Issues in Porting TTS to Minority Languages

Ksenia Shalonova and Roger Tucker

Outside Echo (Local Language Speech Technology Initiative)
HP Labs, Filton Rd, Stoke Gifford, Bristol BS34 8QZ UK
{ksenia, roger}@outsideecho.com

Abstract

We describe issues that are arising in the Local Language Speech Technology Initiative (LLSTI) where we are porting TTS to languages where commercial organisations are reluctant to take the risk. Currently Hindi, Ibibio, Swahili, Tamil and Zulu are being developed. We propose that the TTS development process can be considered as an optimal start for linguistic documentation of minority languages. Possible solutions for obtaining formalised linguistic knowledge on different levels are discussed.

1. Introduction

1.1. LLSTI Project

There are a number of commercial TTS companies, all of which are steadily expanding the number of languages they offer according to likely markets (Multilingual Text-to-Speech synthesis, 1998). Our current open-source project, the "Local Language Speech Technology Initiative" (LLSTI) is focused on the development of TTS systems (including training program) for those countries, where the market is unproven and economically poor, and there is little hope of a commercial organisation taking the risk.

The goal of LLSTI project is to enable engineers & linguists without any prior experience of TTS to be able to produce good quality, deployable systems, in a reasonable timeframe. The general approach is to provide a set of tools, which are as language-independent as possible, to provide some basic training, and then to guide partners through the development (porting) process.

To be able to carry out this approach successfully, we worked top-down, carrying out the following tasks:

- o To understand from the start what TTS problems have to be solved.
- o To find out what information is available for each language in reference works and (reliable) publications
- o To extract TTS-related knowledge into database

- ❑ To identify technological gaps to be filled in
- ❑ To develop (semi-)automatic tools to solve the TTS problems in an integrated way
- ❑ To investigate possibilities for re-use of modules from existing languages

There is enormous benefit in making all results freely available. This enables a community of interest to be formed, with different people working on different languages and parts of the system, according to their own expertise and interest. LLSTI is committed to enabling and supporting this open-source approach (Tucker and Shalonova, 2004).

1.2 TTS Development as the Way of Linguistic Documentation for Minority Languages

The development of the language-specific modules (grapheme-to-phone converter, morpho-syntactic analyser etc.) in a TTS system are one way of formalising linguistic knowledge, and thus can be considered a form of documentation for minority languages. The modules have the benefit that they can be used as the basis for a range speech and language applications in that language – Text-to-Speech, Machine Translation, ASR and etc. (see Figure 1).
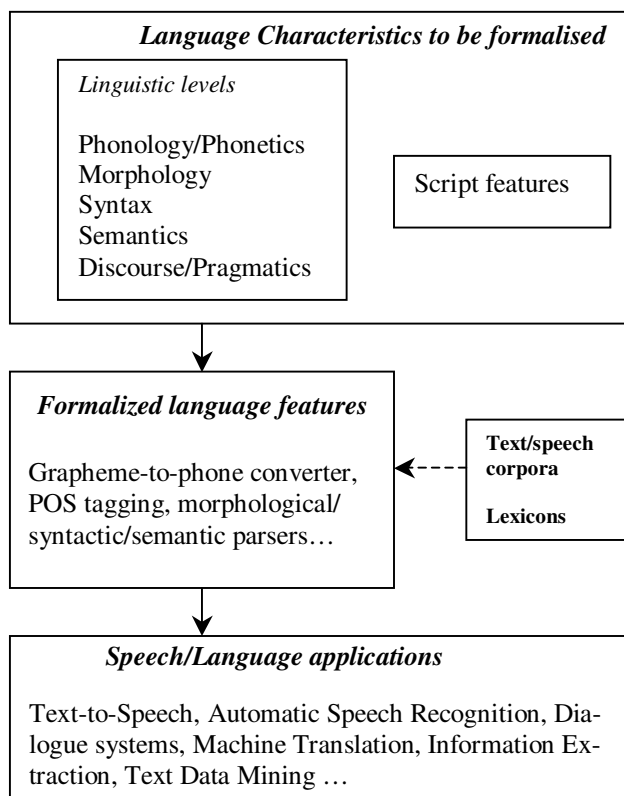

Figure 1: Formalisation of language features for technological transfer between Speech/Language Applications

Formalised linguistic features or language-specific modules can be obtained from annotated text and speech corpora. Data-Driven approaches in Linguistics can be considered as a method both for obtaining new or updating linguistic information.

As the development of TTS systems incorporates formalised knowledge for speech generation on all linguistic levels (phonology/phonetics, morphology, syntax and semantic/pragmatics), it can be considered as an optimal start point for linguistic documentation of poorly investigated languages. As mentioned above, certain linguistic modules and corpora produced for TTS, can be re-used in other applications.

1. 3. TTS Development Complexity Score

Performing a survey of languages and scripts used worldwide has enabled us to put what information is available in a database, and identify the problems which will be encountered in building TTS for them. We rank the languages by the TTS development complexity score (Shalonova and Tucker, 2003). This database has formed the foundation of the local language TTS program. The TTS-related complexity for a particular language is calculated by summarising all its script and language feature scores.

Languages / Linguistic Features	Hindi	Ibibio	Tamil	Zulu	Swahili
Tones (Cues for tone assignment)	0	2	0	2	0
Lexical stress (Cues for lexical stress assignment)	0 (1)[1]	0	0	0	0
Secondary stress or rhythm	0	0	0	0	2
Morpho-syntactic characteristics	1	1	1	1	1
Morphological characteristics (derivation)	1	1	1	1	1
Proper syntactic characteristics	0	0	0	0	0
Other characteristics		2[2]		2	

Table 1: Complexity for Language Features

Languages / Script Features	Hindi	Ibibio	Tamil	Zulu	Swahili
Capitalisation	1	0	1	0	0
Consisting Grapheme-to-phoneme rules	1	0	0	1	0
Symbols for loan words	0	0	-0.5	0	0
Symbols for stress	0	0	0	0	0
Symbols for tones[3]	-	0	-	0	-
Punctuation marks	0	0	0	0	0

[1] The existence of Lexical stress in Hindi is disputable.
[2] Terraced tone system related to grammatical characteristics
[3] In combination with the field *Tones* in the table *Languages Features*.

Languages / Script Features	Hindi	Ibibio	Tamil	Zulu	Swahili
Spaces between words	0	0	0	0	0
Homographs	0	1	0	1	1
Other characteristics	0	0	0	0	0

Table 2: Complexity for Script Features

Score / Languages	Intelligibility (basic)	Intelligibility (full)
Ibibio	5	7
Hindi	2 (3)	4 (5)
Swahili	0	5
Tamil	0	2.5
Zulu	6	8

Table 3: Summarized Complexity score

In Table 1 and Table 2 we score Hindi, Ibibio, Swahili, Tamil and Zulu regarding script and language complexity, in Table 3 – we summarise the scores for evaluating the complexity for creation of TTS systems with basic and full intelligibility. By basic intelligibility we mean generally correct grapheme-to-phoneme conversion & stress; full intelligibility also has correct secondary stress, homograph disambiguation etc. The same scale can be applied to all languages worldwide. It should be noted that some linguistic information for particular languages is either missing or contradictory.

2. Characteristics of the developed TTS systems

We are currently developing TTS systems with the following characteristics:
- concatenative TTS
- diphone as a minimal speech unit (as we are using data-driven approach, larger units such as triphones, words and word combinations can be selected from the Speech Database)
- data-driven approach in speech database creation
  - multiple candidates per diphone without prosody modification (currently)
  - combination of a rule-based and data-driven approach – context-sensitive diphones with further prosody modification (in the future)
- uses Festival/Festvox as a basis (http://festvox.org)

In addition to the modules currently available in Festival/Festvox, we provide the following modules:
- Phonetically balanced algorithm for creating an optimal set of sentences for creating Speech Database
- Language-independent Morpho-Syntactic analyser
- Language-independent Intonation extraction/modelling system
- Improved pitch marking tool

- Evaluation procedure tools for testing.

We also support the run-time engine Flite for professional deployment of Festival voices (crucial for telephone applications that need to process several calls at a time and for Windows or PDA-based applications).

The basic training provided currently centers on two major topics:

- speech and text annotation/segmentation
- TTS development overview.

Courses have been held in Bangalore (India) and Bielefeld (Germany). The last course was held with the support of Prof. Dafydd Gibbon.

3. TTS-related problems on different linguistic levels

The current LLSTI partners are developing TTS for the following languages: Hindi, Ibibio, Swahili, Tamil and Zulu. The problem with the languages under development is the lack (or very small amount) of speech and text corpora. It is one of the reasons why their linguistic structure is very poorly investigated, especially such linguistic levels as phonetics (including prosody). Below we present problems on different linguistic levels that we have already experienced while developing TTS systems.

3.1. Phonology/Phonetics

3.1.1. Selecting a normative speaker

TTS systems should normally generate speech that will be accepted by most local people for whom the synthesis is actually developed. For this reason speech databases for TTS are usually recorded by speakers with normative pronunciation. It is not straightforward to define what is normative speech for a language with various dialects (one dialect is normally considered to be the normative pronunciation). The standard pronunciation can be determined by several ways:

- The speech of broadcast readers of central TV/Radio stations can be considered as standard.
- Socio-linguistic study can be carried out. This type of research requires a lot of effort – plenty of recordings and their analysis. Nevertheless, it is the most reliable method as it allows verifying changes in speech culture and thus defining the normative speech (pronunciation standard typically changes significantly over a 20-30 year period).
- "Compulsory" appointment – the speech of a particular person (professor, writer, actor…) can be defined as standard.

For the current project the first method (a broadcast reader/actor of a theatre in a capital city) is taken as a start point as it is the easiest to handle.

For our partners we provide a document for speaker selection procedure that describes the set criteria to be taken into account. It is interesting to notice that for European languages the speaker should normally have a loud and distinctive voice, whereas in Ibibio culture, for example, it is very insulting to speak loudly, so the synthesis will have to replicate a quiet voice with the corresponding voice quality.

3.1.2. Optimal Allophone/Phone inventory and G2P rules for TTS Speech Database

In order to create an optimal speech database for TTS, it is necessary to go through an iterative procedure where segmentation/annotation of the recorded speech material for the speech database itself can change both the phoneme/phone set of a language and grapheme-to-allophone(phone) rules. In this case the data-driven approach for Voice font creation (where the speech database contains a large number of real sentences that have to be segmented/annotated) can be considered as an optimal start point for obtaining new phonetic knowledge about a language.

Ideally, speech corpora required for TTS can be subdivided into 2 parts:

1. Speech corpora for the database itself
2. Speech corpora for phonetic research (including research in prosody).

In our project due to the lack of time we are using only the first type of corpus (approx. 400-600 real sentences) for research purposes and differentiate only between 2 sentence types: declarative sentences and yes/no questions. This strategy is sufficient enough for obtaining preliminary results for segmental (grapheme-phoneme-allophone-phone variation) and suprasegmental (pitch and duration) characteristics.

For each particular language it seems important to find the trade off between the number of allophonic/phone variations (between the number of speech units- diphones) and the degree of detail of acoustic/phonetic transcription required for obtaining natural TTS systems. A great number of phonetic variation in speech due to the influence of nasal consonants, position in a word/phrase/sentence etc. can be represented in the recordings of real sentences for the TTS speech database rather than in the diphone inventory itself. The provided algorithm for choosing Phonetically Balanced Sentences allows taking into account essential phonetic phenomenon while creating the Speech Database.

3.1.3. Implementation of Prosody Rules

Prosody - intonation (pitch variation), duration, stress and syllabification is the most poorly investigated area for the languages under our investigation.

We tried to find an intonation system that is easy to develop from scratch and does not require thorough expertise. Currently we are testing the MOMEL algorithm and INTSINT annotation on several languages (Hirst, (2001). The first results sound promising. We intend to publish them in the near future.

Duration parameters are trained by means of a CART-tree tool (provided by Festival) using the speech database for TTS (400-600 sentences).

It is difficult to define the notion of "lexical stress" for some languages. The difficulty is explained by the fact that acoustically stress is "expressed" by the combination of several parameters such as pitch movement, vowel duration and intensity, which requires thorough phonetic research. For example, there are two contradictory opinions about lexical stress in Hindi - either there is no lexical stress at all or it does exist and depends on the syllable weight. Another problem that is related to lexical

stress is a possible phenomenon of reduced vowels in unstressed syllables. Reduced vowels need to be presented as separate units in the speech database inventory.

3.2. Morpho-syntactic analysis

Currently we are working on the development of a shell for a TTS NLP module. The linguistic specs for this shell are to be filled in by local linguists in India, Kenya, Nigeria and South Africa. The shell is currently a language-independent Morpho-syntactic analyser (details to be published shortly). The analyser has a powerful context-free mechanism that allows to process languages with different morpho-syntactic complexity.

Our experience with Morphological Analyser shows that due to the lack of available lexica for most of our languages, we desperately need a Morphological Learning Tool. This tool has to provide the possibility for obtaining both rules and data (stem and affix dictionaries) on the basis of a limited lexicon (starting from approx. 10.000 units). One of our partners (IIIT Hyderabad) are working on such a tool, but so far it is tested only for Hindi. For Tamil and Swahili a Morphological Analyser is not required for creating a basic TTS system, whereas for Hindi this tool is crucial for prediction of schwa deletion in G2P module.

As for syntactic analysis, in this project we are working on chunking (not full syntactic parser) that will be the basis for assignment of phrases for intonation modeling. We intend to use the same speech database for TTS in order to obtain a preliminary set of rules for phrasing. To the best of our knowledge, phrasing mechanisms for our languages have not been investigated at all. As a start point, we took the algorithms for such commonly investigated languages as English and French (e.g., Black and Taylor, 1994).

Besides phrasing solutions, a syntactic analyser is used in the current project for tone assignment for Ibibio and Zulu. As tones in these languages have grammatical meaning, morpho-syntactic analysis is required.

Linguistic tools related to Semantics and Discourse/Pragmatics are currently not under our development. As the incorporation of such knowledge into TTS will improve Intonation modelling, we hope to deal with this problem in the future.

3.3. Scripts

So far we have not developed TTS for the languages with "complex" scripts such as Arabic with optional vowel marking or Thai with lack of spaces between the words. The only problem that we have experienced is the lack of special symbols for marking tones (basic tones) in Ibibio and Zulu that requires dictionary look-up.

We have experienced an interesting problem with the Ibibio language regarding script and NLP processing. This language does have script, but only a few written texts can be found (mainly several short fairy tails). Prof. Dafydd Gibbon and his team propose creation of written texts/corpora either on the basis of the existing dictionary or by writing down radio broadcasts.

4. Conclusions and future work

In the framework of the LLSTI project we aim to provide solutions for most TTS-related problems that arise on different levels. We are interested in testing our language-independent modules (Morpho-Syntactic Analyser) and techniques (defining of optimal diphone inventory, speaker selection etc.) on a greater number of languages (especially on "lesser investigated" ones).

The lack of linguistic knowledge requires the development of efficient tools (or procedures) for obtaining linguistic rules from scratch. Currently we are testing the Intonation Modelling System for automatic extraction of pitch movements. One of the tasks for the future will be testing different statistical and ML approaches for the TTS modules (e.g., for creating Morphological Learning Tool) and selecting the most appropriate approach on the basis of performance.

5. Acknowledgement

6. References

Black, A. and Taylor, P. (1994). Assigning intonation elements and prosodic phrasing for English speech synthesis from high level linguistic input, ICSLP'94, Yokohama, Japan, pp. 715—718.

Hirst, D.J. (2001). Automatic analysis of prosody for multilingual speech corpora, E.Keller, G.Bailly, J.Terken & M.Huckvale (eds) Improvements in Speech Synthesis, Wiley.

Multilingual Text-to-Speech synthesis. (1998). The Bell Labs approach. Editor R.Sproat. Kluwer Academic Publishers.

Shalonova, K. and Tucker, R. (2003). South Asian Languages in Multilingual TTS-related Database, EACLWorkshop on Computational Linguistics for the Languages of South Asia - Expanding Synergies with Europe, Budapest, pp. 57-63.

Tucker, R. and Shalonova, K. (2004). The Local Language Speech Technology Initiative - Localisation of TTS for Voice Access to Information, Crossing the Digital Divide shaping technologies to meet human needs, SCALLA Conference, Nepal.
http://www.elda.fr/en/proj/scalla/SCALLA2004/tucker.pdf