Computing phonological generalization over real speech exemplars

Robert Kirchner
Linguistics Dept.
University of Alberta
kirchner@ualberta.ca

Roger K. Moore
Dept. of Computer Science
University of Sheffield
r.k.moore@dcs.ac.uk

Draft: December 29, 2008

Abstract

We present an Exemplar-Theoretic confidence-sensitive dynamic programming model of speech production, PEBLS (Phonological Exemplar-Based Learning System), and test it with real acoustic speech signals. We focus on the computational problem of how to generate an output that generalizes over a collection of unique, variable-length signals, without resorting to a priori phonological units such as phones or syllables. We show that PEBLS displays pattern-entrenchment behaviour, central to Exemplar Theory's account of phonologization.

1 Introduction

1.1 The need for an explicit Exemplar Theoretic model

Since Goldinger's (1996, 2000) experiments suggesting memory for speaker voices as part of lexical representation, and Johnson's (1997) seminal application of this idea to speech perception, Exemplar Theory (ET) has attracted steadily increasing interest among phonologists and phoneticians (e.g. Kirchner 1999, Bybee 2001, Pierrehumbert 2001, 2002, Gahl & Yu 2006 and articles contained therein, Port 2007, Gahl 2008). For ET potentially affords elegant accounts of frequency effects, sociophonetic variation, gradient sound change; and more generally, provides a seamless phonetics-phonology interface. Exemplar-based approaches have also attracted recent interest in the automatic speech recognition (ASR) field, for their ability to exploit fine phonetic detail in recognition (e.g. Moore & Maier 2007).

ET's development, however, has been hindered by the lack of an explicit computational speech processing model, capable of applying to real speech data, without which its claims cannot be rigourously tested. The recognition side of the model is not the central problem. A number of exemplar-based recognition models have been put forward, from Johnson's (1997) original X-Mod to the large-vocabulary continuous ASR system of DeWachter 2007. All a recognition model need do is assign a category label (or a sequence thereof) to an input signal based on its similarity to the variously labelled speech exemplars in memory.[1] (For concreteness' sake we assume these categories to be *words*, though they might extend to phrases or whole utterances as well; they do not, for our purposes, include phonological units: segments, syllables, and the like). Most of the interesting phonological phenomena attributed to ET, however, pertain to the production side of the model, or at least crucially involve production as part of the story.

[1] This is an oversimiplification. Moore 2007 argues for a recognition system that includes an analysis-by-synthesis component (and likewise, for a synthesis-by-analysis component in production). In exemplar-based terms, while the recognition system decides, on auditory grounds, what category to assign to the input, it also emulates the production of the input, and uses the resulting articulatory similarity to influence the recognition decision. By giving special weight to self-produced exemplars, this partial analysis-by-synthesis thus induces some speaker normalization of the input signal for recognition purposes.

1.2 The production problem

Production involves a harder problem: generation of a concrete signal (in principle, a motor plan[2]) from a target word category (or a sequence thereof).

Naively, one might suppose that an exemplar-based production system could work simply by selecting some exemplar of the target word and reproducing it verbatim (i.e. playback). The playback method, however, lacks any mechanism for *generalizing* over a set of exemplars, and so its productions are limited to its previous experiences. It thus fails to model many key properties of human speech processing (and many desirable properties of an automatic speech processing system). For example, humans have the capacity to produce words which they have never uttered before, e.g. repeating a word just learned from another speaker. At the point of hearing this new word (and recognizing it as such), the relevant speaker acquires an exemplar encoding her auditory experience of the word, but no corresponding articulatory experience. Without articulatory information for this word, no motor plan can be "played back" as output to the speaker's vocal tract. This deficiency can only be overcome by generalizing: in ET terms, forming a motor plan based on subsequences of exemplars of other words with similar auditory cues. In fact, the generalization issue is pervasive in speech production. Consider production of a word in some previously unencountered syntactic or pragmatic context, e.g. where it is subject to some phrasal phonological process; where it receives contrastive stress; or where a whispered or shouted production of the word is felicitous. Again, an adequate production model needs to generate a *composite* output – one that blends together some exemplars of the target word with contextually appropriate subsequences grabbed, perhaps, from exemplars of other word categories. More generally, Pierrehumbert 2001 shows that, in an exemplar-based production model without generalization, categories (word, phone, or any other level) increase their variances with each iteration of the production-perception loop, leading to massive collapse of the categories.[3]

Pierrehumbert therefore proposes generation of an output by averaging over a group of exemplars, namely some randomly selected exemplar of the target category, and its neighbours within a certain distance radius. However, Pierrehumbert applies this model – the most explicit exemplar-based production model to date – only to low-dimensional static data. Pierrehumbert's model can readily be extended to higher-dimensional data. But it is not clear how it might be extended to real speech, which, in addition to being multi-dimensional, is variable-length time-series data. To recap, the production system needs to be able to generalize, but how can it generalize over a collection of unique, variable-length speech signals?

One response to this problem, adopted (but not computationally fleshed out) in Pierrehumbert 2002, is to appeal to less time-variable units, such as phones (= segments in the phonology literature). Phones can be characterized, albeit crudely, in terms of relatively static phonetic targets. Thus, if our exemplar system parses signals into phone as well as word categories, we can pool together all exemplars of, e.g. /s/, reduce these to fixed-dimensional vectors representing the phone "target" (perhaps with contextual target measurements as well), abstracting away from temporal variation within the exemplars. We can now generate an output based on an average of these fixed-dimensional vector values. However, this segmentation into a priori phonological units seems contrary to the spirit of ET. Categories, to the extent that they play a role in speech processing, should emerge bottom-up from comparison over the exemplars. This approach also fails to do justice to the rich dynamic structure of speech.

1.3 Roadmap

Rather than segmenting the dynamic signal into quasi-static chunks, one might adopt a dynamic model ab initio. In section 2 below, we present such a dynamic exemplar-based production model: PEBLS (Phonological Exemplar-Based Learning System). In section 3 we report results of an experiment testing PEBLS' pattern generalization capacity with real speech. We further show, in a second experiment, that PEBLS propensity for generalization increases with iteration, thus capturing *pattern entrenchment*, one of the core

[2]The modelling results presented below, alas, do not include motor plans, only acoustic data. If we had articulatory data, we could simply add them as further dimensions to the exemplars, and PEBLS, with only minimal modification, could incorporate this information in its computation. Hofe and Moore's (2008) development of an animatronic model of the vocal tract promises to make articulatory data easier to acquire in future.

[3]This is under the assumption that outputs are subject to some non-deterministic variation from their inputs, as a consequence of their implementation by a physical system, namely the vocal tract (or 'slips 'twixt brain and lips').

properties attributed to ET in the literature, but never before demonstrated with real speech data. Finally we discuss parallels between this conception of ET and Optimality Theory.

2 PEBLS

2.1 Framing the problem

To generate an output for a given word, PEBLS begins, as in Pierrehumbert's model, by randomly selecting an exemplar from this word class for use as the *input*.[4] Following Pierrehumbert's terminology, the remainder of the exemplars are the *cloud*. (In the results presented below, we arbitrarily restrict clouds to other exemplars of the same word category.[5]) The clouds thus contain collections of exemplars which are more-or-less similar, but never identical, to the input.

The production problem can now be cast as finding an optimal *alignment* between between the input and the cloud. That is, the output is constructed from subsequences of the cloud exemplars which more-

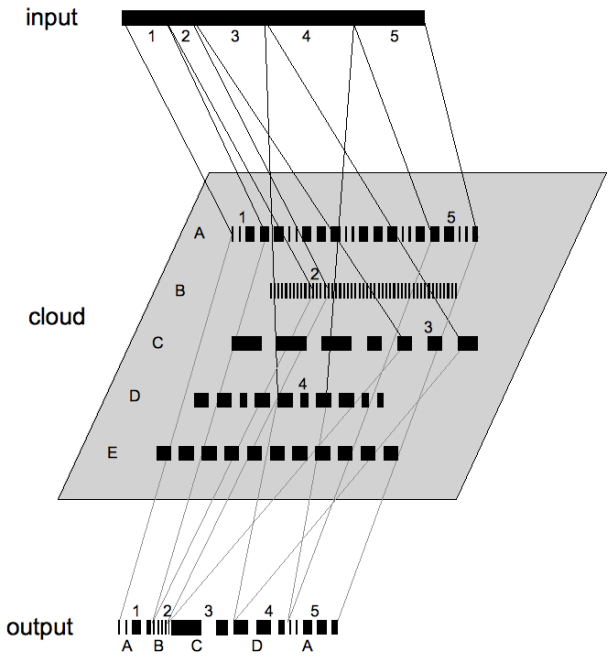

Figure 1: Output as alignment of input with cloud. Numbers indicate corresponding subsequences within the input and cloud, and the concatenation of these subsequences to form the output. Letters show the particular exemplar from which each subsequence was taken.

or-less correspond to subsequences of the input, and which more-or-less reflect typical subsequences (i.e. generalizations) within the cloud, as schematically represented in Figure 1. The trick lies in specifying an alignment criterion that can find these subsequences.

2.2 Dynamic time warping

Dynamic time warping (DTW) provides a computational technique for optimally aligning two variable-length signals A and B, locally stretching or shrinking subsequences within A to best fit B, or vice-versa (see

[4]This method, generating an output based on a particular input exemplar, was chosen to highlight PEBLS' similarities and differences with Pierrehumbert's 2001 model. It is not, however, crucial to PEBLS; we have also developed a version of the model in which the input is simply a vector indicating which word-class is to be activated.

[5]As we are ultimately interested in capturing phonological generalizations that transcend individual lexical items, this is a restriction that we are eager to get away from in future research.

generally Sankoff & Kruskal 1983). Since PEBLS builds upon this technique, it bears some examination. Firstly, DTW presupposes some meaningful measure of distance between timepoints of each of the signals to be aligned. For concreteness' sake, assume we are aligning two speech spectrograms, A and B. Each spectrogram is a series of spectral frames, and we can take the Euclidean distance between each frame of A and each frame of B to construct a distance matrix.

DTW (like all dynamic programming) works by breaking a complex problem down into possible sub-solutions, and for each sub-solution, asking "what's the optimal *sub*-sub-solution from which this sub-solution could have been reached?" and recording the results. In classic DTW, each sub-solution corresponds to a cell in the distance matrix, which can be reached from at most three other cells: by deletion, insertion, and substitution. Cell *(i,j)* of Figure 2, for example, can be reached from *(i,j-1)* (i.e. insertion of a frame of B,

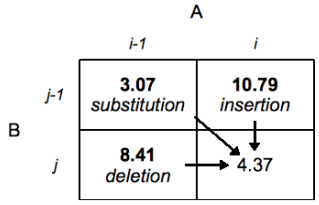

Figure 2: Fragment of a hypothetical distance matrix, illustrating choices for the originating cell for *(i,j)*. Distances in boldface are cumulative.

relative to A), from *(i-1,j)* (deletion of a frame of B, relative to A), or from *(i-1,j-1)* (substitution: advancing a frame in both A and B). The *cumulative* distance of *(i,j)* is computed as

$$D_{i,j} = d_{i,j} + \min(D_{i,j-1}, D_{i-1,j}, D_{i-1,j-1}) \tag{1}$$

where D denotes cumulative distance, and d denotes raw distance. In this case, substitution has the lowest cumulative distance (3.07), so we add this "cost of getting there" to the the raw distance 4.37, the "cost of being there", replacing the raw distance value with the cumulative distance, 7.44 in this cell. We also record the *decision*: which cell has the minimum cumulative distance (i.e. $\arg\min(D_{i,j-1}, D_{i-1,j}, D_{i-1,j-1})$). Because the algorithm proceeds iteratively from upper left to lower right in the matrix, the cumulative distances of the three possible originating cells are always recorded before they are needed for computing the cumulative distance of the current cell. Once all the cumulative distances have been computed, we can trace the decision from which the bottom-right corner cell was reached, then the decision from which *that* cell was reached, iteratively, until we reach the upper-left corner of the matrix. This iterative traceback procedure gives us the alignment, provably the globally minimal distance path through the matrix.

2.3 The intra-cloud transition network

DTW aligns a whole signal with another whole signal: because the choice at every step is limited to insertion, deletion and substitution, the path is monotonic, moving more or less diagonally from upper left to lower right. DTW therefore cannot align, for example, both corresponding parts in tokens of *housework* and *workhouse*. In PEBLS, however – particularly as we wish to avoid a priori segmentation of exemplars into phonological units – we must crucially be able to find alignments of subsequences of one exemplar with subsequences of another exemplar, as suggested in Figure 1. That is, we must be able to pool data on a less-than-whole-exemplar basis.[6] In principle, then, we allow alignment of any frame of the input with any frame of any exemplar within the cloud, transitioning forward or backward in time within any given exemplar, or from part of one exemplar to another. Intuition suggests, though, that some transitions are better than others, namely transitions similar to those instantiated within the cloud. More precisely, if the input contains the frame sequence $\langle p, q\rangle$ while the cloud contains frames r and s (in any location), then the alignment of $\langle p, q\rangle$ to $\langle r, s\rangle$ is permissible to the extent that

[6]Indeed, with this ability, PEBLS can handle whole-utterance exemplars, using its alignment method to find words within longer stretches of speech. This feature becomes necessary if we want to model patterns of phrasal phonology, or the sorts of lexicalization of high-frequency phrases discussed by Bybee 2001. Cf. Tucker & Tremblay (2008) showing onset latencies in a reading task continuously correlating (inversely) with word n-gram probability within several large speech corpora.

- p is similar r,
- q is similar to s, and
- there is a sequence $\langle r, s' \rangle$ or $\langle r', s \rangle$ within an exemplar in the cloud s.t.
  - r' is similar to r, or
  - s' is similar to s.

To compute this permissibility, we construct an intra-cloud transition network: a similarity matrix of the entire cloud to itself, offset by one frame. Similarity of frames *i,j* is related to distance as

$$s_{i,j} = \exp\left(-cd_{i,j}\right) \tag{2}$$

where c is a parameter that scales the steepness of drop-off (following Johnson 1997). Cell *(i,j)* of this matrix thus encodes not the similarity of frame i to j, but the *similarity of* i *to the frame that immediately precedes* j (or, equivalently, the similarity of j to the frame that immediately follows i).[7] By means of this transition network, PEBLS takes into account not only how the input aligns with each exemplar in the cloud, but how the cloud aligns with itself – getting emergent structure from self-similarity within the data.

The algorithm now proceeds, as in DTW, by computing a $U \times V$ cumulative matrix for the alignment of the input (V frames long) with the intra-cloud transition network t (size $U \times U$, with U frames in the whole cloud).[8] As a first pass at the problem, assume that the cumulative similarity S of the v^{th} frame of the input to the u^{th} frame of the cloud is given by

$$S_{u,v} = \max_{i=1}^{U}\left(s_{i,v-1}t_{i,u}\right) + s_{u,v} \tag{3}$$

(Because we have switched from distance to similarity, we now choose based on max rather than min.) Within the max function of the first term, the getting-there score is the cumulative similarity, previously computed, for the *(v-1)*$^{\text{th}}$ frame of the input to the i^{th} frame of the cloud, times the transition network score for moving from frame i to frame u: that is, a good originating point is one with a high cumulative similarity score thus far, and whose transition value into frame u is also high. The second term corresponds to the being-there score, the raw similarity of frame u to v.[9] Finally, the decision is given by

$$\arg\max_{i=1}^{U}\left(s_{i,v-1}t_{i,u}\right) \tag{4}$$

2.4 Confidence sensitivity

The model presented thus far finds the maximum similarity alignment between input and intra-cloud transition network. It thus solves the technical problem of how to generate a concrete speech output from a collection of variable-length speech exemplars. What we want, though, is an alignment that *generalizes* over the cloud (see section 1.2 above), reflecting frame sequences which are in some sense prototypical of the cloud. To highlight this difference, consider a cloud of exemplars, predominantly, but not uniformly, reflecting some phonological pattern, e.g. intervocalic spirantization. If we select as input a token containing a non-spirantized intervocalic sequence, the presence of even a single pattern-violating exemplar in the cloud licenses transitions from vowel to plosive to vowel, notwithstanding the aberrancy of this subsequence relative to the rest of the cloud; and since the non-spirantized subsequence best matches the input, this is the alignment which will be chosen by the maximum similarity criterion. To capture the generalization effect, we need a different criterion: the "getting-there" score should include some measure of the frequency of similar subsequences within the cloud. This problem is analogous to the statistical notion of *confidence*

[7] This network is thus analogous to the transition matrix of an ergodic hidden Markov model, albeit with a unique state for every observation.

[8] Unlike DTW, PEBLS does not allow deletion or insertion. It finds some frame in the cloud as a substitution for every frame in the input.

[9] Though we are not concerned, for present purposes, with recency effects, a la Pierrehumbert 2001, this factor could be incorporated into the model, simply by multiplying the word recency value by the being-there score in eq. 3.

that a particular sample reflects the distribution of an underlying population. We calculate this confidence-sensitive measure by *hierarchically clustering*[10] the whole (U-point) vector of getting-there scores from the previous frame (still calculated as the product of the cumulative and transition scores, as in eq. 3) at each dynamic programming step (under the assumption that similar subsequences will have similar getting-there scores). We identify the optimal cluster w according to the following criterion:

$$w = \arg \max_{i=1}^{2U-1} \left(\frac{\mu_i N_i}{\sigma_i^2 + 1} \right) \tag{5}$$

where μ_i is the mean getting-there score, N_i the size, and σ_i^2 the variance, of cluster i. The optimal getting-there score is then μ_w (the mean of the optimal cluster), and the decision is

$$\arg \min_{i=1}^{U} (|u_i - \mu_w|) \tag{6}$$

i.e. the originating cell whose getting-there score is closest to the optimal cluster mean. The criterion thus involves a potential trade-off between similarity (which figures into the getting-there score) and density (i.e. size over variance): a high-similarity but atypical alignment may lose to a somewhat lower-similarity alignment if drawn from a higher-density cluster.

3 Experiment I: Output generation for multiple clouds

3.1 Hypotheses

We were interested whether (a) as a threshold matter, PEBLS' generated appropriate outputs for given target words, which could be resynthesized into reasonably natural sounding speech; and (b) PEBLS' outputs showed *generalization*, focussing on a pattern of allophonic intervocalic /k/ spirantization.

3.2 Method

We recorded ten tokens each of the first author saying (in randomized order) a set of (mostly) nonsense words, shown in Table 1. These words consisted of voiceless velar obstruents [k,x] flanked by vowels [i,e,æ] or

Pattern-conforming		Pattern-violating	
Intervocalic [x]	Non-intervocalic [k]	Intervocalic [k]	Non-intervocalic [x]
æxæ	æks	ækæ	æxs
æxe	ækt	æke	æxt
æxi	eks	æki	exs
exæ	ekt	ekæ	ext
exe	iks	eke	ixs
exi	ikt	eki	ixt
ixi	skæ	iki	sxæ
ixæ	ske	ikæ	sxe
ixe	ski	ike	sxi

Table 1: Word list, in IPA transcription

consonants [s,t], yielding nine word-types each of the velars in intervocalic and non-intervocalic position, or

[10] Hierarchical clustering is an algorithm for efficiently finding all possible similarity-based groupings of a set of data points. Specifically, we use agglomerative average-linkage clustering. See http://www.resample.com/xlminer/help/HClst/HClst_intro.htm. The average-linkage method has the advantage of clustering based on the same statistics (mean, variance, and cluster size), that we use in the confidence-sensitivity score. For a vector of U data points, the number of possible clusters is $2U$-1.

eighteen types each that conform to, or violate, a pattern of allophonic spirantization of /k/ in intervocalic position. Eighteen clouds were then constructed, consisting of

- all ten tokens of each of the pattern-conforming words, plus
- one token each of the pattern-violating words.

Each of the clouds thus reflects a strong, albeit variable pattern of [x] in intervocalic position and [k] in non-intervocalic position. We operationalize the notion of generalization as follows: if an input is selected which violates the spirantization pattern, and it is fed through PEBLS, with the corresponding cloud constructed as above, and the resulting output conforms to the pattern, notwithstanding the input, then PEBLS has generalized the pattern. If, however, the output violates the pattern, remaining faithful to the input, then PEBLS has not generalized the pattern.

The recordings were made with an Andrea NC7100 head-mounted USB microphone in a quiet office environment, directly to a computer hard-drive, at 41.5 KHz. The audio signals were preprocessed into frames of thirteen mel-frequency cepstral coefficients (MFCCs) using Slaney's (1998) Auditory Toolbox in Matlab.[11] Formant synthesis parameters were also computed from the audio signals, using Holmes' (1988) formant synthesis software, on the same timescale as the MFCCs. We could thus match each MFCC frame in the cloud with its formant synthesis parameters, and used the latter to resynthesize audio signals from PEBLS' outputs. The similarity drop-off parameter c (see eq. 2) was set to 30. In addition, a similarity threshold of 0.1 was imposed on the transition network, to speed up computation.

For each of the eighteen clouds, each of the nine pattern-violating tokens not included in the cloud was successively selected as input, for which PEBLS generated an output. For purposes of comparison, outputs were also generated for each of the ten pattern-conforming tokens, using a leave-one-out procedure in constructing the clouds. The resulting MFCCs were transformed into 40-point filter bank (quasi-spectrographic) representations by multiplying by the discrete cosine transform matrix (see Slaney 1998).

We measured mean energy during the medial consonant[12] of the outputs. The global minimum was rescaled to zero. High values reflect a spirantized output, whereas low values reflect stop closure.

3.3 Results and discussion

A few illustrative spectrograms (Figure 3b and d) show that PEBLS' outputs meet a threshold level of adequacy: they are appropriate outputs for the given target words. Resynthesized audio signals of these inputs and outputs are available at http://www.ualberta.ca/~kirchner/PEBLS. The outputs are natural-sounding speech.[13] They further show generalization of the intervocalic allophonic spirantization pattern.

More general results are shown in Figure 4. For every intervocalic cloud (the top row of boxes in Figure 4), the majority of outputs show fricative allophones of the medial velar consonant; whereas in non-intervocalic clouds, the outputs are predominantly stops. Broadly speaking, then, the results show generalization of the allophonic spirantization pattern instantiated in each cloud. In some words (/æ_æ/, /i_i/, /æ_s/, /e_t/, /s_æ/), the outputs uniformly adhere to the pattern; whereas in others, the outputs vary in their pattern conformity. In the less interesting case of selection of pattern-conforming inputs, the outputs (not shown here) uniformly conform to the pattern.

[11] Our choice of MFCCs is not crucial to the model. PEBLS can handle any sort of signal, provided it allows for reasonably distance measures. MFCCs are standard in ASR, and have generally been found to yield more useful distance results than e.g. spectrograms, due to the independence of the coefficients.

[12] The consonant boundaries were visually identified based on onset and offset of aperiodic energy in the case of fricatives (or abrupt shifts in the energy's frequency, if flanked by another fricative), and onset and offset of closure in the case of stops. Release bursts were not included in the stop measurements.

[13] This result is not as impressive as it may seem. With c set at 30, transitions to immediate successor frames have much higher values than other transitions; consequently, the optimal alignment turns out to be a straight line through a particular exemplar in the cloud. Not so trivially, though, PEBLS' confidence-sensitive criterion ensures that the particular exemplar chosen is a representative one.

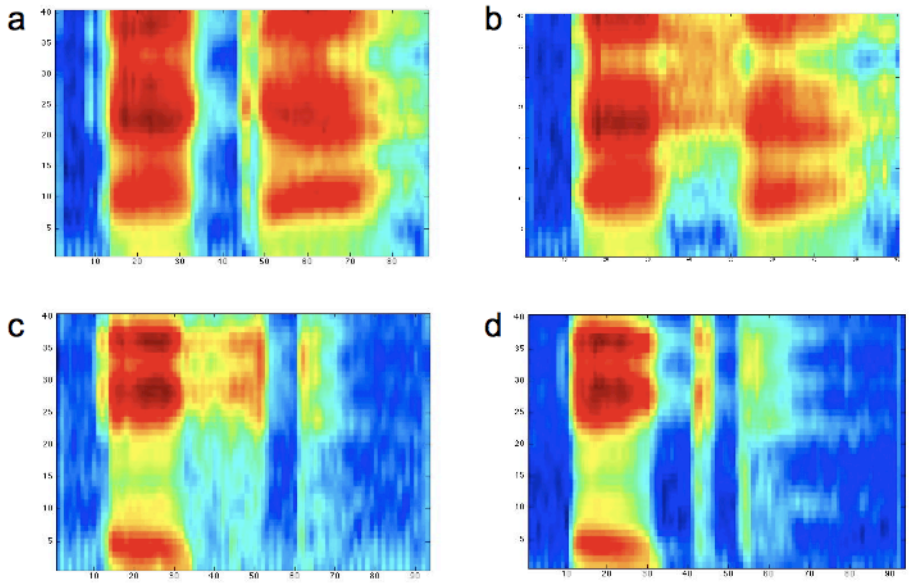

Figure 3: Filter bank spectrograms of input tokens of [ækæ] (a) and [ext] (c), and resulting PEBLS outputs (b and d, respectively). The outputs both show generalization of the patterns in their clouds: (b) by substituting a fricative interval for the input stop in intervocalic position, and (d) by substituting a stop closure interval in place of the input fricative in non-intervocalic position.

4 Experiment II: Iterative production

4.1 Hypothesis

Putting together the results of the Experiment I,

- When a pattern-conforming input is selected, the output uniformly conforms to the pattern.
- When a pattern-violating input is selected, the output conforms to the pattern in at least in some cases.

It should thus be the case that, as the system generates outputs iteratively, adding each new output to the cloud and then randomly selecting another exemplar from within the cloud as the new input, the word type should show a progression toward uniform adherence to the pattern, i.e. pattern entrenchment.

4.2 Method

Iteration with PEBLS' current input selection method, however, is problematic. Addition of self-produced outputs introduces new tokens in the cloud with particular frames, or even long sequences of frames, which may *exactly* match frames of the input. In PEBLS, these exact matches seem to trump confidence sensitivity. This technical problem can be overcome, though, by adding a modicum of normally distributed, variance-scaled random noise to the MFCCs of each output as it is appended to the cloud (indeed, variable deformation of outputs is a crucial part of Pierrehumbert's model, see fn. 3, though we acknowledge that our method is a crude way to implement this idea).

With this modification, we tested PEBLS' productions for /e_e/ (one of the still variable clouds in Experiment 1), starting with the original cloud plus the results of Experiment 1 (with both pattern-conforming and violating inputs), and then iterating with random selection of inputs. Results were measured as in Experiment 1.

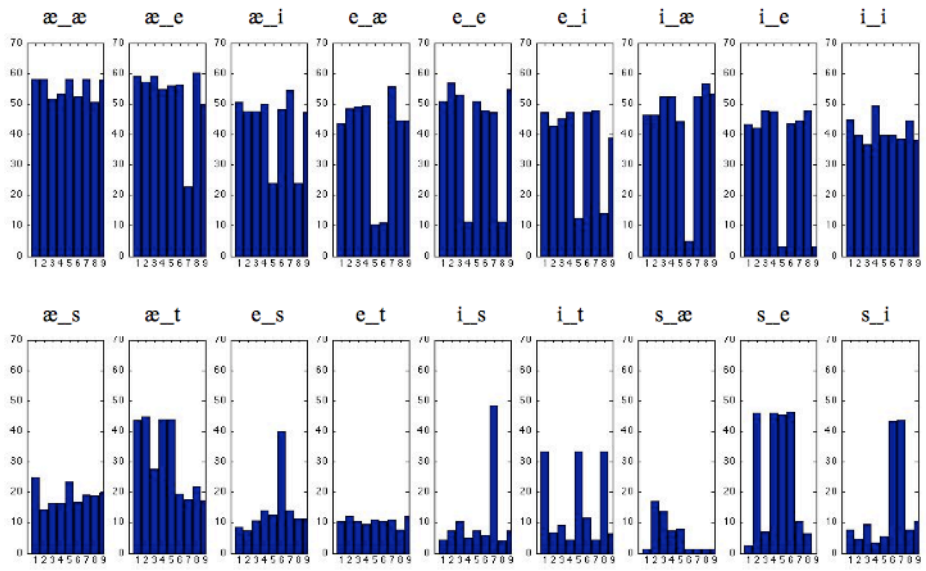

Figure 4: Mean energy of medial consonant in PEBLS' outputs. Each box shows results for a given word cloud. Within each box, the bars show results for each of 9 pattern-violating inputs. High values (>30) reflect fricatives, low values, stops.

4.3 Results and discussion

The hypothesis that PEBLS would model pattern entrenchment was confirmed. The results in (Figure 5)

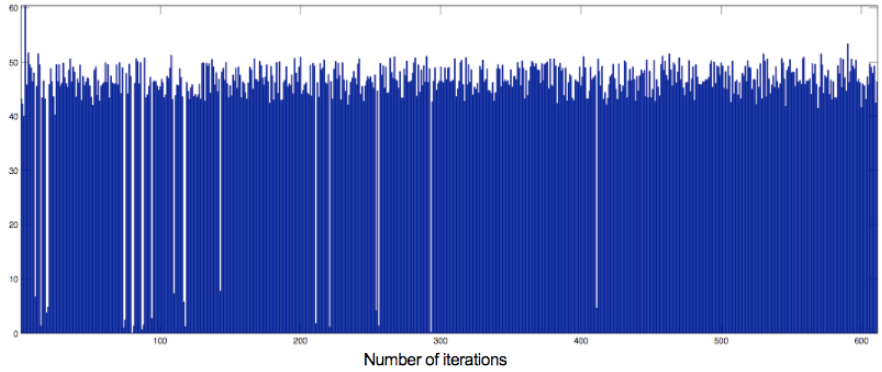

Figure 5: Mean energy of medial consonant in iterative productions of /e_e/.

show intermittent stop outputs which begin to taper off after about 100 iterations, ceasing altogether after the 411th iteration, and continuing with only fricative outputs for 200 iterations thereafter. We infer that, for this word, after these iterations, the spirantization allophone has become obligatory.

5 Conclusions

PEBLS provides a solution (though perhaps better solutions remain to be discovered) to the modelling problem which has hindered the development of ET, namely how to generate a composite output from of a set of unique, variable-length signals.

The notion of pattern entrenchment in exemplar dynamics has been a central claim of the ET literature.

It is the sum and substance of the ET story on where phonology comes from – how categorical, stable (i.e. quasi-symbolic) behaviour arises from numerical signals. PEBLS provides the first explicit model of this emergent effect with real speech signals.

The next step in this research programme is to show generalization outside the word class. That is, expanding the cloud to include all other exemplars in the corpus, we hope to show that a pattern of, e.g., intervocalic spirantization, strongly instantiated in most of the word types, can be extended by PEBLS even to outputs for word types with intervocalic contexts which initially contain only non-spirantized exemplars, i.e. lexical diffusion of the spirantization pattern. It should further be the case in PEBLS that this lexical diffusion occurs more readily to word types of low token frequency.

Finally, we note that, inasmuch as PEBLS computes a global optimization for the output, there exist deep parallels to Optimality Theory (or more directly, to Harmonic Grammar). The alignment described in section 2 is analogous to OT enforcement of correspondence constraints. A more elaborated version of PEBLS would include soft constraints reflecting phonetic pressures as part of the optimization criterion, e.g. an energy minimization imperative, analogous to Pierrehumbert's lenition bias, but also analogous to OT markedness constraints "grounded" in ease of articulation, cf. Kirchner 1998. In PEBLS then, as in OT, phonological patterns would arise from conflict between constraints favouring current patterns (including patterns within the word-class, as with IO-faithfulness), and constraints favouring phonetic naturalness. ET, however, computes over numeric signals rather than symbolic representations, thus providing a seamless phonetics-phonology interface.

References

[1] Bybee, J. (2001) *Phonology and language use.* Cambridge University Press.

[2] DeWachter, M. (2007) *Example based continuous speech recognition.* Doctoral dissertation, Katholieke Universiteit Leuven.

[3] Gahl, S. (2008) *Time* and *Thyme* are not homophones: The effect of lemma frequency on word durations in spontaneous speech. *Language* 84:3, 474-496.

[4] Gahl, S. & A. Yu (2006) Introduction to the special issue on exemplar-based models in linguistics. *The Linguistic Review*, 23:3, 213.

[5] Goldinger, S. D. (1996). Words and voices: Episodic traces in spoken word identification and recognition memory. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 22, 1166–1183.

[6] Goldinger, S. D. (2000) The role of perceptual episodes in lexical processing. In A. Cutler, J.M. McQueen, and R. Zondervan (eds.) *Proceedings of SWAP (Spoken Word Access Processes)*, Nijmegen, Max Planck Institute for Psycholinguistics. 155-159.

[7] Hofe, R. & R. K. Moore (2008) Towards an investigation of speech energetics using 'AnTon': an animatronic model of a human tongue and vocal tract. *Connection Science*, 20 (4): 319-336.

[8] Holmes, J. (1988) *Speech synthesis and processing.* Van Nostrand Reinhold.

[9] Johnson, K. (1997) Speech perception without speaker normalization. In K. Johnson & J. Mullennix (eds) *Talker variability in speech processing.* San Diego: Academic Press.

[10] Kirchner, R. (1998). *An effort-based approach to consonant lenition.* Doctoral dissertation, UCLA. (Published by Routledge, 2001).

[11] Kirchner, R. (1999). Preliminary thoughts on phonologization within an exemplar-based speech-processing system, *UCLA Working Papers in Linguistics* (Papers in Phonology 2), M. Gordon, ed., 1, 205-231.

[12] Moore R. K. (2007) Spoken language processing: piecing together the puzzle. *J. Speech Communication*, Special Issue on Bridging the Gap Between Human and Automatic Speech Processing, v. 49, 418-435.

[13] Moore, R.K. & V. Maier (2007) Preserving fine phonetic detail using episodic memory: automatic speech recognition with MINERVA2, *Proc. ICPhS,* Saarbruchen.

[14] Pierrehumbert, J. (2001) Exemplar dynamics: word frequency, lenition, and contrast. J. Bybee & P. Hopper (eds.), *Frequency effects and the emergence of linguistic structure*, 137-157, Amsterdam: John Benjamins.

[15] Pierrehumbert, J. (2002) Word-specific phonetics. In Carlos Gussenhoven & Natasha Warner (eds.), *Papers in Laboratory Phonology VII*, Berlin: Mouton de Gruyter. 101-140.

[16] Port. R. (2007) How are words stored in memory? Beyond phones and phonemes. *New Ideas in Psychology* 25, 143-170.

[17] Sankoff, D. & J. Kruskal (1983) *Time warps, string edits and macromolecules.* CSLI Publications.

[18] Slaney, M. (1998) Auditory Toolbox version 2, Technical Report #1998-010. Interval Research Corporation, http://cobweb.ecn.purdue.edu/~malcolm/interval/1998-010.

[19] Tucker, B. & A. Tremblay (2008) Effects of transitional probability and grammatical structure on the production of four-word sequences. Poster presented at Mental Lexicon 6 Conference, Banff, Oct. 2008.