Neural network models for language acquisition

Micha Elsner
OSU Linguistics

Thanks to:

Cory Shain (OSU)

Shain and Elsner 2019 "Measuring the perceptual availability of phonological features during language acquisition using unsupervised binary stochastic autoencoders"
Elsner and Shain 2017 "Speech segmentation with a neural encoder model of working memory"

Kasia Hitczenko and Naomi Feldman (UMD)
Stephanie Antetomaso (OSU)

Hitczenko et al 2018 "How to use context to disambiguate overlapping categories: The test case of Japanese vowel length"
Antetomaso et al 2017 "Modeling phonetic category learning from natural acoustic data"

and Herman Kamper (Stellenbosch), Aren Jansen (Google), Sharon Goldwater (Edinburgh)

Into language acquisition at OSU?

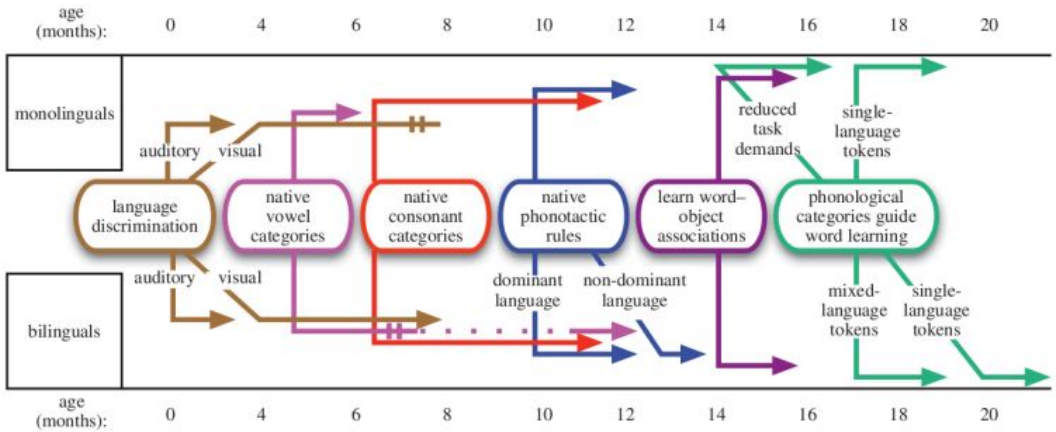
Lacqueys interdisciplinary reading group

- 9am every Wednesday
- Present your work! Read papers! Eat little oranges!

The Buckeye Language Network (BLN)

- Umbrella group for language at OSU
- Can nominate you for an award, feature your work in a flash talk series, connect you with mentors

Phonetic and phonological abilities in young infants … suggest *rapid* and *powerful* learning mechanism



(Werker, Byers-Heinlein and Fennell 09)

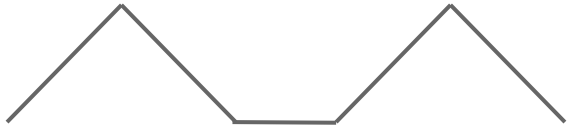
Experiments tell us *what* is learned, not always *how*

A classic study by Maye et al (2002) showed that infants could use **statistical properties** of speech to form protocategories

A process called **distributional learning**

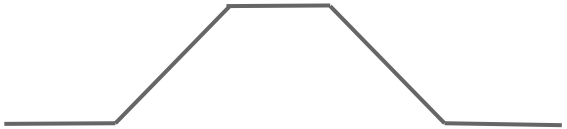
Maye teaches infants minilanguages

Group 1 hears two categories



more like ta … more like da

Group 2 hears one category



more like ta … more like da

After a few minutes...

Test perception of the contrast

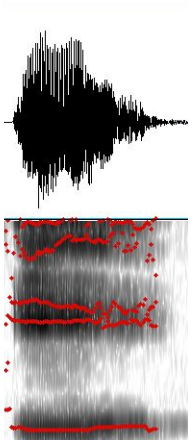
Infants in group 1 detect the change better!

Can we simulate this kind of learning?

We can test hypotheses about how it works by building models

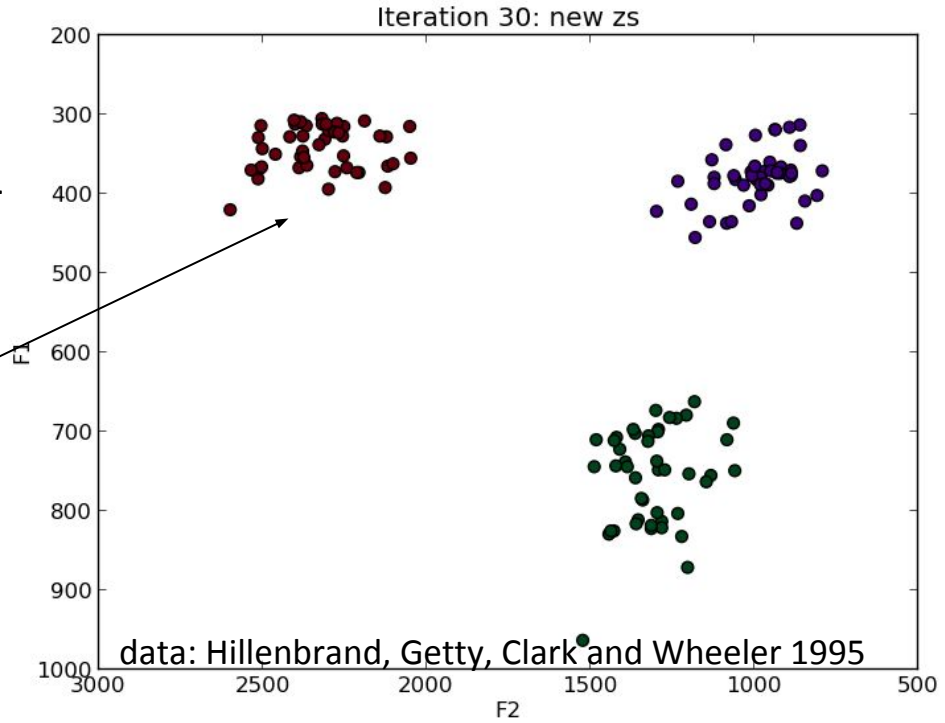
We'll use these models to learn from real data
And see where they succeed and fail

Computer models to the rescue?



i

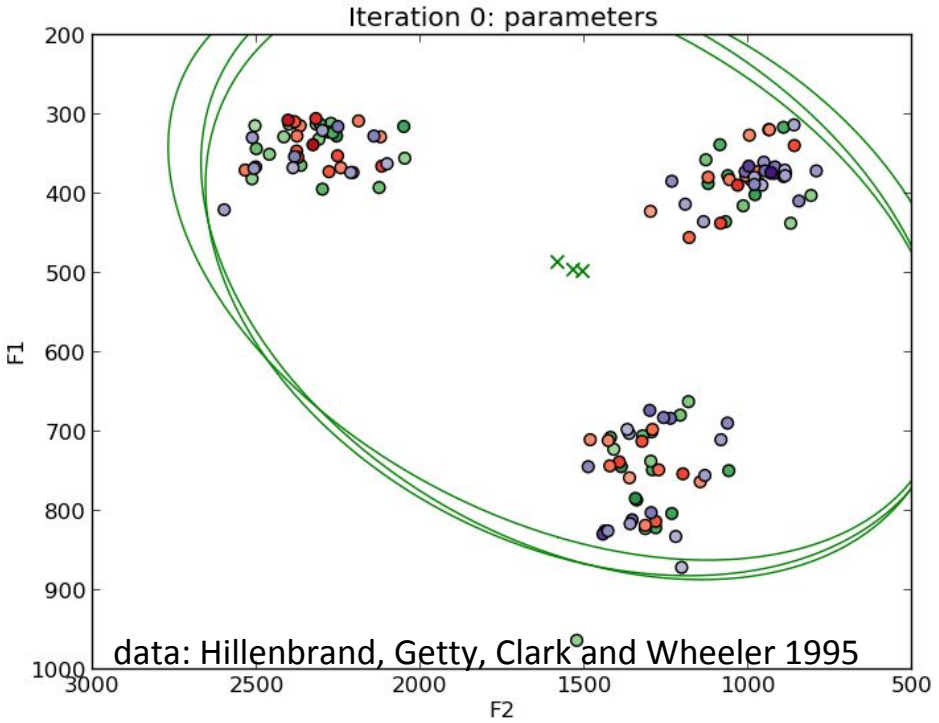
formant 1



formant 2

approach of de Boer and Kuhl 2003,
see also Vallabha et al 2007

Start with random category system



formant 1

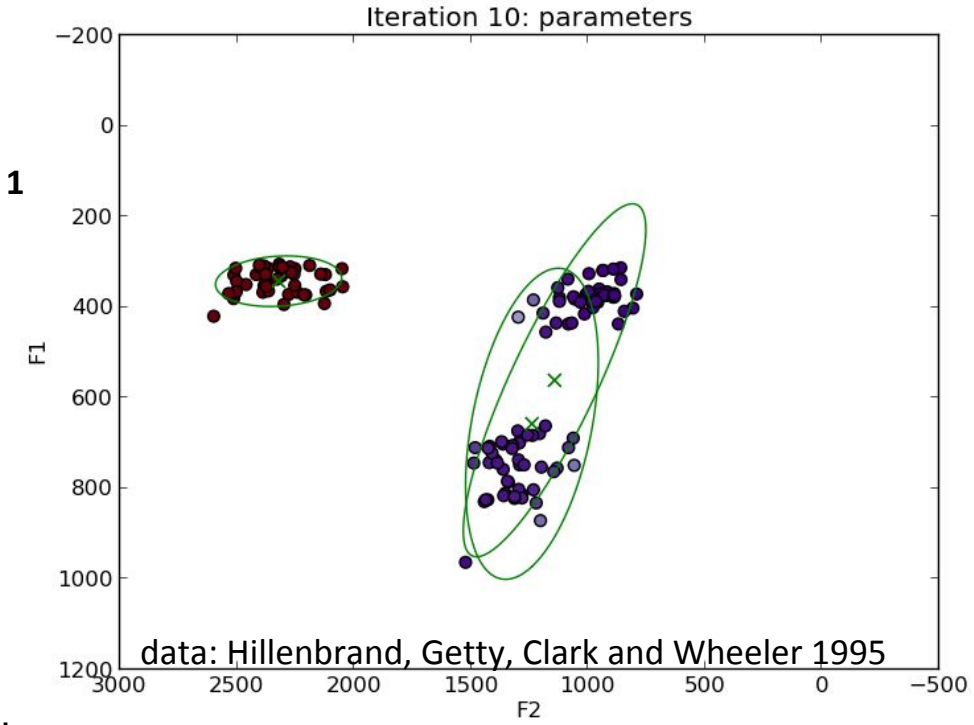
formant 2

approach of de Boer and Kuhl 2003,
see also Vallabha et al 2007

Slowly improve the representations

Categorize each sound based on the current category system

Then update the category representations

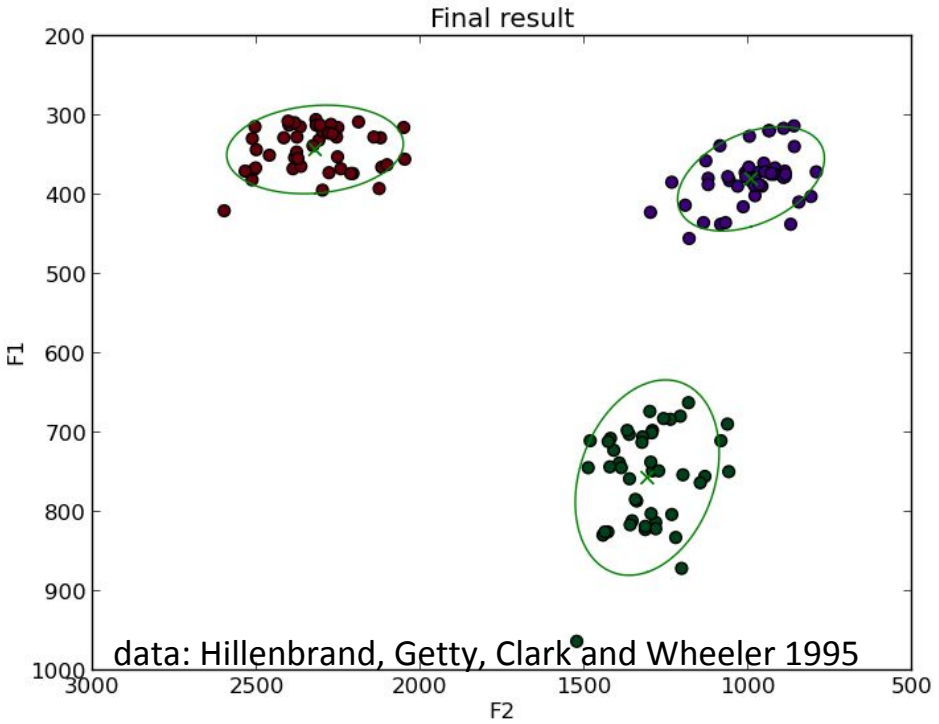


formant 1

formant 2

approach of de Boer and Kuhl 2003, see also Vallabha et al 2007

Eventual success



formant 1

formant 2

approach of de Boer and Kuhl 2003,
see also Vallabha et al 2007

This works fine for toy systems

Works well when:

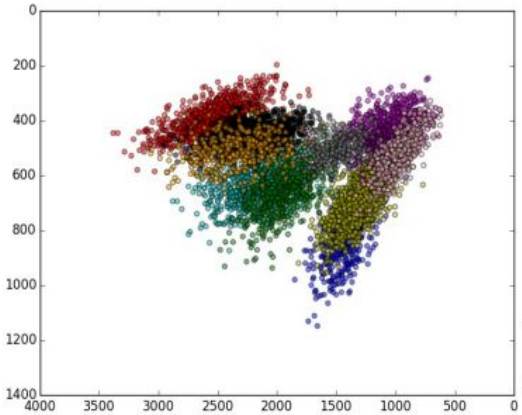
Few relevant dimensions

Categories are allowed to overlap, but not too much

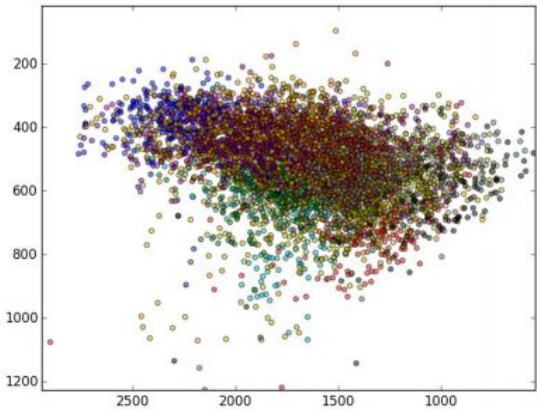
Sound categories have simple elliptical shape

But real life is highly variable

American English from the lab:
(Hillenbrand et al 1995)

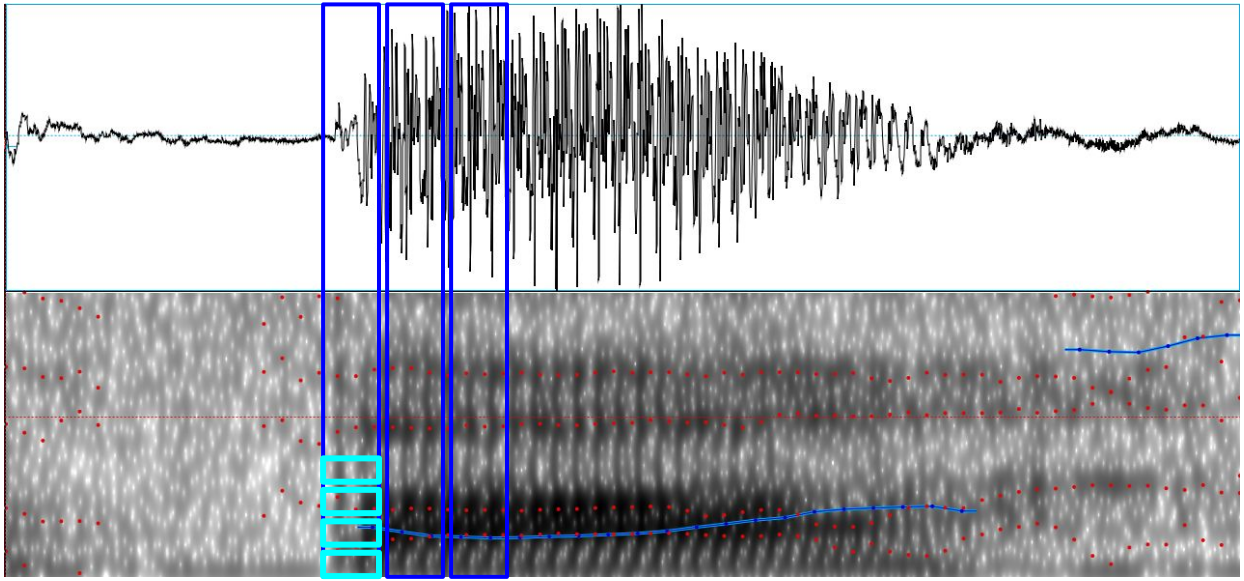


American English from the wild:
Buckeye speech corpus (Pitt et al 2005)



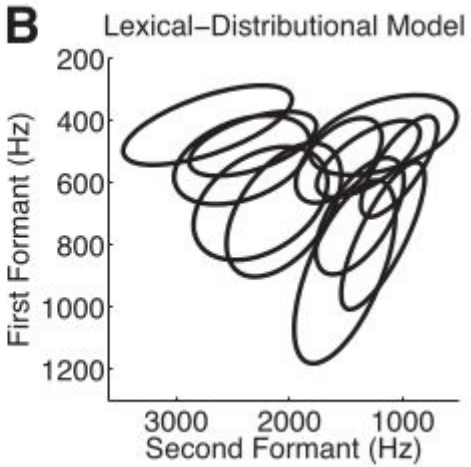
And very high-dimensional

With highly correlated features!

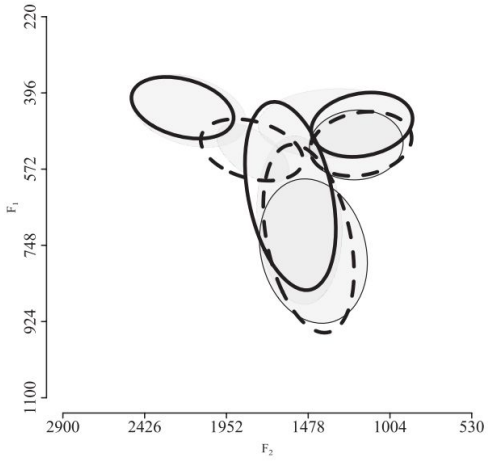


Proposed solution: more features

Adding *word-level* information helps to compensate for the overlap between English vowel categories
(Feldman et al 2013)



Adding *consonant context* compensates for coarticulatory effects in Inuktitut
(Dillon et al 2013)



But in the end, we concluded something else was necessary

Some of our studies:

Japanese vowel length contrast can't be learned from conversation, and normalizing the data doesn't help (Hitczenko et al 2018)

Real English vowels overlap too much to learn (Antetomaso et al 2017)

Learning the vocabulary alongside the categories does not disambiguate all the English vowel contrasts (Elsner et al 2016)

Piling on covariates causes too many problems

Dependence: All the variables are correlated, in complicated and hard-to-model ways

There are too many correlations to learn each one separately

Relevance: Not all predictable variation is phonologically meaningful

I don’t sound like you, but that doesn’t mean I should have my own, speaker-specific phonemes

“Interaction terms”

Typically, we deal with potentially-correlated features using interaction terms

But these are like tribbles: they grow exponentially!

from wikimedia commons

Too much data is required
And we’ll probably learn some spurious effects

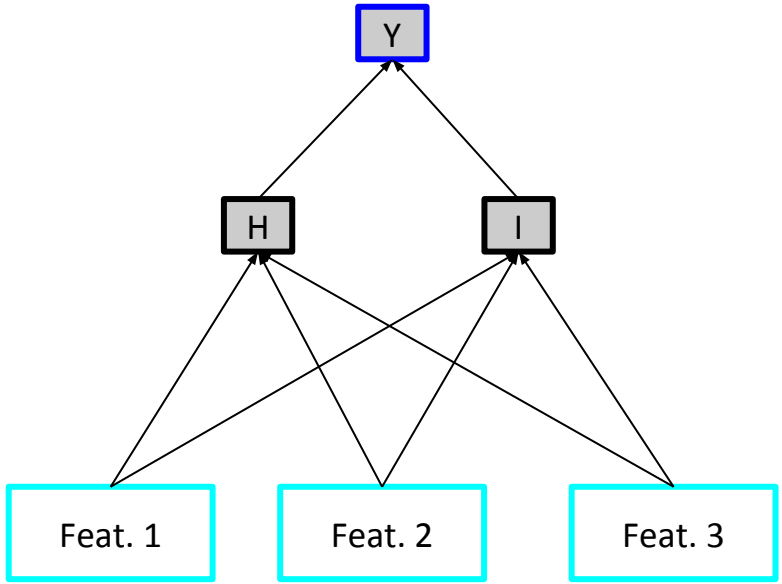
Limit the number of interactions

A neural network is given a fixed number of intermediate variables…

These can summarize any combination of features 1, 2 and 3

These represent learned *abstractions* which summarize intermediate conclusions from the low-level cues

The model picks its own interaction terms

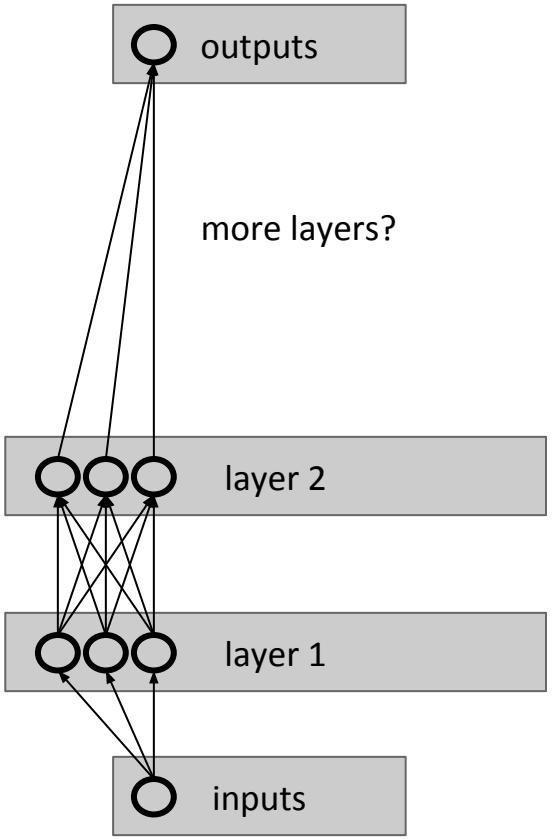


Multilayer network

More complicated network topologies are common…

Take advantage of structure in the data:

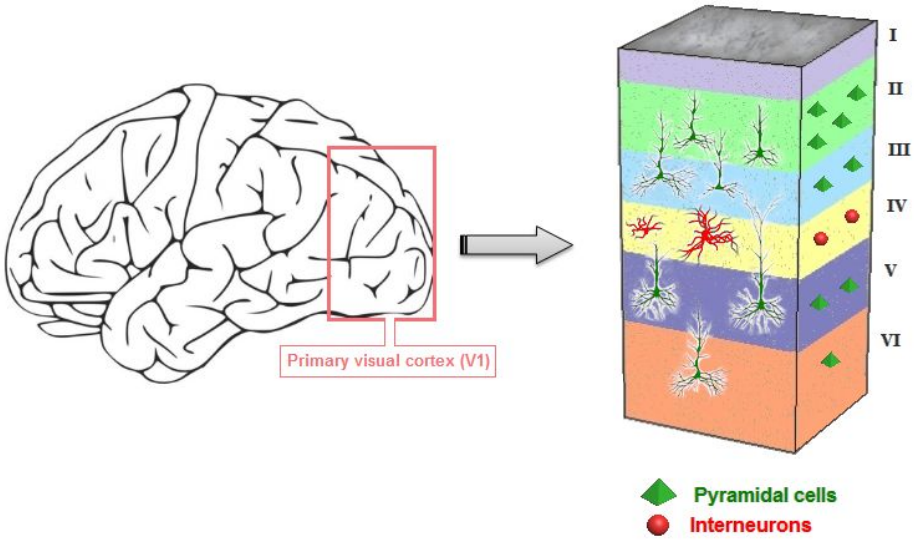
- Temporal (speech ms. by ms.)
- Spatial (nearby frequency bands)
- Source of data (my voice vs. yours)



Biological analogies?

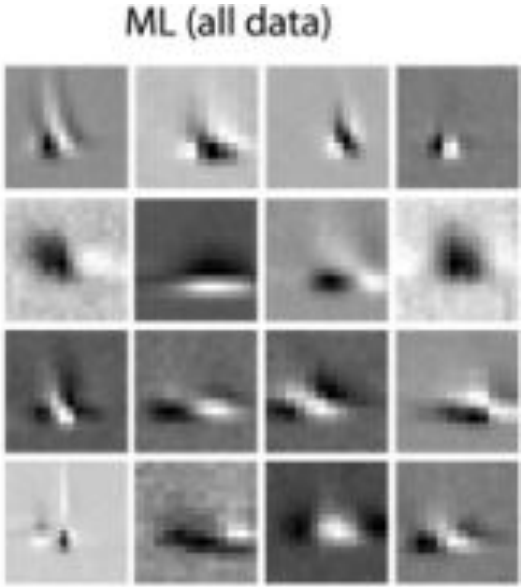
The visual cortex processes input in layers…

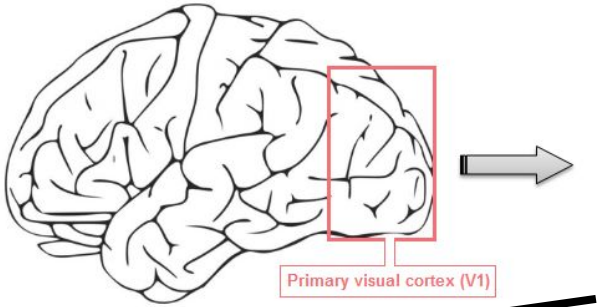
Lower layers detect “low-level” features; higher layers are more abstract

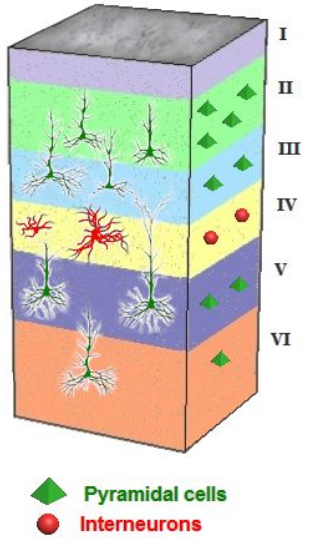


Adaptation and Neuronal Network in Visual Cortex
Lyes Bachatene, Vishal Bharmauria and Stéphane Molotchnikoff

Biological analogies?







Receptive Field Inference with Localized Priors
Mijung Park and Jonathan Pillow

Adaptation and Neuronal Network in Visual Cortex
Lyes Bachatene, Vishal Bharmauria and Stéphane Molotchnikoff

Model design

We want to use networks to model human language learning...

We'll try to deal with **dependence** and **relevance** issues by tuning:

The **learning objective**: what the network predicts

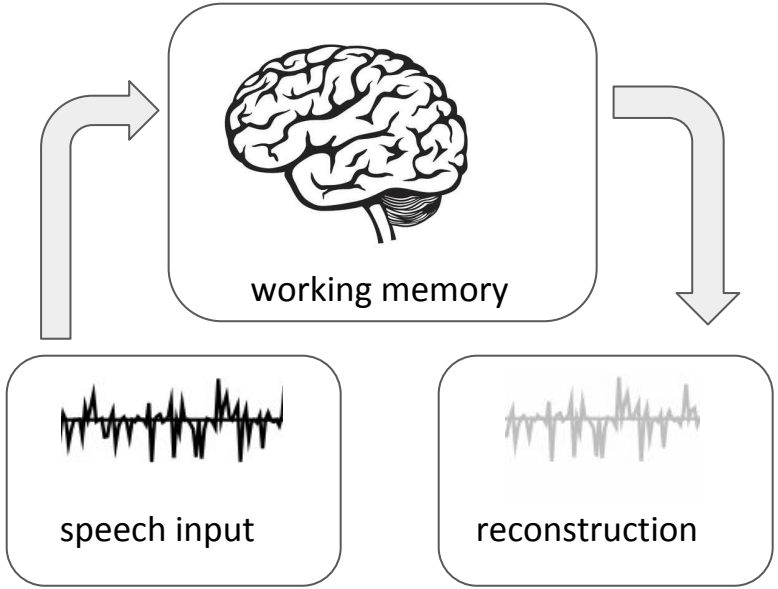
The **inputs**: what features we give it

The **internal structure** of the intermediate layers

Learning as memorization

People have limited working memory, especially for fine phonological details

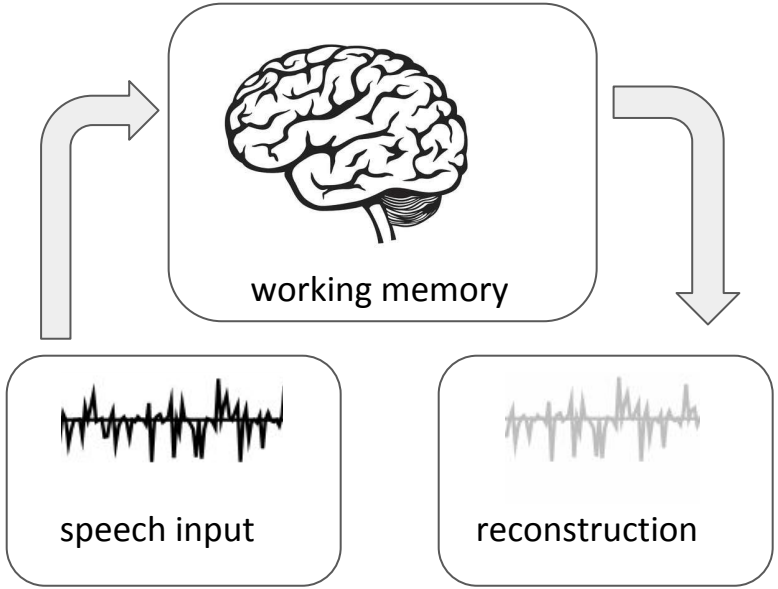
This makes it difficult to remember speech in a language you're unfamiliar with



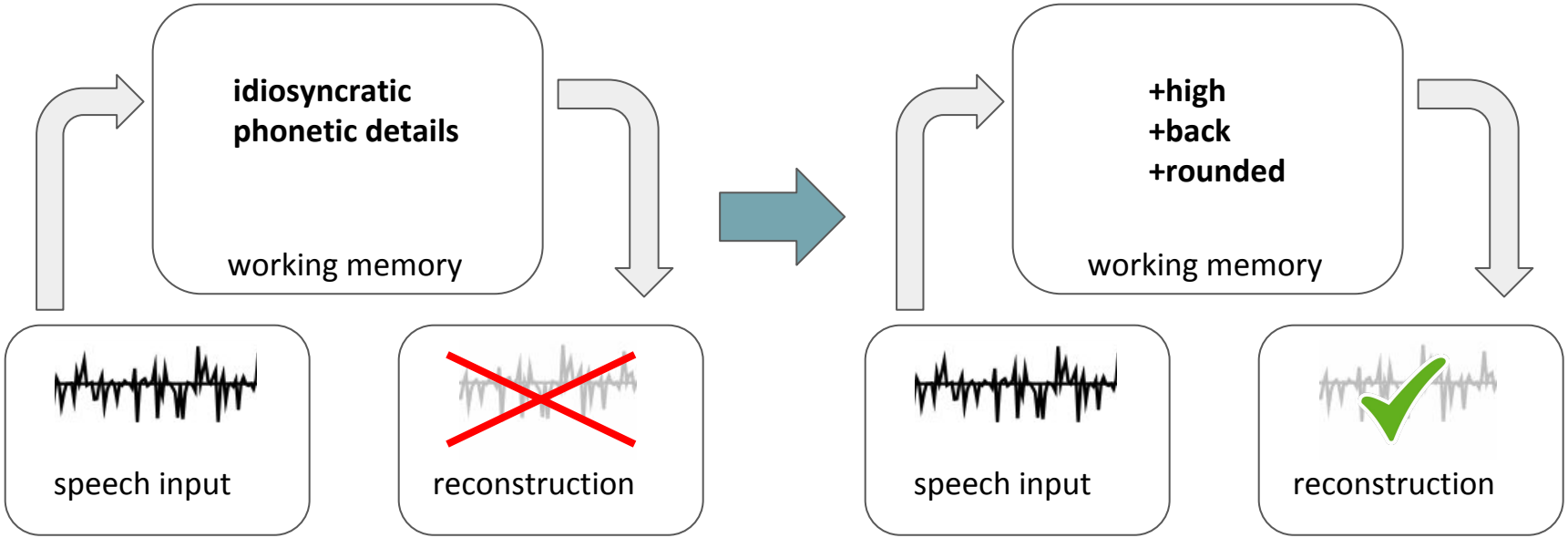
Learning as memorization

Perhaps if you try to memorize well with limited space…

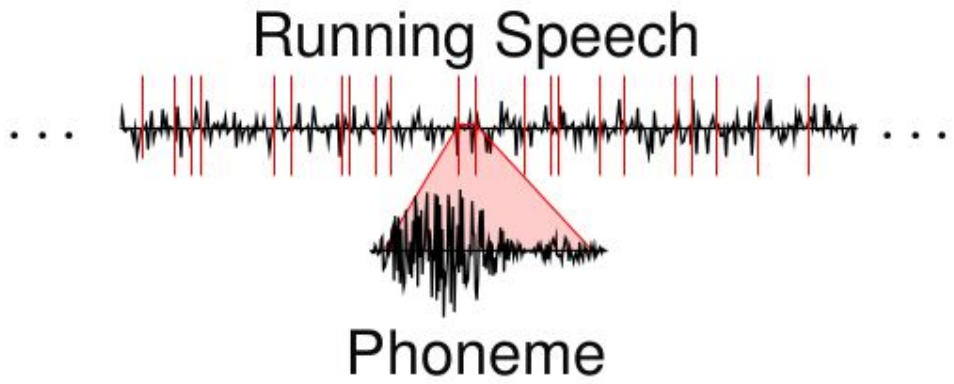
This will *force* you to learn some linguistic distinctions?



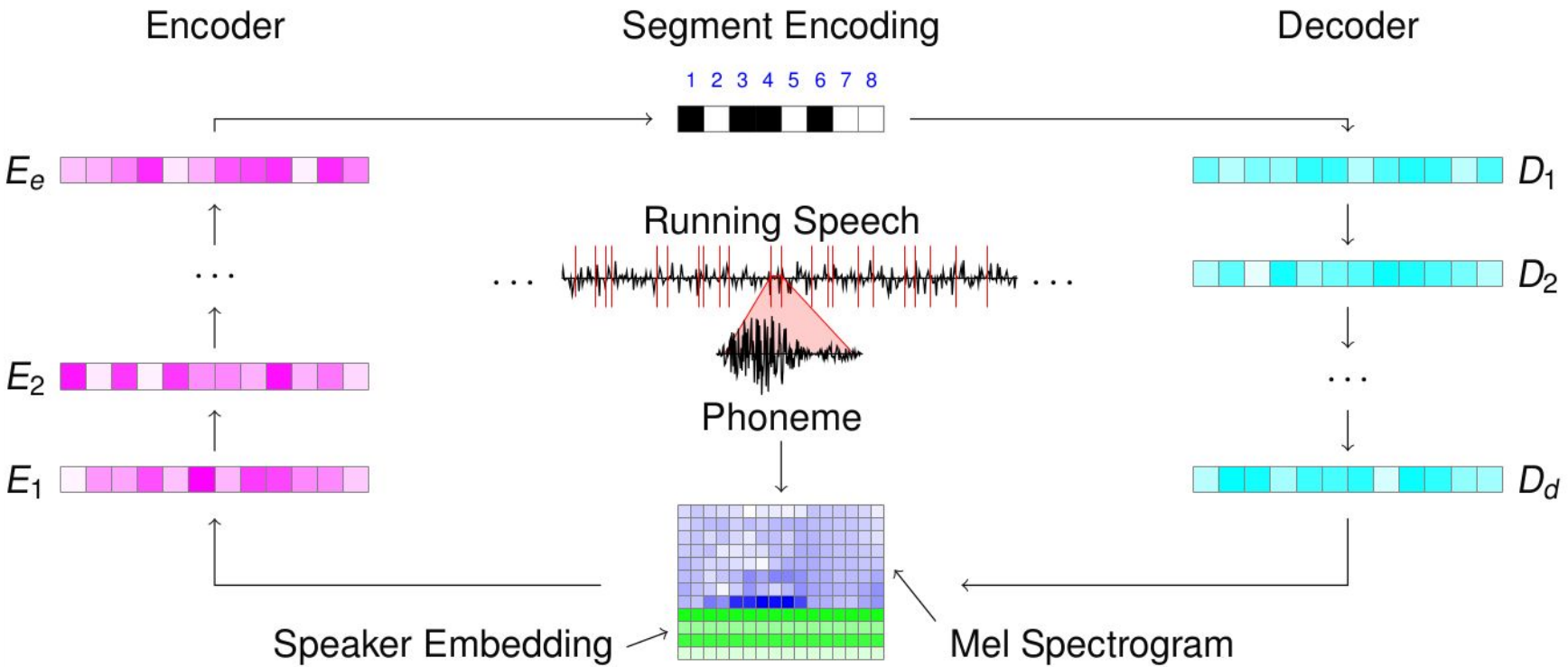
Learning as memorization



System design



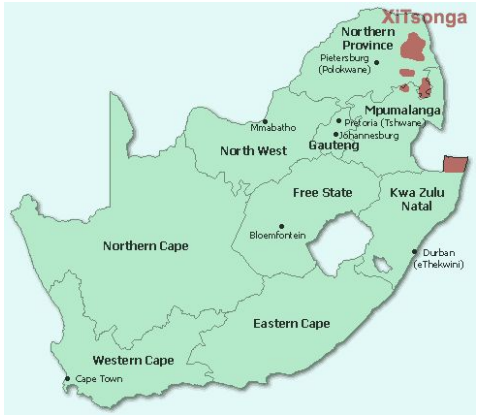
System design



Clustering effectiveness

American English

Xitsonga: Bantu

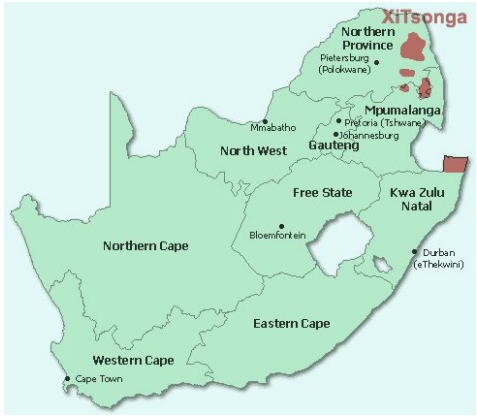


Zerospeech 2015 Challenge: Versteegh et al

Clustering effectiveness

American English

Xitsonga: Bantu



Homogeneity: 27%
Completeness: 18%

same label -> same phone

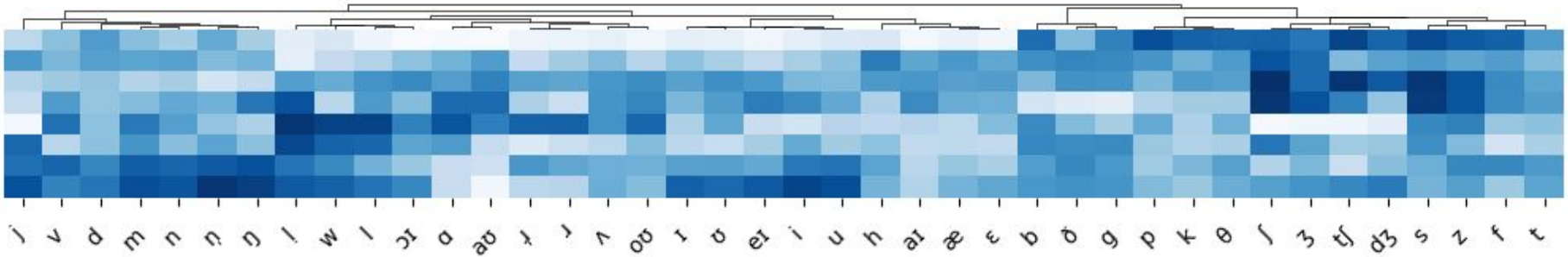
same phone -> same label

Homogeneity: 46%
Completeness: 27%

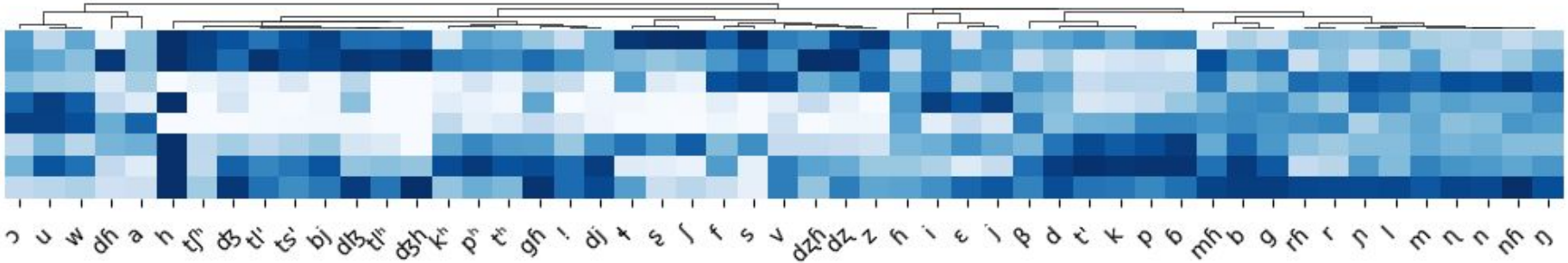
Zerospeech 2015 Challenge: Versteegh et al

Cluster analysis

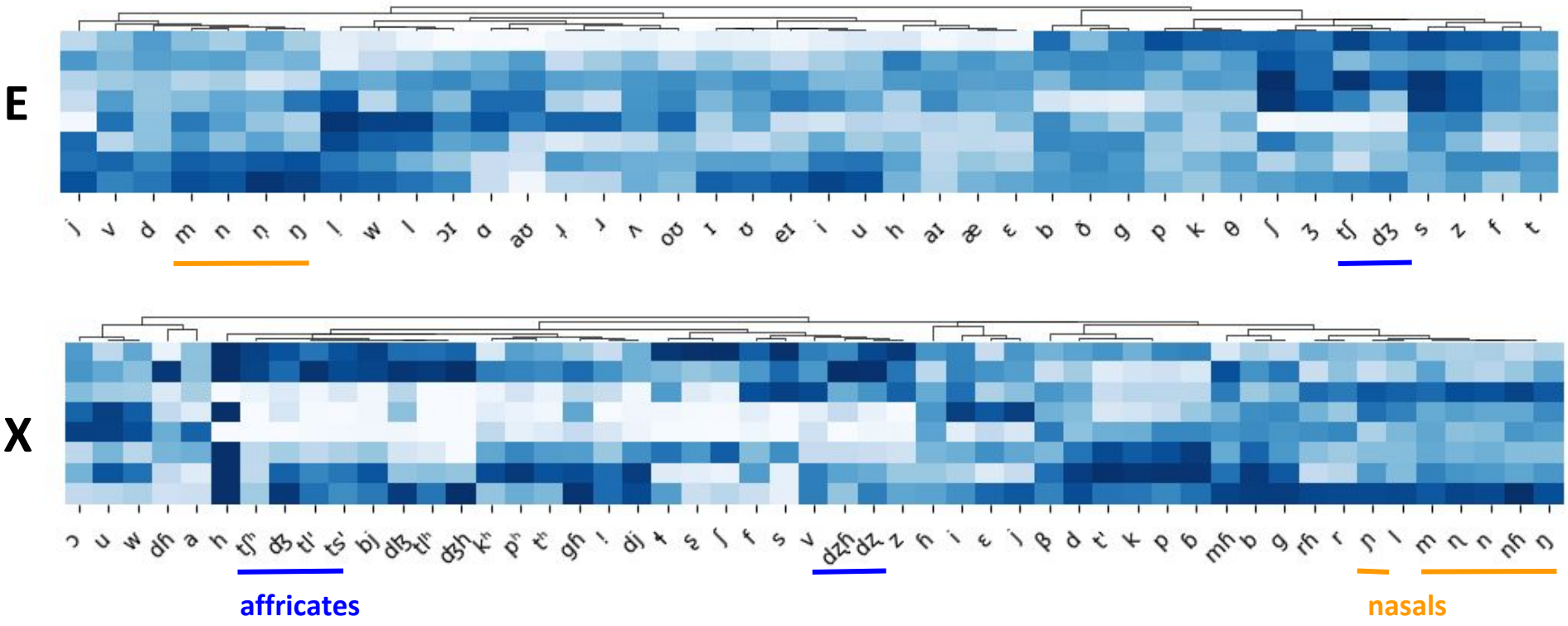
E



X



Cluster analysis



X

voice	94
sonorant	92
continuant	86
consonantal	86
approximant	86
syllabic	84
dorsal	83
strident	81
low	80
front	73
high	67
back	66
round	66
labial	65
coronal	65
tense	63
delayed release	62
anterior	55
nasal	51
distributed	38
constr. glottis	29
lateral	26
labiodental	17
trill	15
spread glottis	12
implosive	1

E

voice	89
sonorant	87
approximant	82
continuant	81
consonantal	78
syllabic	74
dorsal	71
strident	68
coronal	63
anterior	61
delayed release	55
front	55
high	49
tense	45
back	44
nasal	41
labial	37
low	37
distributed	33
stress	33
diphthong	33
round	27
lateral	25
labiodental	14
spread glottis	7

X

voice	94
sonorant	92
continuant	86
consonantal	86
approximant	86
syllabic	84
dorsal	83
strident	81
low	80
front	73
high	67
back	66
round	66
labial	65
coronal	65
tense	63
delayed release	62
anterior	55
nasal	51
distributed	38
constr. glottis	29
lateral	26
labiodental	17
trill	15
spread glottis	12
implosive	1

voicing
p: -voice
b: +voice

vowels vs consonants
t: -sonorant
a: +sonorant

E

voice	89
sonorant	87
approximant	82
continuant	81
consonantal	78
syllabic	74
dorsal	71
strident	68
coronal	63
anterior	61
delayed release	55
front	55
high	49
tense	45
back	44
nasal	41
labial	37
low	37
distributed	33
stress	33
diphthong	33
round	27
lateral	25
labiodental	14
spread glottis	7

X

voice	94
sonorant	92
continuant	86
consonantal	86
approximant	86
syllabic	84
dorsal	83
strident	81
low	80
front	73
high	67
back	66
round	66
labial	65
coronal	65
tense	63
delayed release	62
anterior	55
nasal	51
distributed	38
constr. glottis	29
lateral	26
labiodental	17
trill	15
spread glottis	12
implosive	1

voicing
p: -voice
b: +voice

vowels vs consonants
t: -sonorant
a: +sonorant

vowel place features
i: -back
u: +back

E

voice	89
sonorant	87
approximant	82
continuant	81
consonantal	78
syllabic	74
dorsal	71
strident	68
coronal	63
anterior	61
delayed release	55
front	55
high	49
tense	45
back	44
nasal	41
labial	37
low	37
distributed	33
stress	33
diphthong	33
round	27
lateral	25
labiodental	14
spread glottis	7

Good news and bad news

The system learns to distinguish vowels from consonants very well

Some within-class distinctions can be acquired as well…

Others (including that annoying English vowel space) cannot

Presumably these depend on **top-down** evidence about **words** and **phonological environments**

Integrating larger units

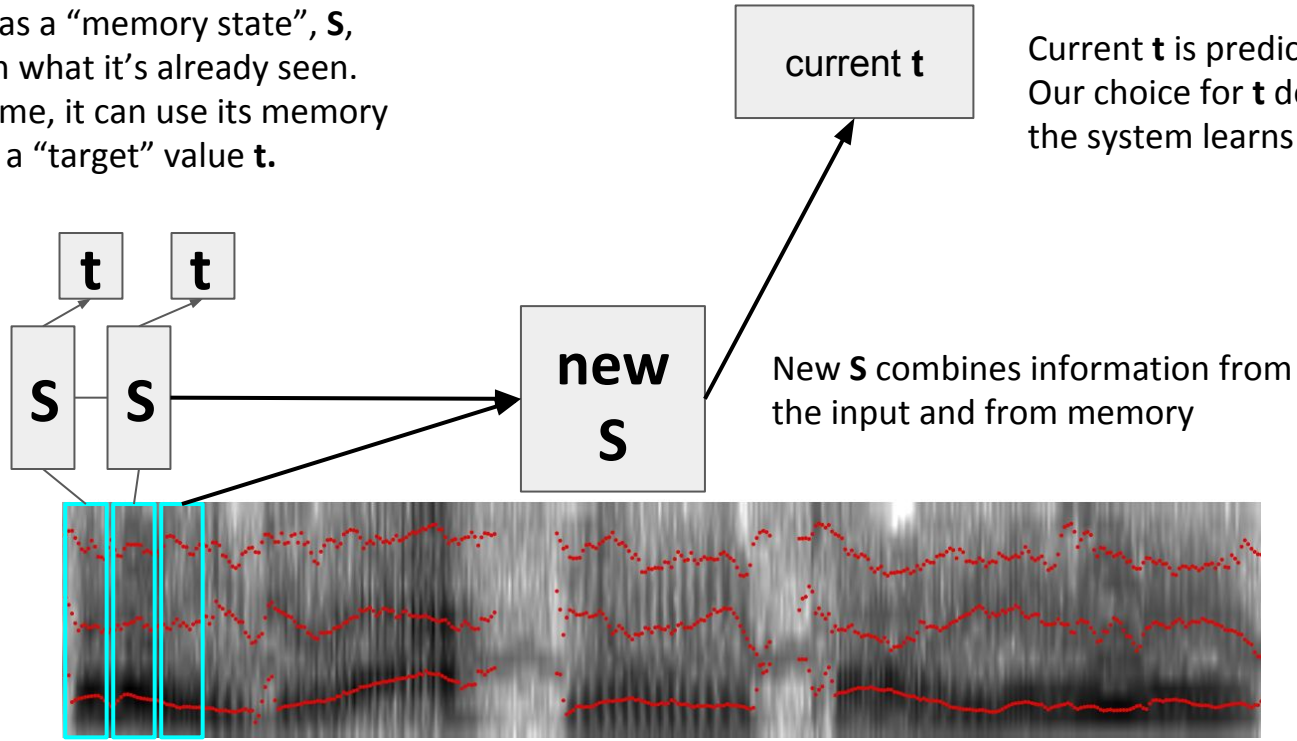
This system represents single phones, and relies on pre-labeled **phone boundaries**

We really want to process **unsegmented** speech

This could allow us to learn higher-level units, but it’s also more difficult!

Basic sequential network

Model has a “memory state”, **S**, based on what it’s already seen. At any time, it can use its memory to guess a “target” value **t.**



Current **t** is predicted based on **S**. Our choice for **t** determines what the system learns to remember.

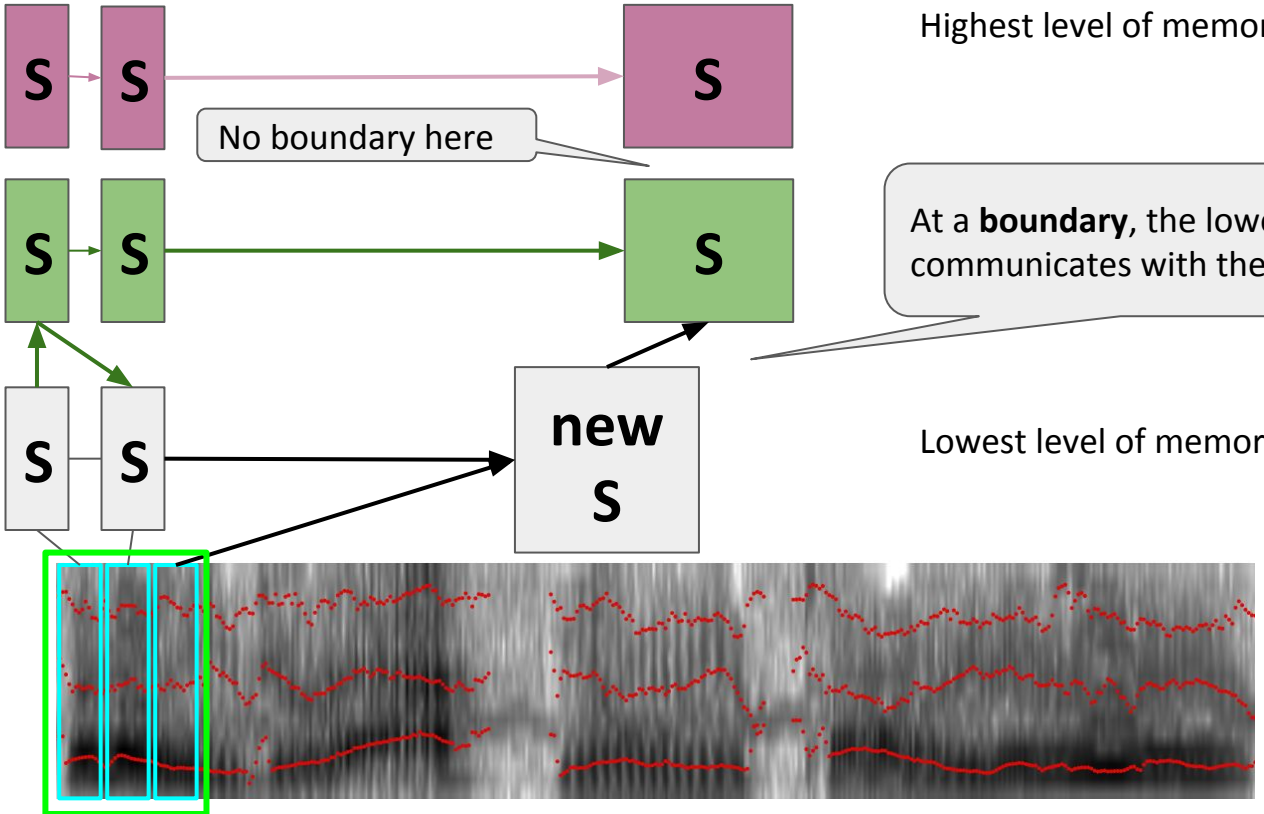
New **S** combines information from the input and from memory

Segments and hierarchy

This basic setup provides a **label** for each frame in the utterance (like the previous model)

But it doesn’t “chunk” the speech into coherent segments

Hierarchical sequential network

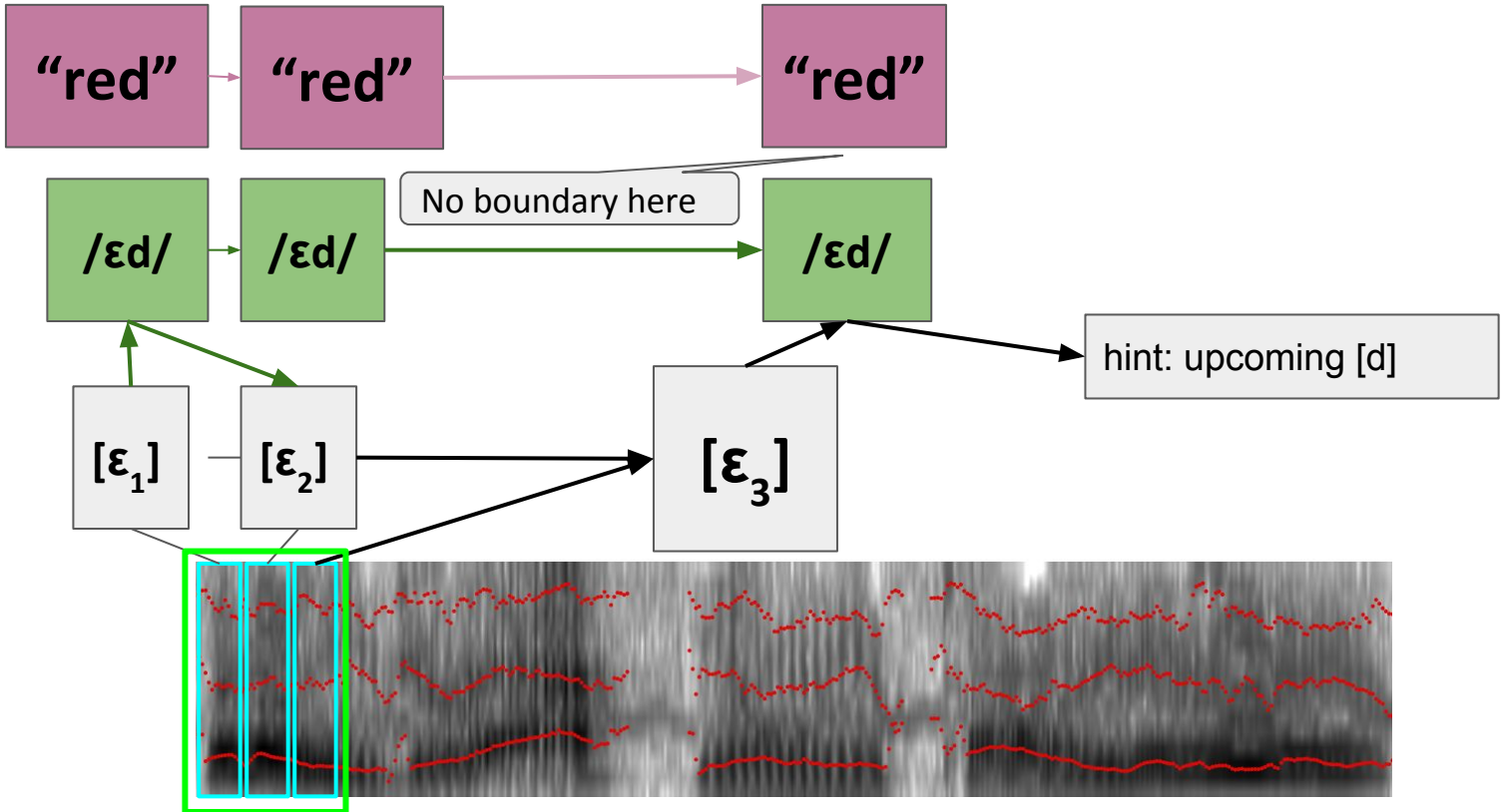


Highest level of memory (largest chunks)

At a **boundary**, the lower level communicates with the upper

Lowest level of memory (input-level)

How is this meant to work?

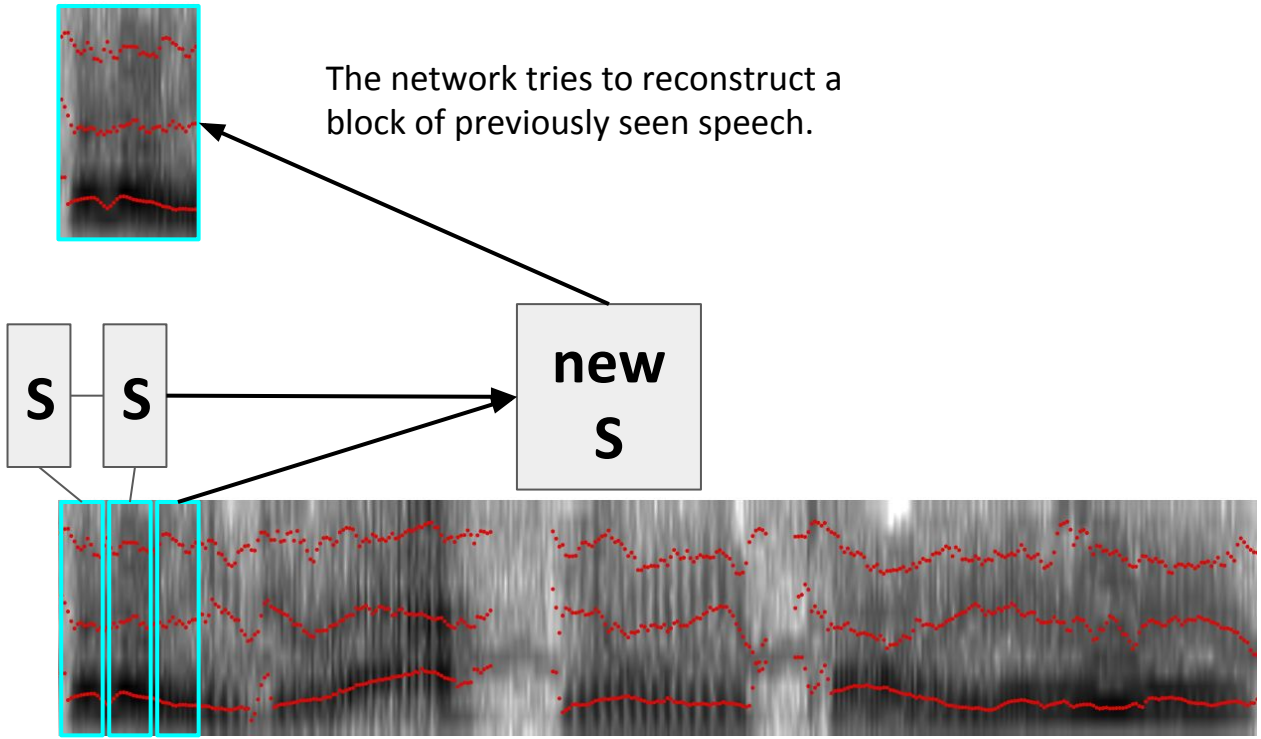


In practice, we don’t know what the levels will do

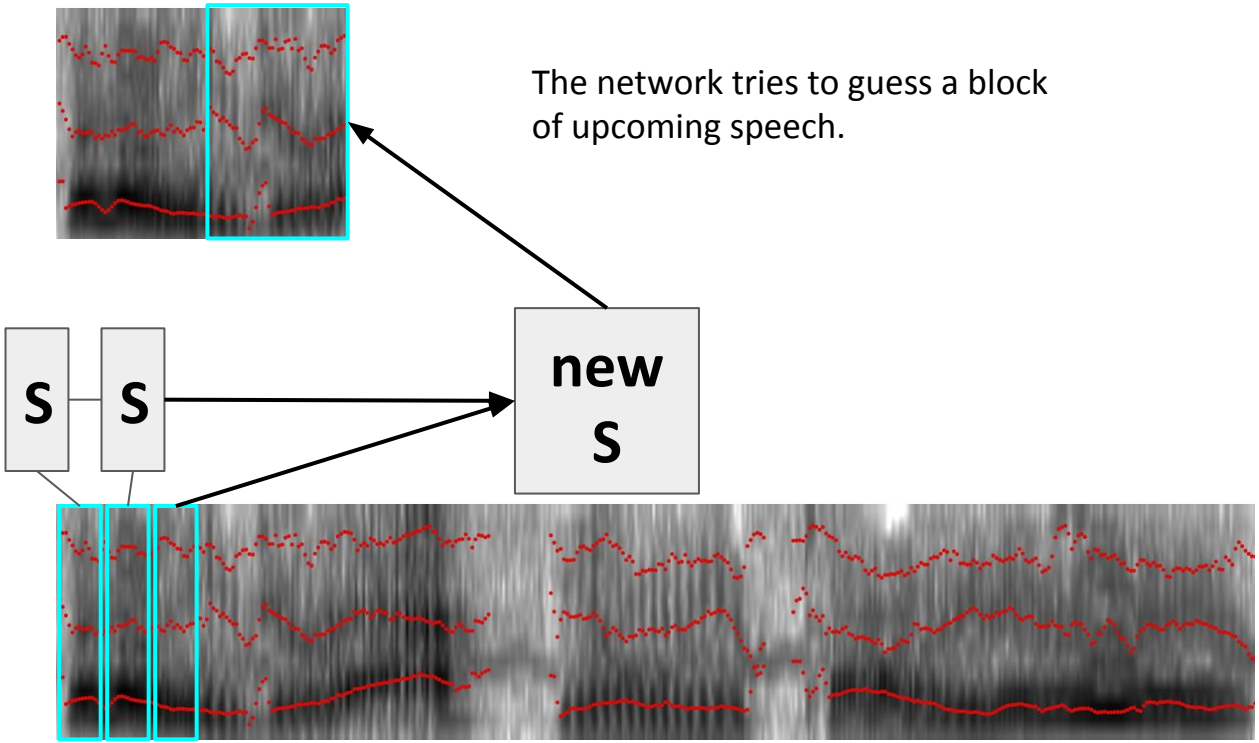
We can try to manipulate what the model learns in different ways

Starting with its learning objective (choice of **target**)

Memory in sequential model



Prediction in sequential model



Factoring out non-linguistic cues

How do we get rid of speaker identity?

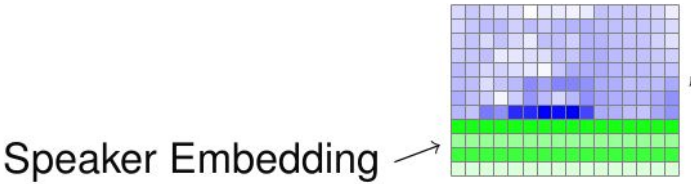
Infants are capable of identifying speakers in their native language (Johnson et al 2011)

But even young infant perception is not speaker-specific (Bergelson and Swingley 2017)

Dealing with speaker identity

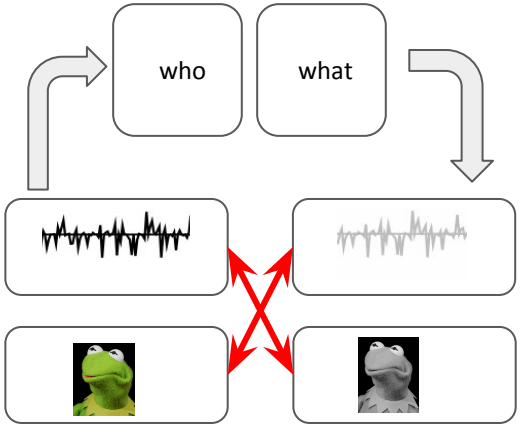
Provide a speaker ID

Hopefully, the model isolates correlations between speaker and sound within a few dimensions of its representation.



Adversarial learning

Split the model's representation in two, and punish the model if the "linguistic" part is correlated with the speaker



Preliminary conclusions

Both memory and prediction objectives contain information about linguistic abstractions

Using *some* method to combat speaker specificity leads to higher-quality representations

- The “adversarial” technique is effective at removing speaker information
- We still don’t know if it’s linguistically helpful

Brent 1999

Sample segmentations (transcribed speech)

yu want tu si **D6bUk**

lUk D*z **6b7** wIT hIz h&t

&nd 6**d Ogi**

yu want tu **lUk&t** DIs

lUk&t DIs

h&v 6**d rINk**

oke nQ

WAts DIs

WAts D&t

WAt Iz It

lUk k&n yu tek It Qt

tek It Qt

yu want It In

pUt D&t an

D&t

yEs

oke

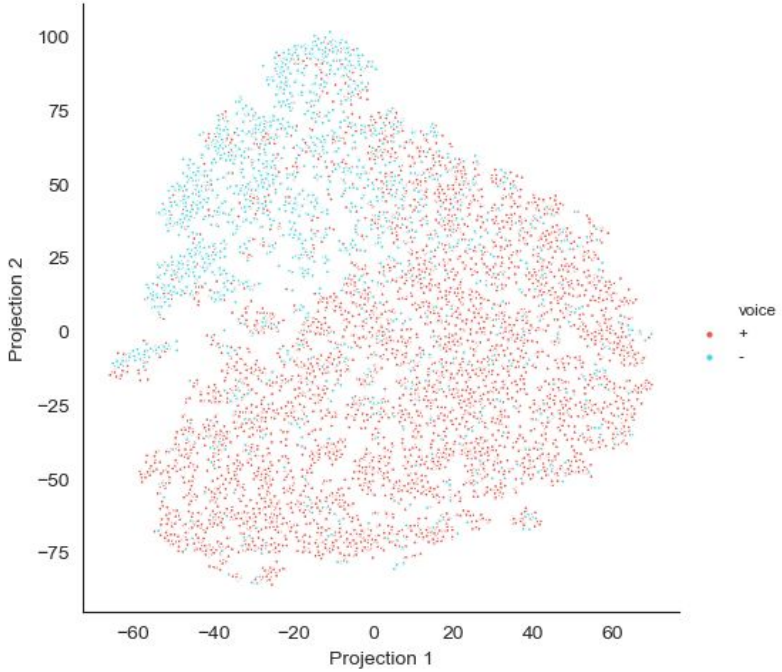
op~ It Ap

tek D6 dOgi Qt

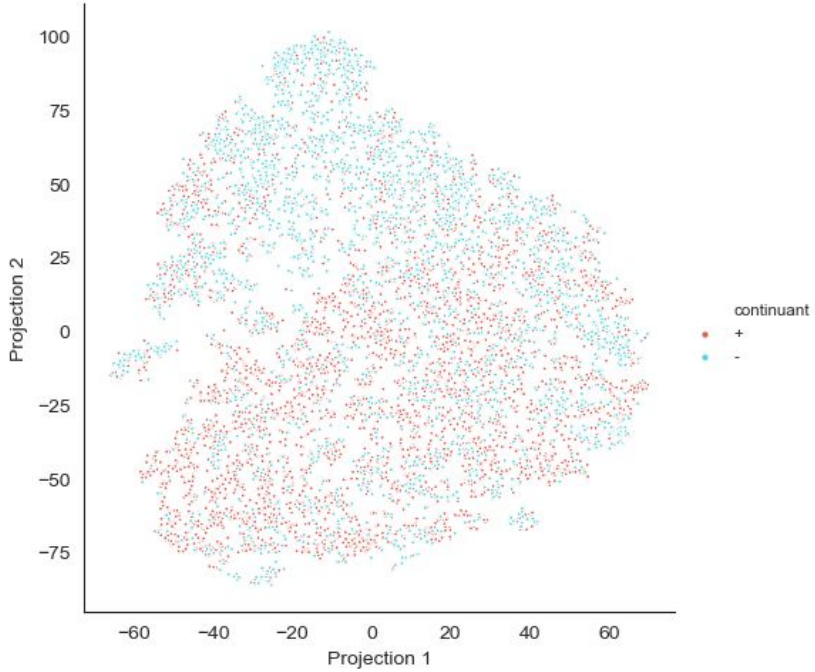
9**T INk** It wIl kAm Qt

Visualizations (audio)

En: +/- voice



En: +/- continuant



Preliminary conclusions

Systems that segment seem to learn roughly similar generalizations to the system with fixed boundaries

But not quite as well (yet)

Segmentation accuracy hovers in the high 50% range, but we believe we will do better soon

Conclusions

Simplistic models of distributional learning don’t appear to capture the flexibility and robustness of infant learning

Techniques that build hierarchical representations might be able to help

Low-level objectives (such as memorization) are capable of extracting considerable phonological information from raw audio

Thank you!

Thanks again to my collaborators, especially Cory Shain.

This work was supported by NSF #1422987

A Google Faculty Fellowship

And an equipment grant from NVIDIA