# Tamil IT! : Interactive Speech Translation in Tamil

**K.M. Ganesh** < devkmg@rediffmail.com> **&**
**S. Subramanian** <mailsurya@lycos.com>

*Department of Computer Science and Engineering*
*P.S.G. College of Technology, Peelamedu ,*
*Coimbatore. Tamil Nadu, 641004, India.*

Under the Guidance of
Dr. Vasu Renganathan, Department of South Asia Regional Studies,
University of Pennsylvania, USA.

---

**Abstract**

The Tamil IT! (Interactive Translation) speech translation system is intended to allow unsophisticated users to communicate across the Tamil ↔ English language barrier, without strong domain restrictions, despite the error prone nature of current speech and translation technologies. Achieving this ambitious goal depends in large part on allowing the users to interactively correct recognition and translation errors. We briefly present the Multi Engine Machine Translation (MEMT) architecture, describing how it is well suited for such an application. We then describe our incorporation of interactive error correction throughout the system design. We are currently in the process of developing a Tamil ↔ English system based on this architecture.

## A Brief Overview of Tamil Language Analysis

Like any other language analysis process, Tamil language analysis also involves morphological analysis, syntax analysis and semantic analysis. Tamil is a Morphologically rich language. Most of the grammatical functions are embedded into the word in the form of inflections.

### *Morphological Analysis*

Here is an example of Tamil morphological analysis.

| | |
|---|---|
| ஏறினேன் | ஏறு ன் ஏன் |
| eeRineen | eeRu in een |
| (climbed) | (Verb) (Past tense) (I Person+Singular+Neuter) |

In the above example the word eeRineen (climbed) has three morphemes viz.(1) eeRu [Verb for climb], (2) in [Past tense marker] and (3) een [GNP marker]

### *Syntax Analysis*

Syntactically, Tamil is a head final language. Information like the tense, gender, number and person can be found embedded within the inflected verb (predicate). The parse tree for the sentence :-

நான் மரத்தின் மேல் ஏறினேன்
(I climbed the tree), with the following rules

S -> NP VP
NP -> PRO | NOUN
VP -> NP VERB
is given below.

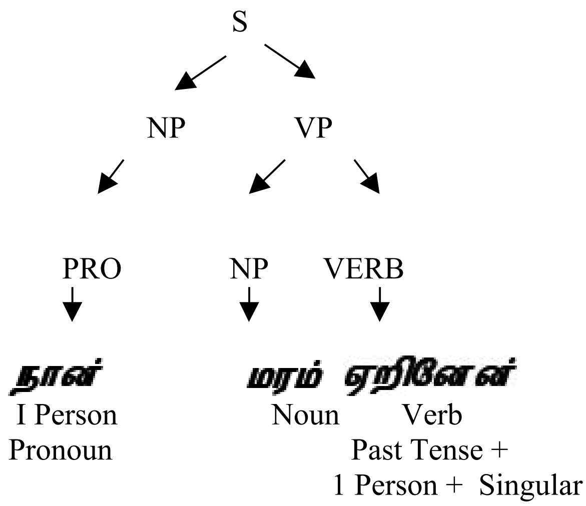


**Figure 1** : Sample Parse Tree

***Semantic Analysis***

Semantics is the study of meanings of the language. The issues in Semantic analysis include, Word sense disambiguation, anaphora resolution, representation of meaning in some logical form so that this can be used across different languages etc. The difficulty in semantic analysis is the element of World knowledge that the machine should be able to apply to the problems.

***Sense disambiguation***

As stated earlier, multiple words in a language pose a challenging task for Machine Translation. The words apply across the languages.

For example, the word 'sentence' has atleast two meanings in Tamil, one 'வாக்கியம்' in 'phrase' sense and the other 'தீர்ப்பு' in 'judgement' sense.

In Tamil, the word 'வெள்ளி' has atleast three meanings, which are listed as:
01 – Silver metal
02 – Planet Venus
03 – Friday

## Tamil IT! : A Overview

The Tamil IT! project is designed to explore the feasibility of creating a wearable bi-directional speech translation systems. The speech understanding component used is the Sphinx II HMM based speaker independent continuous speech recognition system (Huang et al., 1992; Ravishankar, 1996), with techniques for rapidly developing acoustic and language models for new languages (Rudnicky, 1995). The machine translation (MT) technology is the Multi Engine Machine Translation (MEMT) architecture (Frederking and Nirenburg, 1994), described further below. The speech synthesis component is a newly developed concatenative system (Subramanian, Ganesh, 2001). Tamil IT! thus involves research in MT, speech understanding and synthesis, interface design, as well as wearable computer systems.

A major concern in the design of the Tamil IT! system has been to cope with the error prone nature of both current speech understanding and MT technology, to produce an application that is usable by non translators with a small amount of training. We attempt to achieve this primarily through user interaction: wherever feasible, the user is presented with intermediate results, and allowed to correct them. In this paper, we will briefly describe the machine translation architecture used in Tamil IT! (showing how it is well suited for interactive user correction), describe our approach to speech recognition and then discuss our approach to interactive user correction of errors in the overall system.

## Multi Engine Machine Translation

Different MT technologies exhibit different strengths and weaknesses. Technologies such as Knowledge Based MT (KBMT) can provide high quality, fully automated translations in narrow, well defined domains (Mitamura et al., 1991; Far_ well and Wilks, 1991). Other technologies such as lexical transfer MT (Nirenburg et al., 1995; Frederking and Brown, 1996; MacDonald, 1963), and Example Based MT (EBMT) (Brown, 1996; Nagao, 1984; Sato and Nagao, 1990) provide lower quality general purpose translations, unless they are incorporated into human assisted MT systems (Frederking et al., 1993; Melby, 1983), but can be used in non domain restricted translation applications. Moreover, these technologies differ not just in the quality of their translations, and level of domain dependence, but also along other dimensions, such as types of errors they make, required development time, cost of development, and ability to easily make use of any available on_line corpora, such as electronic dictionaries or online bilingual parallel texts. The Multi Engine Machine Translation (MEMT) architecture (Frederking and Nirenburg, 1994) makes it possible to exploit the differences between MT technologies.

As shown in Figure 2, MEMT feeds an input text to several MT engines in parallel, with each engine employing a different MT technology. Each engine attempts to translate the entire input text, segmenting each sentence in whatever manner is most appropriate for its technology, and putting the resulting translated output segments into a shared chart data structure (Kay, 1967; Winograd, 1983) after giving each segment a score indicating the engine's internal assessment of the quality of the output segment.

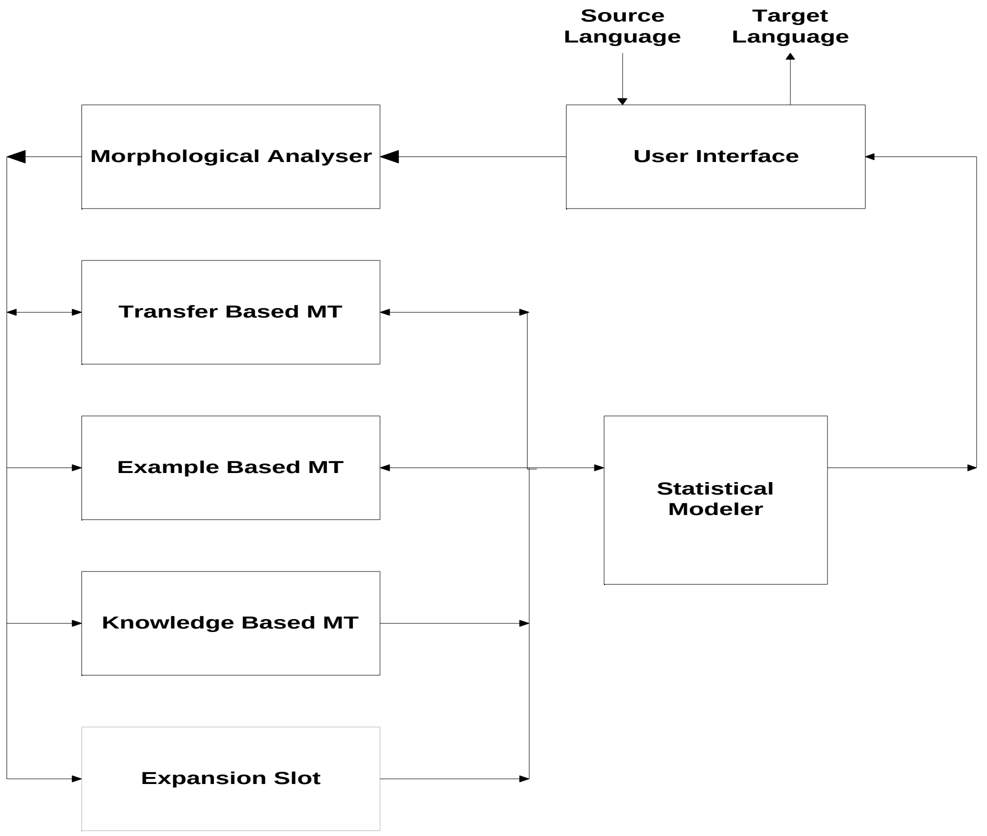


Figure 2 **: MEMT Architecture**

These output (target language) segments are indexed in the chart based on the positions of the corresponding input (source language) segments. Thus the chart contains multiple, possibly overlapping, alternative translations. Since the scores produced by the engines are estimates of variable accuracy, we use statistical language modeling techniques adapted from speech recognition research to select the best overall set of outputs (Brown and Frederking, 1995; Frederking, 1994). These selection techniques attempt to produce the best overall result, taking the probability of transitions between segments into account as well as modifying the quality scores of individual segments. The use of the MEMT architecture allows the improvement of initial MT engines and the addition of new engines to occur within an unchanging framework. The only change that the user sees is that the quality of translation improves over time. This allows interfaces to remain stable, preventing any need for retraining of users, or redesign of inter operating software. The EBMT and Lexical Transfer based MT translation engines used in Tamil IT! are described elsewhere (Frederking and Brown, 1996). For the purposes of this paper, the most important aspects of the MEMT architecture are:

- The initially deployed versions are quite error prone, although generally a correct translation is among the available choices.
- The unchosen alternative translations are still available in the chart structure after scoring by the target language model.

### Speech recognition for Tamil

Contemporary speech recognition systems derive their power from corpus based statistical modeling, both at the acoustic and language levels. Statistical modeling, of course, presupposes that sufficiently large corpora are available for training. It is in the nature of the Tamil IT! system that such corpora, particularly acoustic ones, are not immediately available for processing. As for the MT component, the emphasis is on rapidly acquiring an initial capability in Tamil, then being able to incrementally improve performance as more data and time are available. We have adopted for the speech component a combination of approaches which, although they rely on participation by native informants, also make extensive use of preexisting acoustic and text resources. Building a speech recognition system for a target domain or language requires models at three levels (assuming that a basic processing infrastructure for training and decoding is already in place): acoustic, lexical and language. We have explored two strategies for acoustic modeling. *Assimilation* makes use of existing acoustic models from a language that has a large phonetic overlap with the target language. This allows us to rapidly put a recognition capability in place. Of course, such overlaps cannot be relied upon and in any case will not produce recognition performance that approaches that possible with appropriate training. Nevertheless it does suggest that useful recognition performance for a large set of languages can be achieved given a carefully chosen set of core languages that can serve as a source of acoustic models for a cluster of phonetically similar languages. The selective collection approach presupposes a preparation interval prior to deployment and can be a follow on to a system based on assimilation. The goal is to carry out a limited acoustic data collection effort using materials that have been explicitly constructed to yield a rich phonetic sampling or the target language. We do this by first computing phonetic statistics for the language using available text materials, then designing a recording script that exhaustively samples all diphones observed in the available text sample. While the effectiveness of this approach depends on the quality (and quantity) of the text sample that can be obtained, we believe it produces appropriate data for our modeling purposes. Lexical modeling is based on creating pronunciations from orthography and involves a variety of techniques familiar from speech synthesis, including letter to sound rules, phonological rules and exception lists. The goal of our lexical modeling approach is to create an acceptable quality pronouncing dictionary that can be variously used for acoustic training, decoding and synthesis. System vocabulary is derived from the text materials assembled for acoustic modeling, as well as scenarios from the target domain. Finally, due to the goals of our project, language modeling is necessarily based on small corpora. We make use of materials derived from domain scenarios and from general sources such as newspapers (scanned and OCRed), text in the target language available on the Internet and translations of select documents. In combination, these techniques allow us to create working recognition systems in very short periods of time and provide a path for evolutionary improvement of recognition capability. They clearly are not of the quality that would be expected if conventional procedures were used, but nevertheless are sufficient for providing cross language communication capability in limited domain speech translation.

### User Interface Design

As indicated above, our approach to coping with error prone speech translation is to allow user correction wherever feasible. While we would like as much user interaction as possible, it is also important not to overwhelm the user with either information or decisions. This requires a careful balance, which we are trying to achieve through early user testing.

In order to achieve communication, the users currently can interact with Tamil IT! in the following ways:

- Speech displayed as text: After any speech recognition step, the best overall hypothesis is displayed as text on the screen. The user can highlight an incorrect portion using the touch_ screen, and re-speak or type it.
- Confirmation requests: After any speech recognition or machine translation step, the user is offered an accept/reject button to indicate whether this is ``what they said''. For MT, back_ translations provide the user with an ability to judge whether they were interpreted correctly.
- Interactive chart editing: As mentioned above, the MEMT technology produces as output a chart structure, similar to the word hypothesis lattices in speech systems.

## Conclusion

We have presented here the Tamil IT! speech translation system, with particular emphasis on the user interaction mechanisms employed to cope with error prone speech and MT processes.

## Acknowledgments

We wish to thank the Dr. Vasu Renganathan, University of Pennsylvania, USA, our project guide for giving his support and guidance. Also We wish to thank Dept. of C.S.E., P.S.G. Tech, India for their support.

## References

1 **Vasu Renganathan, 2001,** Computational Processing of Agglutinative Languages. (Unpublished)

2. **Subramanian and Ganesh, 2001,** Design and Implementation of a Tamil Text to Speech Engine. In Proceedings of TI 2001.

3. **Ralf Brown. 1996,**. Example_Based Machine Translation in the Pangloss System. In Proceedings of the 16th International Conference on Computational Linguistics (COLING_96).

4. **Ralf Brown and Robert Frederking. 1995** Applying Statistical English Language Modeling to Symbolic Machine Translation. In Proceedings of the Sixth International Conference on Theoretical and Methodological Issues in Machine Translation (TMI_95), pages 221--239.

5. **David Farwell and Yorick Wilks. 1991**. Ultra: A Multi_lingual Machine Translator. In Proceedings of Machine Translation Summit III, Washington, DC, July.

6. **Robert Frederking. 1994**. Statistical Language Models for Symbolic MT. Presented at the Language Engineering on the Information Highway Workshop, Santorini, Greece, September.

7. **Robert Frederking, D. Grannes, P. Cousseau, and S. Nirenburg. 1993 ]** . An MAT Tool and Its Effectiveness. In Proceedings of the DARPA Human Language Technology Workshop, Princeton, NJ.

8. **Robert Frederking and Ralf Brown. 1996**. The Pangloss_Lite Machine Translation System. In Proceedings of the Conference of the Association for Machine Translation in the Americas (AMTA).
9. **Robert Frederking and Sergei Nirenburg. 1994.** Three Heads are Better than One. In Proceedings of the fourth Conference on Applied Natural Language Processing (ANLP_94), Stuttgart, Germany.
10. **Xuedong Huang, Fileno Alleva, Hsiao_Wuen Hon, Mei_Yuh Hwang, Ronald Rosenfeld. 1992** The SPHINX_II Speech Recognition System: An Overview. Carnegie Mellon University Computer Science Technical Report CMU_CS_92_ 112.
11. **Martin Kay. 1967** Experiments with a powerful parser. In Proceedings of the 2nd International COLING, August.
12. **Kevin Lenzo. 1997** Personal Communication. R. R. MacDonald. 1963. General report 1952_ 1963 (Georgetown University Occasional Papers in Machine Translation, no. 30), Washington, DC.
13. **A. K. Melby. 1983** . Computer_assisted translation systems: the standard design and a multi_level design. Conference on Applied Natural Language Processing, Santa Monica, February.
14. **Teruko Mitamura, Eric Nyberg, Jaime Carbonell. 1991.** Interlingua Translation System for multilingual Document Production. In Proceedings of Machine Translation Summit III, Washington, DC, July.
15. **M. Nagao. 1984**. A framework of a mechanical translation between Japanese and English by analogy principle.
16. **Sergei Nirenburg. 1995**. The Pangloss Mark III Machine Translation System. Joint Technical Report, Issued as CMU technical report CMU_CMT_95_145.
17. **Mosur Ravishankar. 1996** . Efficient Algorithms for Speech Recognition. Ph.D. Thesis. Carnegie Mellon University.
18. **Alex Rudnicky. 1995** . Language modeling with limited domain data. In Proceedings of the ARPA Workshop on Spoken Language Technology. San Mateo: Morgan Kaufmann, 66_69.
19. **S. Sato and M. Nagao. 1990** . Towards memory based translation. In Proceedings of COLING_90, Helsinki, Finland.
20. **Terry Winograd. 1983** . Language as a Cognitive Process. Volume 1: Syntax. Addison_Wesley.