# TONGUES: RAPID DEVELOPMENT OF A SPEECH-TO-SPEECH TRANSLATION SYSTEM

*Alan W Black*[1], *Ralf D. Brown*[1], *Robert Frederking*[1], *Rita Singh*[2], *John Moody*[3], *Eric Steinbrecher*[3]

[1]Language Technologies Institute, Carnegie Mellon University,
[2]School of Computer Science, Carnegie Mellon University,
[3]Lockheed Martin Systems Integration, Owego, NY

## ABSTRACT

We carried out a one-year project to build a portable speech-to-speech translation system in a new language that could run on a small portable computer. Croatian was chosen as the target language. The resulting system was tested with real users on a trip to Croatia in the spring of 2001. We describe its basic components, the methods we used to build them, initial evaluation results, and related significant observations. This work was done in conjunction with the US Army Chaplain School; chaplains are often the only personnel in a position to communicate with local people over non-military issues such as medical supplies, refugees, etc. This paper thus reports on a realistic instance of rapidly deploying and field-testing a speech-to-speech translator using current technology.

## INTRODUCTION

With speech recognition, synthesis and translation beginning to work well enough for small tasks, this paper describes a short project to build a portable speech-to-speech translation system in a new language. We describe its basic components, the methods we used to build them and related significant observations. The end system was tested with real users on a trip to Croatia in the spring of 2001.

This work was done in conjunction with the US Army Chaplain School. Army chaplains are often among the advance party of troop deployment. In many cases, the chaplains are the only personnel in a position to communicate with local people over non-military issues such as medical supplies, refugees, etc. Often the chaplain has no knowledge of the local language, and due to immediacy requirements, no human translator is available. Thus the chaplain must communicate as best possible, perhaps without even a bilingual dictionary.

Given this domain, our task was to build a speech-to-speech translation system that could run on a small portable computer that will aid conversations between a chaplain and a native. Such a task requires

- speech recognition systems for English and the target language
- speech synthesizer for English and the target language
- a translation system from English to the target language, and the reverse translation system
- an interface that allows these components to be used effectively in communication

The entire project, from start to finish, was allowed to take only one calendar year, including initial contractual arrangements, hiring language experts, etc. All of these systems had to run on a single small sub-notebook computer, in a reasonable time; this added further interesting constraints on the project.

For topical reasons, Croatian was chosen as the target language. Although spoken by around 5 million people, it does not command enough economic weight that the commercial speech and language community has produced recognizers, synthesizers and translation engines for it. Thus it is a realistic language type to use as an example. From a testing standpoint, although Croatia is still of interest to the US military, there are no current hostilities there, thus enabling a realistic field-test under safe conditions.

## PREPARATION

As we were to build these system in a short period and on a small budget, data driven approaches were the only reasonable method. Such techniques must be used for each of the three core components: machine translation, speech recognition engines, and speech synthesis engines.

Thus at the very start we arranged to record a number of chaplains in role-playing conversations of the type we expected the device to encounter. Fortunately, the chaplains were familiar with role-playing exercises, and all had relevant field experiences to reenact. Both sides of the conversations were in English. These were digitally recorded with head mounted microphones at 16KHz in stereo (one speaker on each channel), as this was closest to the intended audio channel characteristics of the eventual system. In all, we recorded 46 conversations, ranging from a few minutes to 20 minutes length. In total there was 4.25 hours of actual speech.

These conversations were then hand-transcribed at the word level, identifying false starts, filled pauses and the complete words.

Next the transcriptions of these English-English conversations were translated into Croatian by native Croatian speakers by hand.

This data provided the basic information from which we could boot strap the rest of the speech-to-speech translation system.

## COMPONENTS

### Recognition

For speech recognition, we used the CMU Sphinx II system [8], a relatively light-weight recognizer that works in real time even on machines with relatively small memory and modest-speed processors. For Automatic Speech Recognition (ASR) to work we need to build two basic types of models. **Acoustic Models** which model the acoustic phonetic space for the given language and **Language Models** which model the probability of word sequences. In addition to these models we also need two lexicons one for English and one for Croatian that map words to their pronunciations.

For the English acoustic models, we could have used existing acoustic models trained from similar wide-band speech, but as there were not any readily available conversational wide-band speech databases in the intended domain, it was felt better to train on the chaplain dialogs directly rather than use existing models and some form of adaptation. Although such adaptation techniques may have been beneficial and feasible for English, we knew that for the Croatian no such data was available, and part of this exercise was to develop speech-to-speech translation systems for languages that did not already have speech resources constructed for them. Thus for English we took **only** the 4.25 hours of chaplain speech and directly trained semi-continous HMM models for Sphinx2.

For the English language model we required a larger collection of in-domain text. We used the dialog transcriptions themselves but also augmented that with text from chaplain handbooks that were made available to us. Although we knew we could provide better recognition accuracy by using more resources, we were interested in limiting what resources were necessary for this work, and also (see below) we found the trained models from this data adequate for the task.

Building Croatian models was harder. As we were aware that our resource of Croatian speakers was limited, and they had less skill in carrying out full word transcription of conversational speech, we wished to find a simpler, less resource-intensive method to build Croatian acoustic models. From the the translated chaplain transcripts, we wished to select example utterances that when recorded would give sufficient acoustic coverage to allow reasonable acoustic models to be trained. To do this, we used a technique originally developed for selecting text to record for speech synthesis [2]. By using the initially developed Croatian speech synthesizer, we could find the phonemes that would be used to say each utterance. We then ran a greedy selection algorithm that selects utterances that would best cover the acoustic space [2]. From a list of several thousand utterances, we selected groups of 250 utterances that were phonetically rich. These sets were then read by a number of Croatian speakers. Using read speech avoided the process of hand-transcription of the speech, though it does make it less like the intended conversational speech. Due to the relative scarcity of native Croatian speakers, we recorded only 15 different speakers, of which 13 were female and 2 were male. This resulted in a gender imbalance, which was not however observed to affect the system's performance greatly. In all, a total of 4.0 hours of Croatian speech was collected. This data **alone** was then used to train new acoustic models for Croatian.

For both English and Croatian recognition systems, semi-continuous 5-state triphone HMMs were trained. The number of tied states used in each case was commensurate with the amount of training data available. Although the English models did have explicit modeling of filled pauses (non-linguistic verbalized sounds such as "um", "uh" etc.), none were trained for Croatian. This was partially because the recorded speech was read, and had minimal spontaneous speech phenomena such as filled pauses.

Language models in both cases were word-trigrams built with absolute discounting. The language-model vocabularies consisted of 2900 words for English and 3900 words for Croatian. In pilot experiments with heldout test sets, the word error rates were found to be below 15% for English and below 20% for Croatian.

We note that as the utterances used in the training were not spontaneous, the system was more easily confused by hesitations and filled pauses. However in the actual user tests of the system this proved to be less of a problem than we expected. As turns in a conversation through a speech-to-speech translation system are slower and less spontaneous compared to single language conversations, speakers were more careful in their delivery than they might be in full conversations.

### Synthesis

For English synthesis, we used an existing English voice in the Festival Speech Synthesis System [3]. Although there may have been a slight advantage in building a targeted synthesizer for conversations, it would not have been significantly different in quality. A few lexicon additions were made for the particular domain, but the existing English voice was essentially used unchanged.

For Croatian, it was necessary to build a complete new speech synthesis voice. To do this, we used the tools available in the CMU FestVox project [1], which is designed to provide the necessary support for building new synthetic voices in new languages. Synthetic voices require: text processing, lexicons, a method for waveform synthesis, and prosodic models.

In this case, the text processing was minimal, as the type of language being given to the synthesizer was fairly regular, since it would be generated by the translation system (or the Croatian recognizer).

Luckily, orthographic-to-phoneme rules for Croatian are relatively easy and could be written by hand, so building a lexicon was much easier than it might be for some other languages. (The same lexicon and letter-to-sound rules were used by the recognition engine).

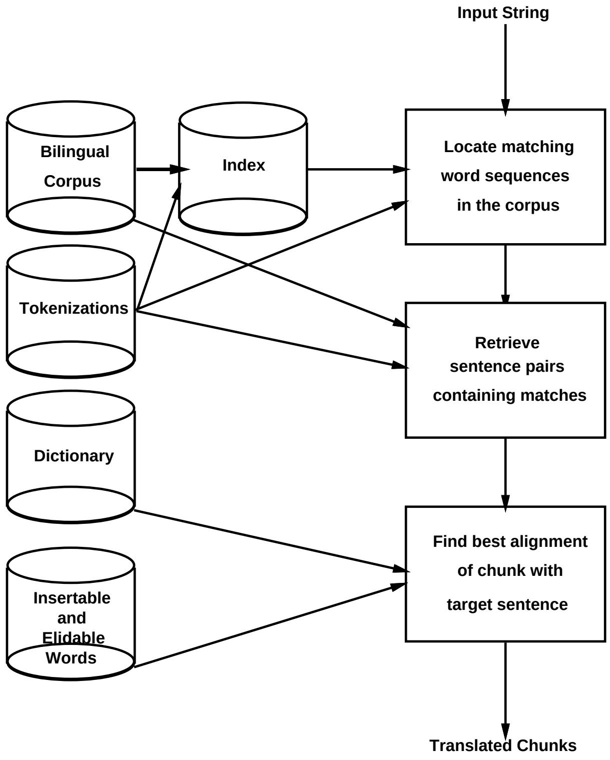
The waveform synthesis was done using a constrained version of general unit selection techniques. From the translated utterances from chaplain dialogs and other Croatian text, we selected 1000 utterances that best covered the phonetic space (using the technique more fully described in [2]). These were spoken by a native male Croatian and automatically labelled by a simple dynamic time warp technique using cross-linguistic prompts (as decribed in [1]). These were then hand corrected.

The final required piece was a set of prosodic models for Croatian; we found a very simple rule-based method of phrasing adequate for this domain (mostly shorter sentences). We trained duration models from the recorded Croatian speech, which worked well. However, the intonation model was harder. We found that a model trained from the relatively small amount of speech in the Croatian database did not produce a good intonation model. Thus we fell back on a different technique: we simply used our English intonation model modified to the range of our Croatian speaker. On listening tests, native Croatians preferred this over the natively-trained model. For other languages such short cuts may not be so acceptable.

The resulting quality–although not always fluent–was understandable almost all the time, and much better than a standard diphone synthesizer. It also readily captured the voice quality of the original Croatian speaker.

### Machine Translation

Again due to the requirement of rapid development, data-driven approaches were preferred. Thus we used a Multi-Engine MT (MEMT) system [7], whose primary engines were an Example-Based MT (EBMT) engine [4] and a bilingual dictionary/glossary. Carnegie Mellon's EBMT system uses a "shallower" approach than many other EBMT systems; examples to be used are selected based on string matching and inflectional and other heuristics, with no deep structural analysis. The MEMT architecture uses a trigram language model of the output language to select among competing partial translations produced by several engines. It is used in this system primarily to select among competing (and possibly overlapping) EBMT translation hypotheses.


Input String
Bilingual Corpus
Index
Locate matching word sequences in the corpus
Tokenizations
Retrieve sentence pairs containing matches
Dictionary
Find best alignment of chunk with target sentence
Insertable and Elidable Words
Translated Chunks

EBMT Architecture

For translation into Croatian, we incorporated a finite-state word reordering mechanism, applied during the language model-driven selection of partial translations, to place clitics in a cluster in the appropriate location. (Croatian syntax requires a very specific ordering of clitics in a cluster in a specific position in the sentence.)

The training corpus for the EBMT engine consisted of the translated chaplain dialogs plus pre-existing parallel text from the DIPLOMAT project [6] and newly-acquired parallel text from the web. The dictionary/glossary engine used both statistically-extracted translations and manually-created entries. The English trigram model already existed, and had been generated from newswire and broadcast news transcripts. Finally, the Croatian trigram model was built from the Croatian half of the EBMT corpus, some Croatian text found on the web, and the full text of some sixty novels and other Croatian literary works (in total, approximately six million words).

### Integration and Interface

Simply stringing together a recognizer, translator, and synthesizer does not make a very useful speech-to-speech translation system. A good interface is necessary to make the parts work together in such a way that a user can actually derive benefit from it. Using our experience from the earlier DIPLOMAT system, we designed the interface to be asymmetric, with the Croatian side being as simple as possible, and any necessary complexity handled on the English side, since the chaplain would be trained and practiced in using the system.

We included back-translation, to allow a user with no knowledge of the target language to better assess the quality of the translation. We also included several user-requested features, such as built-in pre-recorded instructions and explanations for the Croatian (since the Croatian speaker is completely naive regarding the device and the chaplain's intentions), emergency key phrases (such as "Don't move!"), and enhancements such as being able to modify the translation lexicon, so that the system could be tuned to more specific tasks.

The final system ran on a Windows-based Toshiba Libretto, running at 200MHz with 192MB of memory. At the time of the project (2000) this was the best combination of speed and size that was readily available. The system was equipped with a custom touchscreen, so that the Croatian-speaker would not need to type or use a mouse at all. Aware that the system may be used in situations where the non-English participant may be unfamiliar with the technology, we include a microphone/speaker handset that looks like a conventional telephone handset. This has the advantage of provided a close-talking microphone, thus making speech recognition easier, and coming in a format that will be familiar to most people.

## EVALUATION

In April 2001, a group organized by the US Army Chaplain school took two versions of the device to Zagreb, where it was tested with non-English-speaking Croatians. A number of scenarios were prepared in English and Croatian, and were given to each participant to act out using the translation device. The scenarios were in the intended domain, involving refugees, medical supplies and getting general directions.

In all, 21 dialogs took place, between different Croatian speakers and one of 5 chaplains. After the test, the Croatian participants were given a questionnaire to fill out. Their overall impression was as follows:

| Overall | |
|---|---:|
| Good | 5 |
| OK | 11 |
| Bad | 3 |

Our own observations of the basic system were that it did actually work to a level that was useful about one half of the time (it was not clear in advance that this necessarily would be the case). The participants were capable of communicating through the system and real information was transferred.

However, as expected, there were a number of specific problems. One that we noted immediately was a frustrating slowness of communication, due to required user clarifications, though it was much faster than if a bi-lingual dictionary were the only translation device available.

On asking the participants to identify the most difficult problems, they replied as follows:

| **User difficulties** | |
|---|---:|
| grammar/case | 5 |
| loudspeakers | 4 |
| translation | 3 |
| recognition | 2 |
| synthesis | 2 |
| speed | 1 |

Hardware issues with the volume of the built-in speakers were a clear (and easily solvable) problem. But other aspects of the core technology were both harder to identify and harder to fix. The system includes a facility to allow the user (typically the chaplain) to explicitly add new words and phrases to the system so that common errors can be minimized over time. Although this facility was not used often, it is clear that supporting a greater level of adaptation would allow the device to become more useful over time.

Unlike the English recognizer, the Croatian recognizer did not support filled pauses and hesitations. The effect was that extra short words (typically function words) were often erroneously hypothesized by the system. As the system displayed what was being recognized, it was easy for the speaker to delete those extra words by hand, which they often did. However the speakers also learned to speak more fluently and less conversationally as they used the system, improving recognition accuracy.

Similarly, we asked what they found easy:

| **What works?** | |
|---|---:|
| short sentences | 10 |
| nothing | 4 |

It was quickly discovered by most participants that the system did not translate long, rambling sentences well. Short, direct sentences were much more likely to produce good translations. This was not surprising, given the limitations of the platform and the deliberate limiting of development time to see if such limitations still allowed a useful translation device. We were actually pleased to see that the system provided adequate coverage for successful translation of unrehearsed, naive dialogues.

Other specific observations we noted were that the users could not easily identify where the problems lay with the system. (For example, if speech recognition produced and displayed a correct transcript, and then translation produced an unacceptable result, they would usually *respeak* the same utterance using the same words.) Thus even if we provided separate user methods to add words to the recognizer, language model and translation engine, it is clear that the user would not be able to identify which part (or parts) need to be updated. As we feel that such systems need to provide methods of adaptation in the field, it is clear that the interface presented to the user to offer that adaptation needs more work.

Although there were problems with the volume of the output through the small built-in speakers on the device, which many users commented on, mistakes in the synthesizer were often erroneously attributed to the translator (and vice versa).

A second observation was that the participants continued to use speech and did not resort to the alternative typing interface (although they were clearly aware of it), and only resorted to typing as a last resort. This may have been due to the fact the participants were told to use the speech-to-speech translation device rather than have the more abstract goal of successful communication by the best means. The very small keyboard on the (required) small device may also have been a factor.

Use of system in Croatia

Further details of the evaluation are described in [5].

## CONCLUSION

As one of the goals of this work was to rapidly develop a speech-to-speech translation system we also wished to account for the effort spent in building this system. Although the work took part over one calendar year, not everyone was working full time on the system during that period. In total there were 6 technical staff involved (the authors of this paper), each bringing their particular expertise. We estimate that in total there was about 2 person-years total effort from the senior staff. In addition to this there was also part-timer Croatian informants, chaplains and some student helpers. We also should note that some of the translation datra used to train the system was collect for a previous project.

Most of the basic systems that were used in development of this system already existed and this was basically a test of how well they perform on new data. However some problems with the tools were found and some new development was carried out. Interestingly it is the organization of data collection, scheduling translators and labellers that is actually one of the most time consuming parts.

If this technique were to be applied to some new language we believe less resources would be required, though we do not want to claim that each new language would be the same as the previous, and hence different possibly non-trivial problems may appear when moving these techniques to new languages.

This project shows how a relatively simple speech-to-speech translation system can be rapidly and successfully constructed using today's tools. The system was indeed constructed in less than one year. The results of the 2001 evaluation in Croatia indicated that, while the system was not ready for actual field use, it was actually impressively close to that level of performance, and worthy of further development to achieve that capability.

## REFERENCES

[1] A. Black and K. Lenzo. Building voices in the Festival speech synthesis system. http://festvox.org, 2000.

[2] A. Black and K. Lenzo. Optimal data selection for unit selection synthesis. In *4rd ESCA Workshop on Speech Synthesis*, Scotland., 2001.

[3] A. Black, P. Taylor, and R. Caley. The Festival speech synthesis system. http://festvox.org/festival, 1998.

[4] R. Brown. Example-based machine translation in the Pangloss system. In *Proceedings of COLING-96*, pages 169–174, Copenhagen, Denmark, 1996.

[5] R. Frederking, A. Black, R. Brown, J. Moody, and E. Steinbrecher. Field testing the tongues speech-to-speech machine translation system. LREC 2002, 2002.

[6] R. Frederking, A. Rudnicky, C. Hogan, and K. Lenzo. Interactive speech translation in the diplomat project. *Machine Translation Journal*, special issue on spoken language translation, 2000.

[7] R. E. Frederking and R. D. Brown. The Pangloss-Lite Machine Translation System. In *Proceedings of the Second Conference of the Association for Machine Translation in the Americas (AMTA)*, pages 268–272, Montreal, Quebec, Canada, October 1996.

[8] X. Huang, F. Alleva, H.-W. Hon, K. Hwang, M.-Y. Lee, and R. Rosenfeld. The SPHINX-II speech recognition system: an overview. *Computer Speech and Language*, 7(2):137–148, 1992.