# The Lessac Technologies System for Blizzard Challenge 2010

*Rattima Nitisaroj, Reiner Wilhelms-Tricarico, Brian Mottershead, John Reichenbach, Gary Marple*

Lessac Technologies, Inc., USA

{rattima.nitisaroj, reiner.wilhelms, brian.mottershead, john.reichenbach, gary.marple}@lessactech.com

## Abstract

For Blizzard Challenge 2010, Lessac Technologies built its first British English voice from the provided full database. To enhance methods for target cost calculation and unit selection, instead of traditional phonetic symbols, we used a more fine-grained set of Lessemes to label units and applied the Hierarchical Mixture of Experts model to map linguistic features to acoustic parameters. The evaluation results show that we performed relatively well on similarity to the original speaker, and comparable to most systems with respect to naturalness. The high word error rate suggests that we need to improve on signal processing for concatenation.

**Index Terms**: speech synthesis, Blizzard Challenge, Lesseme

## 1. Introduction

Lessac Technologies has developed an approach for concatenative speech synthesis in which expression, voice, and style are fundamental. Prior evaluations demonstrate that our text-to-speech system yields near human-quality expressive speech for General American English. Participation in the Blizzard Challenge gives us an opportunity to reach a larger and wider pool of listeners, and to compare our system with others to know where we are and which areas we need to improve. This is our first entry to the challenge and our first time building a British English voice. The next section provides a description of our text-to-speech system. Section 3 explains the process of building the 'rjs' voice for our Blizzard participation. Results from the listening test and related discussion can be found in Section 4. The final section concludes the paper.

## 2. Lessac Technologies Text-to-Speech System

Similar to other systems, Lessac Technologies text-to-speech system consists of two main components: the front-end, which takes plain text as input and outputs a sequence of graphic symbols, and the back-end, which takes the graphic symbols as input to produce synthesized speech as output. In what follows, we visit the properties that distinguish our system from others and, we believe, play an important role in producing expressive synthesized speech.

### 2.1. Use of Lessemes

Successful production of natural sounding synthesized speech requires developing a sufficiently accurate symbolic set of sound representations that can be derived from the input text, and that relate the input text to be pronounced with the corresponding synthesized speech utterances that are heard by the listener. Rather than adopting traditional symbolic representations, such as IPA, SAMPA, or ARPAbet, Lessac Technologies has derived an extended set of symbolic representations called Lessemes from the phonosensory symbol set for expressive speech as conceived by Arthur Lessac [1]. The Lesseme system for annotating text explicitly captures the musicality of speech.

In their basic form and meaning, Lessemes are symbolic representations that carry in their base form segmental information just like traditional symbolic representations. To be able to describe speech more accurately and to include in the symbol set information that is not carried by a typical phonetic symbol, each base Lesseme can be sub-typed into several more specific symbols which then represent phonetic information found in traditional phonetic symbols plus descriptors for co-articulation and supra-segmental information. Acoustic data demonstrate different properties of a set of Lessemes which are normally collapsed under one phonetic label in other systems [2].

At present, for the General American English, with the present Lesseme specification, there can be as many as 1,500 different Lessemes. Compared to other sets of representations which usually contain about 50 symbols, Lessemes allow more fine-grained distinction of sounds. Units of the same type share closely similar acoustic properties. By having supra-segmental information directly encoded in Lessemes, we believe our system can target available units for concatenation better than a system with a relatively impoverished intonation annotation scheme. This should be useful especially when trying to produce expressive speech from a very large database.

### 2.2. Front-end with extensive linguistic knowledge

The front-end which derives Lessemes from plain text input is a rules-based system. The rules are based on expert linguistic knowledge from a wide variety of fields including phonetics, phonology, morphology, syntax, light semantics, and discourse. Simplistically, the LTI front-end labels text, building from, at the lowest level, letters, spaces and punctuation marks. These letters, spaces and punctuations are interpreted by the front-end, and assembled as syllables, words, phrases, sentences, and paragraphs to be spoken, along with context-aware labeling for appropriate co-articulations, intonation, inflection, and prosodic breaks.

First, the input text is processed by a syntactic parser which generates the most likely syntactic tree for each sentence, and tags words with part-of-speech (POS) information. In the next step, words are transcribed by use of a pronunciation dictionary into base Lessemes accompanied by lexical stress. Homograph disambiguation based on POS tags takes place at this step. Subsequent processing steps modify the base Lessemes by making successive decisions based on the overall phrase and sentence structure. In particular, prosodic breaks are inserted in meaningful places by taking into consideration factors such as punctuation, phrase length, syntactic constituency, and balance. In most phrases, an operative word is marked which carries the highest pitch

prominence within the phrase. In addition, Lessemes are assigned inflection profiles and one of two degrees of emphasis. Context-based co-articulations across word boundaries are also captured. The result is a full Lesseme for each sound which encodes expressive intonational content in addition to segmental information found in traditional phonetic symbols. Once the front-end process on a plain text has been completed, a Lesseme stream is delivered to the signal processing back-end.

### 2.3. Voice database construction

In addition to the machine readable form used as the input to the signal processing back-end, Lessemes are also used in creating new voices, namely to automatically generate a human readable graphic output stream which can be thought of as annotated text plus a musical score, as illustrated in figure 1.

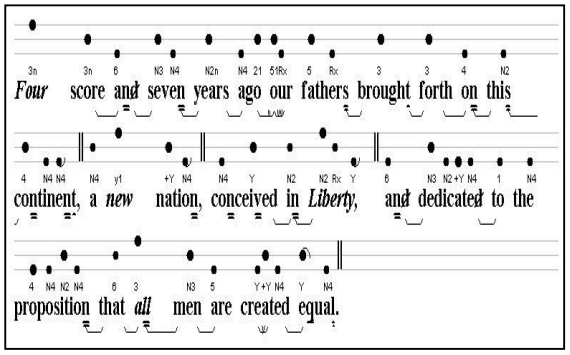


Figure 1: *Lessac Technologies annotated text*

In the annotation, vowel orthographic forms are designated with Arthur Lessec's phonosensory symbols. Consonant orthographic forms are marked with information indicating whether the consonant is sustainable (double underlined) or percussive, i.e. pronounced with a brief contact within the mouth (single underlined), as well as how the consonant is linked to the next sound in connected speech. The musical notation above the orthographic forms depicts 'notes' of an intonation pattern that a person with sufficient voice training can follow. Each syllable corresponds to a note. Higher notes are pronounced with higher pitch. Large notes define stressed syllables while small notes refer to unstressed syllables. Some notes are further specified with an inflection, which reflects a particular shape of pitch movement within the syllable.

During the voice database construction, the text to-be-recorded is first processed by the front-end, yielding the stream of Lessemes. The resulting stream is then transformed into a human readable form, as seen in figure 1, which we use as the combined script and score for a trained voice talent during recordings to construct a voice model. The way the voice talent records the prompts is controlled by the annotated text and musical score. The recordings of the prompts are then segmented and labeled with the same Lessemes that underlie the script and score that the voice talent followed. The fact that the same Lessemes are output for the voice talent script as well as the labeling of the database creates a direct link between each speech snippet and its Lesseme label, thus a high degree of correspondence between the symbols and the sounds as actually recorded by the voice talent. Such high degree of symbol-to-sound correspondence is not guaranteed in the typical voice database construction, where the voice talent sees only plain text and the subsequent recordings are labeled with the symbols generated by the front-end.

### 2.4. Hierarchical Mixture of Experts for mapping linguistic features to acoustic parameters

To enhance methods for target cost calculation and unit selection, we apply the Hierarchical Mixture of Experts (HME) model [3] [4] to learn a direct relationship or mapping between the Lesseme representation of the input text and the ideal acoustic observables measures in the recordings.

A functional diagram of the HME model is shown in figure 2.

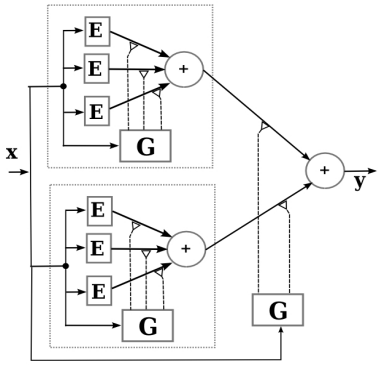


Figure 2: *Hierarchical Mixture of Experts model. (E: experts, G: gates, x: input, y: output)*

The HME model applied to the problem of mapping prosodic features to acoustic observables makes use of the interpretation of the model as a parameterized mixture of Gaussians. Each expert in the model represents one multi-dimensional normal distribution with a variable expectation vector that depends on the input x. The parameters for each expert also include a covariance matrix that is estimated and updated during the training. Each block of experts in a group or clique (3 experts in each of 2 cliques in the figure) together with a gating network represent one mixture of Gaussians whereby the mixture coefficients are computed in the gates as a function of the input. Multiple groups of experts can be combined by another gate in a similar way. The complete network represents a mixture of Gaussians whose parameters are trained from pairs of known input and output. During the learning process, the parameters in the experts and gates are adjusted so that, for a given known input x, the probability of obtaining the desired known output y is maximized.

In our application of the HME model, the input x includes the linguistic features and the output y are acoustic observables, which include MFCC, F0, duration, and intensity. The model is applied recurrently, which means that the acoustic observables predictions for one sound are included in the input x for the prediction of the next y.

We use supervised learning with the HME model to map linguistic feature sequences to acoustic parameters. The structure of the model is shown in figure 3. The system steps through a sequence of Lessemes and predicts for each Lesseme the vector of acoustic parameters that specify the unit, whereby the input to the model consists of the feature information of the previous, the current and the next two Lessemes. Further, by feeding back the previously predicted acoustic parameter vectors as input to the model, the model

becomes partially auto-regressive. This facilitates the learning task because the model only has to learn to predict the current acoustic vector conditioned on the last two acoustic vectors and the input linguistic features. Learning proceeded in two phases. Initially, the looped-back input to the model is the actual acoustic vectors until the model begins to converge. Then, training is continued by having the predictions for the last two time slots become inputs for the prediction of the current time slot. Learning then proceeds by repeatedly processing a large number of sentences in the database, until the error variance minimizes at the valley.

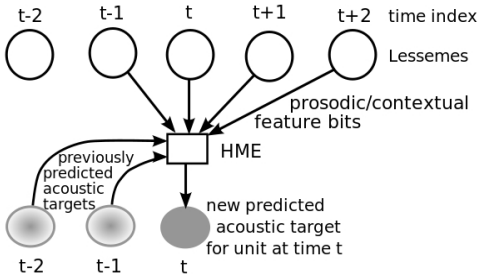


Figure 3: *Recurrent and partially auto-regressive prediction of intonation contour and other acoustic targets by HME*

During the target cost calculation process, we compute the cost as the distance of the acoustic parameters of a candidate unit from the ideal trajectory, which is in turn directly predicted from the linguistic feature variables.

# 3. Building 'rjs' Voice

Blizzard Challenge 2010 provides participants with several tasks for English and Mandarin. We did not participate in any of the Mandarin tasks because, although we do not believe that the approach would be fundamentally a lot more complicated for Mandarin than for English, we have not done the up-front work that would be required to produce a Mandarin text-to-speech engine. We chose to participate in the English task of building a voice from the British English 'rjs' database with 4014 utterances (EH1) as our system has been designed to work with a very large database. From the experience, more than 15 hours of recordings give very good synthesis quality, while the results considerably deteriorated with a smaller database of 4-hour recordings. The 'rjs' database is the largest database provided in this challenge. It contains approximately 6 hours of recordings.

## 3.1. Transcription to Lessemes

Although the phonetic transcriptions for all the utterances were provided, we decided not to use them because our entire system is driven by Lessemes. Although it is possible to align the provided phonetic symbols with Lessemes, our experience showed that the process took time and involved a number of modifications to the system to be able to handle all one-to-many and many-to-one correspondences. Thus, we only ran the utterances through the front-end to generate Lessemes and did not do further mapping from the provided phonetic symbols to Lessemes.

Regarding the dictionary, we used an American English pronunciation dictionary to transcribe the words into Lessemes. A British English Lesseme dictionary has yet to be developed. For the current task, there is enough overlap between the two to justify the use of an American English dictionary to produce British English speech.

For a handful of words that were not in the dictionary, we manually added them to the dictionary. Our system has the letter-to-sound rules, but since the 'rjs' database is relatively small in comparison with the database we normally work with, we wanted to include as many quality units as possible and thus did not let letter-to-sound rules spoil some units.

As seen in Section 2, the use of Lessemes allows our system to have more control over the prosodic aspects of speech. However, in the current exercise, we did not have the recordings done by a Lessac trained voice talent reading the annotated prompts and score similar to Figure 1. The correspondence between Lessemes and the recorded sounds can only come from the performance of the front-end. In more recent work, we have used the Lesseme front-end parser to process text for the annotation of already existing recordings in order to create multiple additional new voices. The synthesized output appears to capture much of the prosodic quality found in the original recordings, and we expected similar results in the speech synthesized from the 'rjs' voice.

## 3.2. Automatic segmentation

Segmentation was done based on features that were extracted by filtering the speech waves through a bank of gamma tone filters, followed by extracting the envelope amplitude for each channel and low-pass filtering both the amplitude and the sample differentials of the channels, providing 48 channels in total which where then reduced by using the first 10 coefficients from a principal component analysis. The ehmm model in speech tools was used for processing the segmentation. For the provided speaker data rjs, the method did slightly better (but not significantly better) than the standard method of segmentation based on mceps and delta mceps, which we tried first. As we do for other voices, in order to reduce the total number of states for the EHMM, we collapsed several of the Lesseme classes used in our annotation into larger super classes, in such a way that 339 states remained.

## 3.3. Database creation

As our phonetic labeling and concept of the prosodic structure is different from the information found in the provided utterance files, we did not use the provided utterance files, but generated our own. We modified Festival feature functions to produce relevant linguistic features at segment, syllable, word, and phrase levels based on the Lessemes and prosodic breaks that the front-end output. The end time of each unit came from the label files produced by the automatic segmentation. As for the acoustic parameters, we extracted MFCC, F0, duration, and intensity. All the linguistic features and acoustic parameters were collected into a binary catalog file, which was then used to train the HME model offline and called by the synthesizer during run-time.

## 3.4. Synthesizer

While Lessemes help narrow the pool of candidates for unit selection and enable more precise targeting, labeling units with Lessemes can lead to the problem of non-existing or sparse units of particular labels in the database, especially the small database. We handled the problem by incorporating a set of fail-over rules. Whenever the target Lesseme has a very limited number of or no matching candidates in the database, the rules look for closely matched Lessemes, e.g., those with a different inflection or pitch level, to include among the candidates for the target and join cost calculations.

Similar to [5], our join cost calculation discourages joins between sonorant sounds. The join penalty varies depending on the types of joining sonorants. For example, the join between two vowels get higher penalty than the join between a vowel and an onset lateral sound.

After the best units are selected, they were put together with very simple time-domain. Two signals are concatenated, the new unit to the preceding sound, by a blend-over, which makes use of a blending function similar to a hyperbolic tangent function, but approximated by two continuously connected third order polynomials. The signal is multiplied by the blending function which gradually changes from 1 to 0 for the first signal by the time-reversed blending function that gradually increases from 0 to 1 for the second signal. No attempts were made to alter intensity or fundamental frequency of the original speech signals.

# 4. Results and Discussion

Seventeen systems participated in the EH1 task (building a voice from the full dataset). During the online evaluation of the task, listeners were asked (i) to judge how similar a system is to the original speaker, (ii) to provide mean opinion scores (MOS) representing how natural or unnatural the utterances from the news and novel domains sound, and (iii) to transcribe the semantically unpredictable sentences (SUS) they heard. The listeners included paid participants, volunteers, speech experts, native and non-native English speakers. Results for our system in comparison with a standard Festival unit-selection system and others are presented below.

## 4.1. Similarity to original speaker

With respect to how similar synthesized speech is to the original speaker's speech, Lessac Technologies is one of the six systems with the median score of 4 on the 5-point scale. The Festival Benchmark system also falls within this group. Pairwise Wilcoxon signed rank tests reveal that two systems score significantly higher than us. Figure 4 illustrates a comparison among the natural speech, the Festival system, the average of all systems, and our system.

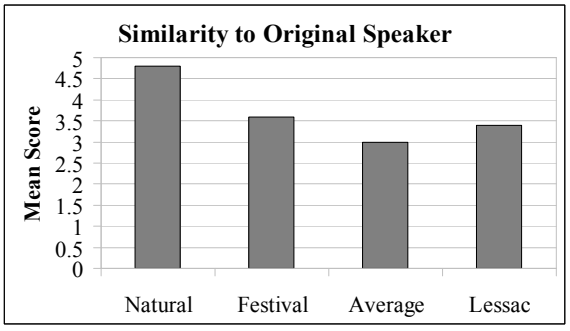


Figure 4: *Mean scores for similarity to original speaker*

Although the Festival system achieves a higher mean score than us, the pair-wise tests do not exhibit a statistically significant difference. Compared to the score averaged from all the participating systems, the utterances produced by our system sound more similar to the natural speech.

As previously mentioned, we used an American pronunciation dictionary to build and synthesize the British English voice. In some cases, this could result in low similarity ratings as some sounds, especially vowels, representing an American accent may be selected for concatenation. We believe that when a Lesseme dictionary for British English becomes available for use in text-to-speech synthesis, the similarity ratings will improve.

## 4.2. Naturalness

A 5-point scale MOS was used to evaluate how natural synthesized speech sounds. Three systems achieved the median score of 4. Lessac Technologies is among the ten systems with the median score of 3. Within this group, the performance of our system does not significantly differ from three other systems, including the Festival Benchmark system, and we score significantly better than the remaining six systems. A comparison among the natural speech, the Festival Benchmark system, the average of all systems, and the Lessac Technologies system is provided in Figure 5.

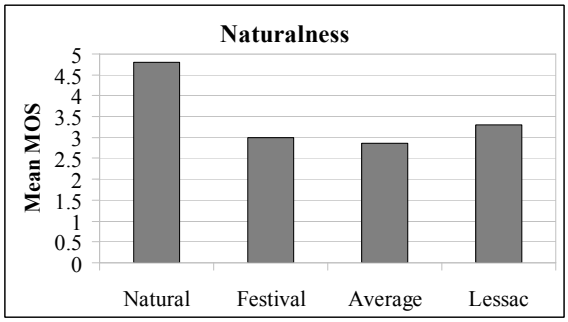


Figure 5: *MOS for Naturalness*

The naturalness of our synthesized speech appears to be comparable to what was produced by other systems, including the Festival Benchmark system. However, when compared to natural speech, there is still a lot of room for improvement. It would be very useful for us to have the results for individual sentences so we can perform further analyses, especially on those sentences with low scores.

## 4.3. Word error rates

Figure 6 demonstrates the word error rates for the natural speech, the Festival system, the average of all systems, and our system respectively.

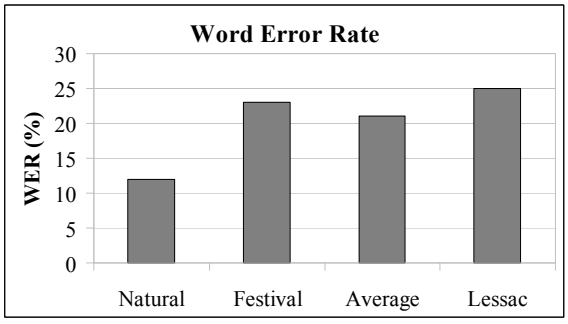


Figure 6: *Word error rates for SUS*

The first thing to note is that the natural speech received 12% word error rate. This confirms the close relationship between the identity of a given word and the semantics of its context. Deprived of meaningful context, listeners cannot perfectly identify the words they hear.

Regarding the performance of our system in the SUS test, the figure shows that we are very far from the ideal. We rank

behind other twelve participating systems with respect to the word error rate. We think that the poor performance came from our very simple signal processing method, described in section 3.4. In many concatenation points, the method cannot get rid of glitches or low-level reverberations, which distract listeners and make it difficult to identify the words they hear, especially when the sentence does not provide the necessary semantic clues.

## 5. Conclusions

Our weakest point currently is in the signal processing for concatenation, and admittedly we have neglected that aspect relative to the others. We are doing fine on naturalness. That we are getting relatively good results for similarity to the original speaker is promising, and it gives us some confidence that it is worthwhile to try to represent and capture in the synthesis model idiosyncratic properties of the original voice that are not directly represented by known explicit models. Instead, we introduced Lessemes, which carry both segmental and supra-segmental information, and chose methods of machine learning using a simple but sufficiently comprehensive model that may be able to discover some of these properties and represent them for the for the context dependent prediction of all acoustic feature variables, while making few assumptions about the nature of the relationship between acoustic signal parameters and perceived prosody. The participation and evaluation by the Blizzard challenge was very helpful for us, even though we would have liked to have our system tested not just on short phrases and semantically unpredictable sentences, but much more on reading a short story of a few pages. Longer synthesized speech could be used in a listening comprehension test, which would be where prosody and expressiveness play a larger role.

## 6. References


[1] Lessac, A., The Use and Training of the Human Voice: A Bio-Dynamic Approach to Vocal Life, McGraw-Hill, 1996.

[2] Nitisaroj, R. and Marple, G. A., "Use of Lessemes in text-to-speech synthesis", in M. Munro, S. Turner, A. Munro, and K. Campbell [Eds], Collective Writings on the Lessac Voice and Body Work: A Festschrift, Llumina Press, 2010.

[3] Jordan, M. I. and Jacobs, R. A., "Hierarchical Mixtures of Experts and the EM Algorithm", Neural Computation, 6:181-214, 1994.

[4] Ma, J., Xu, L. and Jordan, M. I., "Asymptotic Convergence Rate of the EM Algorithm for Gaussian Mixtures", Neural Computation, 12:2881-2900, 2000.

[5] Kominek, J., Bennett, C., Langner, B. and Toth, A., "The Blizzard Callenge 2005 CMU Entry: A Method for Improving Speech Synthesis Systems", Proceedings of Interspeech 2005, 85-88.