

Dorr, B.J. 1993. ***Machine Translation: A View from the Lexicon.***
The MIT Press, Cambridge.

A review by

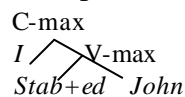
Cecilia Hemming
Department of Languages, University College of Skövde
Swedish National Graduate School of Language Technology, GSLT

Introduction

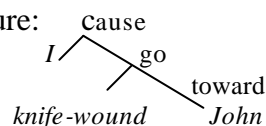
In the book *Machine Translation: A View from the Lexicon*, Bonnie Jean Dorr describes a general solution to handle cross-linguistic divergences¹ within an interlingua (IL) based bidirectional translation system, called UNITRAN. Dorr claims that not only syntactic but also lexicon principles should be parameterized as it allows the system to use two levels of processing: syntactic and lexical-semantic. Both operate on language-independent knowledge that is parameterized to encode language specific information. The same parser and generator are used for all languages (English, Spanish and German in this case). Each language's grammar is automatically produced as a result of precompilation of the parameter settings, mirroring the language specific characteristics. The parameters provide a concise mean to specify the grammar for each language involved and this makes the system easy to design and extend.

Function overview

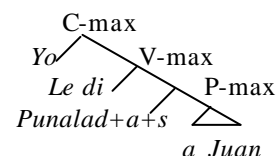
A system input [*I stabbed John*] is first morphologically analysed [*I stab+ed*] then the parser of the syntactic processing module provides a source-language syntactic structure to the lexical-semantic processor:



The lexical-semantic unit processes the structure:



The generator then gives the target-language syntactic structure:



Which finally passes through the morphological synthesiser [*Yo le di puñaladas a Juan*].

¹ According to Dorr a translation divergence arises when the natural translation of a source language results in a very different form in the target language.

The interlingua

The IL used can be said to be an intermediate variant when compared to, on the one hand representations that differs radically from their corresponding syntactic structures, and on the other hand those that do not allow a principled translation mapping because of their language specificity. The representation is based on an extended version of the Lexical Conceptual Structure (LCS) proposed by Jackendoff. The main argument for using the IL is that it allows a representation of surface syntactic distinctions at a level that is independent from the underlying meanings in the source and target languages. All language specific characteristics are specified separately by means of parameter settings. This is a clear advantage compared to both the direct and the transfer translation approach. The former needs specific word-for-word replacements for each language pair. The latter is dependent of intermediate transfer representations for each source and target language. The IL operates on the syntactic as well as the lexical-semantic level. Thanks to the separation of these two knowledge types, constraints can be applied at different representation levels. The mapping between the interlingua and the syntactic structure is realized by a generalized linking routine that defines the mapping between the structures. The routine links for instance a syntactic head X to the logical subject position X' and relates a syntactically external position W to the logical subject position W'.

In the LCS approach the semantic representation is a subset of conceptual structure and includes types like Event and State, which are specialized into *primitives* like GO, STATE, ORIENT, etc. Each action (e.g. GO) and entity (e.g. PERSON) is associated with a representation that is both conceptually plausible and systematically related to a syntactic structure. In the mapping between the interlingua and the syntactic structure there are certain positioning conventions to retain in order to preserve the uniformity, see the definition of the structure below:

$$[T(X') X', \\ ([T(W') W'], \\ [T(Z'_1) Z'_1], \dots [T(Z'_n) Z'_n] \\ [T(Q'_1) Q'_1], \dots [T(Q'_m) Q'_m]])]$$

X' is the logical head, W' the logical subject, Z'_1, ... Z'_n the logical arguments and Z'_1, ... Z'_m are the logical modifiers. An example:

I stabbed John ⇔ *Yo le di puñaladas a Juan* (*I "him" gave knife-wounds to John*)

$$[Event\ CAUSE \\ ([Thing\ I], \\ [Event\ GO_{Poss} \\ ([Thing\ KNIFE-WOUND], \\ [Path\ TOWARD_{Poss} \\ ([Position\ AT\ ([Thing\ KNIFE-WOUND], [Thing\ John])])])])]$$

X in the syntactic structure corresponds to the verb *stabbed*, X' to the logical head CAUSE, T(X') to Event, W' to the logical subject *I*, T(W') to Thing, etc.

Dorr points to the difference of this framework's goal compared to earlier lexical-semantic approaches like that of Schank or Lakoff. They wanted to provide an exhaustive decomposition of word meaning into a finite set of semantic/conceptual primitives defining a

finite collection of necessary and sufficient conditions. The current approach instead aims to provide a systematic relation between the structure of meaning and the structure of language on the surface, without attempting to convey all details of causal relationships involved in the word meaning. That is, as Jackendoff puts it, to define the “typicality conditions”: for instance that the verb *climb* usually incorporates the meaning of *up* or that *X causing Y to die* usually implies that *X killed Y*. In a generative semantics framework, *kill* is presupposed to be derived from a deep structure syntactic representation of *cause to die*. This is not the fact with the current approach (neither with Jackendoff’s), where the representation is constructed on the basis of lexical items’ properties and their potential realizations in the surface syntactic structure.

Morphological Processing

The morphological processing in UNITRAN is taken care of by an extended version of the two-level Kimmo system, originally proposed by Kimmo Koskenniemi. The core idea of this model is that it is reversible: the same lexicon and grammar description can be used both for analyses and generation. All morphological rules are represented as finite-state transducers. The current version includes access to the lexicon during generation, see figure 2. for an overview of the organization.

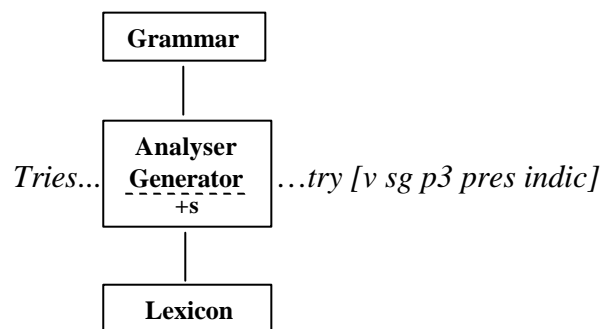


Figure 2. The organization of the modified version of Kimmo, used in UNITRAN.

Syntactic Processing

The syntactic processing module of the system contains the parser, the generator, and GB principles and parameters. In the parsing mode the structure-building operations are executed separately but in parallel with the operations applied by the GB constraint module. The structure builder (based on the Earley-algorithm²) first constructs skeletal phrase structures without information about agreement, abstract case, thematic roles, argument structure, etc. It

² The Earley algorithm uses a dynamic programming approach to implement a parallel top-down search. Dynamic programming in this regard refers to that subtrees, for each constituent in the input, systematically are stored in tables as they are discovered. A stored subtree can then easily be looked up and re-used for all parses calling for that constituent. When complete, the tables contain all subtrees needed to construct the complete tree.

is then the role of the GB module to enforce the conditions of well-formedness (agreement and case filter, etc.) and to add the above mentioned missing information, see figure below.

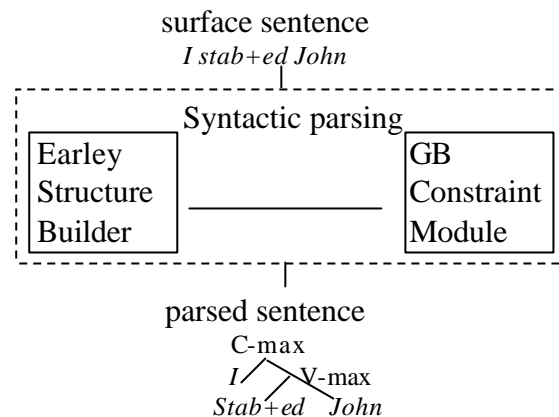


Figure 1. Overview of the syntactic processing module in UNITRAN

The type of translation divergences taken care of in this module is characterized by syntactic properties of the different languages, independently of the actual lexical items used. The syntactic component's parameters are defined in seven separate modules: **X-bar** determines for instance the order of the sentence constituents and also which specifiers and adjuncts that are allowed; **Government** reveals allowed types of government and allowed preposition standing; **Bounding** determines movable specifiers/adjuncts and constraints the distance between trace-antecedent pairs; **Case** ensures that noun phrases are assigned abstract case; **Trace** check for proper government of empty elements, as null-subjects for instance; **Binding** determines co-reference relations among noun phrases; and eventually **q** that assigns the different thematic roles.

Lexical-Semantic Processing

The author means that the lexical-semantic basis for the system handles many divergences between the languages studied and that they can be divided into a small number of *divergence types*. The framework makes use of a linguistically based classification of translation divergence types that can be formally defined and systematically resolved. A divergence is said to occur either when there is an exception to the linking routine between syntactic and lexical-semantic positions or the relation between a lexical semantic type and the syntactic category. To capture these exceptions the lexicon has markers that specify syntactic realization information with lexical items. These markers can be thought of as parameters to the system's LCS component. There are seven such parameters used: :INT, :EXT, :PROMOTE, :DEMOTE, *, :CAT and finally :CONFLATED, all of which is intended to resolve different types of divergences.

The :DEMOTE-parameter, as one example, allows the generalized linking routine (see the section about *interlingua*) to be overridden by relating a logical head to a syntactic adjunct position (the logical head is thus "demoted", placed further down, in the syntactic structure). The English sentence: *I like to eat* is translated into the German sentence *Ich esse gern* (I eat likingly), whereby the English main verb *like* corresponds to the German adjunct *gern*. In the lexicon the root LCS for *gern* is thus associated with the Y-argument through a :DEMOTE-

marker, and this marker forces the German logical head of the composed LCS to be mapped to a syntactic adjunct position. See figure 3. for an overview of the lexical-Semantic processing.

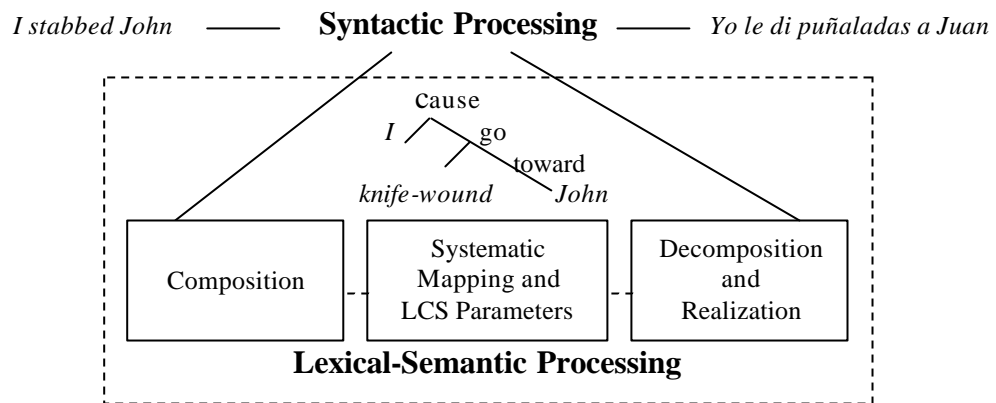


Figure 3. Overview of the lexical-semantic processing module in UNITRAN

Problems to solve

Dorr also points to continuing research in some issues. One problem is that the described version of the UNITRAN system operates on one sentence at the time and thus neither can use discourse information nor situational expectations or domain knowledge in the translation process. The author stresses though that the LCS-based framework is intended to be a supplement rather than a substitution for knowledge-based techniques. A drawback of the system is that the LCS framework lexicon demands a lot of work: it took more than one person-month to define 150 words of the three languages used. As for other systems in this area it is also difficult to realize automatic distinction between prepositions, especially when it comes to spatial orientation. The LCS representation ought to be accordingly extended to handle such distinctions.