An Adaptation of Lexical Conceptual Structure to Multilingual Processing in an Existing Text Understanding System

Bonnie Glover Stalls
Robert S. Belvin
Alfredo R. Arnaiz
Christine A. Montgomery
Robert E. Stumberger

Language Systems, Inc.
{glover,robin,arnaiz,chris,res}@lsi.com

Abstract

The extension of an existing text understanding system to multilingual translation using an interlingual approach prompted the search for a language-independent means of expressing predicates and predicate relations. The approach adopted was Lexical Conceptual Structure (LCS), as put forth by [2, 3] and [1]. LCS incorporates notions of event structure, argument relations, and core meaning into a concise language-independent formalism that preserves structure at the same time that it allows for divergences among languages. The task of installing LCS proved simple and straightforward, given the text understanding system's explicit representation of sentential content and capability of indexing the analyzed constituents of the sentence to the appropriate slots in the LCS.

1 Introduction

Language Systems, Inc. (LSI)'s core automated text understanding technology has been applied over the last three years to multilingual processing in a voice-to-voice translation system that we are developing for the Air Force. Originally designed as an application-independent language analyzer, LSI's Data Base Generation (DBG) system is a flexible, modular system that produces a knowledge representation of the events and entities in a text, incorporating syntactic, semantic, discourse, and other relevant information. The DBG system has been adapted to a number of different applications, including data base update for space event reports [4] and for air activities messages [14], message fusion for radiotelephone traffic [6], data base generation for reports of terrorism in Latin America (MUC-3 and MUC-4)[5, 8] and for the transfer of microelectronics technology (MUC-5) [7], and most recently, the project for which we have developed the multilingual processing capability that we are describing here, machine-aided voice translation [9,10, 11].

2 Multilingual Processing and LSI's DBG System

The Machine-Aided Voice Translation (MAVT) project, now in its second phase of prototype development, is being designed to assist English-speaking Air Force personnel in interacting with speakers of Spanish, Arabic, and Russian. The MAVT testbed consists of three subsystems: speech recognition, language processing, and speech generation. Like the voice-to-voice English → Spanish → English system developed in the first phase of the MAVT project, the system currently under development is a speaker-independent continuous speech translation system, processing query-response interactions in a military domain. As with the previous system, the language processing

functions—understanding, translation, and generation—are performed by LSI's DBG natural language processing system.

Much of the extension of the DBG system for the MAVT project has necessarily focused on multilingual capabilities. In the first phase of the project, the DBG system already had in place a multilingual syntactic parser that was used for Spanish and English. This parser will be used to parse Arabic and Russian as well. DBG produces, as output of the understanding phase of processing, a knowledge representation of the sentence. This knowledge representation is an application-independent data structure of related event and entity frames based on the predicates and arguments of the sentence and derived from an underlying frame-based concept hierarchy. These frames, called *templates* in the DBG system, represent the knowledge contained in a sentence. In translation, this structure serves as the end product of analysis of the source language (hereafter SL) sentence, and the basis for target language (TL) lexical selection and generation processing.

The DBG knowledge representation thus functions as a kind of intermediate or *interlingual* (henceforth, IL) construct. A true *IL* approach does not not rely on direct transfer or direct links between languages but requires a language-independent representation of the data, which can then be used to translate the sentence into any language that the system can handle. The IL approach thus eliminates the need to develop a separate, direct interface between every potential source-target language pair because each language interfaces only with the language-independent IL representation.

From the commencement of the MAVT project, LSI's approach has been *interlingual* in that it assumes that the selection of lexical items in the TL should be based on links to an intermediate structure, the concept hierarchy, rather than on direct or hard links between words in the source and target languages. Thus the words corresponding to the same basic meaning in each language are linked to common concept nodes. These links are present in each event and entity template in the knowledge representation. For some lexical categories, e.g., nouns, this works well. But where cross-category relations are important, as in verbs, which express predicate-argument relations, the lexical properties are much more complex. In a multilingual system, incorporating lexical-semantic information for words associated with a given concept for all of the different languages would greatly increase the complexity of the hierarchy. The concept hierarchy primarily represents meaning relations between concepts of the same category rather than representing the unique properties of the meanings of the individual words associated with those concepts, or the meaning relations and structural requirements of the words in sentences. A great deal of additional syntactic and semantic checking would be needed to ensure the compatibility of a potential TL word with the meaning and structural requirements of the TL sentence.

3 Requirements of an Interlingual Representation

The system we are developing includes a language-independent representation of verbal predicates, as well as prepositions and deverbal nouns. We do not attempt to give an IL representation for nouns, but rather than creating hard links among nouns in different languages, we link them to a point in the concept hierarchy. In this way, we can still translate nouns which do not have an exact equivalent in the TL by checking adjacent nodes in the hierarchy.

We have concentrated our interlingual effort on predicates for two reasons: 1) there are no well-developed theories of noun meaning which are feasible to implement (although see Pustejovsky [13] for a sketch of noun meaning which seems to hold promise for systems such as the one we envision)

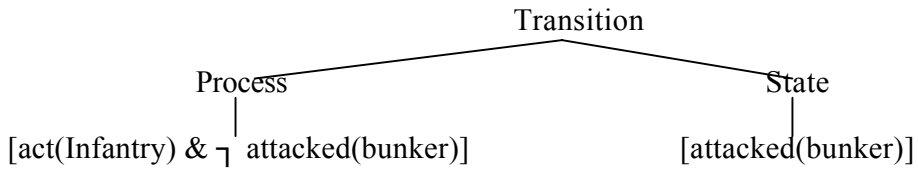
and 2) translation of predicates across multiple languages without the use of a well-defined] component is cumbersome and full of pitfalls.

Several alternatives for the IL representation were considered: what we will call the *0-role approach,* the *event-structure approach,* and the *lexical conceptual structure approach.* The first two do not come close enough to having the ability to uniquely identify a given verbal predicate.

For example, in a sentence like "The infantry attacked the bunker", the θ-role approach would represent the relevant part of the lexical entry for "attack" as (Agent,Patient). In addition, the lexical entry would carry the core predicate relation "attack". Similarly, in "John ate the apple" the relevant part of the lexical entry would contain exactly the same thematic *(θ-)* roles, as well as the core relation "eat". The problem with this is that there is no language-independent way to relate the core meanings of these verbs with their equivalents in other languages. In order to know, e.g., that "attack" corresponds to Spanish "atacar", there must be a hard link between the two verbs. One cannot rely on the θ-roles alone, since there are many verbs which have (Agent,Patient) as their associated roles (we have only mentioned two). One might assume that somewhat better matches could be achieved by enriching the vocabulary of 0-role labels. This is a difficult, perhaps impossible task, since there is still no widely accepted proposal for how large this vocabulary should be, let alone what particular labels it should contain.

The event-structure approach also establishes well-defined classes of verbal predicates, based on event-semantic grounds. However, it again does not come close enough to uniquely identifying verbal predicates. Let us return to the example of "attack" and assume an event structure framework like that outlined in Pustejovsky [13]. Since this verb denotes an *accomplishment* type of event, a sentence like "The infantry attacked the bunker" would have a representation like the following:

(1) Event Structure for *attack*



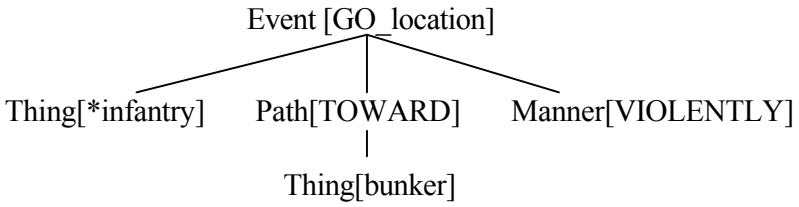
This event structure is paraphrasable as "The infantry performed some action such that there was a change of state wherein the bunker's state previous to the action was non-attacked, and the bunker's state after the action was attacked". This representation captures the fact the infantry was the agent in a deliberate act and the bunker underwent a change of state as a result; however, notice that it relies on the past participle (stative) form of the verb itself to express the core meaning. One should note that these proposals were not devised as a tool for machine translation; hence, our objections are not problems intrinsic to the theories themselves.

In order to have a sufficiently rich IL representation, we need a way of integrating event-structure features, θ-role or argument structure features, and a vocabulary which expresses the core meaning of the predicates in a language-independent way. This is precisely the reason we settled on lexical conceptual structure (LCS) representations; that is, they integrate features of the first two approaches while providing a set of basic conceptual elements which can serve as language-independent metalinguistic building blocks. The theory of LCS we employ is derived from Jackendoff [2, 3] and was first implemented as a component of a translation system by Dorr [1].[1]

[1] [l] defines a mapping of LCS positions into θ-roles, and she addresses the classical problems of divergence in translation, demonstrating how LCS can help to resolve them.

Our system currently represents the sentence "The infantry attacked the bunker" with the partial LCS shown below (the general schema for a primitive predicate in the tree below is PREDICATEJield, which indicates that the primitive predicate PREDICATE is to be interpreted as applying in the semantic field "_field". The main LCS predicates are CAUSE, LET, DO, GO, STAY, BE, ORIENT, and GO-EXT and the field types are circumstantial, existential, identificational, locational, possessive, and temporal):

(2) LCS for *attack*



This representation is paraphrasable as "There is an event in which the thing 'infantry' goes from some unspecified location toward a thing 'bunker' in a violent fashion". The LCS encodes the argument structure (0-role content, selection and subcategorization), as well as the eventive nature of the verb, and the core meaning is sufficiently decomposed to facilitate transfer to other languages. There are, however, some limitations to LCS with regard to lexical selection. Meaning differences that can be captured by means of selectional features on the arguments (e.g., <projectile> in the LCS for "fire" in Section 4) or by simple manner modifiers (VIOLENTLY, above) are easily represented in the LCS. But an important aspect of the verb "attack" which is *not* represented in this structure is the negative effect on the object; otherwise, the event might just be a threat. This characteristic of the verb can be captured in a separate "tier", (Jackendoff's *action tier)* as follows:

(3) Action Tier: [AFF^{-} ([infantry],[bunker])]

This simple predicate indicates "affectedness" with a negative result (the superscripted minus sign on the AFFECT predicate). The combined information of the "thematic tier" and Jackendoff's action tier would yield the richer representation necessary for the lexical selection process.

4 Incorporation of LCS structures into the existing DBG system

DBG is a modular system that analyzes text in progressive stages. The output of each stage of processing is a data structure that then serves as input to the following stage. In the DBG multilingual processing system, there are three stages of SL analysis of a sentence that precede the IL template representation; the IL representation is then followed by three stages of TL generation. The three stages of SL analysis are: a) lexical identification and morphological analysis; b) syntactic parsing; and c) semantic parsing. The three stages of TL generation mirror in part the SL analysis: they are x) lexical selection and semantic parsing; y) syntactic parsing; and z) morphological inflection.

At the heart of processing are the three intermediate translation stages: the SL semantic parse (c, above), the IL templates, and the TL semantic parse (x, above). It is into the data structures output by these modules that we have inserted the LCS. These data structures are essentially of the same type: sets of attribute-value pairs related to other pairs by means of indexing. This kind

(4) English Source Semantic Parse

[The tanks fire at the enemy]

mainpred (1.0)	=	INDEX (1.1)
utt. type (1.1)	=	declarative
predicate (1.1)	=	fire
slcs (1.1)	=	E[CAUSE(T[1.2], E[GO_loc(T[<projectile>] P[TO(T[1.3])])])]
tense (1.1)	=	past
voice (1.1)	=	active
aspect (1.1)	=	perfective
mood (1.1)	=	indicative
modality (1.1)	=	neutral
polarity (1.1)	=	positive
ext-arg (1.1)	=	INDEX (1.2)
obl-arg (1.1)	=	INDEX (1.3)
noun (1.2)	=	tank
person (1.2)	=	3rd
gender (1.2)	=	zero
number (1.2)	=	plural
class (1.2)	=	inanimate
determiner (1.2)	=	the
preposition (1.3)	=	at
prep object (1.3)	=	INDEX (1.4)
noun (1.4)	=	enemy
person (1.4)	=	3rd
gender (1.4)	=	zero
number (1.4)	=	singular
class (1.4)	=	human collective
determiner (1.4)	=	the

E = Event, T = Thing, P = Path

(5) Instantiated Interlingual Templates

EVENT		report [1]
class	:	meta
application	:	mavt-adm
domain	:	domestic mission
corpus	:	example
date	:	13 apr 1994
social situation	:	formal
genre	:	interrogation screening
event	:	[1.1]

EVENT		[1.1]
verb	:	fire
clcs	:	E[CAUSE(T[1.1.1], E[GO_loc(T[<projectile>], P[TO(T[1.1.2])])])]
text status	:	main clause
utterance type	:	assertion
discourse status	:	foreground
class	:	primary
status	:	critical
time	:	precedes [1]
voice	:	active
aspect	:	perfective
modality	:	neutral
polarity	:	positive
[*external*] arg1	:	[1.1.1]
[*oblique*] arg2	:	[1.1.2]

ENTITY		External [1.1.1]
nucleus	:	tank
class	:	military vehicle
number	:	plural
sex	:	zero
person	:	3rd
definiteness	:	definite

ENTITY		Oblique [1.1.2]
nucleus	:	enemy
class	:	human collective
number	:	singular
sex	:	zero
person	:	3rd
definiteness	:	definite

(6) Spanish Target Semantic Parse

[Los tanques le dispararon al enemigo.]

mainpred (1.0)	=	INDEX (1.1)
utt. type (1.1)	=	declarative
predicate (1.1)	=	disparar
slcs (1.1)	=	E[CAUSE(T[1.2], E[GO_loc(T[<projectile>], P[TO(T[1.3])])])]
tense (1.1)	=	past
voice (1.1)	=	active
aspect (1.1)	=	perfective
mood (1.1)	=	indicative
modality (1.1)	=	neutral
polarity (1.1)	=	positive
ext-arg (1.1)	=	INDEX (1.2)
dat-arg (1.1)	=	INDEX (1.3)
noun (1.2)	=	tanque
person (1.2)	=	3rd
gender (1.2)	=	masculine
number (1.2)	=	plural
class (1.2)	=	military vehicle
determiner (1.2)	=	el
preposition (1.3)	=	a
prep object (1.3)	=	INDEX (1.4)
noun (1.4)	=	enemigo
person (1.4)	=	3rd
gender (1.4)	=	masculine
number (1.4)	=	singular
class (1.4)	=	human collective
determiner (1.4)	=	el

of structure allows the system to pass on actual sentence chunks, along with associated features of whatever type, e.g., morphological, semantic, pragmatic, in a homogeneous format.

For example, the sentence "The tanks fired at the enemy" is shown below in the translation phase of processing (SL semantic parse, IL templates, and TL semantic parse), with English as the SL and Spanish as the TL. Note that these structures are shown with the LCS (in DBG, as in Dorr [1], CLCS is the "composed LCS", which contains the complete IL representation, including adjuncts; the SLCS (or "satisfied SLCS"), is an intermediate stage wherein the root LCS from the lexicon is instantiated with the arguments of the sentence being processed). The LCS in this case can be paraphrased as "There was an event in which tanks caused projectiles to go toward the enemy."

Because the LCS is a single construct, the LCS value for a given predicate is expressible as an attribute-value pair, so DBG can easily incorporate it in a meaningful way, indexing it to the appropriate predicates and arguments of the sentence. The LCS itself is coindexed with the predicate. The argument slots in the LCS provide hooks with which to link the LCS to its arguments elsewhere in the DBG data structure. As shown in (4-6), these slots are filled with the indexes to the appropriate arguments (or to the appropriate entity templates in the case of the IL representation). In the English and Spanish semantic parses, the SLCS's for the English verb "fire" and the Spanish verb "disparar" are indexed to the main predicate, which has the index number 1.1. In the IL templates, the CLCS for "fire" is part of the event template for "fire". The indexes for the arguments of the verb are inserted into the argument slots (1.2 and 1.3 in the semantic parses corresponding to the indexes for "the tanks" and "the enemy"; 1.1.1 and 1.1.2 indicating the appropriate entity templates in the IL representation). Additional features associated with the predicate and arguments are also coindexed with them, as is usual in the DBG system. For example, time, voice, aspect, modality, and polarity are features associated with events; number, sex, person, and definiteness are features of entities.

Notice that the LCS is arranged as a set of predicates and arguments, following essentially the same form as a syntactic structure. Thus, *CAUSE* is a predicate of type *Event,* which takes as its first argument a *Thing,* and as its second argument an *Event* or *State.* The *Event* which appears in this example is a *GO* event, occurring in the *locative* field. The *GO*_loc predicate takes a *Thing* as its first argument, and a *Path* as its second argument. Finally the *Path* predicate *TO* takes a single *Thing* argument. Notice the feature <projectile> associated with the first argument of the *GO*_loc event.[2] This feature is linked to a point in the concept hierarchy. Had the sentence contained a direct object, the NP would have to be able to unify with this feature. The presence of this feature, though not overtly realized in the syntax, can aid in lexical selection.

The importance of this last point should not be underestimated. Although LCSs allow for a much finer-grained distinction among predicates than the θ-role or event-structure approaches, the LCS entries of a fairly substantial number of predicates will still not be unique unless these features are considered. In fact, the LCS for the verb "fire" is the same as for "throw", save for the feature <projectile/weapon> associated with the first argument of *GO*_loc.

The incorporation of LCS into the translation data structures represents a major improvement over DBG's previous IL representation. The previous representation required putting into the event templates a diverse set of features, including sub categorization information, 0-roles, verbal class

[2] This is a slight over-simplification, in that <projectile> is really treated as standing in a metonymic relation with <weapon>. See also the analysis in [12] of *shoot* and *fire* as complex events in which the weapon functions as intermediary.

information, selectional restrictions, and links into the concept hierarchy. These features made up a kind of laundry list designed to restrict lexical selection in the TL to verbs that will convey the correct meaning and fit into an appropriate structure. Various options were available in case the exact specifications were not met.

Because the LCS is a complex expression that actually encodes lexical-semantic structure, we do not need to recreate branching predicate-argument structures from scratch. A one-dimensional meaning representation, even with many qualifying features, is inadequate in this regard. With the LCS, we can match the predicate-argument structures of trees analyzed at the instantiated IL template stage with the specified predicate-argument structures of lexical items in the TL and at the same time more precisely define the lexical item. In the DBG system, the CLCS is not only used to match predicates in the TL for the purpose of lexical selection, but it also helps to drive the construction of the semantic parse of the TL sentence, which is the first stage in generation. Overall, the LCS encodes enough structure to allow for a relatively straightforward mapping to or from syntax, but is flexible enough so as not to force the source language's syntax onto the form of the target translation.

One way of looking at the problem is as follows: the information required to do translation falls into two very broad categories: that which needs to be structured or ordered in a particular way, and that which does not. For example, θ-roles or argument structure, no matter what one's position on argument structure, need at least some minimal ordering information, and, in many theories, need much more. On the other hand, the Φ-features associated with a given NP, e.g., number and gender, do not need to be ordered in any particular way.

As we have described above, the LCS supplies an efficient matrix into which the different parts of the DBG representation can be mapped in order to build a structure that can serve as the basis for generation. However, the LCS does not encode all of the information which must be considered when generating. For example, as Dorr [1, p.319] recognizes, tense and aspect information is crucial in translation but is not strictly speaking part of lexical-semantic knowledge. Discourse and pragmatic information are also critical to the analysis and translation of text, for example in resolving extra-sentential reference. These features of the sentence, along with various other semantic, discourse, and pragmatic features, are carried along from the source language text analysis stage through to generation. At the target language semantic parse stage (the first stage of generation), DBG has access to all of these SL features of the text in addition to parameterized information about the TL. Therefore, these other features, in addition to the LCS, can be brought to bear on the composition of the TL structure.

5 Conclusion

DBG was originally developed to analyze and extract information from text to create complex data records. It is able to make explicit the relations between constituents and to associate features with the precise constituents to which they apply. These capabilities are extremely valuable in machine translation and complement the more tightly-structured predicate-argument relations encoded within the LCS. We were therefore able to incorporate LCS into our existing DBG system with minimal stress to the system. In turn, LCS provided the missing link required to move beyond analysis into generation, thus transforming DBG into a genuine interlingual translation system.

References

[1] B.J. Dorr. *Machine Translation: A View from the Lexicon.* MIT Press, 1993.

[2] R. Jackendoff. *Semantics and Cognition.* MIT Press, 1983.

[3] R. Jackendoff. *Semantic Structures.* MIT Press, 1990.

[4] C. Montgomery, B. Glover, J. Kuhns, and J. Burge. Automated data base generation. Technical Report RADC-TR-84-146, Rome Air Development Center, 1984.

[5] C. Montgomery, B.G. Stalls, R. Stumberger, and R. Belvin. Description of the DBG system as used for MUC-3. In *Proceedings of the Third Message Understanding Conference,* pages 171-177. DARPA, 1991.

[6] C. Montgomery, B.G. Stalls, R. Stumberger, R. Belvin, and H. Holmback. Message fusion. Technical report, Ballistic Research Laboratory, 1989.

[7] C. Montgomery, B.G. Stalls, R. Stumberger, R. Belvin, N. Li, and S. Hirsh. Description of the DBG system as used for MUC-5. In *Proceedings of the Fifth Message Understanding Conference,* pages 121-135. ARPA, 1993.

[8] C. Montgomery, B.G. Stalls, R. Stumberger, R. Belvin, N. Li, S. Hirsh, and A. Arnaiz. Description of the DBG system as used for MUC-4. In *Proceedings of the Fourth Message Understanding Conference,* pages 197-206. DARPA, 1992.

[9] C. Montgomery, B.G. Stalls, R. Stumberger, N. Li, R. Belvin, and A. Arnaiz. Machine-aided voice translation. In *Proceedings of the 1992 Machine Translation Evaluation Workshop,* to appear.

[10] C. Montgomery, B.G. Stalls, R. Stumberger, N. Li, R. Belvin, A. Arnaiz, P. Shinn, A. DeCesare, and R. Farmer. Machine-aided voice translation. Technical Report Contract No. F31602-90-C-0058/Report No. LSI R93-01, Rome Laboratory/IRAA, 1992.

[11] C. Montgomery, B.G. Stalls, R.E. Stumberger, N. Li, S. Walter, R. Belvin, and A. Arnaiz. Machine-aided voice translation. In *Information Management Collection Processing & Distribution, Dual-Use Technologies & Applications Conference,* pages 96-101. IEEE, 1993.

[12] M. Palmer. General lexical representation for an effect predicate. In *Lexical Semantics and Knowledge Representation, Proceedings of a Workshop Sponsored by the Special Interest Group on the Lexicon.* Association for Computational Linguistics, 1991.

[13] J. Pustejovsky. The generative lexicon. *Computational Linguistics,* 17(4), 1991.

[14] B.G. Stalls, R. Stumberger, and C. Montgomery. Long range air data base generator. Technical report, Rome Air Development Center, 1990.