LING2: A System for Induction

António G. Portela

Departamento de Engenharia Mecanica
Instituto Superior Técnico
Av. Rovisco Pais
1096 Lisboa Codex, Portugal

Abstract

F-LING is a formal language based on the concept of holonic sentence, a 4-word declaration where the first word symbolises the semantic environment.

The holonic sentence, preceded by 2 words (the symbols of the initial sentence and of the holonic sentence), is stored in a repository. The constancy of the number of words (2 + 4) in each sentence is very helpful.

The main purpose of this paper is to show how induction can be done in F-LING.

1 Introduction

F-LING is a formal language based on the concept of holonic sentence, (HOL), a 4-word declaration which is considered the simplest structure conveying some information. The general form of an holonic phrase is:

$$HOL_z \quad Wrd_0 \quad Wrd_1 \quad Wrd_2 \quad Wrd_3$$

The holonic sentence, HOL_z, informs:

- There is a nexus between the words Wrd_1 and Wrd_3;
- The "nexus" is named Wrd_2 and, in general, there are many other pairs having the same nexus;
- The triad Wrd_1, Wrd_2, Wrd_3 holds in the semantic context or referential, symbolised by Wrd_0.

For example:

HOL_z Jack Mary is-maried-to John

- HOL_z is the symbol of the holonic sentence;
- Jack has made the declaration, "Mary is-maried-to John";
- "Jack" is the symbol given to the semantic context;
- There is a nexus between "Mary" and "John";
- ("Mary", "John") is an ordered pair;
- "is-maried-to" is the word that represents the nexus.

F-DIALOG [Portela 88] is an liguistic tool that enables the dialog with F-LING, using natural (idiomatic) language instead of a formal one. The main functions of F-DIALOG are:

1. *Elimination of ambiguities.* A dialog with the user takes place and only terminates when F-DIALOG considers an ambiguous sentence clarified. The dialog may not take place if the sentence is not ambiguous, from F-DIALOG's point of view.

2. *Separation of the unambiguous (clarified) sentence in one or more holonic sentences.* This operation is performed without loss of information and it is possible to reconstruct a sentence with the same "information content" of the original clarified sentence.

3. *Placing all the sentences in a repository (REP).* The standard form of an holonic sentences in the repository is:

$$Sent_z \quad Hol_z \quad Wrd_0 \quad Wrd_1 \quad Wrd_2 \quad Wrd_3$$

   where $Sent_z$ is the symbol of the "clarified ideomatic sentence" and Hol_z is the symbol of one of the holonic sentences derived from $Sent_z$ by F-DIALOG.

4. *Creating a dictionary (DIC), with all the words given to or created by F-DIALOG.*

5. *Revise REP and DIC whenever necessary.* Both, REP and DIC are "infinitely" expansible.

6. *Definition of referent.* F-DIALOG "asks" how sure Wrd_0 is about the statement made and, if possible, a value in an appropriate credibility-scale is requested. F-DIALOG asks also a credibility-value regarding Wrd_0 (the semantic context).

1.1 Example 1

1. Suppose that one states that "the coordinates of point P are 235, 29, and 4"

   After clarification of this sentence by F-DIALOG one obtains:

   $Sent_A$: "the origin of the space is origin-0, 235, 29, and 4 correspond, in origin-0, to the second, third and first coordinates, respectively".

   After de-articulation of $Sent_A$, the following holonic sentences are obtained:

   $Sent_A$ Hol_1 origin-0 4 is first-coordinate
   $Sent_A$ Hol_2 origin-0 235 is second-coordinate
   $Sent_A$ Hol_3 origin-0 29 is third-coordinate

2. Suppose that one states that "origin-0 is an origin for a 3 dimensional linear ortogonal space (3-D-Space) and all coordinates are given in meters".

The de-articulation of this sentence gives:

$Sent_B$ Hol_4 3-D-Space origin-0 is a-origin
$Sent_B$ Hol_5 origin-0 meter is 1-coordinate-unit
$Sent_B$ Hol_6 origin-0 meter is 2-coordinate-unit
$Sent_B$ Hol_7 origin-0 meter is 3-coordinate-unit

1.2 Example 2

$Sent_{J1}$ "John said that Peter is tall"
$Sent_{A4}$ "Don said that Peter is small"
$Sent_{Pl}$ "Anthropologist said that John is a pygmy"
$Sent_{Q3}$ "*Someone* said Don plays basketball"

Will be converted as follows:

$Sent_{J1}$ "John-point-of-view Peter is tall
$Sent_{A4}$ "Don-point-of-view Peter is small
$Sent_{Pl}$ "Antropologist-statement John is a pygmy
$Sent_{Q3}$ "X? Don plays basketball

1.3 Example 3

$Sent_{X2}$ "Mary said if John is gambling then his spending is great"

Converts to:

$Sent_{X2}$ Hol_{24} Mary-statem. John is gambling
80 0 0 0
$Sent_{X2}$ Hol_{25} Mary-statem. John-spend. is great
80 0 0 0
$Sent_{X2}$ Hol_{26} Mary-statem. Hol_{24} implies Hol_{25}
80 0 75 0

Obs.: "Mary" is highly credible (80). The implication (Hol_{26}) is judged by "Mary" as highly credible (75). But the credible-value of both Hol_{24} and Hol_{25} is not given by Mary in $Sent_{X2}$.

2 Repositories

All information, in holonic sentences is stored in the repository (REP) and all words employed are listed in the dictionary (DIC), both of them can increase and be revised indefinitely. Several types of sub-repositories can be extracted from REP, selecting in REP all the holonic sentences, HOLs, that possesse a given set of "words" in a given set of "places" in the HOL. The normal form of an holonic frase in REP is:

<Sent> <Hol> <Word.C> <Word.D> <Word.V> <Word.R>

where:

- <Sent> is a reference to the original sentence;
- <Hol> is the symbol of the holonic sentence;
- <Word.C> is the semantic context;
- <Word.D> is an element of the domain of the "nexus";
- <Word.V> is the designation of the "nexus";
- <Word.R> is an element of the range of the "nexus".

There are four types of objects in a repository:

1. *Type REP-X*: There are 4 distinct cases, depending on the meaning of X (C, D, V, R):
   - REP-C: the semantic contexts <Word.C>;
   - REP-D: the domain elements <Word.D>;
   - REP-V: the verbs <Word.V>;
   - REP-R: the range elements <Word.R>.
2. *Type REP-X-Y*: There are 6 distinct cases: REP-C-D, REP-C-V, REP-C-R, REP-D-V, REP-D-R, REP-V-R. Note that REP-X-Y = REP-Y-X and REP-X-X are meaningless.
3. *Type REP-X-Y-Z*: REP-C-D-V, REP-C-D-R, REP-C-V-R, REP-D-V-R.
4. *Type REP-X-Y-Z-W*: There is at most one case that corresponds to a well defined HOL but this type of sub-repository may be an empty set.

Some sub-repositories are used often and should be "constructed" before the solution to a given problem begins, for example:

- REP-C (with C = C_1 ... C_n)

  This type is usefull in problems, where only the semantic contexts C_1 ... C_n are of interest. It may be more efficient to create, to start with, a sub-repository of the type REP-C.

- REP-C-V (with V = V_v and C = C_s, C_t)

  This type of sub-repository may be useful to study the verb V_v in the semantic contexts C_s and C_t.

- REP-C-V-R (with C = C_c, V = V_v and R = R_a ... R_q)

  To supply the set d_v (domain of V_v) with a topologic structure, this type of sub-repositories are well suited.

All these sub-repositories, may configurate a "list of addreses" of the holonic sentences stored in REP. The amount of memory used is much reduced but the access to REP is inderect.

3 Primitive and Derived Concepts

The primitive concepts in F-LING are the holonic sentences HOL and the words WORD, which are stored respectively in the repository and in the dictionary. Concepts like set, multiple, set element, etc. are derived concepts. A presentation of some typical derived concepts is appropriate:

1. *Sets.* All sub-repositories presented in Section 2 are subsets of REP, and REP is the "universal-set" of HOL. The empty set and REP are sets. The dictionary DIC is the "universal-set" of words (WORD) and there is a word "is-member-of" (a verb) from which a very important type of HOL can be created, namely

   $$Sent_f \ Hol_h \ D_x \text{ is-a-member-of } C_c$$

   Building REP based on "D_x is-member-of C_c", all D_x retained constitute a set (eventualy an empty set) that is a subset of the dictionary DIC. The symbol for a set is "SET" and the above referenced set of words can be represented as

   $$\text{SET } (C_c = \{D_x : D_n, ..., D_w\}$$
   or
   $$= \{D_x : Sent_f \ Hol_h \ D_x \text{ is-a-member-of } C_C\}$$

   the first mode is a simple listing of words, the second is based on the operator "is-a-member-of" that enables the construction of the set.

2. *Relations.* A relation can be exhaustively represented by the set of all pairs (nexus between 2 words) that satisfy a certain verb V_v, (sub-repository of the type REP-V_v). The list may include nexus in various semantic-environments.

   A problem arises when a relation REL has infinite nexus (see Section 5) and the exhaustive representation is not feasible. The strict-order-relation (>=) will be used as an example.

   Let SET = {a, b, ..., n} be a set and ">=" a verb. In REP there are the following holonic sentences:

   H_{ab} b >= a
   H_{ac} c >= a
   ......
   H_{an} n >= a
   H_{bc} c >= b
   H_{bd} d >= b
   H_{bn} n >= b
   H_{mn} n >= m

   "Rules" are not explicitly declared.

3. *Multiples.* Let SET>=C be an ordered set of sets SET_x

   $$\text{SET>=C} = [\{SET_a \ ... \ SET_n\} >=]$$

   We define MLT_x of SET>=C as any ordered set of elements of SET_x and x : [{a ... n} >=]. A multiple (MULT) is an element of the cartesean product of SET_x of SET>=C.

4. *Connectives.* The connective is the general designation of a class of relations used to introduce an algebraic structure in a set. In general a Cartesian product is formed between a set and another set and a relation is defined, as the Cartesian product being the domain and the set the range, for example, max, min, sup, inf, +, *, union, conjuction etc.

   A very simple example, introducing the connective "max", follows: The given set is {0, 1}.

H_1 S-boole 0,0 is-an-ordered-pair-of Cart. Prod.
H_2 S-boole 0,1 is-an-ordered-pair-of Cart. Prod.
H_3 S-boole 1,0 is-an-ordered-pair-of Cart. Prod.
H_4 S-boole 1,1 is-an-ordered-pair-of Cart. Prod.
H_5 S-boole 0,0 Max 0
H_6 S-boole 0,1 Max 1
H_7 S-boole 1,0 Max 1
H_8 S-boole 1,1 Max 1

5. *Functional Relations.* The domain and or the range represent relations, for example:

   Functional: D=relation V=Funct. C=real
   Transform: D=relation V=Transf. C=relation
   Distribution: D=set V=Distr. C=relation

6. *Operators.* The standard form of a relation REL is a "table" of all the nexus which are members of REL (D-domain V-verb R-range). Another form of describing REL is by means of operators:

   direct: (OP-REL), OP-REL { D } >>> R
   converse: (OP-REL!), OP-REL { R } >>> D
   If these operators are given then the above referenced "table", is not needed. The inverse relations are usefull when introducing a topological structure in the domain set of REL.

4 Translation

The basic concept of HOL is described by an ordered set of four words (C D V R). The first element represents the semantic environment or context, the remaining triad (D V R) is a declaration. If the context or semantic environment C is substituted by c-, then the triad (D V R) is converted or "translated" into (d- v- r-), and conveys the same information:

(C D V R) == (c d- v- r-)

A point P in a 2-D plane is spatialy described by two reals (coordinates of point P). Let P_0 be a point with coordinates 20 and 300. This information would be incomplete, without an explicit mention to the "origin" or referential used, for example,

H_1 S-originA 20 is-coordinate1-of P_0
H_2 S-originA 300 is-coordinate2-of P_0

Two types of translation are here presented:

- *Literal or word-to-word translation.*

  H_{11} S-originA translates-to S-originB
  H_{12} 20 translates-to 40
  H_{13} 300 translates-to 150

  H_1 and H_2 are "translated" as follows:

  H_{1-t} S-originB 40 is-coordinate1-of P
  H_{2-t} S-originB 150 is-coordinate2-of P

- *Semantic or sentence-to-sentence translation.*

  H_{s1} H_1 translates-to H_{1-t}
  H_{s2} H_2 translates-to H_{2-t}

In general, these two methods yield diferent results, but are equivalent, in formal languages. The inverse translation does not reproduce the initial holonic sentences, in general. Again the existence of an inverse operation in formal languages is important.

5 Infinite Sets (continuum)

Finite processors are unable to deal with the continuum and the usual method consists in the conversion of the continuum into a discrete form. A table of circular functions is a discrete image of the continuous functions it represents. The output of a program that computes a circular function is also discrete. The choice between using more memory or consuming more time in computation is a practical problem but not a conceptual one.

The "coarsenness" of a table can be reduced by interpolators. In the following, the method of obtaining a discrete image of the continuum will be considered adequate to the type of problem to be solved and this assumption is applicable both to tables and operators (hard or soft).

6 Credibility

Credibility is fundamental in F-LING. This is discussed in [Bourbaki 57]. Here we will just give a sumarised reference.

To a tetrad of words corresponds a tetrad of reals belonging to the set [-1, 1]. The absolute meassures the credibility are 0 (no crebility); 1 (total credibility). Negative numbers stand for "negation", for example, -1 means totaly false. A set of functions over the set of tetrads yields the credibility of the set.

7 Deduction and Induction

7.1 First example

- *Deduction.* The verb "is-maried-to" is declared as a symmetrical relation i.e. if (X is-maried-to Y) then (Y is-maried-to X) or from H_1 X is-maried-to Y we may infer H_2 Y is-maried-to X. Thus, the declaration: (A is-maried-to B) and the symmetry of the "is-maried-to" relation, permits to infer: (B is-maried-to A). This is a typical deduction, after it is completed, the total information in REP has not increased.

- *Induction.* A process of induction applied to the study of properties of the verb "is-maried-to" may be described by the following steps:

  1. *Data retrieval regarding "is-maried-to".* Find in REP, all the HOLs where "is-maried-to" is the verb. Suppose the following list, was obtained:
     A_1 is-maried-to B_6,
     A_2 is-maried-to B_5,
     A_4 is-maried-to B_3,
     B_5 is-maried-to A_2,
     B_7 is-maried-to A_3,
     B_3 is-maried-to A_4.
  2. *Looking for symmetries in list.* The program having the capability, and the initiative, to test various concepts, namely the symmetry of relations, produces a list of all pairs of symmetric HOLs contained in the previous list:
     ((A_2 is-maried-to B_5) and (B_5 is-maried-to A_2)),
     ((A_4 is-maried-to B_3) and (B_3 is-maried-to A_4)).

  3. *Conjecture formation.* The 2 cases of symmetry observed and the lack of existence of counter-examples, may be considered enough to form the following conjecture: *"is-maried-to" is symmetric.*
  4. *Testing the conjecture.* From now on, every new HOL containing "is-maried-to" as a verb will be tested for symmetry. If some new cases are detected and no counter-examples were found, then the conjecture is converted in a "finding" and "is-maried-to" is declared a symmetrical verb and a new rule is included in REP.
  5. *Consolidation.* After step 4, the symmetry of "is-maried-to" is now a rule and this rule can be used in deductive processes whenever needed.

Eventualy, new findings may imply the revision of the rule. The development of the theme, inductive processes, follows.

7.2 Endowment of capabilities

All living things (agents) inherit and or aquire "capabilities" essencial to perform tasks. The construction of a computer language adapted to a certain species of agents must take in account these capabilities.

It would be senseless to provide the computer language with means to describe 3-dimensional objects, if the agents of the species we are trying to simulate are blind and deaf. But if a given species is endowed with the faculty of sensing the magnetic inclination, the language should provide concepts like: "up and down", "distance and velocity" and the necessary rules to process the information aquired and appropriate to an one-dimensional linear space, with the usual topology and the usual operators: order-relation, connectives (+, *), scalar derivatives, etc.

Humans are endowed with so many innate capabilities and further developed by learning, that even a modern natural language and all formal languages so far developed are insufficient to adequately simulate a human being. The method usualy adopted is to provide a basic set of formal procedures and functions, for example,

- A finite set of characteres with a strict order relation and two connectives;
- A finite set of numbers and two not closed connectives (+, *), a mimic of the integer numbers;
- A set of numbers, emulating imperfectly the reals;

- A set of pre-defined functions, on the reals;
- Input and output devices.

The basic set of procedures and rules can be increased, including other specialised functions and or structured sets. This final set (basic and specialised) represents the "inherited capabilities" of the artifact (processor) and emulates a certain "species" real or imagined.

7.3 Paradigms and Reconnoitre

Formal algebic and topological structures and concepts were developed mostly because they serve as models or paradigms. Natural and formal languages are essentially an antropomorphic artifact, adjusted to and inspired in the capabilities and atributes of the human species.

Paradigms can be confronted or compared with the atributes of the real or imaginary objects and beings. For example,

- counting == cardinal of set ; natural numbers;
- precedence (in a tribe or table) == order-relation;
- stereocopic vision == distance, proximity;
- cup to contain a liquid == concave (convex);
- straight line, vertical, horizontal == linear spaces;
- biologic clocks, sideral time == time.

The atributes of a paradigm are invariant and all theorems and inferences are also invariably valid. If an "object" conforms well with a paradigm then "all" atributes of the paradigm can be applied to the "object". The pair (paradigm, to-reconnoitre-it) represents a faculty, inherited or learned, by the artifact and the two are usefull only as a pair.

When we discussed induction, step 2, looking for symmetries, is only feasable if the processor is endowed with the pair (paradigm symmetry, procedure to reconnoitre symmetry).

7.4 Conjectures propounding

The artifact (processor), at each stage of evolution, possesses a set of pairs (Pairset), of the form:

(Paradigm, Reconnoitre procedure)
or
(P, RP)

In REP, there are holonic sentences that can be compared with a Paradigm (P_k) by means of a Reconnoitre procedure (RP_k), the result is given by an element of the range of (RP_k). Usualy the range (RP_k) is the set {Yes, No}, but other sets can be considered, for example, {Yes, Undecided, No}. Systhematicaly or randomly, the processor takes the iniciative to proceed with a reconnaissance of a pair (P_k, RP_k), belonging to the set (Pairset).

The successes and failures are remembered and a decision function (FC) is provided, to enable the processor to propound the paradigm P_k as a conjecture.

7.5 Conjectures testing

This is mainly designed to test the conjecture, giving time to confirm or refute the conjecture. A functor (FT) is provided to perform this function.

7.6 Another example

This example presents another aspect of induction and is based on simple generalization or exsention of the domain of application of a rule. Suppose that John sees a dog (B) bite Jack and declares:

H_{23} John dog (B) bites Jack.

The functor (FCT) may suggest "rules", like: "dog (B) bites humans", or "all dogs bite Jack", or "all dogs bite all humans", etc.

The generalisation is a "risk avoider" posture, that has saved many human lives. The functor declares universally what as been observed once. Eventualy, later, the rule may be disproved so many times, that dogs have to be classified in bad and good dogs and the domain of the rule is reduced to bad dogs.

7.7 Concluding remarks

The final result of the induction is the proclamation of a "new rule" by the artifact (processor). This rule was not given by an external source, namely by the human operator.

A regular application of the inductive method, using all the pairs (P_k, RP_k) members of Srpr, will generate eventualy more "rules", following the introduction of new holonic sentences in the repository. If the repository is "frosen", that is no "new HOLs" are intruced in REP, then, after some time, the production of "new rules" stops.

Human progress is a long story of disproved conjectures: "The Earth is flat and is in the center of the world", "the space dimension is 3", "later 4", "now more then 4", "the atom is indivisible", etc. The statement "all rules are true" is acceptable, provided, the domain of the rule may be an empty set!

8 Conclusion

Humans do not create "ab nihil", the same applies to human artifacts and to F-LING in particular.

- F-LING starts with a Repository REP-0, a Diccionary DIC-0, a set of hardware capacities HW_0.

  In REP-0 are stored the initial holonic sentences (HOLs):

  - HOLs describing Pairset-0, a set of pairs (Paradigm, Reconnoitre-procedures)
  - HOLs describing rules (functions, procedures etc.).

  DIC-0 contains some initial words.

- The development of F-LING results of the contact with the "outside world", by means of the input/output devices. The information received, after interpretation, is stored in REP. By induction, new rules are created, increasing the available information. If needed, declarations not contained explicitly in REP can be deducted from the avaiable data, using the rules thar are in force. F-LING can also be developed by introducing in Pairset-0 new pairs (P-, RP-), that will increase the induction capacity of the processor and new rules will be uncovered.

9 Acknowledgements

I am gratefull to João P. Martins, Ernesto Morgado and Helder Coelho for their comments but the responsibility for all errors and misconceptions the paper may contain is mine.

10 References

Bourbaki N., *Actualites Scientifiques et Industrielles*, 1957.

Eytan M., "Fuzzy Sets: A Topos-Logical Point of View", *Fuzzy Sets and Systems, 5*, 47-67, 1981.

Portela A.G., Dialogos, *Inteligencia Artificial 0*, No.2, 1988.

Russell B., *Introduction to Mathematical Philosophy*, George Allen & Unwin, Ltd., 1930.

Zadeh L., "Fuzzy Sets", *Information and Control, 8*, 338 - 353, 1965.