V. 1991. Fuzzy Logic, Fuzzy Sets, and Natural Languages. *International Journal of General Systems*, 20(1): 83–97.
- Partee, B. 1976. *Montague Grammar*. New York: Academic Press.
- Pedrycz, W., and F. Gomide. 1998. *Introduction to Fuzzy Sets*. Cambridge, Mass.: The MIT Press.
- Peterson, P. 1979. *On the Logic of Few, Many and Most*. *Journal of Formal Logic* 20(1–2): 155–179.
- Shafer, G. 1976. *A Mathematical Theory of Evidence*. Princeton, New Jersey: Princeton University Press.
- Smith, B., and C. Welty. 2002. What Is Ontology? Ontology: Towards a New Synthesis. In *Proceedings of the Second International Conference on Formal Ontology in Information Systems*. New York: Association for Computing Machinery.
- Smith, M. K., C. Welty, and D. McGuinness, eds. 2003. *OWL Web Ontology Language Guide*. W3C Working Draft 31. Cambridge, Mass.: World Wide Web Consortium (W3C).
- Sowa, J. F. 1999. Ontological Categories. In *Shapes of Forms: From Gestalt Psychology and Phenomenology to Ontology and Mathematics*, ed. L. Albertazzi, 307–340. Dordrecht, The Netherlands: Kluwer Academic Publishers.
- Sowa, J. F. 1991. *Principles of Semantic Networks: Explorations in the Representation of Knowledge*. San Francisco: Morgan Kaufmann Publishers.
- Sowa, J. F. 1984. *Conceptual Structures: Information Processing in Mind and Machine*. Reading, Mass.: Addison-Wesley.
- Sukkarieh, J. 2003. Mind Your Language! Controlled Language for Inference Purposes. Paper presented at the Joint Conference of the Eighth International Workshop of the European Association for Machine Translation and the Fourth Controlled Language Applications Workshop, Dublin, Ireland, 15–17 May.
- Sun, R. 1994. *Integrating Rules and Connectionism for Robust Commonsense Reasoning*. New York: John Wiley.
- Yager, R. R. 1991. Deductive Approximate Reasoning Systems. *IEEE Transactions on Knowledge and Data Engineering* 3(4): 399–414.
- Zadeh, L. A. 2002. Toward a Perception-Based Theory of Probabilistic Reasoning with Imprecise Probabilities. *Journal of Statistical Planning and Inference* 105(1): 233–264.
- Zadeh, L. A. 2001. A New Direction in AI—Toward a Computational Theory of Perceptions. *AI Magazine* 22(1): 73–84.
- Zadeh, L. A. 2000. Toward a Logic of Perceptions Based on Fuzzy Logic. In *Discovering the World with Fuzzy Logic: Studies in Fuzziness and Soft Computing* (57), ed. V. Novak and I. Perfilieva, 4–25. Heidelberg: Physica-Verlag.
- Zadeh, L. A. 1999. From Computing with Numbers to Computing with Words—From Manipulation of Measurements to Manipulation of Perceptions. *IEEE Transactions on Circuits and Systems* 45(1): 105–119.
- Zadeh, L. A. 1997. Toward a Theory of Fuzzy Information Granulation and Its Centrality in Human Reasoning and Fuzzy Logic. *Fuzzy Sets and Systems* 90(2): 111–127.
- Zadeh, L. A. 1986. Outline of a Computational Approach to Meaning and Knowledge Representation Based on the Concept of a Generalized Assignment Statement. In *Proceedings of the International Seminar on Artificial Intelligence and Man-Machine Systems*, ed. M. Thoma and A. Wyner, 198–211. Heidelberg: Springer-Verlag.
- Zadeh, L. A. 1984. Syllogistic Reasoning in Fuzzy Logic and Its Application to Reasoning with Dispositions. In *Proceedings International Symposium on Multiple-Valued Logic*, 148–153. Los Alamitos, Calif.: IEEE Computer Society.
- Zadeh, L. A. 1983. A Computational Approach to Fuzzy Quantifiers in Natural Languages. *Computers and Mathematics* 9: 149–184.
- Zadeh, L. A. 1979. A Theory of Approximate Reasoning. In *Machine Intelligence 9*, ed. J. Hayes, D. Michie, and L. I. Mikulich, 149–194. New York: Halstead Press.
- Zadeh, L. A. 1968. Probability Measures of Fuzzy Events. *Journal of Mathematical Analysis and Applications* 23: 421–427.



Lotfi A. Zadeh is a professor in the graduate school and a member of the Department of Electrical Engineering and Computer Science at the University of California, Berkeley, and serves as director of the Berkeley Initiative in

Soft Computing. He received his Ph.D. from Columbia University in 1949 and was promoted to the rank of professor in 1957. In 1959, he moved to the University of California, Berkeley, and served as chair of the Department of Electrical Engineering and Computer Sciences from 1963 to 1968. His first paper on fuzzy sets was published in 1965, and since then his research has focused on the development of fuzzy logic and its applications. Zadeh is a fellow of the American Association for Artificial Intelligence, the Association of Computing Machinery, the Institute of Electrical and Electronics Engineers, and the International Fuzzy Systems Association. He is a member of the National Academy of Engineering and a foreign member of the Russian and Finnish Academies of Sciences. Zadeh has received numerous awards in recognition of his development of fuzzy logic, among them the IEEE Medal of Honor, the ACM 2000 Allen Newell Award, and twenty honorary doctorates. His e-mail address is zadeh@cs.berkeley.edu.

Call for Participation

First Annual General Game Playing Competition

HELD IN CONJUNCTION WITH THE
TWENTIETH NATIONAL CONFERENCE ON ARTIFICIAL INTELLIGENCE

Pittsburgh, Pennsylvania

Sponsored by the American Association for Artificial Intelligence



General game players are computer systems able to accept formal descriptions of arbitrary games and able to play those games effectively without human intervention. General game playing systems are characterized by their use of general cognitive information-processing technologies (such as knowledge representation, reasoning, learning, and rational behavior). Unlike specialized game playing systems (such as Deep Blue), they do not rely on algorithms designed in advance for specific games.

The Competition

The AAAI competition is designed to test the abilities of general game playing systems by comparing their performance on a variety of games. The competition will consist of two phases: a qualification round and a runoff competition.

In the qualification round, entrants will play several different types of games, including single player games (such as maze search), competitive games (such as tic-tac-toe or some variant of chess), games with both competitors and cooperators. In some cases, complete information of the board will be available (as in chess or tic-tac-toe); in others, only partial info will be available (as in battleship). In some cases, the game will be exhaustively searchable (as in tic-tac-toe); in other cases, this will not be possible (as in chess). Players will have to handle all of these possibilities. Entrants will be evaluated on the basis of consistent legal play, ability to attain winning positions,

and overall time; and the best will advance to the second round.

In the runoff round, the best of the qualifiers will be pitted against each other in a series of games of increasingly complexity. The entrant to win the most games in this round will be the winner of the overall competition.

Note that, prior to the competition, players will be told nothing about the games to be played. The rules of all games will be transmitted to the players electronically at the beginning of each game. Game playing systems must be able to read the rules for each game, receive runtime information from the game manager, and inform the manager of its moves.

Prize and Eligibility

A \$10,000 prize will be awarded to the winning entrant. The competition is open to all computer systems, except those generated by affiliates of Stanford University. Sorry, no human players allowed.

More Information

The competition website, (<http://games.stanford.edu>), contains further details, including the description of the underlying framework, the game description language, and the programmatic interfaces necessary to play the games.