An Environment for Quick Ramp-Up Multi-Lingual Authoring

Tod Allman
University of TX, Arlington
701 S. Nedderman Drive
Arlington, TX 76019
TodAllman@aol.com

Stephen Beale
University of Maryland, BC
1000 Hilltop Circle
Baltimore, MD 21250
sbeale@umbc.edu

Abstract

The driving force behind controlled language document authoring systems has been the desire to bypass the knowledge-intensive (and thus time-intensive) and error-prone stage of analyzing the source text. More accurate and deeper analysis of source texts at a lower acquisition cost is possible if the vocabulary and syntax of the input text are kept as simple as possible. The multi-lingual translation system described in this paper capitalizes on this methodology and improves on it in several ways. Our system includes an easy to use machine-assisted semantic analyzer, which automatically produces syntactic and semantic analyses that can be edited by the document author. On the generation side, we provide a "quick ramp-up" grammar acquisition environment, along with a very convenient and novel "visual grammar" interface.

This paper will describe this complete environment for multi-lingual document authoring. The following four aspects of the system will be discussed:

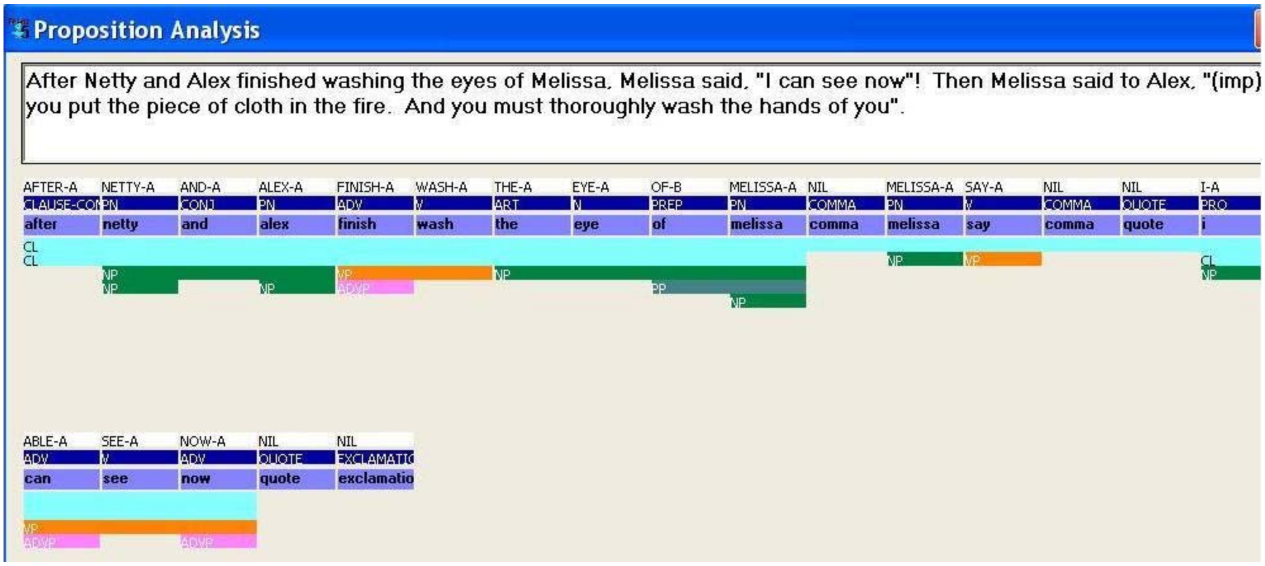
1. The machine-assisted semantic analyzer.
2. The controlled language required by the semantic analyzer.
3. The methodology for quick ramp-up grammar and lexicon acquisition.
4. The multi-lingual text generator and its interface.

This translation system has been used to generate high-quality medical texts in Korean and English. In addition, a large corpus of Biblical texts has been semantically analyzed, with high-quality translations into Korean, English and several minority languages completed. We will report on the native-speaker evaluations of these translations. The "grammar start-up" methodology implemented in this project, along with the convenient visual grammar interface have significantly reduced the knowledge acquisition time needed to produce quality translations in a new language. The machine-aided semantic analyzer, combined with a natural controlled source language has made it possible to produce large quantities of semantically analyzed (and text generation-ready) source texts at a relatively low cost.

1.0 Machine-aided Semantic Analysis

1.1 The analysis environment

The first priority of a document authoring system must be to provide a convenient interface for authors to input text, which must subsequently be analyzed in such a way as to maximize the chances for quality translation into the target languages. We accomplish this goal through the interface shown in Figure 1 below.



After Netty and Alex finished washing the eyes of Melissa, Melissa said, "I can see now"! Then Melissa said to Alex, "(imp) you put the piece of cloth in the fire. And you must thoroughly wash the hands of you".

AFTER-A	NETTY-A	AND-A	ALEX-A	FINISH-A	WASH-A	THE-A	EYE-A	OF-B	MELISSA-A	NIL	MELISSA-A	SAY-A	NIL	NIL	I-A
CLAUSE-CON	PN	CONJ	PN	ADV	V	ART	N	PREP	PN	COMMA	PN	V	COMMA	QUOTE	PRO
after	netty	and	alex	finish	wash	the	eye	of	melissa	comma	melissa	say	comma	quote	i

ABLE-A	SEE-A	NOW-A	NIL	NIL
ADV	V	ADV	QUOTE	EXCLAMATIO
can	see	now	quote	exclamatio

Figure 1: The machine-assisted semantic-analysis interface

The author inputs the text using the controlled English that we will describe below. The machine-assisted analysis program then displays the results of its initial analysis. Each input word is analyzed as follows (from top-to-bottom in Figure 1):

- word sense (semantics)
- part of speech
- root/citation form
- syntactic dependencies (indicated by color bars)

A simple morphological analyzer is used to find the root form and part of speech of each input word. The syntactic analysis is visually displayed using colored bars. Part-of-speech disambiguation and syntactic analysis are performed using a simplified version of the analysis system described in (Beale, submitted). Word sense disambiguation is currently accomplished by choosing the sense most commonly used for the root word (in its currently displayed part of speech) in all previously analyzed texts.

The interface provides for easy editing of each of the four types of analysis. The root word, part of speech and word sense can all be changed simply by clicking on the appropriate box and selecting a different choice. If a new root form is desired that is not in the list of possible choices, a dialog box pops up that enables the user to enter the new root and identify the appropriate suffixes for the word entered in the text box. The expected case frame for verbs can also be entered. This information is then stored. After changing any of the syntactically related information, the syntactic analysis is automatically updated. New word senses can be added in a similar manner.

The opportunity to add new information to the sources of knowledge used for analysis has been allowed up to this point because of the close cooperation between the text authors and the developers of the target language grammars. For our applications, we expect this cooperation to continue. We also expect that other document authoring applications will benefit from the flexibility that such cooperation affords. We plan on adding a mechanism that will permit the target language grammar and lexicon developers to quickly identify and add target language realizations for any semantic inputs that were added by the document authors. However, there are certainly applications for which the ability to add word senses would not be appropriate and can be restricted.

The syntactic analysis proposed by the system and represented by the colored bars can also be easily changed. The most common change concerns the location of attachment sites, especially for prepositional phrases. Phrase attachments can be moved by clicking on the phrase and dragging it to a new attachment point. The starting and ending points of any phrase can be easily changed, and as a last measure, any phrase can be deleted and new ones added.

In practical terms, once the few special features of the controlled language are learned, texts can be input by the document author and subsequently automatically analyzed by the system with almost no need for the author to post-edit any syntactic analysis, except for PP attachments (which default to the nearest possible attachment site, unless the verb explicitly expects it in its case frame). The main task for the document author is to check that the word senses are correct. Most words have only one word sense. The user quickly learns which common polysemous words, such as "of" (see below), must be handled on a regular basis. Figure 2 shows an example of manual sense disambiguation for the word "wash" as used in the example sentence above.

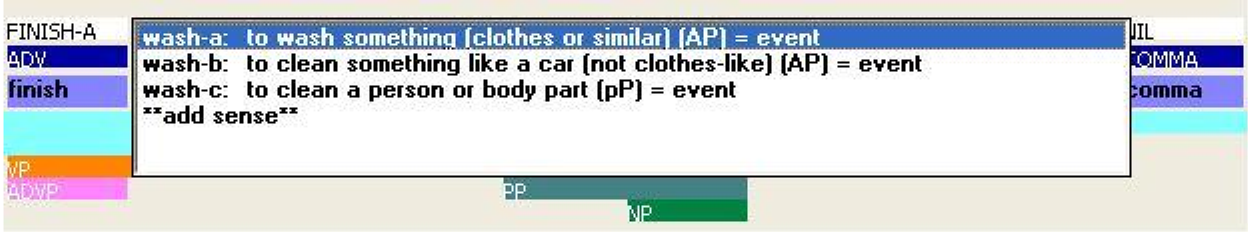

FINISH-A
ADV
finish
VP
ADVP
wash-a: to wash something [clothes or similar] (AP) = event
wash-b: to clean something like a car [not clothes-like] (AP) = event
wash-c: to clean a person or body part (pP) = event
add sense
IL
COMMA
comma
PP
NP

Figure 2: An example of semantic ambiguity. The analyzer chooses a sense, which can be changed by author by clicking on the chosen word sense.

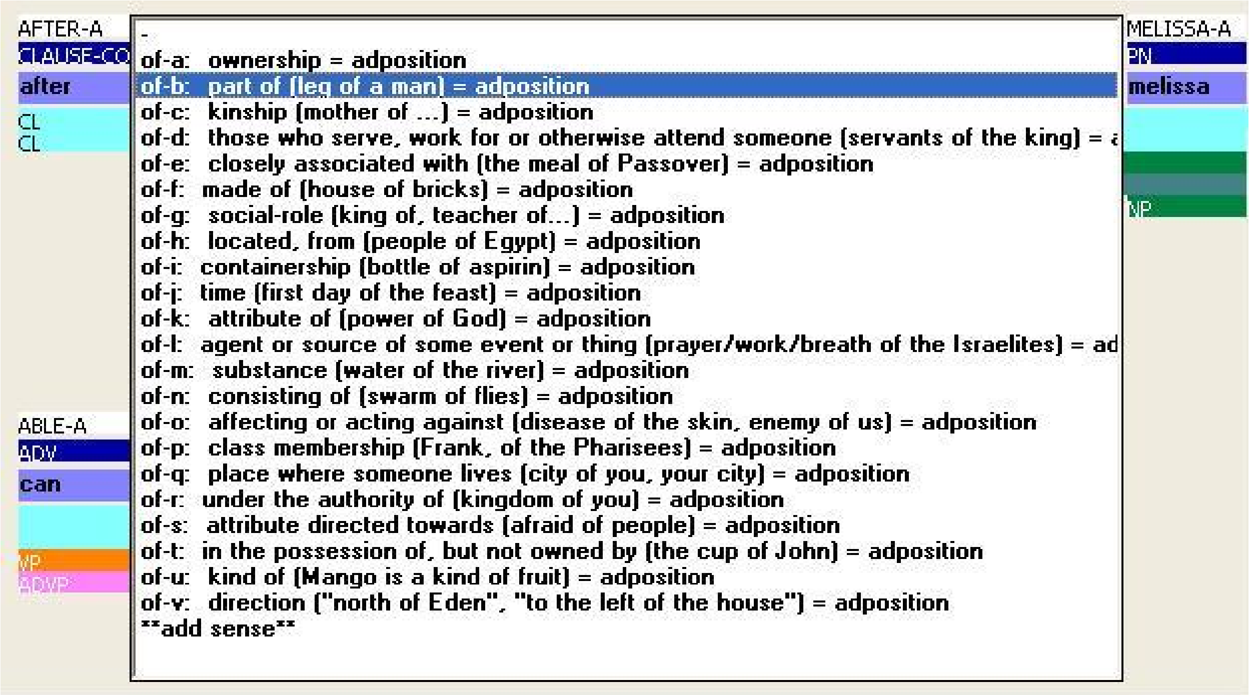
The analyzer also adds various semantic features to individual words based on such syntactic features as tense and number. There is a mechanism for changing these semantic features, although it is rarely needed.

1.2 The controlled language

A few of the features of the controlled language we enforce can be seen in the text box in Figure 1:

- We do not allow possessive nouns (i.e. 's), but require the use of "of" ("the eyes of Melissa"). This is primarily because we want to be able to specify the precise semantics of the relationship. See Figure 3 for the list of possible semantic relations from which the sense of "of" must be chosen for each occurrence. A similar list describes the possible meanings of the English verb "be".
- The use of pronouns is allowed, but the document author is trained to use them only in cases where they are semantically unambiguous. With experience, we have learned that the target text's naturalness can be dramatically improved by specifying ahead of time which nouns can be safely referred to by pronouns. A conservative use of pronouns by

the document author, with an eye trained to spot those situations that are semantically unambiguous, has proven valuable in this project. A mechanism for viewing (and changing) the analyzer's default linking of the pronoun to its antecedent is provided. This link is needed for target languages in which the syntactic features of pronouns are different than in English. In addition, the English pronouns have word senses that distinguish them based on number and exclusivity (for example, there are we-pl-exclusive and we-pl-inclusive word senses, along with dual and trial distinctions).



-
of-a: ownership = adposition
of-b: part of (leg of a man) = adposition
of-c: kinship (mother of ...) = adposition
of-d: those who serve, work for or otherwise attend someone (servants of the king) = a
of-e: closely associated with (the meal of Passover) = adposition
of-f: made of (house of bricks) = adposition
of-g: social-role (king of, teacher of...) = adposition
of-h: located, from (people of Egypt) = adposition
of-i: containership (bottle of aspirin) = adposition
of-j: time (first day of the feast) = adposition
of-k: attribute of (power of God) = adposition
of-l: agent or source of some event or thing (prayer/work/breath of the Israelites) = ad
of-m: substance (water of the river) = adposition
of-n: consisting of (swarm of flies) = adposition
of-o: affecting or acting against (disease of the skin, enemy of us) = adposition
of-p: class membership (Frank, of the Pharisees) = adposition
of-q: place where someone lives (city of you, your city) = adposition
of-r: under the authority of (kingdom of you) = adposition
of-s: attribute directed towards (afraid of people) = adposition
of-t: in the possession of, but not owned by (the cup of John) = adposition
of-u: kind of (Mango is a kind of fruit) = adposition
of-v: direction ("north of Eden", "to the left of the house") = adposition
add sense

Figure 3: Interplay of controlled language and semantic analysis: the case of "of"

- Imperatives, yes-no questions and content questions are marked directly in the text by (imp), (yn-ques) or (ques). The actual sentence is then entered in its declarative form. For example, in the text box in Figure 1, notice that the subject "you" is included in the imperative. For content questions, an appropriate pronoun such as "who" or "where" is placed in the clause constituent that is being questioned.
- Other standard restrictions (like those described for the Kant system in (Baker, et al., 1994) and (Mitamura, 1991)) are employed, such as disallowing reduced relative clauses.

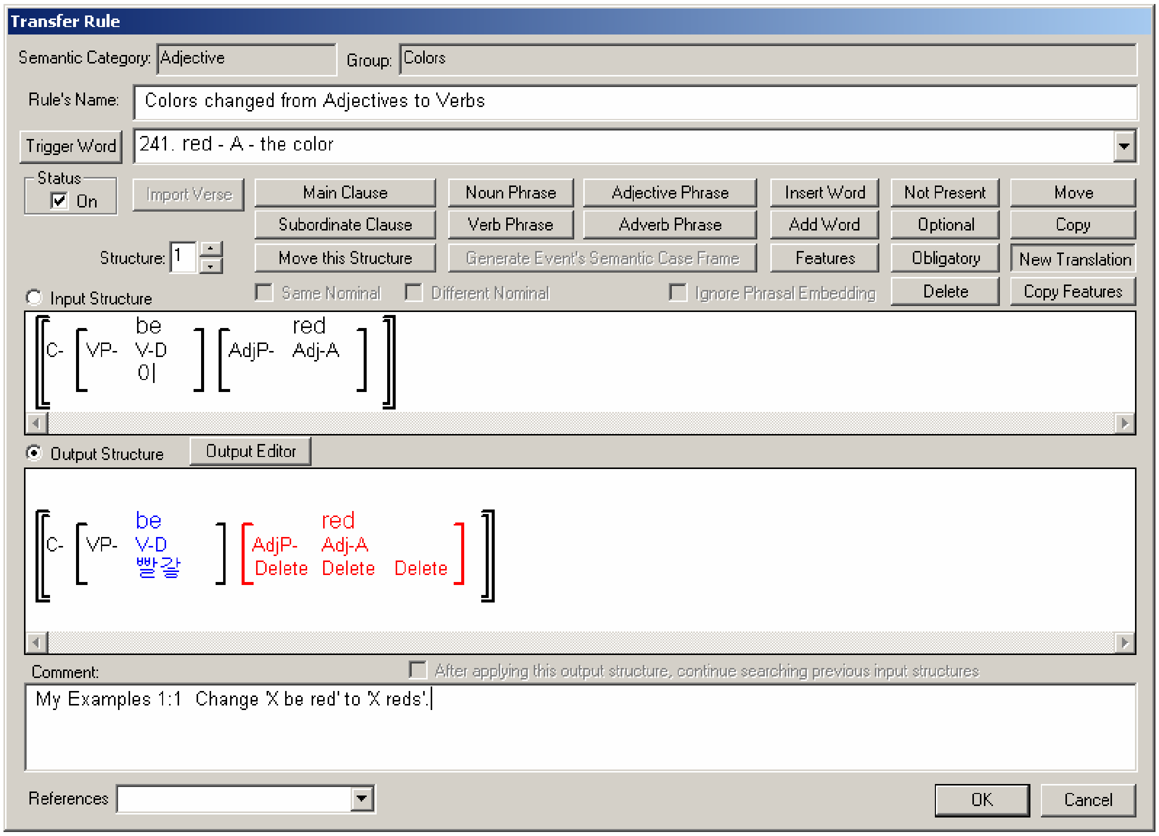
2.0 The Multi-lingual Text Generator and Interface

In this section we discuss the target language knowledge acquisition process along with a brief overview of the translation process. The text generator has been tested with English, Korean, Jula (spoken in West Africa) and Kewa (a clause chaining language spoken in Papua New Guinea)[i]. Korean, Jula and Kewa differ conceptually and structurally from English, yet in all cases the generated text has been well understood, grammatically perfect, and semantically equivalent to the original text.

2.1 The generation environment

The text generator that has been integrated into this system was designed to be extremely flexible yet very easy to use. The generator is capable of producing text for any language regardless of how radically that language differs structurally or conceptually from the source. The knowledge sources required for generation consist of a target lexicon and grammar. The target language acquirer is presented with all the semantic concepts that make up the set of semantically analyzed texts to be translated. The target words and expressions must then be entered into predefined and user-defined syntactic categories, and are automatically linked to the concepts in the Text Meaning Representations (TMR). A target grammar must then be entered that will first transform the TMR into target language structures and then synthesize the proper surface forms. These processes will be briefly described next.

The transfer grammar performs mechanical operations on the TMR in order to change it into a new underlying representation that is appropriate for the target language's descriptive grammar. These mechanical operations include inserting new constituents into the TMR, deleting constituents, moving constituents, copying constituents, and setting or copying features. It is the transfer grammar that performs all of the case frame adjustments, generates grammatical relations from semantic roles, builds clause chains with medial and final verbs, etc. After the transfer grammar has been executed, the TMR has been transformed so that it contains the appropriate words, features and structures for the target language. The descriptive grammar then takes the output generated by the transfer grammar and synthesizes the appropriate surface forms.


Transfer Rule

Semantic Category: Adjective

Group: Colors

Rule's Name: Colors changed from Adjectives to Verbs

Trigger Word: 241. red - A - the color

Status: ☑ On

Structure: 1

□ Same Nominal □ Different Nominal □ Ignore Phrasal Embedding

○ Input Structure

[C- [VP- be V-D 이] [AdjP- red Adj-A]]

● Output Structure

[C- [VP- be V-D 빨갈] [AdjP- Delete red Adj-A Delete Delete]]

Comment:

□ After applying this output structure, continue searching previous input structures

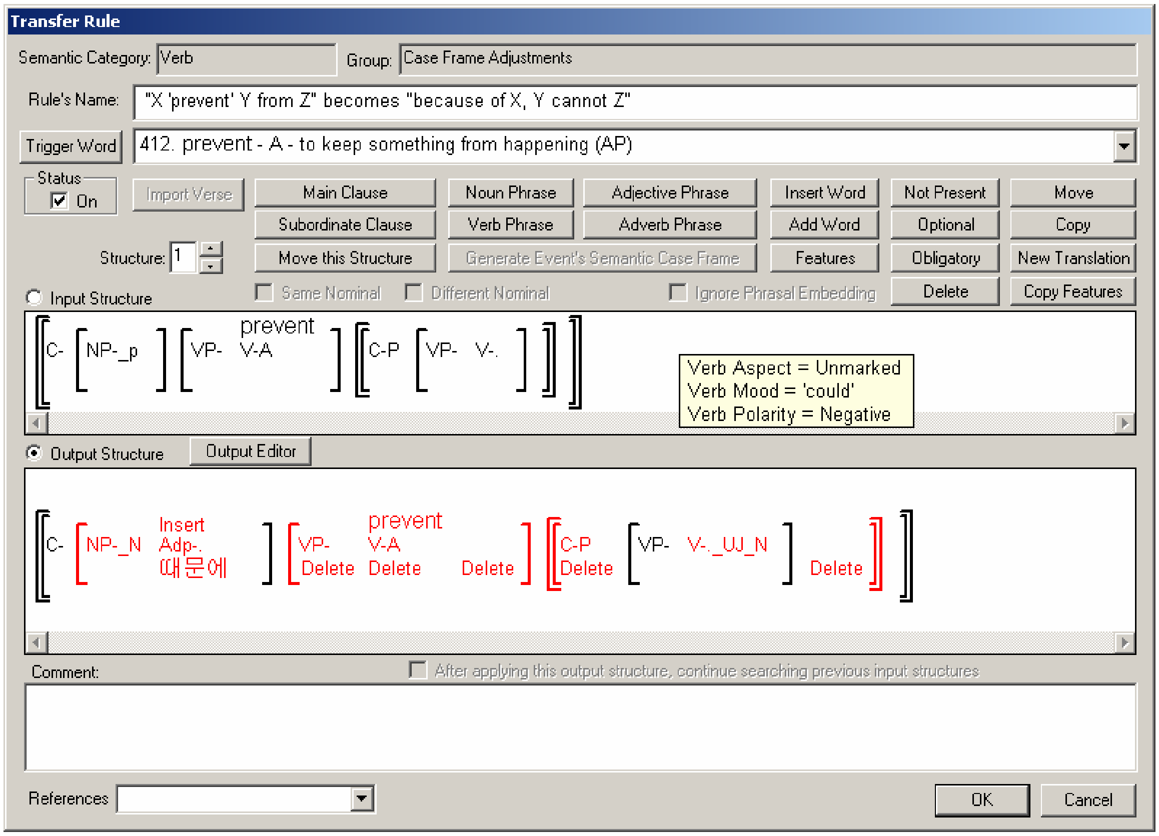
My Examples 1:1 Change 'X be red' to 'X reds'.

References:

Figure 4. Korean Transfer Rule for RED

To illustrate the function of the transfer grammar, a simple transfer rule for Korean is shown in Figure 4. Korean has a verb that is somewhat equivalent to the English verb ‘to be’, but it also has a more natural way of expressing color with a verb that means ‘be red’. Therefore the transfer grammar for Korean needs a rule that will look for constructions in the TMR containing [X be red]. That rule will then delete the adjective phrase and change the verb from ’to be’ to the Korean verb meaning ‘to be red.’ Since all colors in Korean are handled this way, this rule can be modified so that it contains a table that will make the necessary corrections for each of the colors. The user can set up tables whenever different concepts need to be handled in similar ways.

For a slightly more complex example, consider the transfer rule for PREVENT in Figure 5. Korean does not have a lexical equivalent for the English concept PREVENT[ii]. However, by restructuring the proposition, the semantic equivalent can be formed. Consider the sentence *Mary prevented John from reading the book.* The Korean equivalent is *Because of Mary, John was unable to read the book.* A transfer rule can perform the case frame adjustment for the event PREVENT to generate a new underlying proposition that is semantically equivalent to the original but is more suitable for Korean. The visual grammar interface will automatically present the standard case frame for PREVENT in the “input structure” and will copy it to the “output structure.” The grammar writer can then make the necessary modification to the output structure. Because of space limitations, we will gloss over the specifics of the representation language.


Transfer Rule

Semantic Category: Verb

Group: Case Frame Adjustments

Rule's Name: "X 'prevent' Y from Z" becomes "because of X, Y cannot Z"

Trigger Word: 412. prevent - A - to keep something from happening (AP)

Status ☑ On

Structure: 1

☐ Same Nominal ☐ Different Nominal ☐ Ignore Phrasal Embedding

○ Input Structure

[C- [NP-_p] [VP- prevent V-A] [C-P [VP- V-.]]]

Verb Aspect = Unmarked
Verb Mood = 'could'
Verb Polarity = Negative

● Output Structure

[C- [NP-_N Insert Adp-. 때문에] [VP- Delete prevent V-A Delete Delete] [C-P Delete [VP- V-._UJ_N] Delete]]

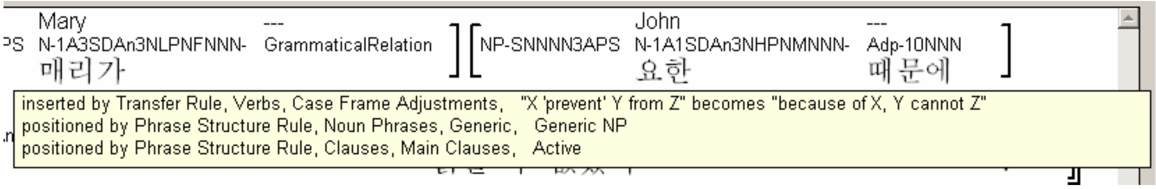
Comment:

☐ After applying this output structure, continue searching previous input structures

References:

Figure 5. Korean Transfer Rule for PREVENT

During the translation of a sentence, the system keeps track of all the rules that participate in the generation of each particular constituent. After a short passage has been generated, the user can rest the cursor on each constituent and see which rules were involved in the synthesis of that particular constituent. Shown below in Figure 6 is another popup menu that shows the rules that were involved in inserting and positioning the Korean word for 'because.' As can be seen in the popup, the rule that inserted this word is the rule that was discussed in the previous paragraph for PREVENT. The only other rules that affected this particular word were phrase structure rules, which are used to set up the linear order of constituents.

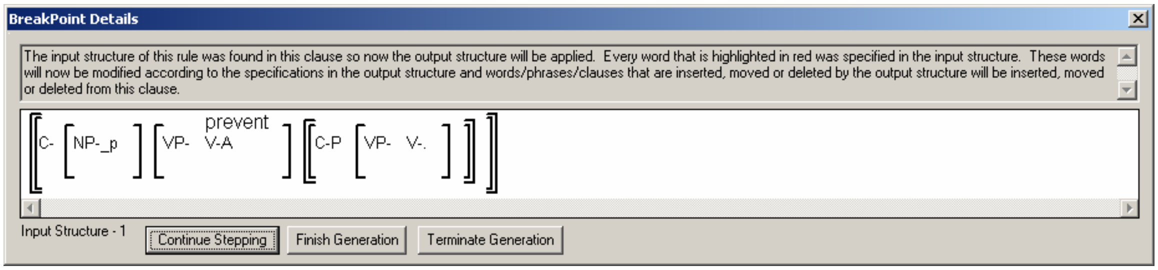


inserted by Transfer Rule, Verbs, Case Frame Adjustments, "X 'prevent' Y from Z" becomes "because of X, Y cannot Z"
positioned by Phrase Structure Rule, Noun Phrases, Generic, Generic NP
positioned by Phrase Structure Rule, Clauses, Main Clauses, Active

Figure 6: Pop-up showing rules that generated때문에 'because'

If any of these rules were not functioning properly, the user would right click and a new dialog box listing these rules would appear. The user could then select a rule from that dialog and edit it accordingly.

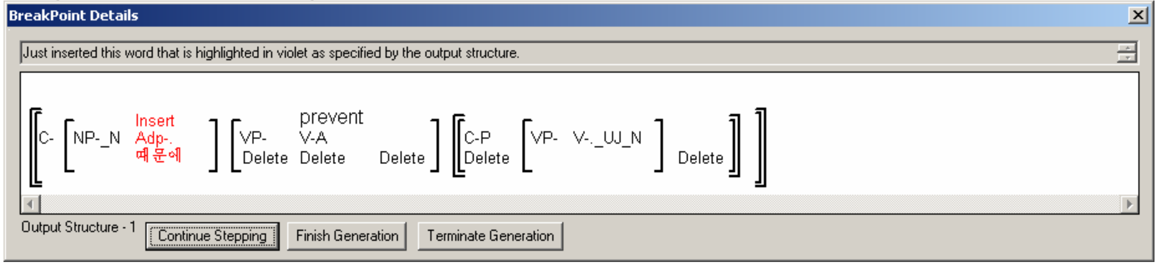
The system also contains a "grammar debugger." This debugger lets the user specify a breakpoint in the grammar. After the user clicks the Generate button, the system executes all of the rules that precede the breakpoint. The system then stops the execution and lets the user step through the following rule watching each decision that the system makes. Shown below in Figure 7 is the breakpoint dialog as it appears when the input structure for the PREVENT rule has been found.



The input structure of this rule was found in this clause so now the output structure will be applied. Every word that is highlighted in red was specified in the input structure. These words will now be modified according to the specifications in the output structure and words/phrases/clauses that are inserted, moved or deleted by the output structure will be inserted, moved or deleted from this clause.

Figure 7: Grammar Debugger explaining its current step

As the user continues stepping through the grammar execution process, the breakpoint dialog continues to show and explain each step. Shown below in Figure 8 is the breakpoint dialog as the generator is inserting the Korean word for'because' into the text.



Just inserted this word that is highlighted in violet as specified by the output structure.

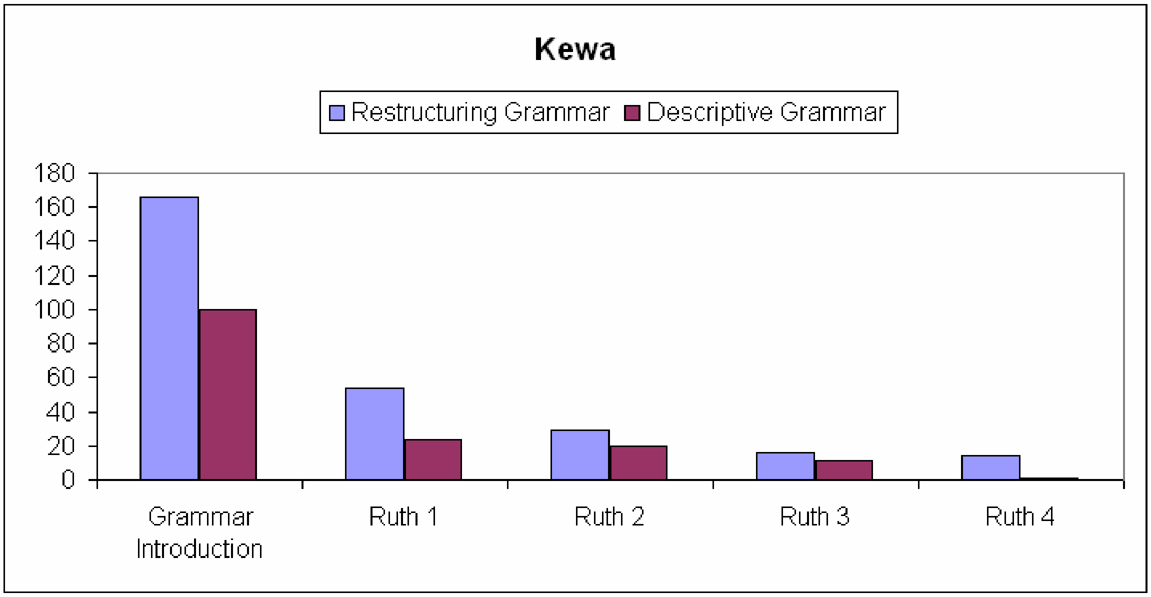
Figure 8: Grammar Debugger inserting a word

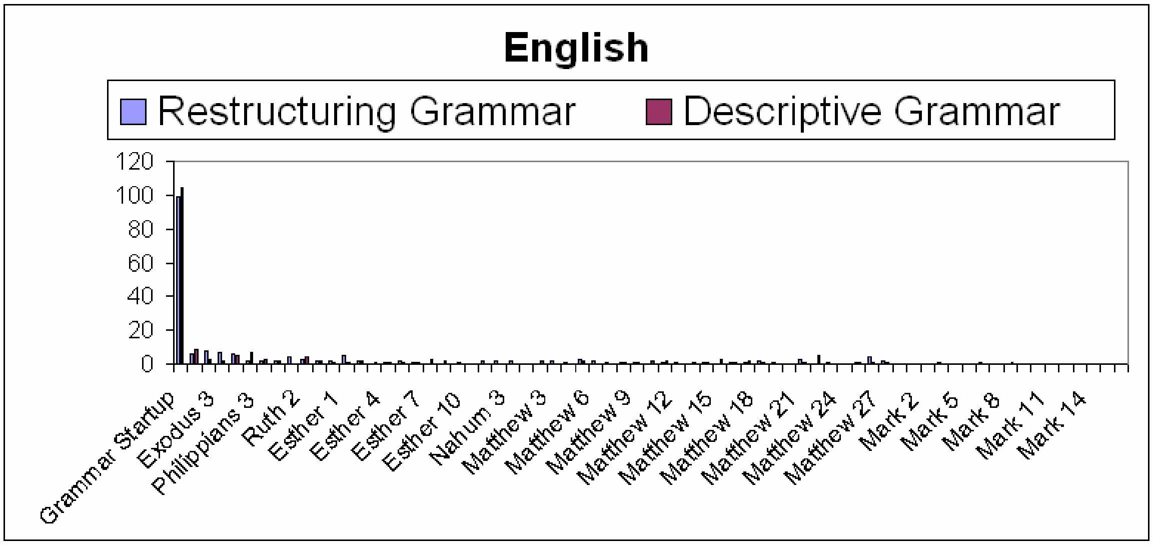
By integrating the grammar editor, debugger and execution modules, the user is able to quickly and easily develop his grammar so that it generates the desired target text. Additional tools which help the user develop his grammar will be described in the next section.

2.2 The quick ramp-up grammar acquisition process

This generator has several additional features which help users build their grammars very quickly. By far the most common task performed by the transfer grammar is case frame adjustments. In order to help users build their case-frame-adjustment rules quickly, the system will, upon request, automatically create the skeleton for hundreds of case frame adjustment rules, each one containing the input case frame for an event in the ontology. Then the user need only enter the necessary adjustments into the output structure of each rule. Other common tasks that must be performed by the transfer grammar have been loaded into pre-written transfer rules which users can turn on or off. For example, OBJECTs in the TMRs are marked for Number, with the possible values being Singular, Dual, Trial, Quadrial and Plural. If a particular language only distinguishes Singular and Plural, the user can select a text preprocessor that converts Dual, Trial and Quadrial to Plural.

In order to further facilitate the development of the target grammars, a Grammar Introduction has been developed. This consists of approximately 300 basic propositions (or clauses) and culminates in a short narrative discourse. Each of the propositions illustrates a particular feature, concept or construction that is found in the TMRs. These propositions illustrate a variety of verbal aspects and moods, relative clauses formed on a variety of semantic roles, patient propositions (object complements) formed with a variety of matrix events, different types of adverbial clauses, different types of questions, etc.. After developing the grammar rules for these basic propositions, the user will have built a solid foundation for his grammar. To emphasize the utility of this Grammar Introduction, we present below two bar charts showing the number of rules that were required for the Grammar Introduction, and how many additional rules had to be entered in order to translate subsequent chapter of text. As can be seen, the number of new rules per chapter drops off dramatically after the Grammar Introduction has been completed.





Future development of this project will include the addition of a semi-automatic grammar acquisition module. This module will prompt users to enter responses to very specific questions. The module will then analyze the answers and propose rules that the user will be able to edit and save in his grammar.

2.3 Native-speaker evaluation of generated texts

This project has been used to generate fairly substantial amounts of text in English, Korean, Jula and Kewa. In every case, readers of the texts have said that the texts are easily understandable, grammatically perfect, and have the same semantic content as the original TMRs. They have also been able to back-translate the generated texts into English. However, the texts produced by this project are lacking in naturalness. For example, when generating the English draft of a short story about preventing eye infections, the program repeatedly produced the phrase “your two eyes are …” That phrase is perfectly understandable, but it would be more natural to say “both of your eyes are …” If the final product must be polished and natural, editors can make the necessary minor adjustments. Two separate experiments have shown that experienced translators are able to edit the generated texts into publishable forms in less than a third of the time they would have needed to manually translate the text. The participants in these experiments have said that they prefer to edit the drafts produced by this system rather than produce the translations manually.

Conclusion

We believe that the system we have described in this paper takes advantage of the controlled language document authoring methodology in a unique and valuable way. The simple, yet effective machine-aided semantic analyzer allows for very accurate semantic analysis of even large texts in a relatively small amount of time. The visual grammar and quick ramp-up methodology have been used to produce generation systems quickly in several languages from diverse language families. And most importantly, the translations produced by this system have been judged by native speakers to be nearly flawless.

Baker, K. et al. (1994), "Coping with Ambiguity in a Large-Scale Machine Translation System," Proc. of COLING-94, Kyoto, Japan, 1994, pp. 90-94.

Beale, Stephen and Sergei Nirenburg. 2003. Just in time grammar. Submitted to HLT-NAACL-03.

Mitamura, T., Nyberg, E. and Carbonell, J. (1991), "An Efficient Interlingua Translation System for Multilingual Document Production," Proc. of MT Summit III, Washington, DC, 1991.

[i] All Korean data courtesy of Dr. Baek Sung Choi. All Jula data courtesy of Randy Groff. All Kewa data courtesy of Dr. Karl Franklin.

[ii] Korean does have two words, one which means “block or prevent” and one which means “interfere or prevent.” However, neither of these words is used in daily language because they both have strong negative connotations and they both have a much narrower range of meaning than the English word ‘prevent.’ Another Korean word exists that is used when discussing preventative maintenance.