Automated English-Korean Translation for Enhanced Coalition Communications

Clifford J. Weinstein, Young-Suk Lee, Stephanie Seneff, Dinesh R. Tummala, Beth Carlson, John T. Lynch, Jung-Taik Hwang, and Linda C. Kukolich

This article describes our progress on automated, two-way English-Korean translation of text and speech for enhanced military coalition communications. Our goal is to improve multilingual communications by producing accurate translations across a number of languages. Therefore, we have chosen an interlingua-based approach to machine translation that readily extends to multiple languages. In this approach, a natural-language-understanding system transforms the input into an intermediate-meaning representation called a semantic frame, which serves as the basis for generating output in multiple languages. To produce useful, accurate, and effective translation systems in the short term, we have focused on limited military-task domains, and have configured our system as a translator's aid so that the human translator can confirm or edit the machine translation. We have obtained promising results in translation of telegraphic military messages in a naval domain, and have successfully extended the system to additional military domains. The system has been demonstrated in a coalition exercise and at Combined Forces Command in the Republic of Korea. From these demonstrations we learned that the system must be robust enough to handle new inputs, which is why we have developed a multistage robust translation strategy, including a part-of-speech tagging technique to handle new words, and a fragmentation strategy for handling complex sentences. Our current work emphasizes ongoing development of these robust translation techniques and extending the translation system to application domains of interest to users in the military coalition environment in the Republic of Korea.

The United States military operates worldwide in a variety of international environments that require language translation. Translators who can interpret military terminology are a scarce commodity in countries such as the Republic of Korea (R.O.K.), and U.S. military leaders there support the development of bilingual machine translation. Although U.S. and R.O.K. military personnel have been working together for more than forty years, the language barrier still significantly reduces the speed and effectiveness of coalition command and control. During hostilities, any time saved by computers that can quickly and accurately translate command-and-control information could provide an advantage over the enemy and reduce the possibility of miscommunication with allies.

Machine translation has been a challenging area of research for four decades, as described by W.L. Hutchins and H.L. Somers [1], and was one of the original problems addressed with the development of

computers. Although general, effective solutions remain elusive, we have made substantial advances in developing an automated machine-translation system to aid human translators in limited domains, specifically for military translation tasks in the Combined Forces Command (CFC) in Korea. Our strategy to enhance the probability of success in this effort has been threefold: first, to build upon the tremendous advances in the research and development community over the past decade in natural-language understanding and generation, machine translation, and speech recognition; second, to carefully choose limited but operationally important translation applications to make the task manageable; and third, to facilitate user interaction with the translation system, so that the primary goal is not a fully automated translator but an aid that helps the human translator be more effective.

Machine-Translation Background

The pyramid diagram of Figure 1 shows source-language analysis along the left side and target-language generation along the right side, and three machine-translation strategies: interlingua, transfer, and direct. Most machine-translation strategies cut off the source-language analysis at some point along the way, and perform a bilingual transfer. The interlingua approach is different. It eliminates a bilingual transfer phase by producing a language-independent meaning representation called the interlingua that is directly usable for target-language generation. In addition, it greatly facilitates the development of a multilingual system, because the same interlingua can be used to generate multiple target languages. Although achieving a language-independent interlingual representation is a difficult challenge for general domains, the interlingua approach offers significant advantages in limited domains.

Direct translation systems do little source-language analysis, proceeding immediately to a transfer. They produce a word-for-word translation, much like an automated bilingual-dictionary lookup. The resulting translation generally does not have proper word order, syntax, or meaning in the target language, although it may be of some help to a user.

Transfer systems perform some intermediate form of analysis, then proceed to a bilingual transfer. The SYSTRAN translation system, which has been used in our project for Korean-to-English translation, falls into this category. Transfer systems vary greatly in the quality of translation output and, for multilingual applications, require substantial additional effort in analysis and generation for each language pair. The advantage of a state-of-the-art transfer system like SYSTRAN is that it produces translations for a wide range of input texts and does not require a limited domain. When compared to an interlingual approach, however, the transfer system has a disadvantage: the translations produced, although better than word-for-word direct translations, often do not capture the correct syntax or meaning of the input text.

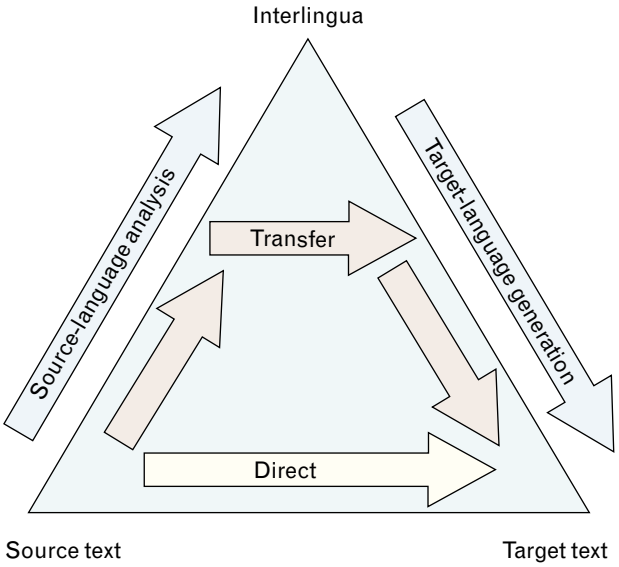


FIGURE 1. Pyramid illustrating the relationships among interlingua, transfer, and direct approaches to machine translation. The interlingua approach differs from the other two by producing a language-independent meaning representation called the interlingua that is directly usable for target-language generation.

CCLINC Translation-System Structure

The architecture for our translation system, presented in Figure 2, consists of a modular, multilingual structure including language understanding and language generation in English and Korean. We refer to this translation system as the common coalition language system at Lincoln Laboratory, or CCLINC. The system input can be text or speech. The understanding

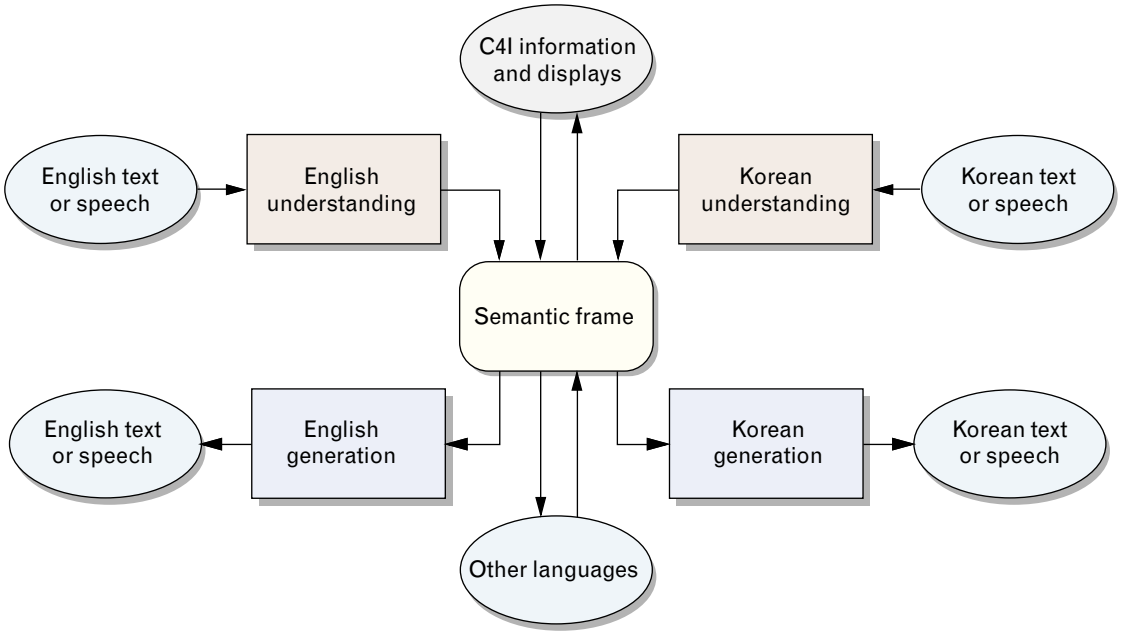


FIGURE 2. Architecture of the common coalition language system at Lincoln Laboratory (CCLINC). The understanding modules convert Korean or English input into a language-independent meaning interlingual representation known in this case as a semantic frame. The use of semantic frames allows the CCLINC system to extend to multiple languages. The meaning representation in the semantic frame could also be used to provide two-way communication between a user and a Command, Control, Communications, Computing, and Intelligence (C4I) system.

module of CCLINC converts each input into an interlingual representation. In CCLINC, this interlingual representation is called a semantic frame. In the case of speech input, the understanding module in Figure 2 performs speech recognition and understanding of the recognition output. Our current speech-recognition system and its performance on speech translation are described in a later section. Although our original work on this project involved speech-to-speech translation [2], we have recently emphasized text translation [3] in response to the priorities of U.S. military users in Korea. An ongoing effort by Korean researchers in English-to-Korean text translation is described in Reference 4.

The CCLINC translation system provides feedback to the originator on its understanding of each input sentence by forming a paraphrase in the originator's language. For example, when an English speaker enters a sentence into the system, the sentence is first transformed into a semantic frame by the English-understanding module. Then the English-generation module produces a paraphrase of what the system understood, which can be verified by the originator before the Korean-generation module provides the translation to the receiver. Figure 2 illustrates how the interlingual approach expedites the extension of the system to multiple languages. For example, adding Japanese to the English-Korean system requires Japanese-understanding and Japanese-generation modules, but the English and Korean modules do not change. Successful system operation depends on the ability to define a sufficiently constrained yet useful vocabulary and grammar as well as the application of powerful understanding and generation technology so that a high percentage of input sentences can be understood. Figure 2 also shows a two-way connection between the translation system and a Command, Control, Communications, Computing, and Intelligence (C4I) system. Because the translation system involves the understanding of each input, C4I data and displays based on this understanding can be periodically updated and users can request information through the C4I system while communicating with other people via translation.

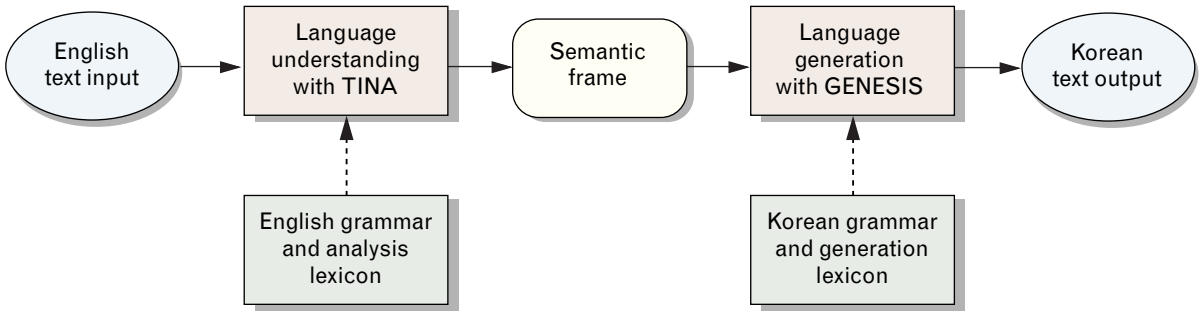


FIGURE 3. Process flow for English-to-Korean text translation in CCLINC. The TINA language-understanding system utilizes the English grammar and analysis lexicon to analyze the English text input and produce a semantic frame representing the meaning of the input sentence. The GENESIS language-generation system utilizes the Korean grammar and generation lexicon to produce a Korean output sentence based on the semantic frame.

This article deals mainly with our work in English-to-Korean text translation. Although the CCLINC translation system is general and extendable, most of our work to date has focused on English-to-Korean text translation because it is the application of most interest to U.S. forces in Korea. Our work has also included two-way English-Korean translation of both speech and text. We have started developing an interlingua-based Korean-to-English translation subsystem in CCLINC. (Our previous Korean-to-English system was developed by SYSTRAN, Inc., under a subcontract.) Our initial work on this project included translation from English speech and text to French text [2].

Input sentence: *0819 z uss sterett taken under fire by a kirov with ssn-12s.*

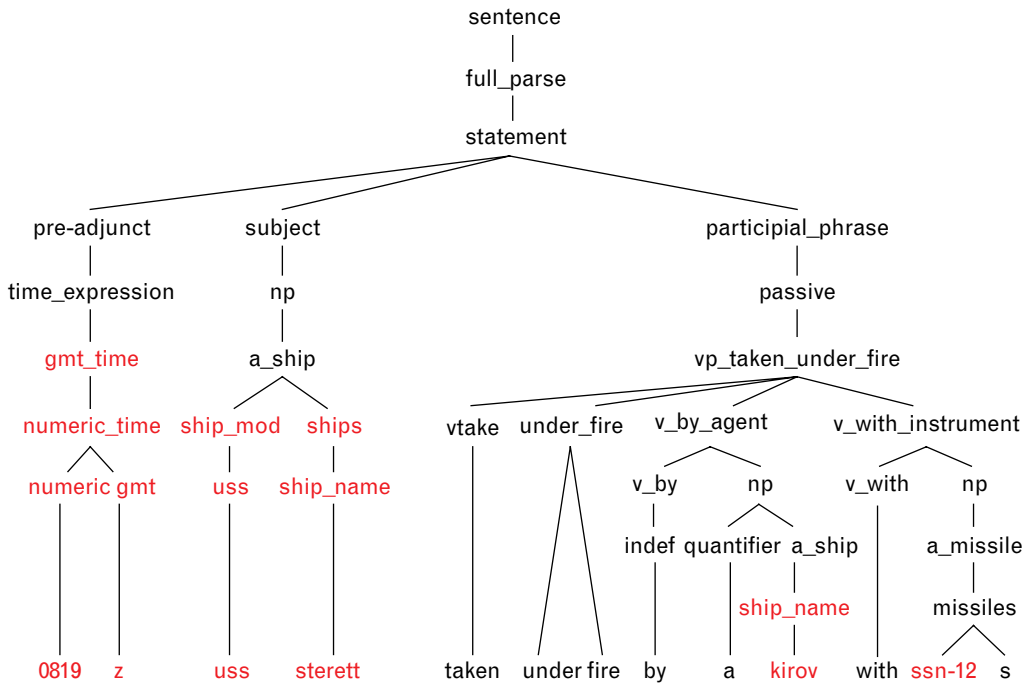


FIGURE 4. Parse-tree example based on English input sentence. The parse tree represents the structure of the input sentence, and is represented in terms of both general syntactic categories, such as the subject or participial phrase, and domain-specific semantic categories, highlighted in red, of material being translated, such as the ship name.

System Process Flow and Example for English-to-Korean Text Translation

Figure 3 illustrates the process flow of English-to-Korean text translation in CCLINC. The core of CCLINC consists of two modules: the language-understanding system, TINA [5], and the language-generation system, GENESIS [6]. Both modules were originally developed by the Spoken Language Systems group at the MIT Laboratory for Computer Science, under Defense Advanced Research Projects Agency (DARPA) sponsorship, for applications in human-computer interaction with a variety of languages [7, 8]. Our project was the first to adapt TINA and GENESIS for language translation and to apply these systems to the Korean language.

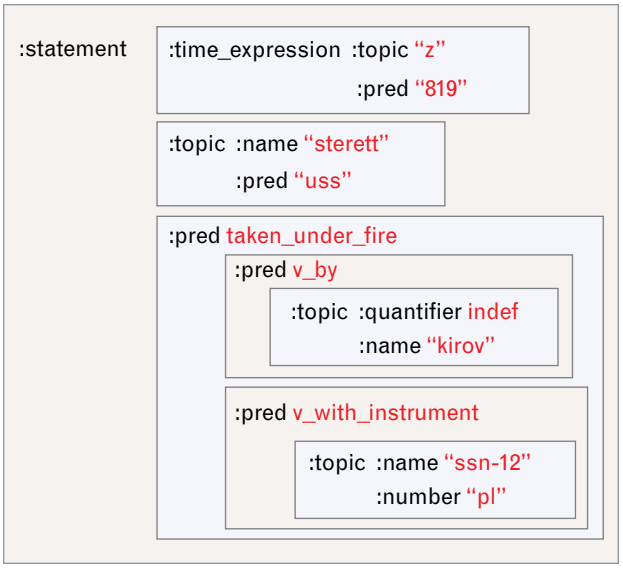
The understanding and generation modules operate from a set of files that specify the source-language and target-language grammars. The modules are mediated by the semantic frame, which serves as the basis for generating output in multiple languages, and can be integrated into the command-and-control information system for database query.

The first task domain that we used to develop our CCLINC translation system consists of a set of messages about simulated naval engagements in the Pacific. To illustrate system operation, we show the roles of the various system modules in translating the following sentence from that domain:

> *0819 z uss sterett taken under fire by a kirov with ssn-12s.*

Given this input sentence, the language-understanding system produces a parse tree, as illustrated in Figure 4. The parse tree, which represents the input sentence structure, is produced automatically by CCLINC. The parse tree identifies grammatical, or syntactic, information such as the pre-adjunct *0819 z*, the subject *uss sterett*, and the predicate *taken under fire by a kirov with ssn-12s*. The parse tree also provides domain-specific information—in this example, *z* stands for Greenwich Mean Time; *sterett* and *kirov* are ship names; and *ssn-12* is a missile name. The categories such as ship and missile name are not standard English grammatical categories, but are domain-specific semantic categories that represent the meaning of the words in this domain of naval messages. These domain-specific categories enable CCLINC to reduce the ambiguity of the input sentence.

The language-understanding system then derives a semantic frame from the parse tree, as shown in Figure 5. As a language-neutral meaning representation of the input sentence, the semantic frame captures the core meaning of the input sentence through three major categories: topic, predicate, and clause. The main action of the sentence—*taken under fire*—is represented by the predicate category, and the entities involved in the action—*sterett, kirov, ssn-12*—are represented by the topic category. The semantic frame also preserves the information that the sentence is a statement rather than a question or command. However, the semantic frame purposely does not retain structural information that tells us how to generate a



Paraphrase:
819 Z USS Sterett taken under fire by a kirov with SSN-12s.

Translation:
8 시 19 분 표준시간 미군함 스테렛이 SSN-12 미사일로 포격되다.

FIGURE 5. Semantic frame, paraphrase, and translation for the example sentence of Figure 4. The semantic frame represents the meaning of the input in terms of fundamental language-neutral categories such as topic and predicate, and is used as the basis for generation of both the English paraphrase and the Korean output sentence. Entries in red in the semantic frame are replaced by the corresponding vocabulary items in the Korean-generation lexicon.

Table 1. Sample English-Korean Language-Generation Lexicon

V1	V	"ha" PRESENT "han" PAST "hayss " PP "hayss" PSV "toy"
indef	D	" "
kirov	N	"khirob"
ssn-12	N	"ssn-12 misail"
sterett	N	"stheret"
take_under_fire	V1	"phokyek"
uss	N	"mikwunham"
z	N	"pyocwunsikan"
v_by	P	"uyhay"
v_with_instrument	P	"lo"

sentence in a particular language with the meaning represented in the frame. This language-generation information needs to be put in by a generation system specific to the target language, as discussed below.

In addition to the three major categories, other categories such as number (singular or plural) and tense (present, past, or future) can be added. Whether we add more categories to the semantic-frame representation depends on how detailed a representation is required. To refine the translation, we can increase the number of semantic-frame categories. In addition, some languages require more elaborate tense or honorific representations than others. The flexibility of the semantic-frame representation makes the TINA language-understanding system an ideal tool for machine translation.

The primary task of the language-generation system is to produce target-language output that captures the meaning represented in the semantic frame in a proper and grammatically correct sentence in the target language. In our translation system, we have both a Korean-generation and an English-generation module. For English source language, the English-generation module must produce a paraphrase of the input in the source language. Both the English paraphrase and the Korean translation are shown beneath the example semantic frame in Figure 5. The paraphrase in this case is essentially identical to the original (except that 0819 is replaced by 819). The Korean output is Hangul text composed from the 24 basic letters and 16 complex letters of the Korean alphabet.

To produce translation output, the language-generation system requires three data files: a lexicon, a set of message templates, and a set of rewrite rules. These files are language-specific and external to the core language-generation system. Consequently, extending the language-generation system to a new language requires creating only the data files for the new language. A pilot study of applying the GENESIS system to Korean language generation can be found in Reference 9. For generating a sentence, all the vocabulary items in the semantic frame such as *z, uss,* and *by* are replaced by the corresponding vocabulary items provided in the lexicon. All phrase-level constituents represented by topic and pred are combined recursively to derive the target-language word order, as specified in the message templates. We give examples below of the data files that are necessary to generate Korean translation output.

Table 1 shows a sample language-generation lexicon necessary to generate the Korean translation output of the input sentence from the semantic frame in Figure 5—*0819 z uss sterett taken under fire by a kirov with ssn-12s.* Words and concepts in the semantic frame are given in the left column of the table, and the corresponding forms in Korean are given in the right column. The Korean forms are in Yale Romanized Hangul, a representation of Korean text in a phonetic form that uses the Roman alphabet [10]. Because the semantic frame uses English as its specification language, lexicon entries contain words and concepts found in the semantic frame with corresponding forms in Korean. (For a discussion about designing interlingua lexicons, see Reference 11.)

In the lexicon, *P* stands for the part of speech preposition; *N* noun; *D* determiner; and *V* verb. Verbs are classified into several subgroups according to grammatical rules that govern which tense forms are used. The first row of the example in Table 1 says that the entry V1 is a category verb *ha* for which the present tense is *han*, past tense *hayss*, past participle *hayss*, and passive voice *toy*.

Table 2. Sample Korean Language-Generation Message Templates

(a) statement	:time_expression :topic *i* :predicate *ta*
(b) topic	:quantifier :noun_phrase
(c) predicate	:topic :predicate
(d) np-uss	:predicate :noun_phrase
(e) np-v_by	:topic :predicate :noun_phrase
(f) np-v_with_ instrument	:topic :predicate :noun_phrase

Message templates are target-language grammar rules corresponding to the input-sentence expressions represented in the semantic frame. The word order of the target language is specified in the message templates. Table 2 gives a set of message templates required to produce the Korean translation output from the semantic frame in Figure 5.

Template *a* instructs that a statement consists of a time expression followed by the topic, which in turn is followed by the predicate (corresponding to the verb phrase). The morpheme *i* following *:topic* is the subject case marker, and the morpheme *ta* following *:predicate* is the marker indicating that the sentence is a statement. According to template *b*, a topic (typically equivalent to a noun phrase) consists of a quantifier and the head noun itself. Template *c* says that a verb phrase consists of an object followed by the verb. This template specifies that in Korean the object precedes the verb, as opposed to English, in which the object follows the verb. Also, it illustrates that the predicate category encompasses several syntactic subcategories including a verb and a verb phrase. Template *d* says that *uss* is a predicate embedded under a higher-level predicate. Templates *e* and *f* say that the prepositional phrases headed by the equivalents of *by* and *with* are predicates, and take an object to their left, and are embedded under a higher-level category.

Rewrite rules are intended to capture surface phonological constraints and contractions, in particular, the conditions under which a single morpheme has different phonological realizations. In English, the rewrite rules are used to generate the proper form of the indefinite article, *a* or *an*. Choosing one indefinite article over the other depends on the phonology of the word that follows. For example, if the word that follows starts with a vowel, the appropriate indefinite article is *an;* if the word that follows starts with a consonant, the appropriate indefinite article is *a*. The Korean language employs similar types of morphological variations. In Table 3, the so-called nominative case marker is realized as *i* when the preceding morpheme (*John* in this example) ends with a consonant, and as *ka* when the preceding morpheme (*Maria* in this example) ends with a vowel. Similarly, the so-called accusative case marker is realized as *ul* after a consonant, and as *lul* after a vowel. Because these types of alternations are regular, and it is not possible to list every word to which these markers are attached in the rewrite-rule templates, a separate subroutine written in C-code has been implemented to improve efficiency. For details of other related phenomena in the Korean language, see Reference 12.

User View of System as Translator's Aid

Before proceeding to an extended discussion of the technical operation and performance of our system, we describe its operation as a translator's aid. Figure 6 shows the graphical user interface of our system in the English-to-Korean translation mode. The interface features four windows and five icon buttons. English text is entered in the top window. Input is entered by voice, through the keyboard, or from a file or external message source. To enter a voice input, the user activates the speech recognizer by clicking on the microphone icon and speaks the sentence. The recognized speech appears in the English input window and is

Table 3. Phonologically Conditioned Case Markers in Korean

	Nominative Case	*Accusative Case*
Following consonant	John-*i*	John-*ul*
Following vowel	Maria-*ka*	Maria-*lul*

then treated as text input. To translate a sentence in the input window, the user clicks on the English-to-Korean translation icon (indicated by flags) and the translation appears in the third window from the top. In this example of text translation, the user has activated translation of the sentence that begins *At 0823 z Sterett*. The English paraphrase is shown in the paraphrase window, and the Korean translation of that sentence (in Hangul characters) is shown in the window below the English paraphrase. The user then has an opportunity to edit the Korean translation by using a Hangul text editor. When the translation is acceptable, the user clicks on the check icon, and the translated sentence is moved to the output window at the bottom. Here, the translation of the prior sentence starting with *0819 z USS Sterett* is shown in the output window. If the user wishes to view the translation process in more detail, the parse tree or semantic frame can be viewed by clicking on the tree or frame icons.

In configuring our system as a translator's aid, we provide the user with as much help as possible. If the system is unable to parse and understand the input sentence, a word-for-word translation is provided to the user, consisting of a sequence of word translations from the Korean-generation module. If some of the English words are not in the generation lexicon, the original English word is included in the translation output in the place where its Korean equivalent would have occurred. In both cases, the problem is noted on the output.

The interlingua-based Korean-to-English translation system operates with the same graphical user in-

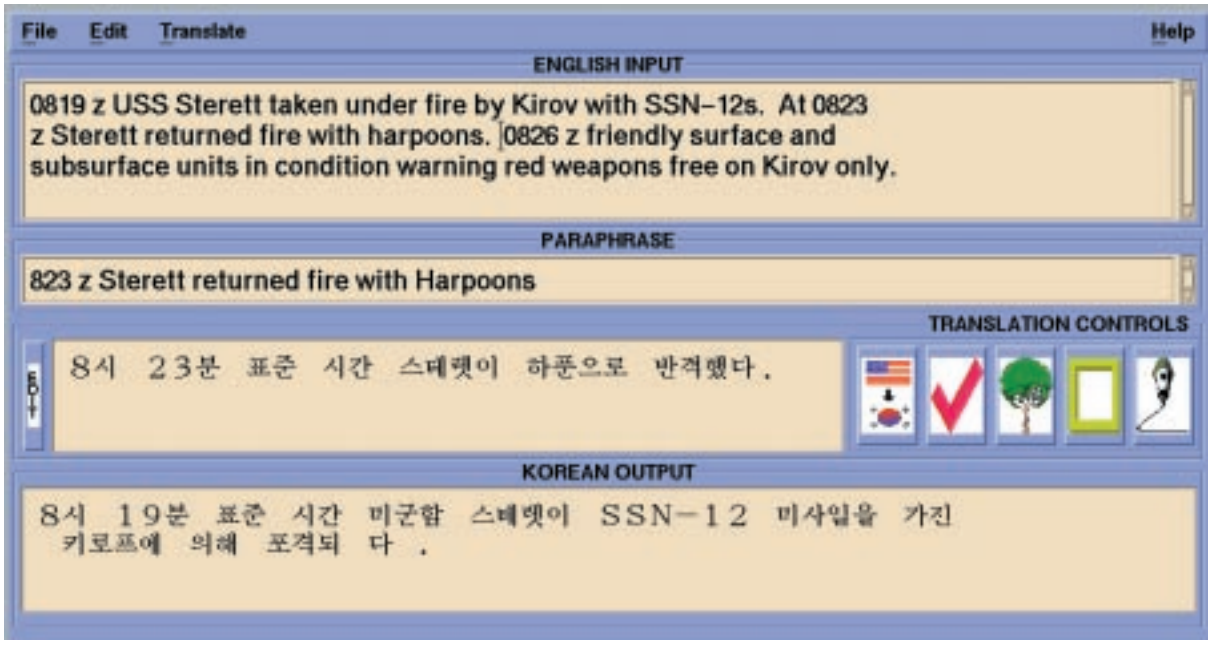


FIGURE 6. Graphical user interface of translator's aid in English-to-Korean translation mode. The input is entered by voice, through the keyboard, or from a file to the top window. The English paraphrase is shown below the input window, and the Korean translation of that sentence (in Hangul characters) is shown in the window below the English paraphrase. The user can edit the translation output by using a Hangul text editor. If the translation is acceptable, the translated sentence can be moved to the bottom window by clicking on the check icon. The parse tree and the semantic frame of the input sentence can be displayed by clicking on the tree and the frame buttons, respectively.

terface, except the U.S. and Korean flags are interchanged in the translation icon, and the input language is Korean. The SYSTRAN transfer-based Korean-to-English translation system, however, does not provide the user a paraphrase, parse tree, or semantic frame.

English-to-Korean System Development on Naval Message Domain: A Domain-Specific Grammar Approach

From June 1995 to April 1996 we trained our system on the MUC-II corpus, a collection of naval operational report messages from the Second Message Understanding Conference (MUC-II). These messages were collected and prepared by the center for Naval Research and Development (NRaD) to support DARPA-sponsored research in message understanding. Lincoln Laboratory utilized these messages for DARPA-sponsored machine-translation research. We chose to use the MUC-II corpus for the following reasons: (1) the messages were typical of actual military messages that our users would be interested in translating, including high usage of telegraphic text and military jargon and acronyms; (2) the domain was limited but useful, so that we felt that our interlingua approach could be applied with reasonable probability of success; and (3) the corpus was available to us in usable form.

The MUC-II Naval Message Corpus

MUC-II data consist of a set of naval operational report messages that feature incidents involving different platforms such as aircraft, surface ships, submarines, and land targets. The MUC-II corpus consists of 145 messages that average 3 sentences per message and 12 words per sentence [13, 14]. The total vocabulary size of the MUC-II corpus is about 2000 words. The following example shows that MUC-II messages are highly telegraphic with many instances of sentence fragments and missing articles:

> *At 1609 hostile forces launched massive recon effort from captured airfield against friendly units. Have positive confirmation that battle force is targeted (2035z). Considered hostile act.*

The messages in this article are modeled after but are not from the MUC-II corpus. For each message, a corresponding modified message has been constructed in more natural English. For example, in the modified version below, words that are underlined have been added to the original message:

> *At 1609 z hostile forces launched a massive recon effort from a captured airfield against friendly units. Friendly units have positive confirmation that the battle force is targeted (2035z). This is considered a hostile act.*

MUC-II data have other features typical of natural text. There are several instances of complex sentences having more than one clause, coordination problems involving conjunctions (and, or), and multiple noun and verb modifiers, as in the following examples:

> Complex sentences—*Two uss lion based strike escort f-14s were engaged by unknown number of hostile su-7 aircraft near land9 bay (island target facility) while conducting strike against guerrilla camp.*

> Coordination problem—*Fox locked on with fire control radar and fired torpedo in tiger's direction.*

> Multiple noun and verb modifiers—*The deliberate harassment of uscgc tiger by hostile fox endangers an already fragile political/military balance between hostile and friendly forces.*

Translation-System Training

For our translation-system training and development, we have used both the original and the modified data, including 105 messages from the MUC-II corpus. These messages, including both original and modified versions, comprised a total of 641 sentences. For additional training material, we added a set of 154 MUC-II-like sentences that were created in an in-house experiment, so that the total number of sentences used in training was 795. This training corpus was divided into four data sets. We trained the translation system by using an iterative procedure in which grammar and vocabulary were developed for the first

set, and then we tested and modified the translation system on subsequent sets.

In our training procedure, we developed analysis rules by hand on the basis of observed patterns in the data. These rules are then converted into a network structure. Probability assignments in the network are obtained automatically by parsing each training sentence and updating appropriate counts [5].

When the translation-system development was completed on the MUC-II corpus, the size of the lexicon was 1427 words for analysis and 1000 words for generation; the size of the grammar was 1297 categories for analysis and 850 categories for generation. The actual number of rules is much greater because TINA allows the sharing, or cross-pollination, of common elements [5]. When the training was complete, the translation system was able to translate 673 of the 795 sentences correctly, for a translation accuracy rate of 84.7%.

Parsing Telegraphic Messages

In developing our system on the MUC-II corpus, we addressed two key problems. First, telegraphic messages induce a greater degree of ambiguity than texts written in natural English. Second, our initial system was unable to parse sentences containing words new to the grammar.

Our solution to the problem of resolving ambiguity in telegraphic messages was applied in initial system development, and is reflected in the accuracy results described above. Additional details of our work in resolving ambiguity are presented in Reference 15. When the rules are defined in terms of syntactic categories (i.e., parts of speech) [16], telegraphic messages with omission introduce a greater degree of syntactic ambiguity than texts without any omitted element. The following examples contain preposition omission:

1410 z (which means "at 1410 Greenwich Mean Time") *hostile raid composition of 19 aircraft.*

Haylor hit by a torpedo and put out of action 8 hours (which means "for 8 hours").

To accommodate sentences with a preposition omission, the grammar needs to allow all instances of noun phrase *NP* to be ambiguous between an *NP* and a prepositional phrase *PP*. The following examples show how allowing the grammar an input in which the copula verb *be* is omitted causes the past tense form of a verb to be interpreted as either the main verb with the appropriate form of *be* omitted as in phrase *a*, or as a reduced relative clause modifying the preceding noun, as in phrase *b*.

Aircraft launched at 1300 z.
(a) Aircraft were launched at 1300 z.
(b) Aircraft which were launched at 1300 z.

Syntactic ambiguity and the resultant misparse induced by such an omission often lead to a mistranslation. For example, the phrase *TU-95 destroyed 220 nm* could be misparsed as an active rather than a passive sentence due to the omission of the verb *was*, and the prepositional phrase *220 nm* could be misparsed as the direct object of the verb *destroy*. The semantic frame reflects these misunderstandings because it is derived directly from the parse tree, as shown in Figure 7. The semantic frame then becomes the input to the generation system, which produces the following nonsensical Korean translation output:

*TU-95-ka 220 hayli-**lul** pakoy-**hayssta**.*
TU-95-NOM 220 nautical mile-OBJ destroyed.

The sensible translation is

*TU-95-ka 220 hayli-**eyse** pakoy-**toyessta**.*
TU-95-NOM 220 nautical mile-LOC was destroyed.

In the examples, *NOM* stands for the nominative case marker, *OBJ* the object case marker, and *LOC* the locative postposition. The problem with the nonsensical translation above is that the object particle *lul* necessarily misidentifies the preceding locative phrase *220 hayli* as the object of the verb. This type of misunderstanding is not reflected in the English paraphrase because English does not have case particles that overtly mark the case role of an *NP*.

Many instances of syntactic ambiguity are resolved

on the basis of the semantic information. However, relying on semantic information requires the parser to produce all possible parses of the input text and forward them to a separate module to resolve the ambiguity, a more complex understanding process.

One way of reducing ambiguity at an early stage of processing without relying on another module is to incorporate the domain-specific semantic knowledge into the grammar. Therefore, we introduce domain-specific categories to restrict the types of phrases that allow omissions. For the example *TU-95 destroyed 220 nm*, we can introduce the following sequence of grammar rules to capture the domain-specific knowledge that a prepositional phrase denoting a location (locative prepositional phrase) allows the preposition *at* to be omitted, and noun phrases that typically occur in a locative prepositional phrase with preposition omission are the ones that denote distance.

locative_PP ->
{in, near, off, on, ...} NP
at_locative

at_locative ->
[at] NP_distance

NP_distance ->
numeric nautical_mile

nautical_mile ->
nm

In the preceding grammar, the first rule states that a locative prepositional phrase *locative_PP* consists of either a preposition (in, near, off, on) and a noun phrase *NP* or it is simply an "at_locative." The second rule says that the prepositional phrase *at_locative* consists of the preposition *at*, which may be omitted as indicated by the brackets and a noun phrase denoting distance *NP_distance*. The third rule states that a distance denoting noun phrase *NP_distance* consists of a numeric expression. The head noun *nautical_mile* is written as *nm* according to the fourth rule. With this grammar, the expression *220 nm* can be correctly understood as a locative prepositional phrase rather than a noun phrase.

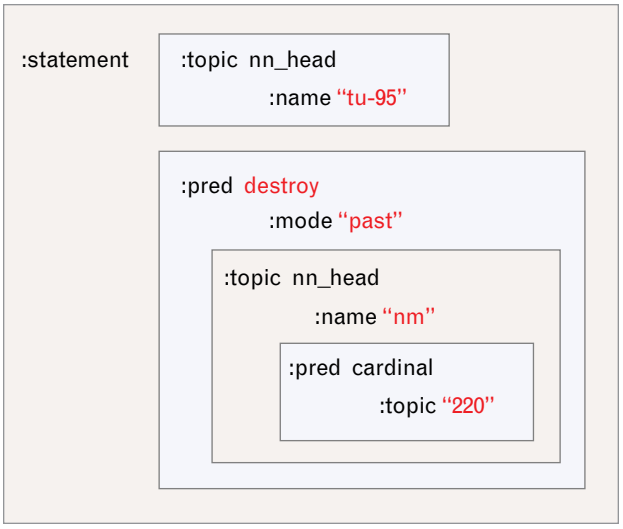
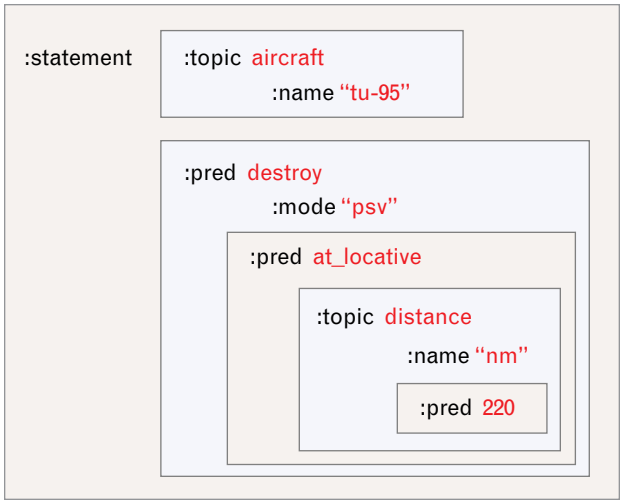


FIGURE 7. Semantic frame for the mistranslation of the input sentence *TU-95 destroyed 220 nm* (which means "TU-95 was destroyed at 220 nm"). The mistranslation occurs because the locative expression *220 nm* is misunderstood as the object of the verb *destroyed*, and the sentence is misunderstood to be in active voice rather than passive voice.

We rely on the training capability of the system to understand the verb "destroyed" as the main verb of the passive sentence in which the verb "was" is omitted, rather than as a verb in a reduced relative clause. Namely, a noun-verb sequence, which is ambiguous between the past tense and past participial form, is more likely to be the subject and the main verb of a passive sentence (i.e., *TU-95 was destroyed*), as opposed to the noun modified by a reduced relative clause (i.e., *TU-95 which was destroyed*).

The introduction of domain-specific semantic grammar and the training capacity of the system allows the input sentence *TU-95 destroyed 220 nm* to be correctly understood as the one equivalent to *TU-95 was destroyed at 220 nm*. Figure 8 shows the semantic frame that reflects the proper understanding. The whole locative prepositional phrase *220 nm* is represented as the predicate at_locative, in which *220 nm* is actually mapped onto the category topic. This semantic frame representation contrasts with Figure 7, which illustrates how the understanding system can mistranslate when no domain-specific knowledge is incorporated into the grammar.



Translation: 티유 95 비행기가 220 해리에서 파괴되었다.

FIGURE 8. Semantic frame for the accurate translation of the input *TU-95 destroyed 220 nm*. Entries in red are replaced by the corresponding vocabulary items in the Korean-generation lexicon. Unlike the semantic frame in Figure 7, the locative expression *220 nm* is understood correctly as the locative expression, and the sentence is translated in passive voice. The correct translation results from the domain-specific knowledge of the grammar and the grammar-training capability of the Korean-understanding system.

Text-Translation Evaluation with Domain-Specific Grammar

After using a data set of 641 training sentences to develop our translation system, we conducted system evaluations on two sets of test sentences that had not been used in training. First, the system was evaluated on a set of 111 sentences comprising 40 messages, called the TEST set. Second, the system was evaluated on another set of data, called TEST′, which was collected from an in-house experiment. For this experiment, the subjects were asked to study a number of MUC-II sentences and create about twenty new MUC-II-like sentences to form data set TEST′. Because our domain-specific grammar at this stage of development could handle only words that had been entered in the grammar, we knew that the performance on TEST, which was certain to contain words unknown to the grammar, would be limited. In creating TEST′, subjects were likely to use words shown to them in the example sentences. Consequently, the percentage of unknown words in TEST′ was lower and the percentage of sentences correctly parsed was greater, as reflected in the following results.

We present evaluation results for our understanding-based translation system on the simple basis of whether correct understanding and generation are achieved. Because our system tends to produce an accurate translation for about 85% of the sentences that are parsed, we have not found it necessary to use more complex evaluation methods like those described in Reference 17. Earlier work in evaluating English-Korean translation systems is described in Reference 18.

Of the 111 sentences in the TEST set, 45 had at least one unknown word, and hence could not be parsed with this domain-specific grammar. Of the remaining 66 sentences, 23 (35%) were parsed, and 20 (87%) of these parsed sentences were correctly translated. However, the system failed on 41% of the new MUC-II sentences in TEST because it could not handle new words at that time. We discuss our solution to the new-word problem in the next section.

The results on the 280 TEST′ sentences were somewhat better because of the much lower frequency of unknown words and the fact that the sentences in TEST′ generally followed the pattern of the training sentences. In TEST′, 41 sentences, or 15%, failed to parse because of the presence of at least one unknown word. Of the remaining 239 sentences, 103 (43%) were parsed, and of these, 88 (85%) were correctly translated.

System Enhancement for New Words: Two-Stage Parsing

Although the language processing is efficient when the system relies on domain-specific grammar rules, some drawbacks exist. Because vocabulary items are entered into the grammar as part of the grammar rules, parsing fails if an input sentence contains new words. For example, the following sentence is not parsed if the word *incorrectly* is not in the grammar:

0819 z unknown contact replied incorrectly.

This drawback was reflected in the initial performance evaluation of our machine-translation system, as discussed previously.

To handle the new word problem, we developed a two-stage parsing strategy. We use domain-specific grammar rules to try parsing on the input word sequence. If parsing fails on the input word sequence because there are words or constructs not covered in the domain-specific grammar, we replace the input words with their parts of speech, and try to parse the part-of-speech sequence by using general grammar rules defined in terms of part of speech rather than individual words.

At the first stage of parsing, the input sentence *0819 z unknown contact replied incorrectly* fails on the domain-specific grammar rules because of the unknown word *incorrectly*. Then part-of-speech tagging takes place, replacing the input word sequence with the corresponding part-of-speech sequence **cardinal** *z* **adjective noun** *replied* **adverb**. At the second stage of parsing, the part-of-speech sequence is successfully parsed, resulting in the parse tree shown in Figure 9. A major difference between the parse tree in Figure 4 and that of Figure 9 is that there are syntactic categories like adjective, noun, and adverb in the lower levels of the latter, whereas the former contains only domain-specific semantic categories at the lower levels. On a closer examination, the input sequence at the second-stage parsing does not consist solely of parts of speech, but of the mix of parts of speech and words. Unless the word is a verb or preposition, we replace the word with its part of speech. By not substituting parts of speech for words that are verbs and prepositions, we avoid ambiguity [15, 19].

Integration of Rule-Based Part-of-Speech Tagger

To accommodate the part-of-speech input to the parser, we integrated the rule-based part-of-speech tagger, developed by E. Brill [20], as a preprocessor to the parser. An advantage of integrating a part-of-speech tagger over a lexicon containing part-of-speech information is that only the former can tag words that are new to the system, which therefore provides a way of handling unknown words.

The rule-based part-of-speech tagger uses the transformation-based error-driven learning algorithm [20, 21]. While most stochastic taggers require a large

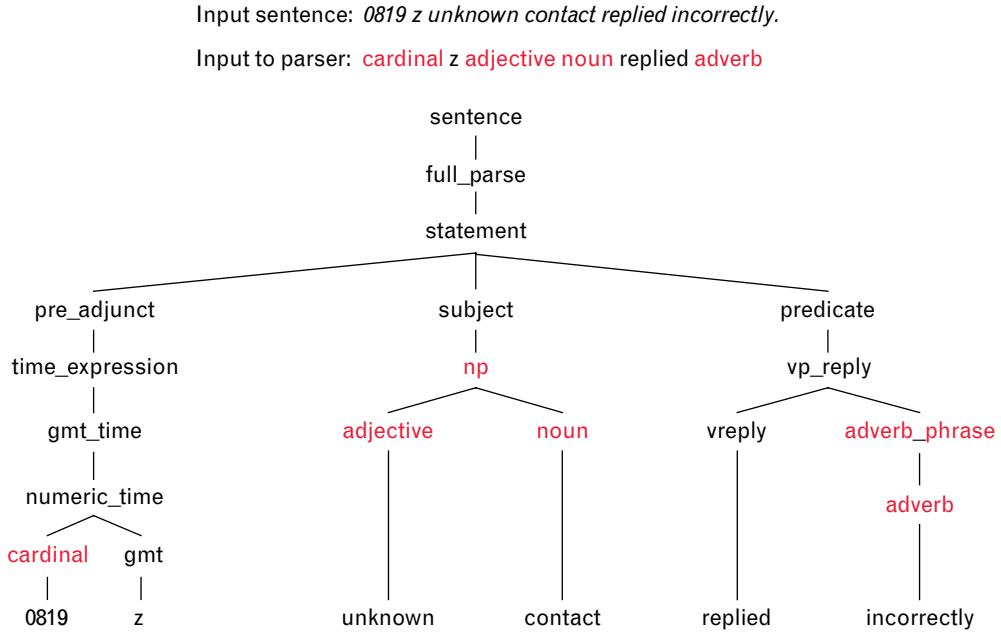


FIGURE 9. Parse tree derived from a mixed sequence of words and part-of-speech tags. The input sentence at the top is converted into the mixed sequence below it by using the part-of-speech tagger. This mixed sequence is the input to the parser. In the parse tree, part-of-speech units are shown in red. When parsing is complete, the part-of-speech units are replaced by the words in the original sentence. For example, *adjective* is replaced by *unknown*, and *adverb* is replaced by *incorrectly*.

amount of training data to achieve high rates of tagging accuracy, this rule-based tagger achieves performance comparable to or higher than that of stochastic taggers, even with a training corpus of modest size. Given that the size of our training corpus is small (7716 words), a rule-based tagger is well suited to our needs.

The rule-based part-of-speech tagger operates in two stages. First, each word in the tagged training corpus has a lexicon entry consisting of a partially ordered list of tags, indicating the most likely tag for that word, and all other tags seen with that word (in no particular order). Every word is initially assigned its most likely tag in isolation. Unknown words are assumed to be nouns, and then cues based upon prefixes, suffixes, infixes, and adjacent word co-occurrences are used to update the most likely tag. Second, after the most likely tag for each word is assigned, contextual transformations are used to improve the accuracy.

We evaluated the tagger performance on the TEST data set both before and after training on the MUC-II corpus. Table 4 presents the results of our evaluations. Tagging statistics before training are based on the lexicon and rules acquired from the Brown corpus and the Wall Street Journal (WSJ) corpus. Tagging statistics after training are divided into two categories, both of which are based on the rules acquired from training data sets of the MUC-II corpus. The only difference between the two is that in one case (after training I) we use a lexicon acquired from the MUC-II corpus, and in the other case (after training II) we use a lexicon acquired by combining the Brown corpus, the WSJ corpus, and the MUC-II corpus.

Table 4 shows that the tagger achieves a tagging accuracy of up to 98% after training and using the combined lexicon. The tagging accuracy for unknown words ranges from 82% to 87%. These high rates of tagging accuracy are largely due to two factors: the combination of domain-specific contextual rules obtained by training the MUC-II corpus with general contextual rules obtained by training the WSJ corpus; and the combination of the MUC-II lexicon with the WSJ corpus lexicon.

Table 4. Rule-Based Part-of-Speech Tagger Evaluation on the TEST Data Set

Training status	*Tagging accuracy*
Before training	1125 ÷ 1287 (87.4%)
After training I	1249 ÷ 1287 (97%)
After training II	1263 ÷1287 (98%)

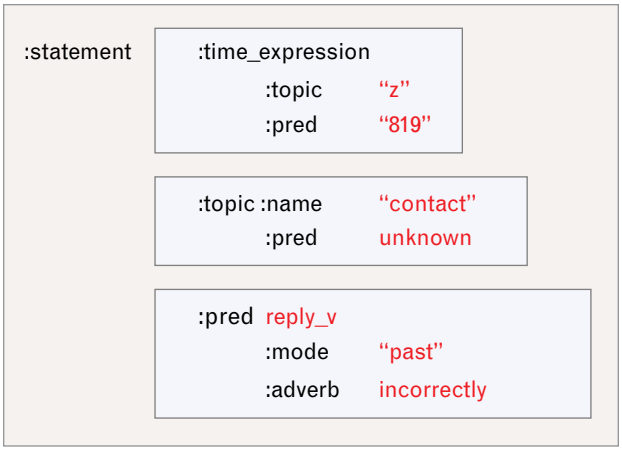
Adapting the Language-Understanding System

The language-understanding system derives the semantic-frame representation from the parse tree. The terminal symbols (i.e., words in general) in the parse tree are represented as vocabulary items in the semantic frame. Once we have allowed the parser to take a part of speech as the input, the parts of speech (rather than actual words) will appear as terminal symbols in the parse tree, and hence as the vocabulary items in the semantic-frame representation. We adapted the system so that the part-of-speech tags are used for parsing, but are replaced with the original words in the final semantic frame. Figure 10 illustrates the semantic frame produced by the adapted system for the input sentence *0819 z unknown contact replied incorrectly*. Once the semantic frame has been produced, as above, generation proceeds as usual.

Summary of Results for English-to-Korean Translation on Naval Messages

After integrating the part-of-speech tagger into the system to implement the two-stage parsing technique, we reevaluated the system on the TEST and TEST′ data. The experimental results show that by adopting a two-stage parsing technique, we increased the parsing coverage from 35% to 77% on the TEST data, and from 43% to 82% on the TEST′ data.

Figure 11 summarizes the results on all training and test sentences (including TEST and TEST′). With the integration of the two-stage procedure that includes the part-of-speech tagger, we have been able to increase the translation accuracy on this domain to 80%. We believe that this level of accuracy, when combined with a fall-back position that provides word-for-word translations for sentences that cannot be parsed, would be of operational value to human translators and would significantly reduce their work



Paraphrase: 819 Z unknown contact replied incorrectly.

Translation: 8시 19분 정체불명의 물체가 부정확하게 응답했다.

FIGURE 10. Accurate semantic frame derived from the parse tree with the part-of-speech input sequence. Entries in red are replaced by the corresponding vocabulary items in the Korean-generation lexicon.

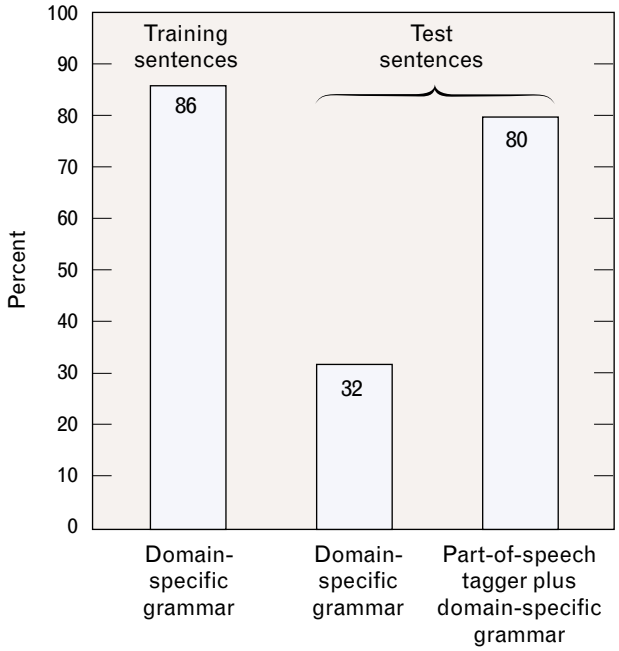


FIGURE 11. Summary of English-to-Korean translation results on the MUC-II training and test data, which includes both TEST and TEST′. Use of the part-of-speech tagger, primarily to solve the unknown word problem, substantially enhances translation performance on the test data.

load. This hypothesis remains to be tested, and to be truly useful the translation also needs to be extended beyond the MUC-II corpus to more operational domains. Work along these lines is described later in this article.

Speech Translation in the MUC-II Domain

Although our primary emphasis in working on the MUC-II domain was text translation, we also developed a speech-translation system for a subset of this domain. In our original speech-translation work we had used a hidden Markov model (HMM) speech recognizer that had been developed earlier at Lincoln Laboratory. For the MUC-II domain, we developed a new HMM speech recognizer by building upon the HMM Toolkit (HTK) software system originally developed at Cambridge University [22, 23]. Given a vocabulary, a grammar, training data, and a number of key parameters of the HMM system, the HTK system can be used to build a speech recognizer.

The speech training data used for the MUC-II speech recognizer was drawn from an independent data source—the TIMIT general English corpus [24]. The HTK system was used to train speaker-independent acoustic triphone models on the TIMIT corpus. Separate gender acoustic models were generated from a total of 233 minutes of data from 326 male and 136 female speakers. The core HMM models were three-state, state-tied models, with one Gaussian per state. Despite the large size of the TIMIT corpus, 22% of the triphone models that occurred in the MUC-II sentences did not occur in TIMIT. For those triphone models, back-off monophone models were used [23].

For speech recognition on the MUC-II corpus, a simple language model was generated in the form of a word-pair grammar (WPG) constructed on the basis of the text of 207 sentences drawn from the MUC-II corpus. A WPG is a special case of a bigram grammar [23]; the WPG specifies the set of words WF that are allowed to follow any given word WI in the vocabulary, and equalizes the probabilities of a transition from a given WI to any of the WF. Many vocabulary items in the MUC-II corpus, particularly naval terms, abbreviations, and acronyms, were not included in our available phonetic dictionaries. Phonetic expansions that were created by hand for about 200 such items were added to the dictionary. In summary, for the purposes of this MUC-II speech-translation ex-

periment, the size of the vocabulary was about 500 words and the perplexity (geometric mean of the number of words that can follow a given word) of the WPG on the data was 6.4 [25].

To test this system, we arranged to have 207 sentences recorded by one male speaker and one female speaker. The speaker-independent acoustic models were completely independent of the test data, but the word-pair grammar was developed for this particular set of sentences. (These speech-translation results were obtained by using the domain-specific MUC-II parsing system, prior to the work on the part-of-speech tagger). Figure 12 shows the performance results for speech-recognition and speech-translation experiments on the 207 sentences. With a word error rate of 7%, the sentence accuracy (percentage of sentences perfectly recognized with no word errors) was 54%. To separate the effects of speech-recognition performance and text-translation performance, we evaluated speech-translation performance only on those sentences which had been translated correctly by the text-translation system. For this group, the percentage of sentences correctly translated (85%) is higher than the percentage of sentences that were perfectly recognized (54%).

The reason for this higher translation rate is that many of the speech-recognition errors are caused by omissions or incorrect recognition of items such as articles or plurals. Our translation system, which had been developed to deal with telegraphic text and to handle number disagreement within sentences, was tolerant of the errors that are often produced by speech recognizers. For descriptions of other work in English-Korean speech translation, see References 26 through 29.

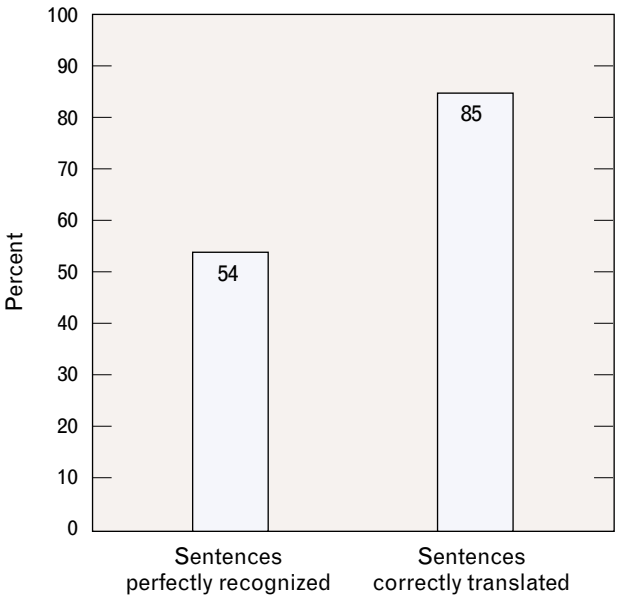


FIGURE 12. Speech-recognition and translation performance on MUC-II naval message data. The sentences averaged 12 words in length and 54% of the sentences were perfectly recognized. Speech-translation performance, shown only for those sentences which were translated correctly by the text-translation system, is 85%, which demonstrates the capability of the parser to handle errors in the text input that it receives.

Korean-to-English Translation

In the early stages of our project, we learned that SYSTRAN, Inc., a company with a long and successful history of work in machine translation, had just embarked on a Department of Defense (DoD)–sponsored project in Korean-to-English translation [30, 31]. Rather than develop the Korean-to-English part of the system ourselves, we chose to gain leverage from that work, and initiated a subcontract with SYSTRAN to adapt their Korean-to-English system to the MUC-II domain. Although this made our two-way system asymmetric in that the SYSTRAN system uses a transfer approach instead of an interlingua approach, we decided that the advantage in expediting development was worthwhile.

To provide a Korean MUC-II corpus, we separately contracted with another organization to produce human translations of 338 MUC-II corpus sentences into Korean. We then supplied this Korean corpus to SYSTRAN for their training, developing, and testing. From this Korean corpus, 220 sentences were used for training and 118 sentences were used for testing. During training, significant changes were made to all modules because the system had never dealt with telegraphic messages of this type. The system dictionary, which had about 20,000 Korean entries but lacked many terms in naval operations reports, was augmented to include the new words in MUC-II. We found that performance on the MUC-II Korean-to-English task was good; 57% of the translations of the test sentences were at least close to

being correct, and for 97% of the sentences, a human translator could extract the essential meaning from the translation output. After the work on the MUC-II corpus, SYSTRAN extended the Korean-to-English system to another domain, which we obtained from a bilingual English-Korean combat officer's briefing course, and which was similar in vocabulary size to the MUC-II corpus. Korean-to-English performance on this domain was similar to the performance on the MUC-II corpus.

For demonstrations, we also developed a small-scale Korean-to-English speech-translation subsystem for the MUC-II domain. We collected training data from two Korean speakers, and using HTK we developed a rudimentary Korean speech recognizer with about a 75-word vocabulary. With this system we were able to demonstrate translation of Korean speech when we fed the output of the Korean speech recognizer into the Korean-to-English translator. This demonstration was of high interest to many observers, but we cautioned them that a great deal of work was still required to make a truly effective Korean speech-recognition and translation system.

Recently, we developed a preliminary working subsystem for interlingua-based Korean-to-English translation that includes a Korean analysis grammar for TINA. This makes CCLINC, to our knowledge, the first machine-translation system that implements two-way, interlingua-based English-Korean translation. Other related ongoing work includes research in Korean language understanding to produce an interlingua representation [32] and transfer-based Korean-to-English translation [33].

System Development on C2W Domain and Treatment of Complex Sentences

While we were carrying out our translation-system development on the MUC-II corpus, we worked with personnel in CFC Korea to obtain data that would be more directly typical of translation applications in that environment. In November 1996, we obtained data for a new task domain in the form of an English and Korean Command-and-Control Warfare (C2W) handbook. The handbook provided us with over two hundred pages of new material in each language, used routinely by CFC, in an electronic format. It contained a vocabulary of 8500 words and 3400 sentences, each with an average size of 15 words.

The new material created challenges. In particular, the sentences were longer and more complex than those in the MUC-II corpus. We were motivated by the C2W corpus to confront some of the difficult challenges in machine translation, which in turn led us to develop a more complete and robust translation system, as described below.

The C2W Data

For the C2W data, we focused our effort on developing a technique to handle complex sentences that includes fragmentation of a sentence into meaningful subunits before parsing, and composition of the corresponding semantic-frame fragments into a single unified semantic frame. Compared to those of the MUC-II corpus, the sentences in the C2W data are much longer and are written in grammatical English:

> *A mastery of military art is a prerequisite to successful practice of military deception but the mastery of military deception takes military art to a higher level.*

> *Although opportunities to use deception should not be overlooked, the commander must also recognize situations where deception is not appropriate.*

> *Often, the skillful application of tenets of military operations-initiative, agility, depth, synchronization and versatility, combined with effective OPSEC, will suffice in dominating the actions of the opponent.*

Such long, complex sentences are difficult to parse. Acquiring a set of grammar rules that incorporate all instances of complex sentences is not easy. Even if a complex sentence is covered by the grammar, a long sentence induces a higher degree of ambiguity than a short sentence, requiring a much longer processing time. To overcome the problems posed by understanding of complex sentences, we have been developing sentence-fragmentation and semantic-frame composition techniques. We briefly describe these techniques below.

Sentence Fragmentation

For sentence fragmentation, the input sentence is first parsed with the Apple Pie Parser, a system developed at New York University. This system runs on a corpus-based probabilistic grammar and produces the parse tree with the highest score among the trees derived from the input [34]. Our sentence-fragmentation algorithm [35] is applied to the Apple Pie Parser output, producing sentence fragments that each form a meaningful unit. Figure 13 provides an example of the Apple Pie Parser output and fragmenter output.

As the Apple Pie Parser output and the fragmented output show, the fragmentation algorithm extracts elements with category labels such as *TOINF* and *SBAR*, each of which form an independent meaning unit [36]. Once a fragment is extracted from the higher-level category, the label of the extracted element is left behind to compose the component semantic frames at a later stage. In Figure 13, two fragments have been extracted from the input sentence—an adverbial clause (*although opportunities to use deception should not be overlooked*) whose category label in the parsing output is *SBAR,* and a relative clause (*where deception is not appropriate*) whose category label is also *SBAR*. Labels of these two extracted elements are left in the first fragment as *adverbc1* and *relclause1*, respectively. Likewise, an infinitival clause whose category label in the parsing output is *TOINF* has been extracted from the adverbial clause, leaving its label *toinfc1* in the second fragment.

Understanding Fragments and Semantic-Frame Composition

Once sentence fragments are generated according to the fragmentation algorithm, each fragment is processed by the language-understanding system, TINA, to produce the parse tree and the semantic frame. The semantic frames of each fragment are then combined to capture the meaning of the original input sentence. Figure 14 illustrates this process. The language-understanding system derives the parse tree and the corresponding semantic frame for each fragment, and the semantic frames for each frame are combined. The combined frame then becomes the input to the GENESIS language-generation system.

Input: *Although opportunities to use deception should not be overlooked, the commander must also recognize situations where deception is not appropriate*

Apple Pie Parser output

```
(S (SBAR although
        (SS (NP opportunities
                (TOINF (VP to
                        (VP use
                            (NPL deception)))))
        (VP should
            not
            (VP be
                (VP overlooked)))))
   -COMMA-
   (NPL the commander)
   (VP must
      (ADVP also
      (VP recognize)
         (NP (NPL situations)
            (SBAR (WHADVP where)
               (SS (NPL deception)
                  (VP is
                     (ADJP not appropriate))))))))
```

adverbc1 comma the commander must also recognize situations **relclause1**
adverbc1 although opportunities **toinfc1** should not be overlooked
relclause1 where deception is not appropriate
toinfc1 to use deception

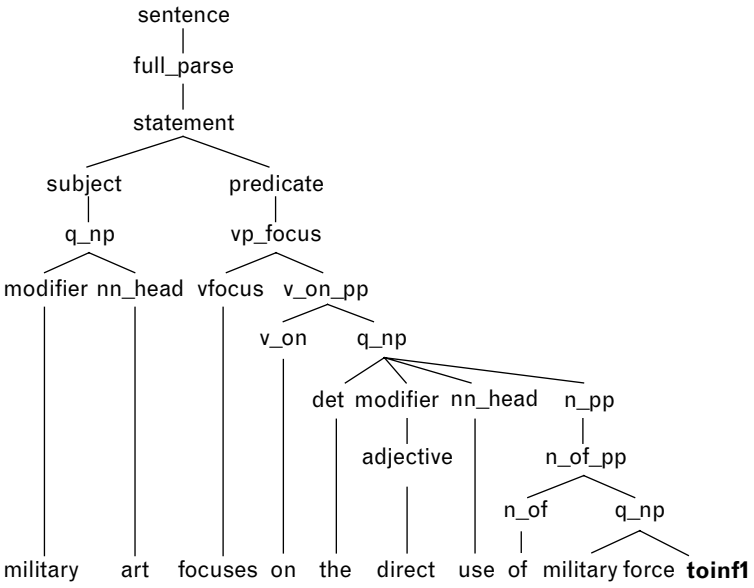
FIGURE 13. Operation of the sentence-fragmentation algorithm. From top to bottom are shown the input sentence, the Apple Pie Parser output, and the four fragments into which the input sentence is broken via the operation of the fragmentation algorithm on the output of the Apple Pie Parser.

Robust Translation System

The robust translation system copes with the technical challenges discussed in the prior section. Figure 15 illustrates the process flow of this system. Given an input sentence, the TINA parser tries to parse the input sentence either on the word sequence or the mix of part of speech and the word sequence. If the parsing succeeds, then TINA produces the semantic frame. If not, the input sentence is fragmented into several subunits. The TINA parser is then applied to each fragment. If parsing succeeds, the semantic frame for the parsed fragment is produced. If not, a

Input: *Military art focuses on the direct use of military force to impose one's intent on an opponent.*

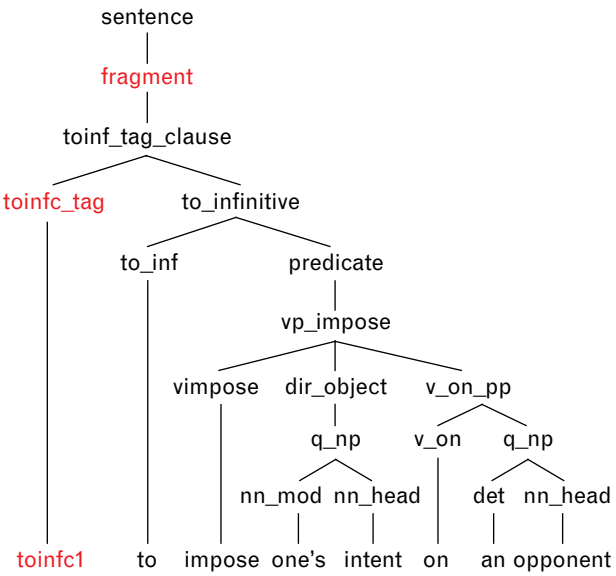
(a) Fragment 1: *Military art focuses on the direct use of military force* **toinfc1**



(b)

```
:statement  :topic art
               :pred military
            :pred focus_v
               :mode "present"
               :number "third"
               :pred v_on
                  :topic use
                     :pred direct
                        :pred n_of
                           :topic force
                           :pred military
                           to_infinitive "toinfc1"
```

(c) Fragment 2: **toinfc1** *to impose one's intent on an opponent*



(d)

```
fragment  :tag "toinfc1"
          :to_infinitive to_inf
          :pred impose
             :mode "root"
             :topic intent
                :pred one's
             :pred v_on
                :topic opponent
```

(e)

```
:statement  :topic art
               :pred military
            :pred focus_v
               :pred v_on
                  :topic use
                     :pred direct
                        :pred n_of
                           :topic force
                              :pred military
            :to_infinitive to_inf
               :pred impose
                  :topic intent
                     :pred one's
                     :pred v_on
                        :topic opponent
```

Paraphrase: *Military art focuses on the direct use of military force to impose one's intent on an opponent.*

Paraphrase:
군사전술은 상대방에 자신의 의도를 강요하는 군사병력의 직접사용에 초점을 둔다.

FIGURE 14. Operation of the robust translation system for parsing and understanding sentence fragments, composing the results into a combined semantic frame, and producing the final translation and paraphrase. In this example, two fragments are processed. The parts of the figure are (a) parse tree 1, (b) semantic frame 1, (c) parse tree 2, (d) semantic frame 2, and (e) combined semantic frame with paraphrase and translation. The labels in red represent the categories that have been extracted by the fragmentation algorithm.

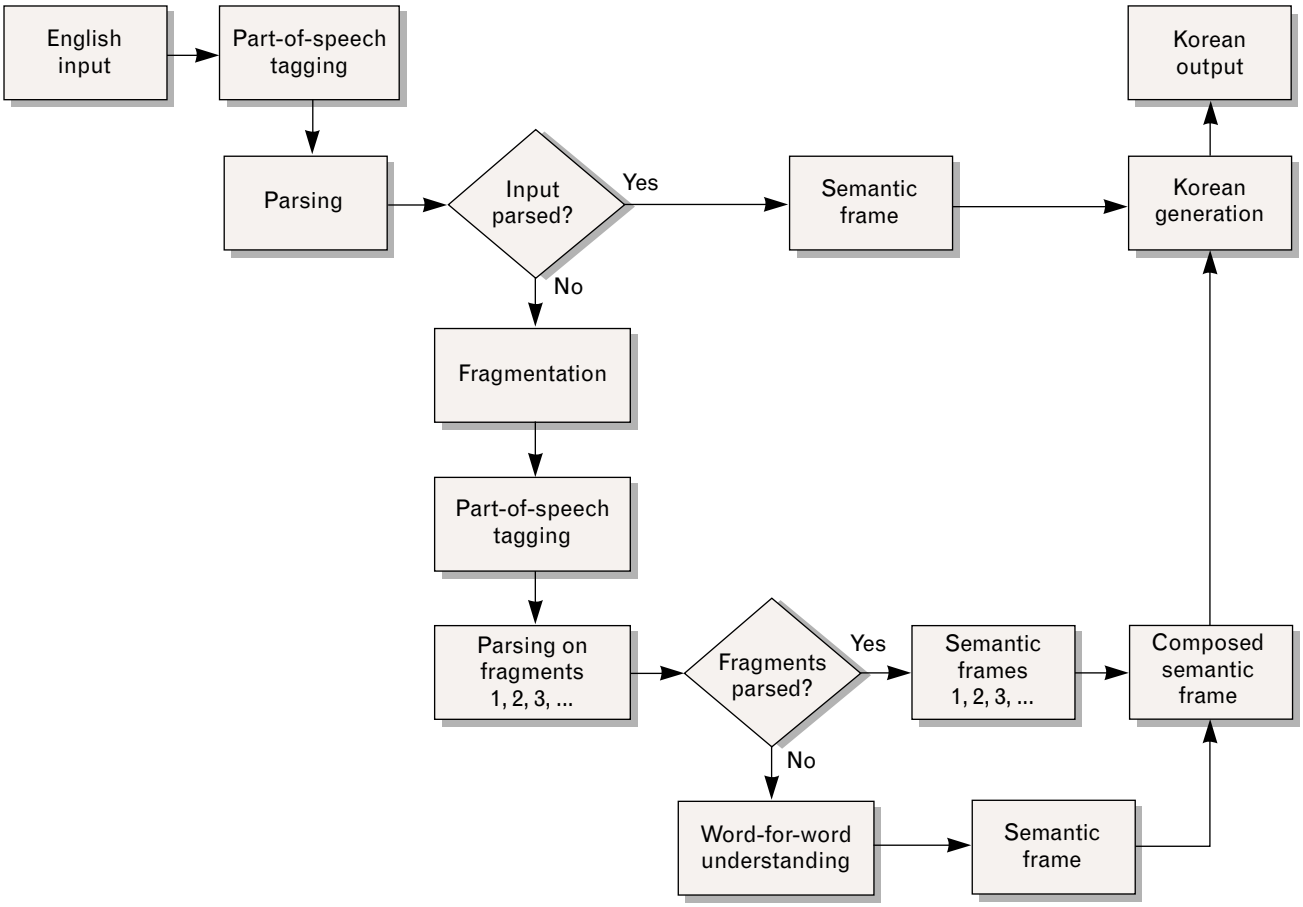


FIGURE 15. Process flow of robust translation system. Given an input sentence, the translation system assigns parts of speech to each word. Parsing takes place with the part-of-speech sequence as input. If parsing succeeds at this stage, the corresponding semantic frame is produced. If parsing does not succeed, the input sentence is fragmented, and parsing takes place on each fragment. Once parsing and semantic-frame generation of all of the fragments has been completed, the semantic frames for the fragments are composed. Generation proceeds with the composed semantic frame as input.

form of word-for-word understanding is applied to the fragment, which results in a semantic frame that serves as a place holder. After all the fragments have been understood, the semantic frames for the fragments are composed, and generation proceeds from the composed semantic frame.

Our initial development of the robust translation system shown in Figure 15 was done on the C2W data, which, as mentioned earlier, included many complex sentences with an average sentence length of fifteen words. With an early version of the robust parser on 286 sentences of C2W data, 158 (55%) of these sentences were fully translated. Of these 158 fully translated sentences, 64 (22%) input sentences were both fully fragmented by the system and fully parsed. This approach has increased the parsing coverage and translation rate on complex sentences in our current translation material of Commander-in-Chief (CINC) daily briefings. We believe this approach provides aid to the user when full translation is not possible.

Software Implementation

Figure 16 illustrates the major modules of the current software implementation of CCLINC. The top-level module of the CCLINC system, the graphical user interface, interacts with both the English-to-Korean and Korean-to-English translation systems. The English-to-Korean translation system consists of three subsystems, namely, speech recognition, language understanding, and language generation. The language-understanding system interacts with two subsystems

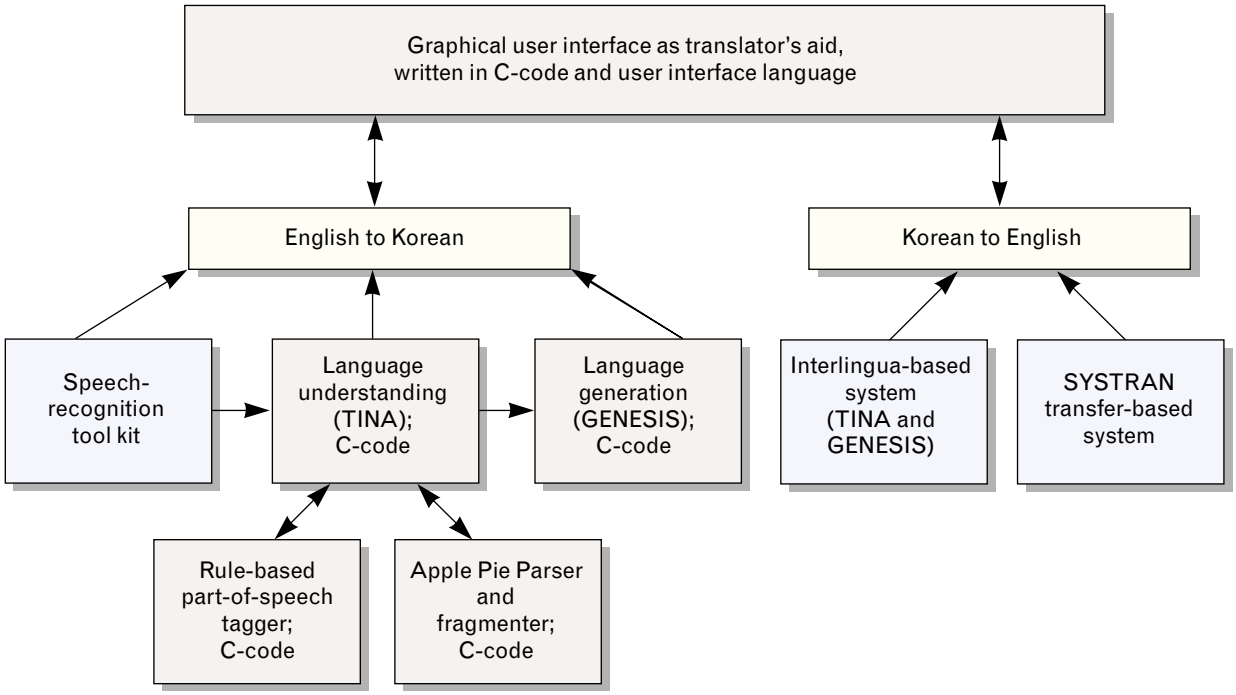


FIGURE 16. Major software modules of the current implementation of the CCLINC automated translation system. The graphical user interface interacts with both the English-to-Korean and Korean-to-English translation systems. The English-to-Korean system consists of three subsystems: speech recognition, language understanding, and language generation. The language-understanding system interacts with two subsystems for robust processing: the rule-based part-of-speech tagger and the Apple-Pie-Parser and fragmenter. The Korean-to-English system consists of two systems that employ different approaches to machine translation: the interlingua-based system being developed at Lincoln Laboratory and the transfer-based system developed by SYSTRAN under a subcontract.

for robust processing: the rule-based part-of-speech tagger to handle unknown words, and the Apple Pie Parser and sentence fragmenter to handle complex sentences. The Korean-to-English translation system includes two subsystems that employ different approaches to machine translation: the transfer-based Korean-to-English system developed by SYSTRAN, and our interlingua-based Korean-to-English system under development.

The translation system operates on UNIX platforms, and has been run on workstations under Solaris and on a Pentium laptop under PC Solaris and LINUX (Solaris and LINUX are versions of UNIX). The system with the part-of-speech tagger and the fragmenter uses about 50 MB of memory and, depending on the size of the data files used by each module, the memory usage varies from 80 MB to 100 MB. The processing times for translation rely on the task domain, the grammar, the length and complexity of the input sentence, and the processor being used. For all the tasks we have run, translation is generally completed within a few seconds per sentence.

As an example, text-translation processing times for the MUC-II domain, with the system running on a 125-MHz Hypersparc workstation ranged from 1.7 sec for an average sentence length of 12 words, to 2.3 sec for a 16-word sentence, to about 4 sec for a complex sentence containing 38 words. For the same processor, English speech recognition in the MUC-II domain runs in about two times real time. We caution the reader that the processing efficiency of a system is determined by various factors that include CPU speed, machine memory size, the size of data-grammar-lexicon files required by the system, and the complexity of the input data, which largely determines the parsing time. For a general introduction to the efficiency issue of different parsing techniques, see Reference 37.

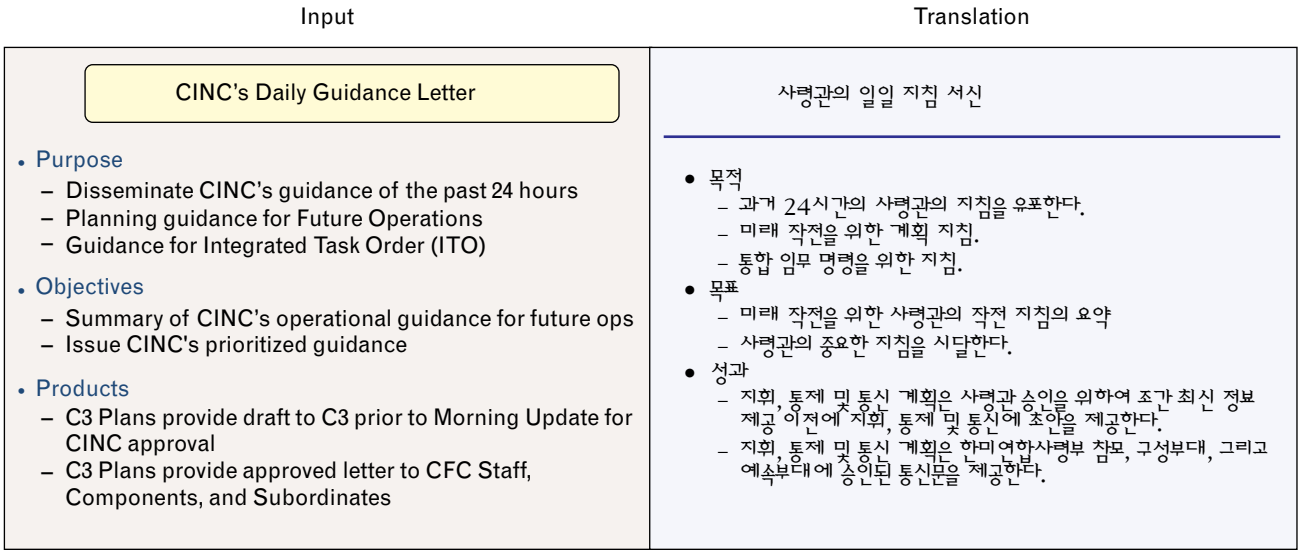


FIGURE 17. Sample slide of Commander-in-Chief (CINC) briefing material, in which each English sentence has been translated by CCLINC. The development of CCLINC to achieve high performance for a large variety of such material is the focus of our current work.

System Demonstrations and Task Definitions

From the outset, this project has focused on developing an automated translation system that would be useful for military coalition forces in Korea. Therefore, we have actively pursued user feedback in demonstrating and testing our technology in the user environment, and iteratively adjusted our efforts to respond to this feedback. These system demonstration and technology activities include our first visit to Korea in September 1994, a system demonstration on board the *USS Coronado* at the June 1996 Rim of the Pacific Coalition (RIMPAC 96) exercises, and a system demonstration at CFC Korea in April 1997 in conjunction with the Reception, Staging, Onward Movement, and Integration (RSO&I) exercises. During these exercises, we tested the system on new operational data comprising intelligence spot reports, intelligence summaries, and excerpts from CINC daily guidance letters and briefings.

The mission was successful in winning support and encouragement from high-ranking officers and military personnel who would be directly working with the system. We held discussions with CFC translators, operations personnel, and flag officers to help us define tractable translation tasks, with the CINC briefings becoming our highest priority. We also brought back samples of key operational material to be used in system development.

As a result of the RSO&I exercises, we are developing the system to translate CINC daily briefings, which consist of slides and speaker notes that are used by a presenter to explain each slide. The speaker notes that accompany a slide include longer, more complex sentences, and hence our robust translation approach of handling complex sentences is critical for translating this material. Our ultimate goal in training the translation system on CINC briefings is to allow CFC personnel to focus more on the content of the briefings than on translation. Figure 17 illustrates a briefing slide in which each of the English sentences has been translated into Korean by our system. Although the translation is accurate for this slide, a substantial amount of system development on similar material is needed before the translation accuracy on new CINC briefing material will be high enough for effective operational use. We plan to bring the system to Korea by spring of 1998 for tests on new CINC briefing material.

Summary and Plans

Substantial progress has been made in automated English-Korean translation. Major accomplishments in this project to date include (1) development and fea-

sibility demonstrations of automated two-way English-Korean text and speech translation for military messages; (2) development of a modular, interlingua-based translation system that is extendable to multiple languages and to human interaction with C4I systems; (3) development of a multistage, robust translation system to handle complex text; (4) development of an integrated graphical user interface for a translator's aid; and (5) several successful demonstrations and technology transfer activities, including participation in the RIMPAC 96 coalition exercise on board the *USS Coronado* and the RSO&I coalition exercises at CFC Korea.

Our plans for the future involve extending the system capability to additional application domains, including translation of operations orders and operations plans. We will expand our recently begun effort in developing an interlingua-based Korean-to-English translation system by using the same understanding-based technology that we have applied to English-to-Korean translation. Ultimately, we hope to integrate the system's understanding capabilities with C4I systems to allow multilingual human-computer and human-human communication. One such application would involve a report translated by the system for communication among coalition partners. The report's meaning, captured in the semantic frame, would be conveyed to the C4I system to update databases with situation awareness information.

Acknowledgments

This project has benefited from the contributions of individuals inside and outside Lincoln Laboratory, and we particularly appreciate the contributions of and interactions with people in the DoD and research communities. We would like to cite the contributions of the following people: Ronald Larsen, Allen Sears, George Doddington, John Pennella, and Lt. Comdr. Robert Kocher, DARPA; Seok Hong, James Koh, Col. Joseph Jaremko, Lt. Col. Charles McMaster, Lt. David Yi, and Willis Kim, U.S. Forces Korea-Combined Forces Command; Beth Sundheim and Christine Dean, NRaD; Capt. Richard Williams and Neil Weinstein, *USS Coronado*, Command Ship of the Third Fleet; Victor Zue, James Glass, Ed Hurley, and Christine Pao, MIT Laboratory for Computer Science, Spoken Language Systems Group; Key-Sun Choi, Korean Advanced Institute for Science and Technology; Ralph Grishman, New York University; Martha Palmer, University of Pennsylvania; and Jerry O'Leary, Tom Parks, Marc Zissman, Don Chapman, Peter Jung, George Young, Greg Haber, and Dennis Yang, Lincoln Laboratory.

REFERENCES

1. W.J. Hutchins and H.L. Somers, *An Introduction to Machine Translation* (Academic, London, 1992).
2. D. Tummala, S. Seneff, D. Paul, C. Weinstein, and D. Yang, "CCLINC: System Architecture and Concept Demonstration of Speech-to-Speech Translation for Limited-Domain Multilingual Applications," *Proc. 1995 ARPA Spoken Language Technology Workshop*, *Austin, Tex., 22–25 Jan. 1995,* pp. 227–232.
3. C. Weinstein, D. Tummala, Y.-S. Lee, and S. Seneff, "Automatic English-to-Korean Text Translation of Telegraphic Messages in a Limited Domain," *16th Int. Conf. on Computational Linguistics '96, Copenhagen, 5–9 Aug. 1996,* pp. 705–710; *C-STAR II Proc. ATR International Workshop on Speech Translation, 10–11 Sept. 1996, Kyoto, Japan.*
4. K.-S. Choi, S. Lee, H. Kim, D.-B. Kim, C. Kweon, and G. Kim, "An English-to-Korean Machine Translator: MATES/EK," *Proc. 15th Int. Conf. on Computational Linguistics* **I**, *Kyoto, Japan, 5–9 Aug. 1994,* pp. 129–131.
5. S. Seneff, "TINA: A Natural Language System for Spoken Language Applications," *Computational Linguistics* **18** (1), 1992, pp. 61–92.
6. J. Glass, J. Polifroni and S. Seneff, "Multilingual Language Generation across Multiple Domains," *1994 Int. Conf. on Spoken Language Processing, Yokohama, Japan, 18–22 Sept. 1994,* pp. 983–986.
7. J. Glass, D. Goodine, M. Phillips, M. Sakai, S. Seneff, and V. Zue, "A Bilingual VOYAGER System," *Proc. Eurospeech, Berlin, 21–23 Sept. 1993,* pp. 2063–2066.
8. V. Zue, S. Seneff, J. Polifroni, H. Meng, J. Glass, "Multilingual Human-Computer Interactions: From Information Access to Language Learning," *Proc. Int. Conference on Spoken Language Processing, ICSLP-96* **4**, *Philadelphia, 3–6 Oct. 1996,* pp. 2207–2210.
9. D. Yang, "Korean Language Generation in an Interlingua-Based Speech Translation System," Technical Report 1026, MIT Lincoln Laboratory, Lexington, Mass., 21 Feb. 1996, DTIC #ADA-306658.
10. S. Martin, *A Reference Grammar of Korean* (Tuttle, Rutland, Vt., 1992).
11. C. Voss and B. Dorr, "Toward a Lexicalized Grammar for Interlinguas," *J. Machine Translation* **10** (1–2), 1995, pp. 143–184.
12. H.-M Sohn, *Korean* (Routledge, London, 1994).
13. B.M. Sundheim, "Plans for a Task-Oriented Evaluation of Natural Language Understanding Systems," *Proc. DARPA Speech and Natural Language Workshop, Philadelphia, 21–23 Feb. 1989,* pp. 197–202.
14. B.M. Sundheim, "Navy Tactical Incident Reporting in a Highly Constrained Sublanguage: Examples and Analysis," Technical Document 1477, Naval Ocean Systems Center, San Diego, 1989.
15. Y.-S. Lee, C. Weinstein, S. Seneff, and D. Tummala, "Ambiguity Resolution for Machine Translation of Telegraphic Messages," *Proc. Assoc. for Computational Linguistics*, *Madrid, 7–12 July 1997.*
16. R. Grishman and J. Sterling, "Analyzing Telegraphic Messages," *Proc. DARPA Speech and Natural Language Workshop, Philadelphia, 21–23 Feb. 1989,* pp. 204–208.
17. J.S. White and T.A. O'Connell, "Evaluation in the ARPA Machine Translation Program: 1993 Methodology," *Proc. Human Language Technology Workshop, Plainsboro, N.J., 8–11 Mar. 1994,* pp. 135–140.
18. W.B. Kim and W.B. Rhee, "Machine Translation Evaluation," MITRE Working Note WN 94W0000198, Nov. 1994.
19. E. Brill and P. Resnik, "A Rule-Based Approach to Prepositional Phrase Attachment Disambiguation," *Proc. COLING-1994, Kyoto, Japan, 5–9 Aug. 1994.*
20. E. Brill, "A Simple Rule-Based Part of Speech Tagger," *Proc. Third Conf. on Applied Natural Language Processing, ACL, Trento, Italy, 31 Mar.–3 Apr. 1992,* pp. 152–155.
21. E. Brill, "Transformation-Based Error-Driven Learning and Natural Language Processing: A Case Study in Part-of-Speech Tagging," *Computational Linguistics* **21** (4), 1996, pp. 543–565.
22. S.J. Young, P.C. Woodland, and W.J. Byrne, *HTK Version 1.5: User, Reference & Programmer Manual* (Cambridge University Engineering Department and Entropic Research Laboratories, Inc., Sept. 1993).
23. P.C. Woodland, J.J. Odell, V. Valtchev, and S.J. Young, "Large Vocabulary Continuous Speech Recognition Using HTK," *Proc. ICASSP '94* **2**, *Adelaide, Australia, 19–22 Apr. 1994,* pp. 125–128.
24. W.M. Fisher, G.R. Doddington, and K.M. Goudie-Marshall, "The DARPA Speech Recognition Research Database: Specifications and Status," *Proc. DARPA Workshop on Speech Recognition*, *Palo Alto, Calif., Feb. 1986,* pp. 93–99.
25. S. Young, "A Review of Large-Vocabulary Continuous-Speech Recognition," *IEEE Signal Process. Mag.* **13** (5),1996, pp. 45–57.
26. A. Lavie, A. Waibel, L. Levin, M. Finke, D. Gates, M. Gavaldo, T. Zeppenfeld, and P. Zhan, "JANUS-III: Speech-to-Speech Translation in Multiple Languages," *ICASSP-97 Proc.* **I**, *21–24 Apr. 1997, Munich,* pp. 99–102.
27. *C-STAR II Proc. ATR Int. Workshop on Speech Translation, Kyoto, Japan, 10–11 Sept. 1996.*
28. C-STAR Web Site, http://www.is.cs.cmu.edu/cstar/
29. J.-W. Yang and J. Park, "An Experiment on Korean-to-English and Korean-to-Japanese Spoken Language Translation," *ICASSP-97 Proc.* **I**, *Munich, 21–24 Apr. 1997,* pp. 87–90.
30. D.A. Bostad, "Aspects of Machine Translation in the United States Air Force," in *AGARD: Benefits of Computer Assisted Translation to Information Managers and End Users, N91-13352, June 1990.*
31. W.J. Hutchins and H.L. Somers, "Systran," in *An Introduction to Machine Translation* (Academic, London, 1992), pp. 175–206.
32. B. Dorr, "LCS-BASED Korean Parsing and Translation," TCN No. 95008, Institute for Advanced Computer Studies and Department of Computer Science, University of Maryland, 1997.
33. H.-S. Park, "Korean Grammar Using TAGs," IRCS Report 94-28, Institute for Research in Cognitive Science, University of Pennsylvania, 1994.
34. S. Sekine and R. Grishman, "A Corpus-Based Probabilistic Grammar with Only Two Non-Terminals," *Fourth Int. Workshop on Parsing Technology*, *Prague, 1995,* pp. 216–223.
35. J.-T. Hwang, "A Fragmentation Technique for Parsing Complex Sentences for Machine Translation," M. Eng. Thesis, MIT Department of EECS, June 1997.
36. V. Fromkin and R. Rodman, *An Introduction to Language,* 4th ed. (Holt, Rinehart and Winston, Fort Worth, Tex., 1988).
37. J. Allen, *Natural Language Understanding*, 2nd ed. (Benjamin/Cummings, Redwood City, Calif., 1995).

CLIFFORD J. WEINSTEIN leads the Information Systems Technology group and is responsible for initiating and managing research programs in speech technology, machine translation, and information system survivability. He joined Lincoln Laboratory as an MIT graduate student in 1967, and became group leader of the Speech Systems Technology group (now Information Systems Technology group) in 1979. He has made technical contributions and carried out leadership roles in research programs in speech recognition, speech coding, machine translation, speech enhancement, packet speech communications, information system survivability, integrated voice-data communication networks, digital signal processing, and radar signal processing. Since 1986, Cliff has been the U.S. technical specialist on the NATO RSG10 Speech Research Group, authoring a comprehensive NATO report and journal article on applying advanced speech technology in military systems. In 1993, he was elected an IEEE Fellow for technical leadership in speech recognition, packet speech, and integrated voice-data network. He received S.B., S.M., and Ph.D. degrees in electrical engineering from MIT.

YOUNG-SUK LEE is a staff member in the Information Systems Technology group, and has been working on machine translation since joining Lincoln Laboratory in 1995. As a principal investigator of the Korean-English translation project, she helps develop and integrate several submodules of the CCLINC system, including English and Korean understanding and generation, part-of-speech tagging, robust parsing, grammar and lexicon acquisition and updating, and graphical user interface. Her main research interest is in the development of interlingual representation with semantic frames for multilingual machine translation and other multilingual applications. Before coming to Lincoln Laboratory, she taught linguistics at Yale University. She received a B.A. degree in English linguistics and literature from Seoul National University, Korea, where she graduated summa cum laude in 1985. She also has an M.S.E. degree in computer and information science and a Ph.D. degree in linguistics from the University of Pennsylvania. She is a member of the Association for Computational Linguistics and the Linguistic Society of America.

STEPHANIE SENEFF is a principal research scientist in the Spoken Language Systems group at the MIT Laboratory for Computer Science. During the 1970s, she was a member of the research staff at Lincoln Laboratory, where her research encompassed a wide range of speech processing topics, including speech synthesis, voice encoding, feature extraction (formants and fundamental frequency), speech transmission over networks, and speech recognition. Her doctoral thesis concerned a model for human auditory processing of speech, and some of her later work has focused on the application of auditory modeling to computer speech recognition. Over the past several years, she has become interested in natural language, and has participated in many aspects of the development of spoken language systems, including parsing, grammar development, discourse and dialogue modeling, probabilistic natural-language design, and integration between speech and natural language. She is a member of the Association for Computational Linguistics and the IEEE Society for Acoustics, Speech and Signal Processing, serving on their Speech Technical committee. She received a B.S. degree in biophysics, and M.S., E.E., and Ph.D. degrees in electrical engineering, all from MIT.

DINESH R. TUMMALA works to expand and adapt machine-translation systems to larger and new domains as a staff member in the Information Systems Technology group. He also develops semi-automated lexicon and grammar acquisition techniques. He joined Lincoln Laboratory in 1993, after researching pattern recognition systems and natural-language interfaces in information retrieval during internships at Digital Equipment Corporation. He received an S.B degree in computer science and engineering and an S.M. degree in electrical engineering and computer science from MIT. He was awarded a National Science Foundation Graduate Fellowship.

BETH CARLSON
is a former staff member of the Information Systems Technology group. She researched and developed algorithms for information retrieval, machine translation, and foreign language instruction before leaving Lincoln Laboratory in February 1997. Prior to this position, she worked for GTE Laboratories in Waltham, Mass., developing speech-recognition algorithms for telephone and cellular applications. She received B.E.E. and Ph.D. degrees in electrical engineering from Georgia Institute of Technology.

JOHN T. LYNCH
worked with the Information Systems Technology group for twelve years before retiring in 1996 to study psychology. His research involved test and evaluation of speech technology systems and machine-translation systems. He also worked on applications for automated speech and text information retrieval and classification. During his last three years, he served as an appointed volunteer ombudsperson. He joined the Optical Communications group at Lincoln Laboratory in 1970 and worked for five years on various aspects of the Lincoln Experimental Satellites (LES) 8 and 9. Then he spent three years at the MIT Center for Advanced Engineering Study as director of Tutored Video Instruction, a continuing education program for industrial engineers that videotapes MIT classes. This effort was followed by two years of developing superconducting signal processing devices with the Analog Device Technology group at Lincoln Laboratory. He then joined the faculty of Boston University as associate professor of electrical engineering for three years before returning to Lincoln Laboratory. He received S.B. and S.M. degrees in electrical engineering from MIT and a Ph.D. degree in electrical engineering from Stanford University.

JUNG-TAIK HWANG
works for JLM Technologies, Inc., in Boston, Mass., as a system architect and consultant, designing solutions to client problems. Prior to joining JLM Technologies, he was a research assistant in the Information Systems Technology group, working on techniques to improve the performance of machine translation of long sentences. He received B.S. and S.M. degrees in computer science from MIT.

LINDA C. KUKOLICH
develops and maintains software systems for the Information Systems Technology group. Previously she developed software for the Optical Communications Systems Technology group. She received a B.S. degree in applied mathematics from MIT.