Computational Phonology, class 2: Neighborhood models

Adam Albright

CreteLing 2022 — July 2022

SCAN ME

creteling2022.computational.phonology.party

Neighborhood models

A different approach to predicting wordlikeness

- Counts we have considered so far are all *combinatoric*
  - Probability of cooccurrence of combinations of sounds (local or non-local) → phonotactic probability
  - Intuition: nonce words sound more plausible if they contain well-supported combinations of sounds
  - E.g., [mɪp] is very likely because many words start with [#m], many words have [mɪ], many words have [ɪp], etc.
- A different type of metric: similarity to existing words
  - Intuition: nonce words sound more plausible if they sound similar to existing words
  - E.g., [mɪp] is very wordlike because it sounds like [nɪp], [mɪt], [mæp], etc.

Neighborhood density

- Neighborhood density: The number of words that differ from target word by one change (Greenberg and Jenkins, 1964; Coltheart et al., 1977; Luce, 1986)
  - Change one segment: *plan* $\sim$ *clan*, *plane*, *plaque*
  - Add one segment: *plan* $\sim$ *plant*
  - Delete one segment: *plan* $\sim$ *pan*
- A crude but widely used estimate of similarity to the lexicon

Greenberg and Jenkins (1964)

Greenberg, J. & J. Jenkins (1964) Studies in the psychological correlates of the sound system of American English. *Word* 20, pp. 157-177.

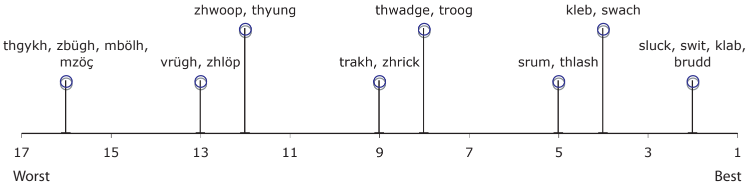
- One of the first attempts to collect systematic data about intermediate degrees of “Englishness”
- Hypothesis: novel words sound better, the closer they are to existing words

Greenberg and Jenkins (1964) (*cont.*)

- Closer = fewer differences, or fewer modifications you have to make to get to the nearest existing word
  - E.g., *clab* could be turned into *slab*, *crab*, *club*, *clam*, or *c_ab* by simply changing one sound
  - *cleb* requires two or more changes (*crab*, *clam*, *cleanse*, etc.)
- Number of similar-sounding words also makes a difference

Greenberg and Jenkins (1964) (*cont.*)

- Made up words predicted to fall along a scale of "Englishness"



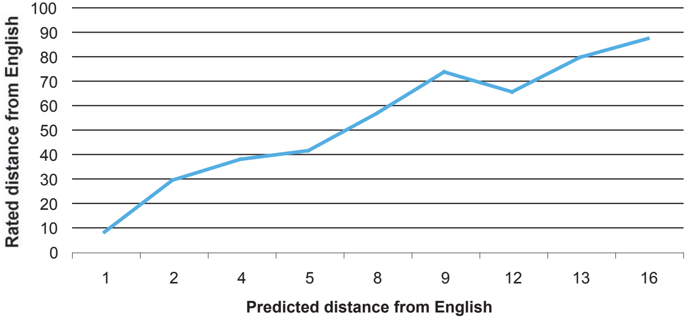
- Asked participants to rate words according to "how far they were from English" (low score = close, high score = far)

Greenberg and Jenkins (1964) (*cont.*)

- Tried various ways: scale of 0–10, arbitrary scales of the subjects’ devising, etc., but always got the same results…

Greenberg and Jenkins (1964) (*cont.*)

- Results: fairly strong relationship between calculated distance and perceived distance



Greenberg and Jenkins (1964) (*cont.*)

(Ratings for distance 5 are artificially low because the made-up word *thlash* was heard by some participants as *flash*, and rated as very English-like)

Crude measure: neighborhood density

- Greenberg and Jenkins: scale reflecting how easily transformed the word is into existing words
- Alternative approach: number of single-edit neighbors (Coltheart et al. 1977; Luce 1986)
  - Neighborhood density (NNB): number of existing words that differ by one phoneme addition, deletion, or substitution
  - Perhaps weighted by frequency, etc. (or similarity: more on this below)

Neighborhood density

- Colab interlude: a script that calculates number of neighbors in CELEX
- Predictions for Albright and Hayes (2003) words

The philosophy behind neighborhood density

Neighborhood effects in lexical/memory-based models

- Claim: reactions to nonce words are not the result of learned statistical knowledge calculated over the lexicon
- Rather, they are the result of an (implicit) attempt to classify the new word
  - Is it English, or not?
  - Items that are similar to many existing words receive lots of support from the lexicon
  - Items that are not similar to any existing words receive no support
- Essentially a by-product of activation of similar words

Exemplar models

Exemplar models

- Data is stored in detailed representations that encode many aspects of the experience
- Over time, a large number of exemplars build up
- New experiences or intents activate similar existing exemplars
- These influence how the word is perceived/categorized/identified/produced

"In order to understand the phenomenon which we usually designate as sound-change, we must get a clear idea of the … processes which operate in the production of groups of sound…In the first place, the movements of the organs of language…; secondly, the series of sensations by which these movements are necessarily accompanied—the 'motory sensation' (*bewegungsgefühl*); thirdly, the sensations of tone produced in the hearers…These sensations are, of course, not merely physiological processes, but psychological as well. Even after the physical excitement has passed away, these sensations leave a lasting psychical effect, viz., in the shape of memory-pictures (*erinnerungsbilder*)…[T]hese set up a connexion of cause and effect between the earlier and later production of the same combination of sounds. The memory-picture left behind by the sensation of the movement carried out before is that which renders possible the

In more modern terms

- Medin and Schaffer (1978) Context theory of classification learning

  "…a probe stimulus functions as a retrieval cue to access information stored with stimuli similar to the probe."

- Motivations for such an approach
  - Classification and prototypicality judgments often influenced by information that is seemingly "extraneous" information, from the point of the relevant rule (e.g., birds, prime numbers)
  - Judgments are gradient—not a binary classification

The Generalized Context Model (Nosofsky, 1986)

Background: Luce choice rule (stimulus identification)

$$\mathrm{P}(\mathrm{response}_j | \mathrm{stimulus}_i) = \frac{\mathrm{bias}_j \times \eta_{ij}}{\sum \mathrm{bias}_k \times \eta_{ik}}$$

- Probability of labeling stimulus$_i$ as item j depends on relative similarity of stimulus to j vs. to other items
- η_{ij} = perceptual similarity between i and j (possibly non-monotonic function of physical distance between i and j)
- $\eta_{ij} = \mathrm{e}^{-sensitivity \cdot distance_{ij}}$, or $\mathrm{e}^{-sensitivity \cdot {distance_{ij}}^2}$ (two options)
- Sensitivity = a parameter (found by fitting)
- Distance: in some space (e.g., string edit distance)

The Generalized Context Model (Nosofsky, 1986)

Context model of classification (=category identification)

$$\mathrm{P}(\mathrm{response}_J|\mathrm{stimulus}_i) = \frac{\mathrm{bias}_J \times \sum_{j \in J} \eta_{ij}}{\sum_K (\mathrm{bias}_K \times \sum_{k \in K} \eta_{ik})}$$

- Probability of classifying stimulus *i* as member of category J depends on relative similarity of *i* to members of category *J* vs. similarity to members of all other categories

Exemplar models

- Simple mechanism, easier to harness for linguistic phenomena than for others
  - Vowel identification (Johnson, and others)
  - Inflectional class (Nakisa et al., 1997; Albright and Hayes, 2003)
- In the case of gradient acceptability of novel words, however, it is not so hard to see how one might frame the problem
- Intuition: if a nonsense word has a lot of existing neighbors, it should sound better
  - The more neighbors, the better
  - The more similar those neighbors are, the better

Generalized Neighborhood Model

Bailey and Hahn (2001) Plausibility of novel word w_i

$$\propto \sum_{w_j \in lexicon} \text{weight}(\text{w}_j) \times \text{perceived sim}(\text{w}_i,\text{w}_j)$$

- As above, perceived similarity depends on physical distance
  - Perceived similarity $= \text{e}^{-\text{distance}}$ (or $\text{e}^{-\text{distance}^2}$, not considered)
- Denominator can be ignored here
  - What would it refer to?
- Weights of words related to frequency (non-linearly?)

Generalized Neighborhood Model (*cont.*)

- Bailey and Hahn propose quadratic fit: $\alpha \times \text{freq}^2 + \beta \text{freq} + \gamma$
- Allows for parabolic relations (monotonic or exponential increase/decrease, or greater/lesser influence of mid-range)

Generalized Neighborhood Model

$$\text{Score of } \mathrm{w}_i = \sum_{w_j \in lexicon} \text{weight}(\mathrm{w}_j) \times \text{perceived sim}(\mathrm{w}_i,\mathrm{w}_j)$$

$$= \sum_{w_j} \text{weight}(\mathrm{w}_j) \times \mathrm{e}^{-sensitivity \cdot d_{i,j}}$$

$$= \sum_{w_j} (\alpha \mathrm{f}_j^2 + \beta \mathrm{f}_j + \gamma) \times \mathrm{e}^{-sensitivity \cdot d_{i,j}}$$

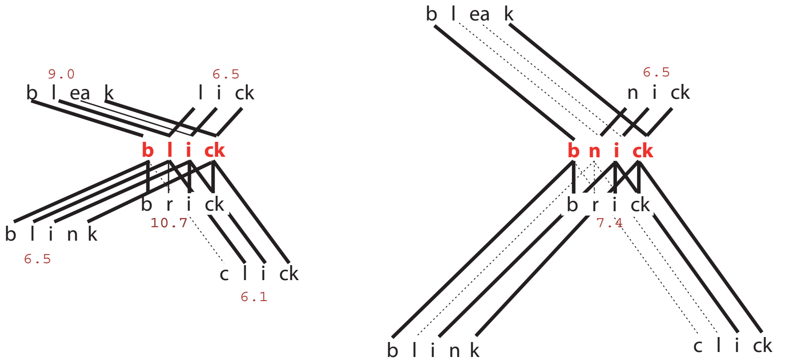
(see Bailey and Hahn, 2001, p. 572)

Generalized Neighborhood Model

How do we measure the perceived similarity of two words?

- Greenberg and Jenkins (and many subsequent researchers) define neighbors simply in terms of number of substitutions
  - Novel *clab*: *cab*, *crab*, *club*, *class* all equally close
- This seems too crude; not all substitutions have equal consequences for similarity of the resulting pairs
- We need a better way to calculate the similarity of two words

Similarity as overlap



- Mismatches depend on relative similarity of segments
  - ☞ E.g., *blick* ∼ *brick* more similar than *blick* ∼ *click*, because *l*∼*r* more similar than *b*∼*k*
- *Bnick* has less overlap with existing words, and mismatches are more serious; gets less support

Calculating similarity

- Goodness of matches between segments is assessed using natural-class based similarity metric of Frisch, Broe, and Pierrehumbert (2004)
- Degree of overlap between whole words is assessed by finding minimum string edit distance

Substitution costs based on similarity

Evidence that humans care about something like featural distance

- Greenberg and Jenkins (1964)
  - How similar are *ba* and *da*? *ba* and *pa*? *ba* and *ta*?
  - Results:
    - *b*∼*d*∼*g* rated very similar to one another
    - *b*∼*p*, *d*∼*t*, *g*∼*k* more similar than *b*∼*t*, *k*∼*p*, etc.
  - Conclusion: one feature change more similar than two feature changes

Substitution costs based on similarity (*cont.*)

- Hahn and Bailey (2005) What makes words sound similar
  - Choose more similar pair: XA (1 change) or XB (2 changes)
    - Onsets: *plimp*∼*flimp* or *plimp*∼*slimp*
    - Codas: *plip*∼*plib* or *plip*∼*plig*
  - Subjects generally go for the one-change pairs

Tversky (1977) Feature contrast model

Distance $=$ shared $-$ unshared features:

$$\mathrm{d}_{a,b} = \theta \cdot f(a \cap b) - \alpha \cdot f(a - b) - \beta \cdot f(b - a)$$

- $a \cap b =$ set of features shared by a and b
- $a - b =$ set of features of a but not b
- $f(a) =$ weighted function of feature contributions
- Parameters θ, α, β depend on task (more important to share at least some attributes? have no differences? etc.)
- Closely related to ratio mode (a la Frisch et al):

$$\mathrm{d}_{a,b} = \frac{f(a \cap b)}{f(a \cap b) + \alpha \cdot f(a - b) + \beta \cdot f(b - a)}$$

The task

- The task:
  - Find a set of feature weights *f*, and parameters θ, α, β, such that sim(*x*,*y*) values for all pairs *x*,*y* match observed values as closely as possible
  - Observed similarities: perceptual confusability, interaction in speech errors, avoidance in root restrictions, judged similarity, etc.
- Challenges
  - Many free parameters (need good search procedure)
  - What is the correct set of features?

Natural class-based similarity

Similarity of pairs of sounds: metric based on natural classes (Frisch et al., 2004)

- Try all possible combinations of feature values
  - All subsets of features, all combinations of values
- See which combinations result in distinct natural classes
  - Collect set of distinct classes
- Sounds are similar if they are grouped together in many natural classes
  - *m* and *n* are both voiced, both nasal, both sonorant, both stops, etc.

$$\text{Similarity} = \frac{\text{shared natural classes}}{\text{shared} + \text{unshared natural classes}}$$

Colab interlude

- Calculate the similarity of vowels in a Spanish-like 5 vowel system using natural-class based similarity
  - i, e, a, o, u
  - Features: [$\pm$high], [$\pm$low], [$\pm$back]
- Verify that the addition of [$\pm$round] does not affect that calculation
- Script to calculate similarity: SimilarityCalculator.pl

Natural classes are not enough

A possibly telling datum from Hahn and Bailey (2005)

- 75% preference for [lɪl]∼[vɪl] (manner + place change) over [lɪl]∼[zɪl] (manner and stridency change)
- Not predicted by Frisch et al model:

Comparison	Shared	Unshared	Similarity
l∼v	5	31	0.161
l∼z	11	32	0.344

- Suggests that stridency matters more than place
  - ☞ See also Coon and Gallagher (2007)

String edit distance

Now that we know the similarity of pairs of segments, we need to figure out how to line up words so that the most similarity segments are in correspondence with one another

- Basic idea: alignment can be calculated by figuring out the smallest number of changes needed to change one string to another
- If two strings share material, don't need to change it
- Unshared material must be deleted, inserted, or substituted

String alignment

Alignments = transformations: *spling* vs. *slink*

s	*p*	*l*	*ɪ*	*ŋ*	—
s	—	*l*	*ɪ*	*ŋ*	*k*

s	*p*	*l*	*ɪ*	*ŋ*	
s		*l*	*ɪ*	*ŋ*	*k*

1. Leave *s* unchanged
2. Delete *p*
3. Leave *l* unchanged
4. Leave *ɪ* unchanged
5. Leave *ŋ* unchanged
6. Insert *k*

- Aligned segments = substituted or unchanged
- Unaligned segments = inserted or deleted
- Goal: use segmental similarity to decide what should be aligned with what

String alignment

The task:

- Analyze correspondence as a sequence of substitutions, insertions, and deletions
- In practice, we usually want the *shortest* sequence of alignments/changes
- That is, the *optimal* alignment
- We'll start first by ignoring phonetic distance, and then incorporate it at the end

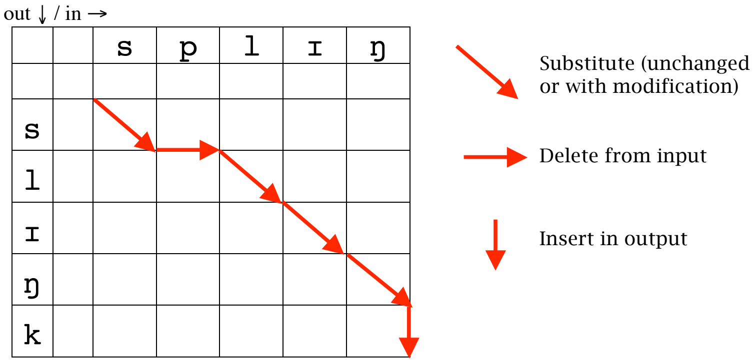
String alignment

Chart to calculate alignment

out ↓ / in →

		s	p	l	ɪ	ŋ
s						
l						
ɪ						
ŋ						
k						

Optimal path



String alignment

Calculating substitution and insertion/deletion costs

out ↓ / in →

		s	p	l	ɪ	ŋ
	0	0.5	1.0	1.5	2.0	2.5
s	0.5					
l	1.0					
ɪ	1.5					
ŋ	2.0					
k	2.5					

↘ Substitute (unchanged or with modification)

→ Delete from input

↓ Insert in output

subst cost	del cost
insert cost	

String alignment

Calculating substitution and insertion/deletion costs

out ↓ / in →

		s	p	l	ɪ	ŋ
	0	0.5	1.0	1.5	2.0	2.5
s	0.5	**0 .5** **.5**				
l	1.0					
ɪ	1.5					
ŋ	2.0					
k	2.5					

↘ Substitute (unchanged or with modification)

→ Delete from input

↓ Insert in output

subst cost	del cost
insert cost	

String alignment

Calculating substitution and insertion/deletion costs

out ↓ / in →

		s	p	l	ɪ	ŋ
	0	0.5	1.0	1.5	2.0	2.5
s	0.5	0 .5 .5	1 .5 .5			
l	1.0					
ɪ	1.5					
ŋ	2.0					
k	2.5					

↘ Substitute (unchanged or with modification)

→ Delete from input

↓ Insert in output

subst cost	del cost
insert cost	

String alignment

Calculating substitution and insertion/deletion costs

out ↓ / in →

		s	p	l	ɪ	ŋ
	0	0.5	1.0	1.5	2.0	2.5
s	0.5	0 .5 .5	1 .5 .5	1 .5 .5	1 .5 .5	1 .5 .5
l	1.0					
ɪ	1.5					
ŋ	2.0					
k	2.5					

↘ Substitute (unchanged or with modification)

→ Delete from input

↓ Insert in output

subst cost	del cost
insert cost	

String alignment

Calculating substitution and insertion/deletion costs

out ↓ / in →

		s	p	l	ɪ	ŋ
	0	0.5	1.0	1.5	2.0	2.5
s	0.5	0 .5 .5	1 .5 .5	1 .5 .5	1 .5 .5	1 .5 .5
l	1.0	1 .5 .5	1 .5 .5	0 .5 .5	1 .5 .5	1 .5 .5
ɪ	1.5					
ŋ	2.0					
k	2.5					

Substitute (unchanged or with modification)

Delete from input

Insert in output

subst cost	del cost
insert cost	

String alignment

Calculating substitution and insertion/deletion costs

out ↓ / in →

		s	p	l	ɪ	ŋ
	0	0.5	1.0	1.5	2.0	2.5
s	0.5	0 .5 .5	1 .5 .5	1 .5 .5	1 .5 .5	1 .5 .5
l	1.0	1 .5 .5	1 .5 .5	0 .5 .5	1 .5 .5	1 .5 .5
ɪ	1.5	1 .5 .5	1 .5 .5	1 .5 .5	0 .5 .5	1 .5 .5
ŋ	2.0	1 .5 .5	1 .5 .5	1 .5 .5	1 .5 .5	0 .5 .5
k	2.5	1 .5 .5	1 .5 .5	1 .5 .5	1 .5 .5	1 .5 .5

↘ Substitute (unchanged or with modification)

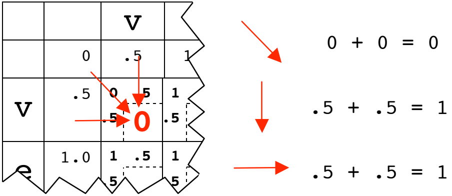
→ Delete from input

↓ Insert in output

subst cost	del cost
insert cost	

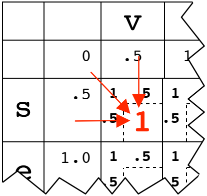
String alignment

Center value = minimum of three corners



String alignment

Center value = minimum of three corners



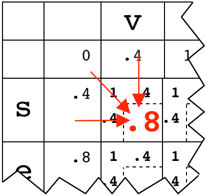
0 + 1 = 1

.5 + .5 = 1

.5 + .5 = 1

String alignment

Center value = minimum of three corners



0 + 1 = 1

.4 + .4 = .8

.4 + .4 = .8

String alignment

Center value = minimum of three corners

out ↓ / in →

		s	p	l	ɪ	ŋ
	0	0.5	1.0	1.5	2.0	2.5
s	0.5	0 .5 .5 0	1 .5 .5	1 .5 .5	1 .5 .5	1 .5 .5
l	1.0	1 .5 .5	1 .5 .5	0 .5 .5	1 .5 .5	1 .5 .5
ɪ	1.5	1 .5 .5	1 .5 .5	1 .5 .5	0 .5 .5	1 .5 .5
ŋ	2.0	1 .5 .5	1 .5 .5	1 .5 .5	1 .5 .5	0 .5 .5
k	2.5	1 .5 .5	1 .5 .5	1 .5 .5	1 .5 .5	1 .5 .5

↘ Substitute (unchanged or with modification)

→ Delete from input

↓ Insert in output

subst cost	del cost
insert cost	

Center value = minimum of three corners

out ↓ / in →

		s	p	l	ɪ	ŋ
	0	0.5	1.0	1.5	2.0	2.5
s	0.5	0 .5 .5 0	1 .5 .5 .5	1 .5 .5	1 .5 .5	1 .5 .5
l	1.0	1 .5 .5	1 .5 .5	0 .5 .5	1 .5 .5	1 .5 .5
ɪ	1.5	1 .5 .5	1 .5 .5	1 .5 .5	0 .5 .5	1 .5 .5
ŋ	2.0	1 .5 .5	1 .5 .5	1 .5 .5	1 .5 .5	0 .5 .5
k	2.5	1 .5 .5	1 .5 .5	1 .5 .5	1 .5 .5	1 .5 .5

↘ Substitute (unchanged or with modification)

→ Delete from input

↓ Insert in output

subst cost	del cost
insert cost	

String alignment

Center value = minimum of three corners

out ↓ / in →

		s	p	l	ɪ	ŋ
	0	0.5	1.0	1.5	2.0	2.5
s	0.5	0 .5 .5 0	1 .5 .5 .5	1 .5 .5 1.0	1 .5 .5 1.5	1 .5 .5 2.0
l	1.0	1 .5 .5	1 .5 .5	0 .5 .5	1 .5 .5	1 .5 .5
ɪ	1.5	1 .5 .5	1 .5 .5	1 .5 .5	0 .5 .5	1 .5 .5
ŋ	2.0	1 .5 .5	1 .5 .5	1 .5 .5	1 .5 .5	0 .5 .5
k	2.5	1 .5 .5	1 .5 .5	1 .5 .5	1 .5 .5	1 .5 .5

Substitute (unchanged or with modification)

Delete from input

Insert in output

subst cost	del cost
insert cost	

String alignment

Center value = minimum of three corners

out ↓ / in →

		s	p	l	ɪ	ŋ
	0	0.5	1.0	1.5	2.0	2.5
s	0.5	0 .5 .5 0	1 .5 .5 .5	1 .5 .5 1.0	1 .5 .5 1.5	1 .5 .5 2.0
l	1.0	1 .5 .5 .5	1 .5 .5 1.0	0 .5 .5 .5	1 .5 .5 1.0	1 .5 .5 1.5
ɪ	1.5	1 .5 .5 1.0	1 .5 .5 1.5	1 .5 .5 .5	0 .5 .5 .5	1 .5 .5 1.5
ŋ	2.0	1 .5 .5 1.5	1 .5 .5 2.0	1 .5 .5 1.5	1 .5 .5 1.0	0 .5 .5 .5
k	2.5	1 .5 .5 2.0	1 .5 .5 2.5	1 .5 .5 2.0	1 .5 .5 1.5	1 .5 .5 1.0

↘ Substitute (unchanged or with modification)

→ Delete from input

↓ Insert in output

subst cost	del cost
insert cost	

Paths with smallest costs

out ↓ / in →

		s	p	l	ɪ	ŋ
	0	0.5	1.0	1.5	2.0	2.5
s	0.5	0 0	.5 .5			
l	1.0			0 .5		
ɪ	1.5				0 .5	
ŋ	2.0					0 .5
k	2.5					.5 1.0

↘ Substitute (unchanged or with modification)

→ Delete from input

↓ Insert in output

subst cost	del cost
insert cost	

String alignment

Finally: phonetically sensible substitution costs

- Substitution cost(x,y) $=1-$ similarity(x,y)
- Choice of aligning vs. inserting/deleting determined by relative cost of indel—e.g., hypothetically with indel$=$.5:

Pair	Similarity	Sub cost	Alignment
b, *p*	.8	.2	Sub *b*$\rightarrow$*p*
b, *v*	.6	.4	Sub *b*$\rightarrow$*v*
b, *m*	.5	.5	Sub b$\rightarrow$m, or delete *b*/insert *m*
b, *f*	.3	.6	Delete *b*/insert *f*

String alignment

Sample of sensible substitution costs

out ↓ / in →

		s	p	l	ɪ	ŋ
	0	0.5	1.0	1.5	2.0	2.5
s	0.5	0 .5 / .5 0	.86 .5 / .5 .5	.84 .5 / .5 1.0	.98 .5 / .5 1.5	.87 .5 / .5 2.0
l	1.0	.84 .5 / .5 .5	.93 .5 / .5 .9	0 .5 / .5 .5	.89 .5 / .5 1.0	.68 .5 / .5 1.5
ɪ	1.5	.98 .5 / .5 1.0	.97 .5 / .5 1.4	.89 .5 / .5 .5	0 .5 / .5 .5	.90 .5 / .5 1.5
ŋ	2.0	.87 .5 / .5 1.5	.83 .5 / .5 1.8	.68 .5 / .5 1.5	.89 .5 / .5 1.0	0 .5 / .5 .5
k	2.5	.84 .5 / .5 2.0	.44 .5 / .5 1.9	.92 .5 / .5 2.0	.97 .5 / .5 1.5	.77 .5 / .5 1.0

↘ Substitute (unchanged or with modification)

→ Delete from input

↓ Insert in output

subst cost	del cost
insert cost	

(Doesn't change anything in this case)

Colab interlude

Running and testing an implementation of the GNM

- Calculate similarity values for English inventory
- See alignment in action
- Use GNM to derive predictions for Albright and Hayes (2003) words

Testing the GNM

Bailey & Hahn (2001)

- Motivating suspicion: many purported effects of phonotactic probability may actually be neighborhood effects
  - In point of fact, NNB and n-gram probability are often highly correlated (why?)
  - Most studies test for effects of one or the other, but don't directly compare the two
  - Previous tests for neighborhood effects hampered by crude definition of neighborhoods
- Strategy
  - Collect ratings for a huge bunch of nonce words
  - Compare predictive power of GNM and n-gram models

Nonce word experiment

Bailey & Hahn (2001)

- Made up 22 *isolates*: 2 edits from any existing word
- Added 250 *near-misses*: modified isolates to create words that had at least one existing neighbor
  - E.g., [dɹɒlf] $\rightarrow$ [dɒlf], [drɪlf], [dɹɒf], etc.
- Collected "wordlikeness" judgments
  - "How typical-sounding is ___?"
  - One orthographic task, one auditory task

Strategy for teasing apart neighbors from sequences

Multiple regression modeling

- Start by considering correlation of many different potential predictors—e.g.,
  - Phonological phone, bigram, trigram transitional probability
  - Orthographic letter, bigram, trigram transitional probability
  - Onset-rhyme transitional probability
  - Number of neighbors (traditional metric)
  - GNM score
- Then considered factors in combination, checking for ability of successive factors to improve on simpler models

Bailey & Hahn results

Individual factors

- Log joint transitional probability (bigram): $r^2=.19$ ($r\approx.44$)
- GNM w/distances based on string alignment: $r^2=.22$ ($r\approx.47$)
    - Slightly better, though care needed since model produces many nearly identical values
- Other metrics not nearly as good (see Table 2, p. 577)

Combining factors

- Sequence probabilities alone: $r^2 = .23$
- Adding traditional NNB: no improvement
- Adding GNM scores: $r^2 = .38$
  - Non-monotonic frequency effect: mid frequencies contribute most
  - When token frequency removed, $r^2 = .36$
- Substantially overlapping predictions

Unique contribution of phonotactics	.09
Unique contribution of GNM	.15
Overlapping predictions	.14

What do we conclude from all this?

- Significant effects of both neighbor activation and also knowledge of probability of sequences (independent of particular words)
  - Contrary to Bailey and Hahn's expectations, neither reducible to the other
- Bailey and Hahn: majority of effect is neighbors, while contribution of phonotactics is relatively smaller
  - Assessment depends on how we credit ambiguous overlapping portion of variance, however
- Role of token frequency also claimed to be significant
  - Effect is extremely small, however, and non-monotonic
  - Response curve is unlike how activation works in lexical access
  - Not clear that this supports the the underlying premise of the exemplar model

Why neighborhoods are unlikely to be sufficient

Why we might have predicted this result from the start

- Certainly, lexical access is a real effect
  - May depend on task, but no reason to preclude possibility of lexical neighbor effects
  - So, not too surprising if we see a neighborhood effect
- However, not all acceptability judgments can be reduced to number of neighbors
  - E.g., [fruː] vs. [sruː]
  - [sruː] has more neighbors, because more licit *sC*-clusters
  - Yet it also contains a phonotactic violation: **sr*

Why neighborhoods are unlikely to be sufficient (*cont.*)

- Undeniable role for sequence probability (whether learned statistically or based on markedness preferences)

Why neighborhoods are unlikely to be sufficient (*cont.*)

- Bailey and Hahn don’t actually test illegal sequences
  - And few studies bother to include *truly* unacceptable words, where sequential models can guarantee consistently low scores

Possible refinements

- Variable indel
  - Segment identity: insertion cost(ə) $<$ insertion cost(*a*)
  - Context: insertion cost(ə)/T___ R $<$ insertion cost(ə)/S___ T
  - Medial vs. peripheral indel (contiguity)
- Incorporating positional/prosodic structure

Relative cost of insertion and deletion

Gentner and Markman (1995)

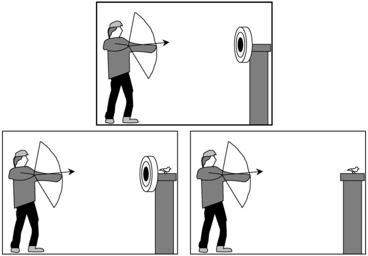


Fig. 5. Insertion and substitution in a scene depicting predicate-argument relations between objects (after Markman & Gentner, 1996, Fig. 4, p. 242).

- Subjects overwhelmingly (88%) prefer inserting bird (left) over swapping target out for bird (right)
- Claim: judgment driven by similarity of aligned elements; little penalty for additional

Relative cost of insertion and deletion

Hahn and Bailey (2005): tested phonological equivalent

- Compared insertion ([fɑːʒ]∼[flɑːʒ], [zɪtʃ]∼[zɪntʃ]) with replacement ([fɑːʒ]∼[lɑːʒ], [zɪtʃ]∼[zɪn])
- Onsets: no preference
- Codas: insertions judged more similar

What this suggests for the model

- Alignments should have relatively low indel cost
  - [sklæm] should get lots of support from *clam*, less from *scram*
- But not *too* small?
  - Bailey & Hahn get best value from GNM with indel $>$ .6

The role of structure

Another robust finding: the importance of position (Nelson and Nelson 1970, Bailey and Hahn 2005)

- Already seen above: sim([fɑːʒ],[flɑːʒ]) $=$ sim([fɑːʒ],[lɑːʒ]), but sim([zɪtʃ], [zɪntʃ]) $>$ sim([zɪtʃ],[zɪn])
- More directly: sim([ʃæʃ], [fæʃ]) somewhat greater than sim([ʃæʃ], [ʃæf])
- Two lines of attack
  - Enhanced perceptibility in coda position enhances differences—VERY UNLIKELY!!! (contradicts results of perceptual studies, and typological facts)
  - Greater psychological weight to VC than CV (‘rhyming’)

Another interesting effect of structure: contrast

Goldstone et al. (1991) Relational similarity and non-independence of features in similarity judgments

- Which is more like △ △ ?

A ○ ○

B □ □

Another interesting effect of structure: contrast

Goldstone et al. (1991) Relational similarity and non-independence of features in similarity judgments

- Which is more like △ △ ?

  A ○ ○

  B □ □

- Which is more like △ △ □?

  A ○ ○ □

  B □ □ □

(What kinds of predictions does this make for comparisons of words?)

Taking stock

Two approaches to modeling phonotactic acceptability

- N-grams: decompose string into subparts
- Exemplar models: align string to stored examples
- Neither one looks all that much like phonological grammars that you may be familiar with!
  - No phonological features, syllable structure, markedness constraints
- Baseline models: help show where further assumptions improve match to humans

Finding likely areas for improvement

- Weaknesses of N-gram models?

Finding likely areas for improvement

- Weaknesses of N-gram models?
  - Local vs. non-local dependencies
  - Differentiating unattested sequences: *bnick* [bnɪk] vs. *bzick* [bzɪk] vs. *nbick* [nbɪk] (all zero)

Finding likely areas for improvement

- Weaknesses of N-gram models?
  - Local vs. non-local dependencies
  - Differentiating unattested sequences: *bnick* [bnɪk] vs. *bzick* [bzɪk] vs. *nbick* [nbɪk] (all zero)
- Weaknesses of GNM?

Finding likely areas for improvement

- Weaknesses of N-gram models?
  - Local vs. non-local dependencies
  - Differentiating unattested sequences: *bnick* [bnɪk] vs. *bzick* [bzɪk] vs. *nbick* [nbɪk] (all zero)
- Weaknesses of GNM?
  - Illegal strings may align well with many existing words
  - Differentiating unattested sequences: *bnick* [bnɪk] vs. *bzick* [bzɪk] vs. *nbick* [nbɪk] (depend on existing words)

References

Albright, Adam and Bruce Hayes. 2003. Rules vs. analogy in English past tenses: A computational/experimental study. Cognition 90, 119–161.

Bailey, Todd M. and Ulrike Hahn. 2001. Determinants of wordlikeness: Phonotactics or lexical neighborhoods? Journal of Memory and Language 44, 568–591.

Coltheart, Max; Eileen Davelaar; Jon Torfi Jonasson; and Derek Besner. 1977. Access to the internal lexicon. In Attention and performance, ed. by Stan Dornic, volume 6, 535–555. Hillsdale, NJ: Erlbaum.

References (*cont.*)

Frisch, Stefan A.; Janet B. Pierrehumbert; and Michael B. Broe. 2004. Similarity avoidance and the OCP. Natural Language & Linguistic Theory 22(1), 179–228. URL http://www.springerlink.com/openurl.asp?genre=article&id=doi:10.1023/B:NALA.0000005557.78535.3c.

Gentner, D. and A. B. Markman. 1995. Similarity is like analogy: Structural alignment in comparison. In Similarity in language, thought and perception, ed. by C. Cacciari, 111–147. Brussels: BREPOLS.

Greenberg, Joseph H. and James J. Jenkins. 1964. Studies in the psychological correlates of the sound system of American English. Word 20, 157–177.

References (*cont.*)

HAHN, ULRIKE and TODD M. BAILEY. 2005. What makes words sound similar? Cognition 97, 227–267.

LUCE, PAUL A. 1986. Neighborhoods of words in the mental lexicon. Technical report, Speech Research Laboratory, Department of Psychology, Indiana University.

NAKISA, RAMIN CHARLES; KIM PLUNKETT; and ULRIKE HAHN. 1997. A Cross-Linguistic Comparison of Single and Dual-Route Models of Inflectional Morphology. In Cognitive Models of Language Acquisition, ed. by P. Broeder and J. Murre. Cambridge, MA: MIT Press.

References (*cont.*)

NOSOFSKY, ROBERT M. 1986. Attention, similarity, and the identification-categorization relationship. Journal of Experimental Psychology: General 115, 39–57. http://www.cogs.indiana.edu/nosofsky/pubs/1986_rmn_jep-g_attention.pdf.

PAUL, HERMANN. 1920. Prinzipien der Sprachgeschichte. Halle: Niemeyer, 5th edition.

Transitional probability

Saffran, Aslin, and Newport (1996) Statistical Learning by 8-Month-Old Infants. *Science* 274, pp. 1926-1928.

- Infants trained on text of 3-syllable “words”
  - E.g., *pabiku*, *tibudo*, *golatu*, and *daropi*

Transitional probability

Saffran, Aslin, and Newport (1996) Statistical Learning by 8-Month-Old Infants. *Science* 274, pp. 1926-1928.

- Infants trained on text of 3-syllable “words”
  - E.g., *pabiku*, *tibudo*, *golatu*, and *daropi*

Example
pabikugolatupabikudaropitibudogolatudaropipabikutibudopabiku
ropigolatupabikugolatutibudodaropitibudo…

Transitional probability

Saffran, Aslin, and Newport (1996) Statistical Learning by 8-Month-Old Infants. *Science* 274, pp. 1926-1928.

- Infants trained on text of 3-syllable “words”
  - E.g., *pabiku*, *tibudo*, *golatu*, and *daropi*

Example
pabikugolatupabikudaropitibudogolatudaropipabikutibudopabiku
ropigolatupabikugolatutibudodaropitibudo…

- Items controlled so no syllable was used in more than one word
- Non-final syllables always followed by the same syllable (*pa* can only come before *bi*, etc.)
- Final syllables could be followed by any other word (*ku* could be followed by *ti*, *go*, or *da*)

Transitional probability

Saffran, Aslin, and Newport (1996) Statistical Learning by 8-Month-Old Infants. *Science* 274, pp. 1926-1928.

- Infants trained on text of 3-syllable “words”
  - E.g., *pabiku*, *tibudo*, *golatu*, and *daropi*

Example

pabikugolatupabikudaropitibudogolatudaropipabikutibudopabiku
ropigolatupabikugolatutibudodaropitibudo…

- Items controlled so no syllable was used in more than one word
- Non-final syllables always followed by the same syllable (*pa* can only come before *bi*, etc.)
- Final syllables could be followed by any other word (*ku* could be followed by *ti*, *go*, or *da*)

Transitional probability

Saffran, Aslin, and Newport (1996) Statistical Learning by 8-Month-Old Infants. *Science* 274, pp. 1926-1928.

- Infants trained on text of 3-syllable “words”
  - E.g., *pabiku*, *tibudo*, *golatu*, and *daropi*

Example

pabikugolatupabikudaropitibudogolatudaropipabikutibudopabiku
ropigolatupabikugolatutibudodaropitibudo…

- Items controlled so no syllable was used in more than one word
- Non-final syllables always followed by the same syllable (*pa* can only come before *bi*, etc.)
- Final syllables could be followed by any other word (*ku* could be followed by *ti*, *go*, or *da*)

Transitional bigram probability

Example

Words: *pabiku*, *tibudo*, *golatu*, and *daropi*

pabikugolatupabikudaropitibudogolatudaropipabikutibudopabiku
ropigolatupabikugolatutibudodaropitibudo…

- Results: after two minutes of exposure, infants can reliably distinguish “words” from “part-words”
  - “Words”: strings of syls that always occur together, like *pabiku*
  - “Part-words”: strings of syls that occur together, but only occasionally (e.g., *kudaro*)
  - Distinguish: prefer to look longer at a speaker playing part-words
- Claim: in order to do this, they must be able to track (somehow) the sequencing of syllable combinations like *pa* and *bi*

Evidence that humans care about transitional probability

Aslin, Saffran & Newport (1998) Computation of conditional probability statistics by 8-month-old infants. *Psych Sci* 9, pp. 321–324.

- Same set-up as before: four words (*pabiku*, *tibudo*, *golatu*, and *daropi*), no repeated syllables
- Small change: two words occurred twice as often as the other two in the training
  - How does this change the syllable bigram probabilities? how about the transitional probabilities?
- Test: tested “part-words” vs. low frequency “words”
- Result: infants still distinguished between the two
  - As before, preferred to look longer at part-words

☞ What statistical differences might they (in principle) have been responding to?