Carnegie Mellon University
Research Showcase @ CMU

Computer Science Department | School of Computer Science

2004

Language Technologies for Humanitarian Aid

Jaime G. Carbonell
Carnegie Mellon University

Alon Lavie
Carnegie Mellon University

Lori Levin
Carnegie Mellon University

Alan Black
Carnegie Mellon University

Follow this and additional works at: http://repository.cmu.edu/compsci

Published In
Technology for Humanitarian Action, Cahill, K. (Ed.), Fordham University Press, 111-138.

This Book Chapter is brought to you for free and open access by the School of Computer Science at Research Showcase @ CMU. It has been accepted for inclusion in Computer Science Department by an authorized administrator of Research Showcase @ CMU. For more information, please contact research-showcase@andrew.cmu.edu.

Language Technologies for Humanitarian Aid

Jaime Carbonell, Alon Lavie, Lori Levin, Alan Black
Language Technologies Institute
Carnegie Mellon University
{jgc,alavie,lsl,awb}@cs.cmu.edu

Introduction: The Need for Speech-to-Speech Translation

Humanitarian aid missions, whether emergency famine relief, establishment of medical clinics, or missions in conjunction with peace-keeping operations, require on-demand communication with the indigenous population. If such operations take place in countries with a commonly-spoken major language, such as English or Spanish, it proves relatively easy to find participating personnel with the appropriate linguistic fluency. However, such is not the case when the operations take place in regions where less common languages are spoken, such as Bosnia (language: Serbo-Croatian), Haiti (language: Haitian Creole), Somalia (language: Somali, a.k.a. "Soomaaliga"), or Afghanistan (language: Pashto, with subpopulations of Urdu and Tadjik speakers). Even in Latin America, where Spanish and Portuguese dominate, there are over 100 indigenous languages, including Quechua in Peru, Aymara in Bolivia, Mapudungun in Southern Chile, and the Tucan languages in the Southern Colombian Putumayo region. Many native speakers of these languages are not versant in either Spanish or Portuguese, especially those in remote mountainous or jungle regions, where the need for medical or educational aid, or protection from organized drug gangs, may be paramount.

The "obvious" solution is to educate relief personnel in the native language of current interest, in order to reach at least a rudimentary level of communicative fluency. However, such education requires time and expense, and must be repeated with all new personnel before they are rotated in. Moreover, sudden needs such as emergency famine relief, coping with a sudden-onset epidemic, or accompanying a peace-keeping mission do not allow for a typical six-month or one-year language education "crash-program". Another potential solution is to ferry a suite of human translators along with the aid workers. That too is fraught with inadequacies, including high expense, exposure of additional personnel to the local dangers (disease, insurgency, etc.), and the sheer difficulty of finding translators for minority languages, let alone enticing them to participate. Therefore, in actual practice, a makeshift combination of approaches is followed, including on-the-job rudimentary language learning, occasionally finding willing translators, and simply going without – a very risky proposition.

This chapter explores a technological solution to the minority-language communication challenge – or at least an important technological ingredient to a combined solution augmenting scarce human translators, if available. That solution entails combining multilingual speech recognition and speech synthesis with new machine translation technologies. Recent advances in all three areas hold significant promise with respect to producing acceptable levels of accuracy, especially for targeted domains (e.g. medical

interviews). Moreover, at Carnegie Mellon University, in conjunctions with our partner spin-off companies, we have integrated and miniaturized the three technologies to produce increasingly functional early prototypes of hand-held speech-to-speech translation devices. These devices operate in three phases:

1. Recognize the spoken language in one language – let us call it the source language – optionally confirming the corresponding text with the speaker to correct potential errors in the speech recognition.
2. Translate the source language text into the second language – the target language – optionally back-translating to the source, in order to detect and correct potential translation errors.
3. Synthesize the translated text into speech in the target language.

If the source-language speaker is illiterate, or if the speaker has gained sufficient confidence on the system's ability to recognize speech and translate, the confirmatory steps are omitted, and the translation proceeds faster, albeit with a potential for undetected errors.

Applying off-the-shelf technology for speech recognition and machine translation proves insufficient for the task. Speech systems are deeply customized to a given language, such as English, and are not easily adaptable for minority languages. Developing standard machine translation technology for a new language-pair (e.g. English to Arabic) requires person-decades of specialized computational linguists, and thus puts the effort beyond the economic reach for humanitarian aid applications. Instead, new technologies aimed at rapid low-cost adaptation to new languages are required; and those are precisely the ones under current investigation, as discussed in this chapter.

The remainder of the chapter is organized into three sections:

- **Speech Recognition and Synthesis** (corresponding to steps 1 and 3 above), including advances permitting rapid adaptation to new language.
- **Machine Translation Technologies** (corresponding to step 2 above), including the AVENUE project for rapid creation of translation systems for minority and endangered languages.
- **Speech-to-speech Translation** (integrating all three steps), including discussion of a progression of projects: JANUS, DIPLOMAT, Speechalator, with increasing capabilities, and discussion of future prospects and applications in humanitarian aid, bi-lingual education, and preservation of endangered languages.

Speech Recognition and Synthesis

Speech is the most natural form of communication for people; however it is far from the easiest form of communication for machines. Over the past 30-40 years the processing of human speech by computer has advanced to the stage that it can now be used effectively in many practical situations. ***Automatic speech recognition*** is the process of converting audio recordings of human speech into text; and ***text to speech synthesis*** is the inverse process of converting text into spoken, fluent audio. Both processes present their own major technical challenges, which we review in this section.

For automatic speech recognition, we must statistically model the acoustic variations that speakers use in speaking their language, as well as filter background noise. Phonemes, the fundamental units of speech, may be spoken in different ways depending on the other phonemes around them. The process is called "co-articulation." For example, the pronunciation of the consonant /s/ is acoustically distinct, although similar, depending on the shape of the following vowel. In a word like "so" the lips are rounded for the /s/, while for "see" the lips are not. Although human ears have learned to deal with such variation without even noticing its existence, automatic computer speech recognition needs to model these variations explicitly in order to recognize every appropriate form of every phoneme in context. Thus, the first step in building sufficient models is to collect examples of such speech in as many contexts as there are variations.

There are other levels of variation too that must be covered. Female and male speech are different, children's speech also differs due to the size and maturity of the vocal tract. People also speak in different styles. Casual or slurred speech and precise speech are quite different; a speech recognition system must be able to handle all forms. Environmental conditions also cause variation. People speak differently when outdoors, versus in a quiet office or on the phone. Linguistic factors also affect speech; a person's dialect, education, and social position can affect pronunciation. Human listeners are good at adapting to the difference in human speech even when heavily accented. Speech recognition engines must often also deal with non-native speakers, both with subtle and strong accents.

Speech output, on the other hand, should be clear and consistent, and may sometimes be based on a single speaker. Issues such as gender of voice and style of voice can, however, be important, and may require a small handful of "voices." For instance, a command voice is need when issuing an order such as "Put down your weapon now!". A more compassionate voice is appropriate when saying: "We are here to help," or asking "Where does it hurt when you walk?".

The Phoneme Level

The first basic task in build speech models for new languages is the definition of a ***phoneme set*** for the target language. Phonemes are the fundamental pieces of speech that make up a language, such as the pronunciation of individual letters. The linguistic definition of a phoneme is a unit that when changed can lead to a new word. For example, the /p/ in the English word "pat" is a phoneme, as if it were to change to /b/ we would get the new word "bat". The International Phonetic Association (IPA,1993) has gone far to define a set of phonemes that cover most of the variations of the languages of the world. But there are still subtle questions that often need to be addressed; even in major dialects of English, that are very well studied, there are questions about how many phonetic distinctions should be made. For example, in British English, the words "Mary", "marry" and "merry" all have different vowels, whereas in American English, for many people, there is no reliable distinction in their pronunciation.

The Lexical Level

Once a phoneme set is defined, the next stage is to construct a lexicon to map from words to sequences of phonemes. For some languages, such as Spanish, where the written form is close to the pronunciation, this phonology-orthography mapping can be done by simple rules. But for other languages with more complex relationships between orthography and pronunciation, such as English and French, a lexicon is required with explicit entries. Even the largest list of words in a language will never be 100% complete. Proper nouns, words borrowed from other languages (e.g. "sushi," "au contraire," "ombudsman," "macho," "insallah"), and neologisms like "gigabyte" or "defillibrator", will always pose new challenges to the most complete of lexicons. Thus, for speech recognition and synthesis we also need to be able to generate the most plausible pronunciation for an unknown word, just like humans do. This we can do by building statistically trained letter-to-sound rules trained from the lexicon. We have used a simple but reliable technique for doing this for a number of languages (Black et al., 1998).

In order to construct the basic lexicon itself we have developed a bootstrap technique (Maskey et al., 2004). In this technique we first hand specify the pronunciations of some 300 common words and then construct a set of letter-to-sound rules automatically from these entries. Then, using text in the target language, we find the most frequent words and test them against this letter-to-sound rule model. If they are correct, we add them to the base lexicon, and, if wrong, we hand correct them, add them to the lexicon and retrain the rules with the additional verified data. By iterating this technique we can quickly construct reliable lexicons even for languages with more opaque orthographic to phonetic relationships.

Acoustic Recognition

In order to build speech recognition acoustic models we must have examples of speech in as wide a variation as possible within the intended use and subject-matter of the recognizer (e.g. medical interviews). Traditionally, speech recognition acoustic models require about 100 hours of recorded and transcribed speech for training the acoustic recognition models. It is crucial that the transcription reflex exactly what was actually said (including repetitions, false starts etc. that are common in even quite careful speech), rather than an idealized or cleaned-up versions, or else the acoustic training will fail to find the sound-to-text correspondences reliably.

The GlobalPhone Project (Schultz, 2002) has reduced the amount of data required to train a speech recognizer in a new language, by using initial models from other languages and then adapting those models for the target language using a much smaller amount of data (Schultz and Wiabel, 2001). The GlobalPhone Project offers not just the ability to build new recognizers in new languages but includes a data collection component that defines and provides tools for non-speech-scientists to collect target language data. GlobalPhone has already collected data from 14 languages and continues to collect data for new languages. This data repository also aids in moving to new languages by growing the common set of cross-language phones and building clusters of related languages.

Although the best results can be achieved by increasing amounts of data from the target language, comparable results can be achieved with relatively small amounts of target language data complementing data available from other languages. Using initial multi-lingual models plus a little as around one hour of transcribed target speech results can achieve results similar to that of collecting tens of hours of speech in that language. This is very important for rapid development of speech-based systems for new languages in emergency-aid situations, as the process of exact acoustic transcription is very slow and detailed – it takes 10 to 20 hours to transcribe exactly one hour of speech.

Text to Speech Synthesis

For speech synthesis, unlike speech recognition, we can often limit our scope to a single voice, or perhaps, just one male and one female voice. The FestVox Project (Black and Lenzo, 2000a) offers tools, techniques and documentation on how to build synthetic voices in new languages reliably, without requiring a computational speech scientist. The technological approach is termed ***concatenative speech synthesis.*** Appropriate small sub-word units of natural speech are selected and concatenated together to form words and new utterances. The quality of the process can approach that of recorded human speech, though unlike recorded speech, it can be used to say unanticipated words, phrases and sentences, for instance those produced by a machine translation system. The design of the recorded database is, however, crucial; it must cover the phonetic and prosodic space.

The data collection process proceeds as follows: We first collect a large amount of text in the target language. Then, we select short sentences (say less than 20 words) that contain primarily high frequency words. Such sentences are typically easy to say; pronunciation errors are thus minimized. We then use the constructed lexicon to convert the text into phoneme strings, as discussed above, and focus our pronunciation training on the sentences with the best phonetic coverage. We repeatedly select sets of phonetically rich sentences until we have identified around 1000 such sentences. In addition to general text we may also include targeted text for the particular application, such as medical interviews. The closer this designed database is to the target utterances the better the quality of the synthesis. In extreme cases we can design the system to cover a targeted domain (Black and Lenzo, 2000b), augmented with standard greetings and transition phrases. The database is recorded by a single native speaker of the target language in a studio quality environment. The data is then automatically labeled using a speaker specific acoustic model to find where all the phoneme boundaries are. For best results these labels are hand corrected, but that is a resource intensive task.

Evaluation of the speech output is extremely important. Just because it may sound Chinese, Greek or Quechua to the builder of the voice does not mean it sounds natural to native speakers. Evaluation including listening tests by natives are used to ensure the quality is acceptable and understandable. A number of different tests explicitly measure phonetic coverage, and domain coverage (Tomokiyo, 2003).

Challenges for Minority Languages

As we cover more languages, our tools and techniques improve. But from this wider coverage, we also learn that there are other factors that can make the construction of speech technology in new languages harder. The top world languages have substantial amounts of written text, linguistic analysis and large volumes of text readily available on-line. As we move to the less spoken languages such resources become more and more scarce. Phonetic systems for minority languages may not be defined, lexicons not readily available, and written texts may be available only on printed or handwritten media.

Whereas the major languages of the world have standardized both their orthographic and phonetic conventions (spelling and pronunciation), the same is not true for many minority languages. For instance, Mapungun, spoken in Southern Chile and Argentina by the indigenous Mapuche population, has several distinct orthographic variants and at least an equal number of phonological ones. Even for majority languages, standardization may be partial or recent. For example, although a well-defined version of Arabic exists, Modern Standard Arabic, this is not normally a spoken language. The people in daily conversation use their own dialects, differing in both pronunciation and lexicon. Building a speech recognizer and synthesizer in Arabic requires first a decision about which dialect(s) to choose. Then, once chosen, we must ensure we build lexicons and orthography-to-phonetic mappings for that dialect rather than simply Modern Standard, even though web-available material is far more abundant for the latter.

There are also socio-linguistic issues in building speech and language models. For instance, many cultures are more gender sensitive than English-speaking ones. The grammar and marking within the language may change, depending on the gender of the speaker, not just that of the addressee. Such language models need to be added to the system so that a female synthesized voice uses appropriate female language, while the male output uses appropriate male terms. Such gender issues can be especially important in dealing with sensitive subjects such as medical interviews that may refer to anatomical concepts, hygiene, diet, family or reproduction.

Another interesting issue is whether the speech-translation system should produce synthetic speech that sounds like a native speaker or with an accent typical of the source language speaker (e.g. American). In the development of a Pashtu synthetic voice we used a US English speaker, trained in phonetics, to mimic the natural Pashtu speech, as no native Pashtu speaker was available. Thus the resulting synthesizer had a slight American accent, and to Pashtun natives sounded non-native, which, surprisingly, they thought was just perfect, as they could see that the original speaker was American, and so his translated voice was appropriate and not deceptive.

Machine Translation Technologies

Machine Translation (MT) (Hutchins and Somers, 1992) has become a popular technology on the Internet. Many web sites offer free automatic translation, and some search engines, such as Google, offer to "translate this page" automatically if the language of the page is different from the language of the user's web browser interface. However, closer inspection reveals that MT is available for very few language pairs. A language pair consists of a source language, the language one is translating from; and a target language, the language one is translating into. Free Web-based translation services are generally available for pairs of major European languages (usually English, Spanish, French, German, Italian, Portuguese, and maybe Russian) and for a few pairs of European and Asian languages (usually Japanese and Chinese). The Compendium of Translation software (http://www.eamt.org/compendium.html) lists all MT software available for sale. In this list we can find more language pairs. However, with a few exceptions, most of the source and target languages are spoken in countries that can provide a large consumer base for MT systems.

Many of the commercial systems available today are the results of person-decades of work, and therefore are developed for language pairs where economic prospects are favorable. Unfortunately, such economic imperatives exclude most minority languages where MT is most needed for humanitarian purposes. For this reason, there is a growing amount of research on producing MT systems for new language pairs quickly and cheaply. After a brief discussion of the state of the art in MT, we present our CMU AVENUE system for building cost-effective MT of minor languages.

Why is MT hard?

A naive concept of translation might involve looking each word up in a dictionary to find an equivalent word in the target language. Consider the translation of a simple sentence like "Me llamo Maria" into English. A word-for-word translation would result in "myself call Maria," which is not a good English sentence. A slightly more sophisticated word-for-word translation might be "Myself I call Maria" taking into account that "llamo" means "I call". Notice that this involves knowledge of morphology, the combinations of prefixes, suffixes, and root words, and that one word in Spanish can correspond to more than one word in English. However, we are still far from the correct translation “My name is Maria”. A major problem is that even closely related languages do not use the same word order. Another problem is lexical ambiguity (e.g. “llamar” means both “to call” and to “be named”, the latter applying to the reflexive construction).

Semantics, the meanings of words and sentences, must also be taken into account. Perhaps the oldest MT joke involves the translation of “The spririt is willing but the flesh is weak” into Russian and back into English as “The wine is good but the meat is rotten.” How can we know weather the word “spirit” refers to a soul or to an alcoholic beverage? A less poetic example involves translating “lock” into Spanish. As a verb, its meaning must be decomposed into “cerrar con llave” (literally: close with key), because there is no corresponding lexical item. These problems are more severe for distant language pairs.

In short, MT is hard because it involves knowledge about morphology, syntax and semantics, for both the target and source languages. Native speakers of a language have this knowledge in implicit form, but coding the knowledge explicitly in computer programs is a deceptively complex task because of the myriad subtleties of language.

What Can Be Done?

The intractability of the translation problem is usually addressed at a practical level, for instance, by specializing for the kind of task for which an MT system is being developed. If the translations are going to be used for dissemination of information, such as technical instructions or public health alerts, then the output must be of very high quality so as to be understood accurately. There are several options in this case: 1) The MT system could interact with a human translator who can catch and correct system errors, thus automation is partial, reducing overall human error, or 2) The semantic domain can be limited so that words do not have so many meanings (for example in texts about food, "spirits" would only refer to beverages), or the input can be restricted to an unambiguous subset of the source language. Different MT systems may be used for assimilation of information, for example, scanning news wire for information about outbreaks of infectious diseases. In this case, the users do not have control over the input to the system, but errors can be tolerated as long as the general idea gets across.

MT systems can also be used for dialogue, and these are the ones of central interest in this chapter. MT systems for spoken language face the additional problem of starting with the imperfect output of a speech recognizer. This tends to reduce the quality of translation. However, since two humans are participating in the conversation, they may be able to detect misunderstandings, correct them, and adjust their language or speaking style to reduce future errors. Speech translation systems therefore do not require full accuracy, but because of the difficulty of the problem, they are usually constrained to a limited semantic domain, such as search and rescue situations, or primary medical care interview, or negotiating travel arrangements.

As a practical problem, MT system development must also take into account the language resources that are available in the source and target languages. Resources can be manifold: text in electronic form, bilingual dictionaries, grammar rule sets, and/or linguists who can write rules for morphology, syntax and semantics. Translation rules written by linguists require substantial time and effort to debug, especially as the number of rules increases and it becomes difficult for rule writers to keep track of the behavior of each rule and the interactions among different rules. It may also be the case that linguists trained in computational transfer-rule writing are not available for some languages.

The dominant MT paradigm is rule-based, but translations can also be calculated automatically from parallel corpora – large volumes of humanly-translated text such as the collected proceedings of the United Nations (Brown et al., 1990, Vogel et al., 2000, Hutchinson et al., 2003). Various algorithms can be used to calculate a translation model – a set of probabilities for translating source language words and phrases into their target language equivalents, augmented with target language models that prefer certain word

sequences over others. For instance, “pichi wentru” in Mapudungun could translate into “young man” or “diminutive man”, but the former is preferred as being a word sequence more typically used in English. This approach is called Statistical MT. Alternatively, Example-Based MT systems compare incoming source text to sentences that are in the parallel corpus. If parts of the new source sentence are similar to parts of previously translated sentences in the parallel corpus, the corresponding translations of the similar parts are assembled into a candidate target sentence. Example-based MT also uses a target language model to select which of potentially many alternative translations is the most probable in the target language.

Corpus-oriented algorithms have the advantage of not requiring human rule writers, and thus can be brought online very quickly, given a large bilingual parallel corpus. But they have some disadvantages, the largest being that large human-translated parallel corpora simply do not exist for minority languages. Smaller corpora produce less accurate translations, and even these seldom exist in sufficient quantity to build meaningful statistical models for minority languages.

The AVENUE Project

AVENUE is a research project aimed at finding quick, low cost methods for developing MT systems for minority or endangered languages, especially focusing on languages that do not have enough resources for corpus-oriented approaches to MT. We also aim to circumvent the large cost in time and money of manually writing a comprehensive set of translation rules. Therefore, AVENUE learns from corpora, but from extremely small linguistically-balanced ones, and instead of learning probabilities, it learns translation rules that can be examined and extended by human linguists if there is a human linguist with adequate training in MT and fluency in source and target languages. Information about morphology and syntax are implicit in the probabilities that are learned by data-oriented methods, but explicit in the rules learned by AVENUE.

AVENUE operates in four stages: learning, translation, decoding, and refinement as shown in Figure-1 below. The details of these four stages are described in Probst et al. (2002) and Lavie et al. (2003). Here we present an overview of the first stage, learning, which has two sub-phases. Development of an MT system for a new language pair starts with the process of elicitation, which produces the data needed for automatic rule learning. Elicitation requires a user who is bilingual in the source and target languages, but does not need to know linguistics and does not need to know how to write rules for an MT system. The elicitation interface is shown in the Figure-2 below. A sentence is presented to the bilingual informant, who then translates it and aligns the corresponding words between the original sentence and its translation. The informant simply clicks on a word in each language and the elicitation interface draws a line connecting them. The elicitation interface also produces an internal representation of the alignments in the form of indices such as (1 1) (first word aligns to first word), (3,5 2) (third and fifth words align to second word, etc.). Words may align with phrases (e.g.: “lock” with “cerrar con llave”) or with multiple disjoint words (e.g. “not” with “ne” and “pas” even the latter two are not always adjacent in French).

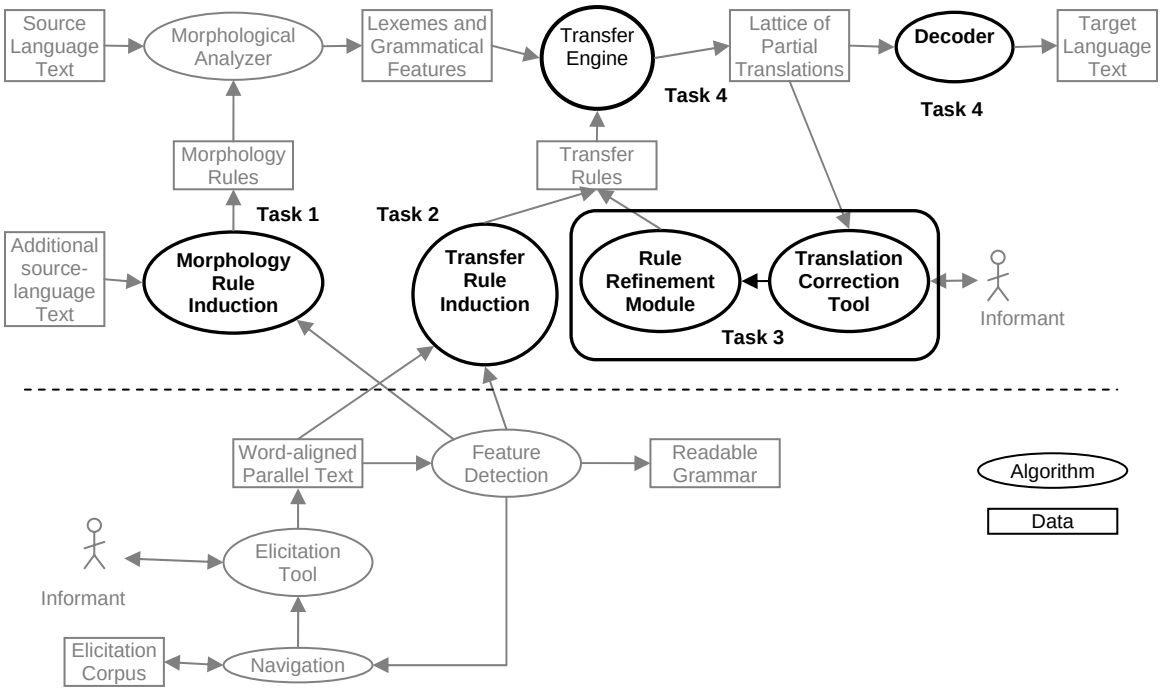

Figure-1: The Architecture of the AVENUE Framework

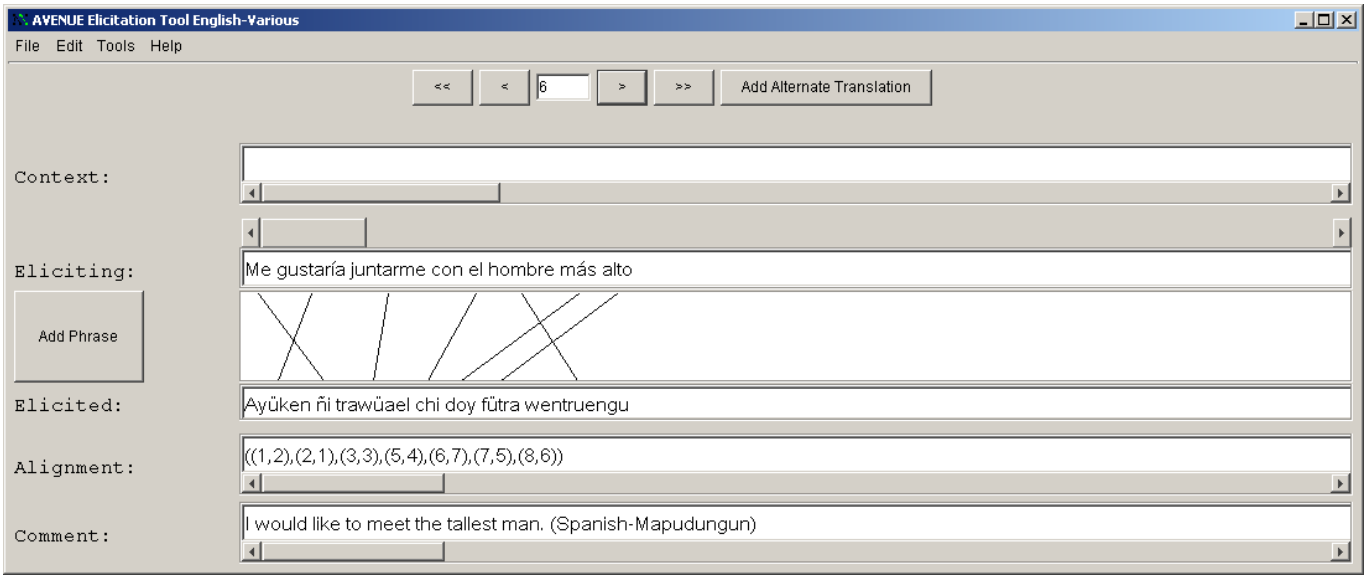

Figure-2: The AVENUE Elicitation Interface

The elicitation interface has been used by Mapudungun speakers from Chile translating from Spanish, Aymara speakers from Bolivia translating from Spanish, Hindi speakers translating from English, and Hebrew speakers translating from English.

The output of elicitation is a small but very useful parallel corpus of a few thousand sentences whose source and target words are carefully aligned. This corpus is the input to the rule learning component. Each portion of the corpus consists of *minimal pairs*, pairs of sentences that differ in only one fundamental linguistic way, such as singular-plural (to elicit pluralization rules), or a noun phrase with and without adjectives to determine whether phrase structures are head-initial or head-final (i.e. whether the adjectives come before or after the main noun in the minority language). The rules learned can be fairly complex, as illustrated below.

Figure-3 below shows an example of a translation rule for Chinese and English. In Chinese, in order to form a question that requests a yes or no answer, a question word 吗 is inserted at the end of the sentence. In English, on the other hand, a yes-no question begins with an auxiliary verb as in *Do the children eat pizza?* In the rule, x0 refers to the Chinese sentence, x1 refers to a noun phrase which is the first element of the Chinese sentence, x2 refers to a verb phrase which is the second element of the Chinese sentence, and x3 refers to the question word. Similarly, y0 refers to the English sentence, y1 refers to an auxiliary verb such as *do* which is the first element of the English sentence, y2 refers to a noun phrase such as *the children* which is the second element of the English sentence, and y3 refers to a verb phrase such as *eat pizza* which is the third element of the English sentence. The rule shows that x1 should be translated into y2 and that x2 should be translated into y3. It also contains various constraints on the source and target language syntax. For example, y3 must contain an infinitive verb such as *eat* rather than a past tense or participial verb such as *ate*, *eaten*, or *eating*.

```
; Rule to transfer Chinese question sentences
      {S,3} ; Unique rule identifier
      ; production rules: SL and TL type and constituent or POS
      sequences
      S::S : [NP VP "吗"] -> [AUX NP VP]
      (
        ; Constituent alignments
        (x1::y2) ; NP to NP
        (x2::y3) ; VP to VP
        ; Parsing (x-side) constraints, build feature structure
        ((x0 subj) = x1) ; Assign NP's features to subj
        ((x0 subj case) = nom)
        ((x0 act) = quest)
        (x0 = x2)
        ; Transfer (xy) constraints
        ((y2 case) = (x0 subj case))
        ; Generation (y-side) constraints
        ; Insert AUX on target side based on
        ; value constraints
        ((y1 form) = do)
        ; Enforce value and agreement restrictions on y-side
        ((y3 vform) =c inf) ; verb must be infinitive
        ((y1 agr) = (y2 agr))
```

Figure-3: An Example Translation Rule

Rules like the one above can be written by a human linguist or can be learned automatically from the output of the elicitation process. In order to learn rules automatically, the AVENUE system must capture two properties of human language syntax, compositionality and generality. Compositionality refers to the composition of larger phrases from smaller ones. For example, a sentence is made from a combination of noun phrases, verb phrases, prepositional phrases, and adverbs. A noun phase can be made from adjective phrases, articles, nouns, prepositional phrases, and possibly also embedded sentences. The sentence in "*Yesterday very big trucks brought the sacks of grain that were needed*" could be described as two adverbs (*yesterday* and *very*), an adjective, a noun, a verb, an article, a noun, a preposition (*of*), a noun, a relative clause marker, and auxiliary verb, and a passive verb. However, it would be better to describe it as an adverb, followed by a noun phrase, a verb and another noun phrase, as shown in Figure-4 below. The reason is that the latter description can also be used to describe other sentences that are similar, but not identical in structure such as *Usually, excess rain ruins crops* or *Unfortunately the truck hit a pothole*. The AVENUE rule learner must therefore be able to recognize which parts of a sentence can be grouped together into noun phrases and prepositional phrases. Then it must be able to hypothesize rules that compose those phrases into sentences. In order to accomplish this in a new language, the phrases of the source language (English or Spanish) are used as a guideline. The words that are aligned to words of the English or Spanish noun phrase are assumed to form a noun phrase in the new language as well. This is not always accurate, but it is usually a good starting point.

Yesterday	adverb
noun phrase	
very big	adjective phrase
trucks	noun
brought	verb
noun phrase	
the	article
sacks	noun
of grain	prepositional phrase
that were needed	embedded sentence (relative clause)

Figure-4: The Compositional Phrase Structure of an English Sentence

This compositional analysis permits us to induce translation rules, such as the example in Figure-3, via a machine learning method called ***seeded version space learning***, which is beyond the scope of this chapter.

Domain-limited Interlingua-based Speech Translation

Evolution of Domain-limited Interlingua-based MT at CMU

The Language Technologies Institute together with the Interactive Systems Laboratory at Carnegie Mellon have been pursuing an ongoing research effort over the past fifteen years to develop machine translation systems specifically suited for spoken dialogue. The JANUS-I system (Woszczyna et al., 1993) was developed at Carnegie Mellon University and the University of Karlsruhe in conjunction with Siemens in Germany and ATR in Japan. JANUS-I translated well-formed read speech in the conference registration domain with a vocabulary of 500 words. Advances in speech recognition and robust parsing over the past ten years then enabled corresponding advances in spoken language translation. The JANUS-II translation system, taking advantage of advances in robust parsing (Lavie, 1996), operated on the spontaneous scheduling task (SST) -- spontaneous conversational speech involving two people scheduling a meeting with a vocabulary of 3,000 words or more. JANUS-II was developed within the framework of an international consortium of six research groups in Europe, Asia and the U.S., known as C-STAR (http://www.cstar.org). A multi-national public demonstration of the system capabilities was conducted in July, 1999. More recently, the JANUS-III system made significant progress in large vocabulary continuous speech recognition (Woszczyna, 1998) and significantly expanded the domain of coverage of the translation system to spontaneous travel planning dialogues (Levin et al, 2000), involving vocabularies of over 5,000 words. The NESPOLE! System (Lavie et al, 2001) further extended these capabilities to speech communication over the internet, and developed new trainable methods for language analysis that are easier to port to new domains of interest. These were demonstrated via a prototype speech-translation system developed for the medical assistance domain called the Speechalator (Waibel et al., 2003).

Overview of the Interlingua-based Approach

Throughout it's evolution over the course of more than fifteen years, the speech translation systems have established a framework based on a common, language-independent representation of meaning, known within the MT community as an ***Interlingua***. Interlingua-based machine translation is convenient when more than two languages are involved because it does not require each language to be connected by a set of translation rules to each other language in both direction. Adding a new language that has all-ways translation with existing languages requires only writing one *analyzer* that maps utterances into the interlingua and one *generator* that maps interlingua representations into sentences. In the context of a large multi-lingual project such as C-STAR or NESPOLE!, this has the attractive consequence that each research group can implement analyzers and generators for its home language only. There is no need for bilingual teams to write translation rules connecting two languages directly. A further advantage of the interlingua approach is that it supports a paraphrase option. A User's utterance is analyzed into the interlingua, and can then be generated back into the user's language from the interlingua. This allows the user to confirm that the system produced

correct interlingua for their input utterance (ie, whether it has correctly understood the sentence prior to translating it, much as a human translator may do). The figure below illustrates interlingua-based MT.

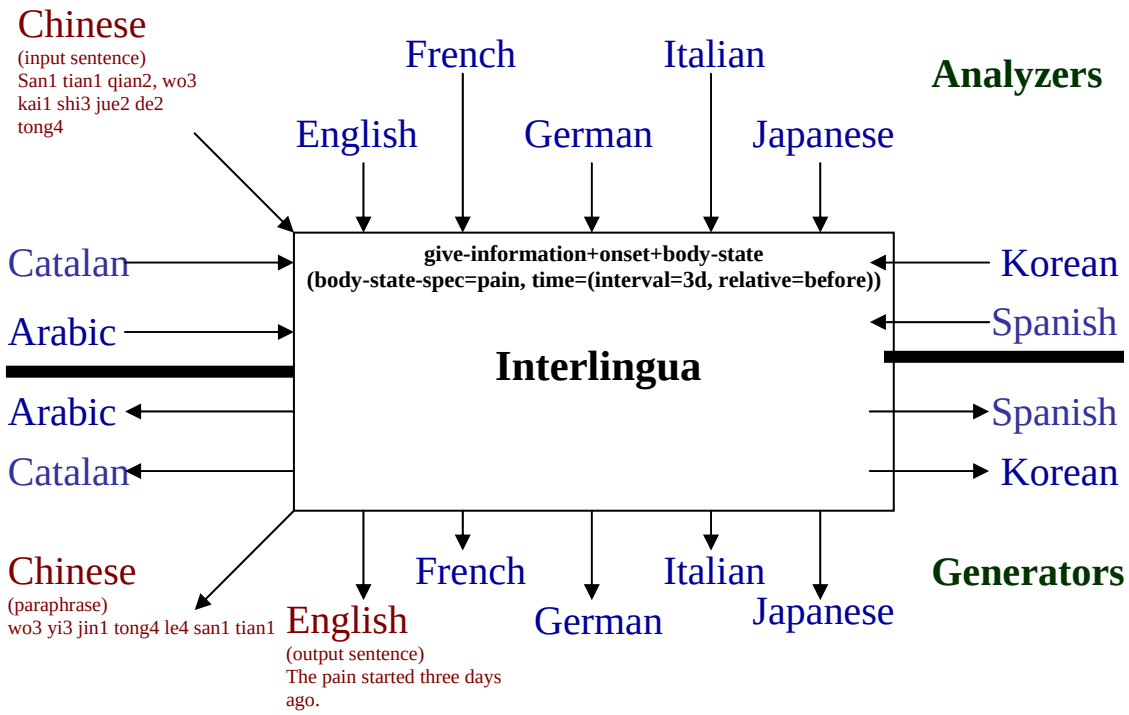

Figure-4: Interlingua-based Machine Translation between Multiple Languages

The main principle guiding the design of the interlingua is that it must abstract away from peculiarities of the source languages in order to account for MT divergences and other non-literal translations (Dorr, 1994). In the travel domain, non-literal translations may be required because of many fixed expressions that are used for activities such as requesting information, making payments, etc. Similarly, in medical assistance, formulaic expressions are often used when eliciting medical information from a patient, or suggesting treatments. The interlingua must also be designed to be language-neutral, and simple enough so that it can be used reliably by many MT developers. In the case of the interlingua systems described here, simplicity was possible largely because the working within task-oriented limited domains. In a task-oriented domain, most utterances perform a limited number of *Domain Actions* (DAs) such as requesting information about the availability of a hotel or giving information about the price of a flight. These domain actions form the basis of the interlingua, which is known as the *Interchange Format*, or IF.

The IF defines a shallow semantic representation for task-oriented utterances that abstracts away from language-specific syntax and idiosyncrasies while capturing the

meaning of the input. Each utterance is divided into semantic segments called *semantic dialog units* (SDUs), and a *Domain Action* (DA) is assigned to each SDU. A DA consists of three representational levels: the *speech act*, the *concepts*, and the *arguments*. In addition, each DA is preceded by a *speaker tag*, to indicate the role of the speaker. The speaker tag is sometimes the only difference between the IFs of two different sentences. For example, "Do you take credit cards?" (uttered by the customer) and "Will you be paying with a credit card?" (uttered by a travel agent) are both requests for information about credit cards as a form of payment. In general each DA has a speaker tag and at least one speech act optionally followed by a string of concepts and/or a string of arguments. In Example-1 below, the speech act is *give-information*, the concepts are *availability* and *room*, and the arguments are *time* and *room-type*. Example-2 shows a DA which consists of a speech act with no concepts attached to it. Finally, Example-3 demonstrates a case of DA which contains neither concepts nor arguments.

Example-1: On the twelfth we have a single and a double available.
a:give-information+availability+room (room-type=(single & double),time=(md12))

Example-2: And we'll see you on February twelfth.
a:closing (time=(february, md12))

Example-3: Thank you very much
c:thank

Figure-5: Examples of Travel Domain Spoken Utterances and their Interlingua Representations.

These DAs do not capture all of the information present in their corresponding utterances. For instance they do not represent definiteness, grammatical relations, plurality, modality, or the presence of embedded clauses. These features are generally part of the formulaic, conventional ways of expressing the DAs in English. Their syntactic form is not relevant for translation; it only indirectly contributes to the identification of the DA.

Language Analysis and Generation

In interlingua-based translation systems, translation is performed by analyzing the source language input text into the interlingua representation, and then generating a string in the target language. Among these, analysis of the source language is the more challenging and difficult task. The richness of language provides for a wide range of ways for humans to express the same basic concept. The same idea can be expressed in many different ways. For example, a doctor querying a patient for the location of a pain or injury could express this using a variety of sentences such as: "*show me where it hurts*" (imperative, command), "*where does it hurt?*" (direct question), "*can you show me where the pain is located?*" (indirect question), "*does it hurt here?*" (yes/no question), etc. Achieving very high levels of coverage of such variations is extremely challenging, even in limited domains. Furthermore, the inherent ambiguity of language is a major obstacle

to accurate analysis of meaning. As the coverage of an analysis system increases to cover more and more variety of vocabulary and structure, ambiguity becomes more pervasive, and the identification of the *correct* meaning becomes significantly more difficult. These problems become yet even harder when dealing with analysis of spontaneous spoken language input. The major additional issues that must be addressed are the disfluent nature of conversational spoken language, the unique grammatical characteristics of spoken language, and the lack of explicit punctuation or even clearly marked sentence boundaries. The imperfect capabilities of speech recognition systems further exacerbate these problems, since some words in the input may have not been recognized correctly. Analyzers for spoken language must therefore be "robust" in the sense that they must be capable of extracting the main meaning expressed in spoken utterances, even when this meaning is embedded in a noisy and imperfect input utterance.

Target-language generation is more straightforward than analysis. Whereas analysis must handle the variation in language in expressing the same concept, generation can suffice with only a single appropriate text generation for any given meaning representation. Moreover, since we have control over the generated text, it can be designed to be fluent and grammatical. Appropriate punctuation and even prosodic markers can be inserted within the generated text, to help produce better pronounced synthesized speech which is more understandable and natural sounding. General text generation frameworks such as GenKit (Tomita and Nyberg, 1988), which were originally designed for text-to-text machine translation have for the most part been equally suitable for target-language generation within speech translation systems, and have been used extensively in the various speech-to-speech translation systems developed at CMU.

Deployment of Portable Speech-Translation Systems

Traditionally, speech-to-speech translation systems have required substantial computing power. Speech recognition benefits from fast processors with substantial computer memory. Translation too, both knowledge-based and statistical, requires significant computing power. Concatenative speech synthesis also generally improves with large databases. Hence, best results are obtained when each process is run on a separate fast processor on a local area network.

However the best form factor for use of such systems in front line medical and refugee situations is a small and portable device. A cell-phone connecting to a central service can be a possibility but cell-phone coverage and quality of transmission present considerable additional challenges. An alternative is to miniaturize the processing, compromising some accuracy for speed and memory size, and through clever engineering reduce the memory footprint of the software, as well as take approximate faster methods versus more exact slower ones.

In our work we have developed scalable systems that can run on large servers and also on consumer PDAs (personal digital assistants like the HP Ipaq). In porting speech translation systems to hand held computers (Waibel et al., 2003), we must modify a

number of key points in the design. Such hand held computers are much less powerful than standard desktop or laptop computers. Although at first Moore's Law, that computers will double in power every 18 months, may be thought as a long-term savior, we find the actual limiting factor is usable battery power. Batteries improve much more slowly than computer or memory chips. Hence, our hand held speech-to-speech translation engine has been specifically design for low-power-consumption chips whose instruction sets exclude floating-point computations. These require significant adjustment to our algorithms, including clever approximation techniques to replace more exact computations.

Additionally the small form factor introduces issues with usability in the field. The system must be light enough to carry, and fast enough to be useful. Although better quality audio can be obtained with a head mounted microphone, this can be impractical in a medical interview or refugee processing scenario. Instead, we use a built-in microphone, and must cope with its poorer quality and ambient noise pickup.

Since the primary user of the system will be a health worker, or other humanitarian aid specialist not necessarily versant with computing or translation, we must design the interface and functionality accordingly. We do give very brief training on system activation, rebooting, and usage, such as speaking in short clear sentences, which we found enhances the system's performance (Frederking et al., 2002).

Illustrative Scenario: Famine Relief in Somalia

Consider a hypothetical scenario where a developing humanitarian crisis calls for a US-led international relief effort: For illustration purposes, let us say that in 2009, after a period of prolonged strife, a consensus government emerges in Somalia, capable of maintaining a certain level of stability. However, the combination of destroyed infrastructure and a drought season are threatening widespread famine again; but this time due to the relative stability, a sizable relief effort starts to be planned for deployment in three to four weeks. However, lack of trained English-Somali translators poses a major potential impasse. A quick search identifies a handful of individuals, three of whom are willing to help, but two are rather elderly (Somali expatriates, now retired professionals in the US), and therefore cannot be safely deployed.

However, all three fluent bilinguals can participate in developing a bi-directional English-Somali speech-to-speech machine-translation system, based on the new PTRANS (Portable TRANSlator) technology just completed in the laboratory after a period of stable funding. The three ex-Somalis are then asked to contribute towards teaching the essence of Somali to PTRANS, both written and spoken. The willing and able individual will later also join the relief deployment as the central translator to help broker agreements with local leaders – though she cannot be in multiple places at once, and therefore more routine translations will be assigned to PTRANS. The first decision is that the translation system will need to focus on the domains of primary medical care (doctor-patient interviews, inoculations, etc.), and in the logistics of food distribution

(roads, directions, warehousing, instructions on delivery, etc.). It would not be feasible to create a general purpose speech-to-speech translation system in four weeks.

In order to train a Somali speech recognizer, several hours of transcribed recorded speech are needed from multiple individuals. All three contribute their speech and transcription, and later a few more Somali speakers are located and asked for a few hours to complete the task. Both male and female speakers are required to train the speech recognizer adequately. Speech synthesis only requires the recordings of one clear speaker (two if both male and female voices are desired), and one of the two retried individuals is selected since he speaks a well-accepted Somali dialect clearly.

Training the Machine Translation part of PTRANS requires a bit more involvement from the Somali-English bilinguals. One is tasked to lexical issues, checking the common words in the electronic bilingual dictionary, and establishing the correspondences between Somali and English inflections. The other two are asked to translate the linguistically-balanced elicitation corpus (see section 3 of this chapter), using the elicitation tool for word and phrase alignments. After two weeks of elicitation, the transfer rule learning method (section 3) extracts and generalizes candidate transfer rules, which are then used to produce test translations of new phrases and sentences. The Somali-English bilinguals check these translations for accuracy, noting which translations are incorrect, and classifying the errors. The learning system uses these corrections to repair and augment the transfer rules, producing a working translation system in the domains of primary care and food-distribution logistics.

The last week is used to integrate, test and further refine the three phases: speech recognition, translation and speech synthesis. Several dozen hand-held PTRANS devices are loaded with the new English-Somali system and distributed to the members of the deployment team. With the departure of the team to Somalia, the PTRANS system work continues for several more weeks, improving the coverage and accuracy of both the speech and translation components. Data relayed back from field units (uploaded at the end of each day) on sentences and phrases it was asked to translate, especially ones where it may have failed to do so correctly. This data permits further refinement of PTRANS to actual field conditions during the deployment.

References

Black, A., K. Lenzo, and V. Pagel. 1998. "Issues in Building General Letter to Sound Rules". In Proceedings of 3rd ESCA Workshop on Speech Synthesis, pp. 77-80, Jenolan Caves, Australia.

Black, A., and K. Lenzo. 2000a. "Building Synthetic Voices: The FestVox Project". http://www.festvox.org

Black, A., and K. Lenzo. 2000b. "Limited Domain Synthesis". In Proceedings of ICSLP-2000. Beijing, China.

Brown, P., J. Cocke, S. Della Pietra, V. Della Pietra, F. Jelinek, J. Lafferty, R. Mercer, and P. Roossin. 1990. “A Statistical Approach to Machine Translation,” *Computational Linguistics*, 16(2). (http://www.aclweb.org/anthology/J90-2002)

Dorr, B. 1994. “Machine Translation Divergences: A Formal Description and Proposed Solution”. *Computational Linguistics 20(4)*, pp. 597--633.

Frederking, R., A. Black, R. Brown, J. Moody, and E. Steinbrecher. 2002. “Field Testing the Tongues Speech-to-Speech Machine Translation System”, in Proceedings of LREC-2002, Las Palmas, Canary Islands.

Hutchinson, R., P. N. Bennett, J. G. Carbonell, P. Jansen, R. Brown. 2003. “Maximal Lattice Overlap in Example-Based Machine Translation", Technical Report CMU-CS-03-138/CMU-LTI-03-174, June 2003.

IPA. 1993. IPA: The International Phonetic Association (revised to 1993) - IPA Chart, Journal of the International Phonetic Association 23.

Lavie, A., 1996. “GLR* : A Robust Grammar-Focused Parser for Spontaneously Spoken Language”. PhD dissertation, Technical Report CMU-CS-96-126, Carnegie Mellon University, Pittsburgh, PA, May 1996.

Lavie, A., C. Langley, A. Waibel, F. Pianesi, G. Lazzari, P. Coletti, L. Taddei and F. Balducci. 2001. “Architecture and Design Considerations in NESPOLE!: a Speech Translation System for E-commerce Applications”. In Proceedings of HLT-2001 Human Language Technology Conference, San Diego, CA, March 2001.

Lavie, A., S. Vogel, L. Levin, E. Peterson, K. Probst, A. Font Llitjós, R. Reynolds, J. G. Carbonell. 2003. "Experiments with a Hindi-to-English Transfer-based MT System under a Miserly Data Scenario." *TALIP* (ACM Transactions on Asian Language Information Processing), Volume 2, Issue 2, June 2003, pages 143-163.

Levin, L., A. Lavie, M. Woszczyna, and A. Waibel, 2000. “The JANUS-III Translation System”. *Machine Translation, 15(1-2).*

Maskey, S., A. Black and L. Tomokiyo. 2004. “Bootstrapping Phonetic Lexicons for New Languages”. In Proceedings of ICSLP-2004, Jeju, Korea, 2004.

Probst, K., L. Levin, E. Peterson, A. Lavie, J. G. Carbonell. 2002. “MT for Resource-Poor Languages Using Elicitation-Based Learning of Syntactic Transfer Rules,” *Machine Translation*, Vol. 17, No. 4, pages 245-270.

Schultz, T., and A. Waibel. 2001. “Language Independent and Language Adaptive Acoustic Modeling for Speech Recognition”, *Speech Communication 35 (1-2)*, pp. 31-51, August 2001.

Schultz, T., 2002. “GlobalPhone: A Multilingual Speech and Text Database Developed at Karlsruhe University”. In Proceedings of the International Conference on Spoken Language Processing (ICSLP-2002), Denver, Colorado, September 2002.

Tomita, M. and E. H. Nyberg. 1988, “Generation Kit and Transformation Kit, Version 3.2: User's Manual”. Technical Report CMU-CMT-88-MEMO, Carnegie Mellon University, Pittsburgh, PA.

Tomokiyo, L., A. Black, and Lenzo, K. 2003. “Arabic in my Hand: Small-footprint Synthesis of Egyptian Arabic”. In Proceedings of Eurospeech 2003. Geneva, Switzerland.

Vogel, S., F. J. Och, C. Tillmann, S. Nie_en, H. Sawaf, H. Ney. 2000. “Statistical Methods for Machine Translation". In: "Verbmobil: Foundations of Speech-to-Speech Translation", pp. 377-393, Wolfgang Wahlster (ed.). Springer Verlag, Berlin, July 2000. (http://www-i6.informatik.rwth-aachen.de/Colleagues/och/VMBUCH.ps)

Waibel, A., A. Badran, A. Black, R. Frederking, D. Gates, A. Lavie, L. Levin, K. Lenzo, L. Mayfield Tomokiyo, J. Reichert, T. Schultz, D. Wallace, M. Woszczyna, and J. Zhang. “Speechalator: Two-way Speech-to-Speech Translation on a Consumer PDA”. In Proceedings of Eurospeech 2003, Geneva, Switzerland.

Woszczyna, M., N. Coccaro, A. Eisele, A. Lavie, A. McNair, T. Polzin, I. Rogina, C. P. Rosé, T. Sloboda, M. Tomita, J. Tsutsumi, N. Aoki-Waibel, A. Waibel, and W. Ward. 1993. “Recent Advances in JANUS: a Speech Translation System”.

Hutchins, W.J. and H. L. Somers. 1992. *An Introduction to Machine Translation* , Academic Press, London, 1992.