Computational modelling of infants’ word acquisition

Jing Liu

Master Thesis proposal

Department of Computer Science, KU Leuven

Research environment

COML

LIIR

- How to reverse engineer **infants'** acquisition of **vocabulary?**

Infants' acquisition of words

- Infants capacity to acquire words

- starts as early as 4 months(recognize their name) [1]

- 6 - 7 months: know the meaning of many common nouns [2 and segment words from fluent speech [3]

- 1 year old: comprehend around 80 words [4]

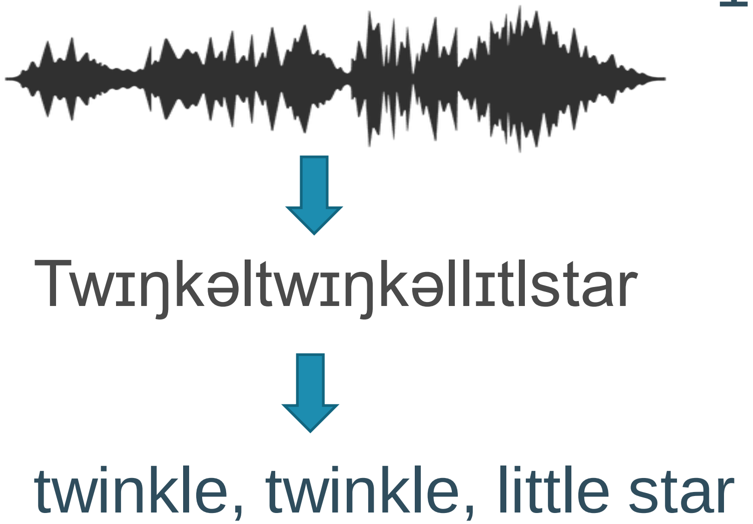
Infants' acquisition of words

- **Infants capacity to acquire words**

- starts as early as 4 months(recognize their name) [1]
- 6 - 7 months: know the meaning of many common nouns [2]

and segment words from fluent speech [3]

- 1 year old: comprehend around 80 words [4]



Twɪŋkəltwɪŋkəllɪtlstar

twinkle, twinkle, little star

Infants’ acquisition of words

Twɪŋkəltwɪŋkəllɪtlstar

twinkle, twinkle, little star

- **Infants capacity to acquire words**

- starts as early as 4 months(recognize their name) [1]
- 6 - 7 months: know the meaning of many common nouns [2]

and segment words from fluent speech [3]

- 1 year old: comprehend around 80 words [4]

- **What mechanism can explain this learning?**

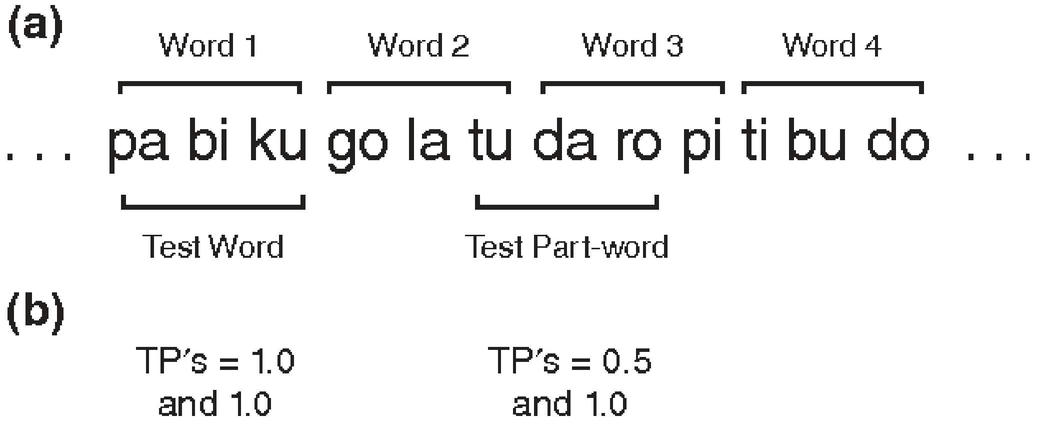
Statistical learning: the ability to extract statistical regularities from the speech input [5]

Statistical learning experiments

- **Highly controlled experimental setting**

Artificial language learning paradigm

- Simplified stimulus: tri-syllabic words
- Transitional probability is controlled

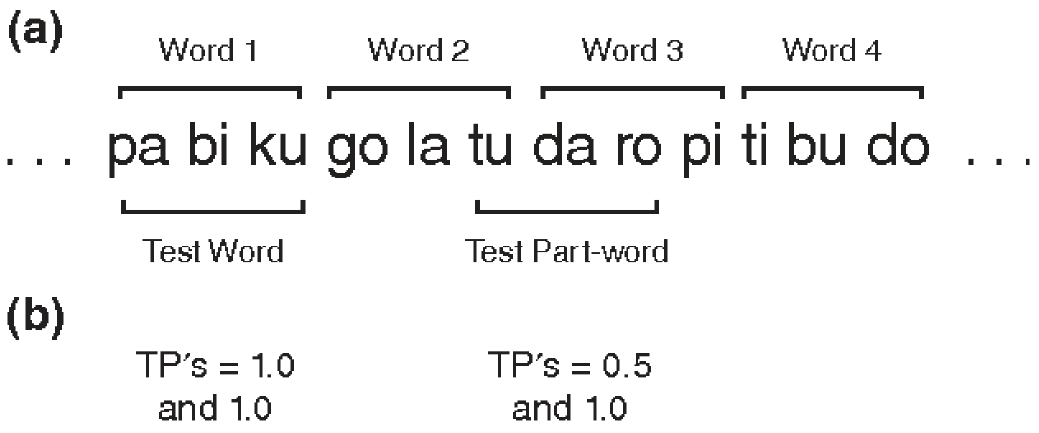


Statistical learning experiments

- **Highly controlled experimental setting**

Artificial language learning paradigm

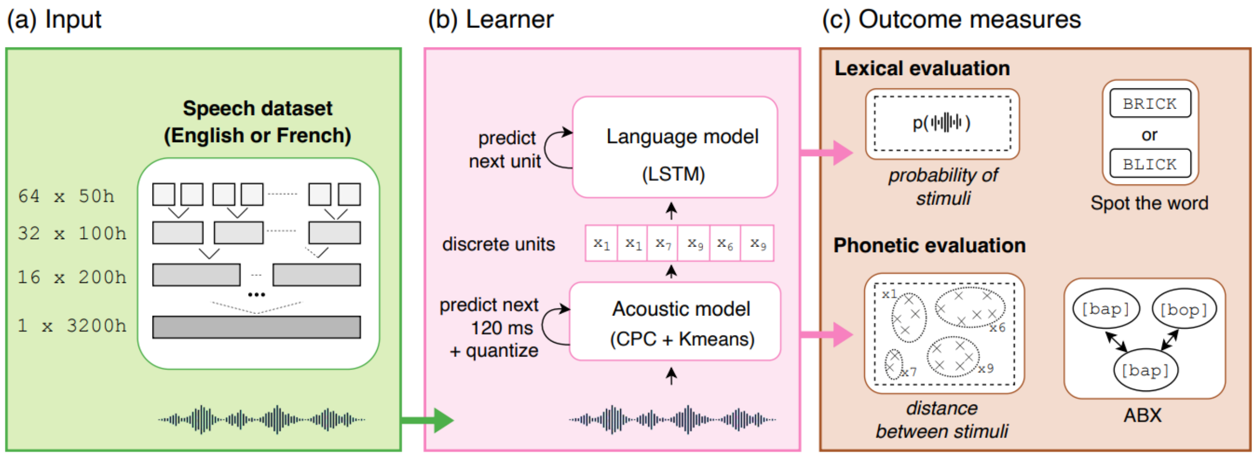
- Simplified stimulus: tri-syllabic words
- Transitional probability is controlled



- **Call for more ecologically valid setting**

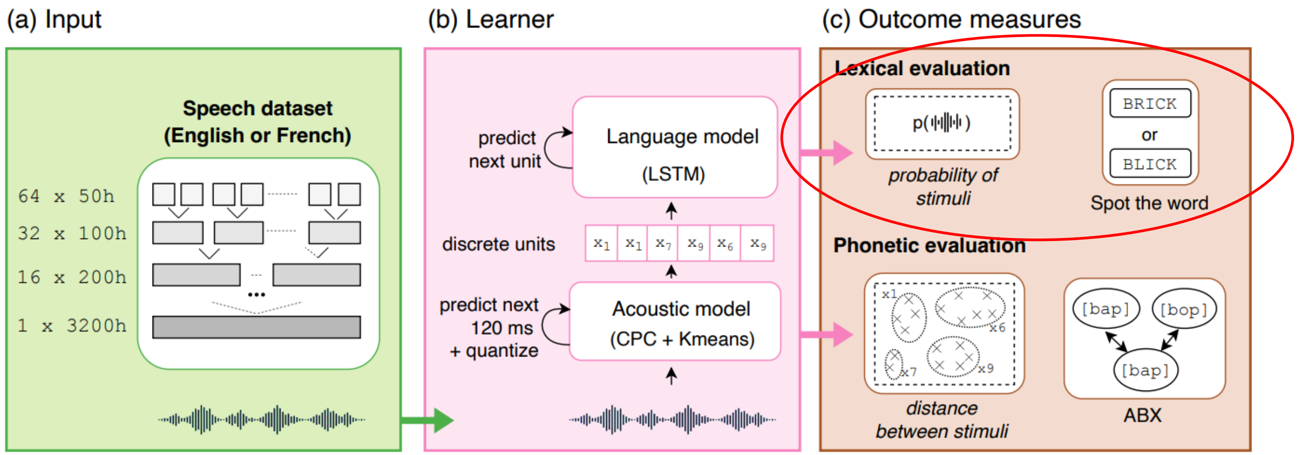
- Raw speech as input
- Few studies on statistical learning framework to bootstrap language
- Self-supervised learning algorithm relying on the statistical learning hypothesis [6]

The proposed model (STELA)



- Input: English audiobooks from LibriSpeech read by native speakers (3200h, 64*50h)
- Acoustic model: Contrastive Predictive Coding (CPC, to predict next 12 frames,120ms)
- Quantizer: K-means (to simulate phonemes)
- Language model: 3-layer LSTM (trained on discretized version of the audio files returned by the Quantizer.

The proposed model (STELA)



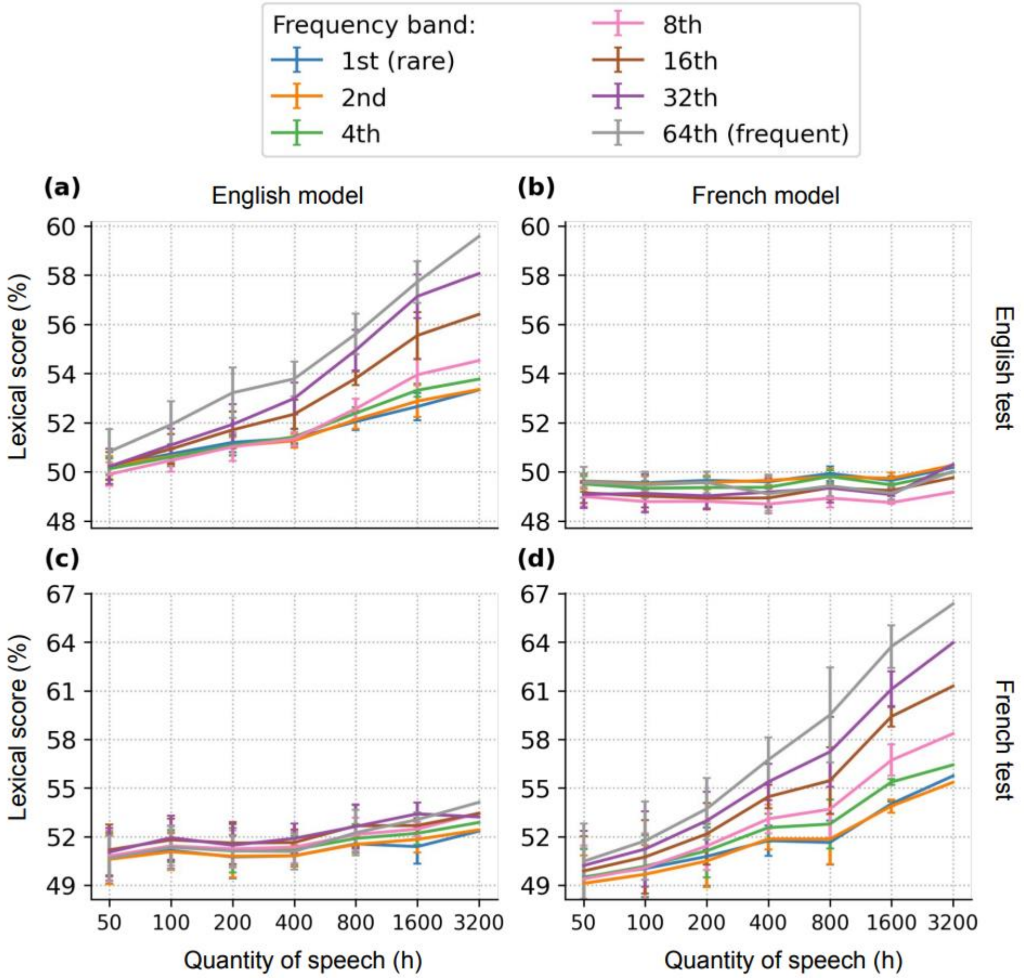
- **Lexical score**

- Spot the word task: present the network with a minimal pair of word and non-word (e.g., ’brick’ versus ’blick’)

- The accuracy score was averaged across all of the pairs in the test set

- Non-words are generated by the Wuggy toolbox

The problem



- **The frequency effect on the lexical task**

Words in the 64th class of frequency are present at least one time in the 50-hours training sets, two times in the 100 hours, Words belonging to the 32th class of frequency are present at least one time in the 100 hours training sets, 2 times in the 200 hours, etc.

- **Training efficiency**

-> non-exponential increase

Hypotheses

- Morphological rule learning

- High acoustic variability

Possible ceiling effect

-> test different types of input data

- The acquisition of proto-lexicons

-> test different segmentation algorithms

- A lack of memory mechanism
  - Episodic memory
  - Long-term memory

Testing different inputs and segmentation models

- **Models**

- Accumulator model (baseline): frequency-based

- STELA: Clustered CPC + LSTM

- DP-Parse: Non-parametric Bayesian model

- **Unit level**

- Word

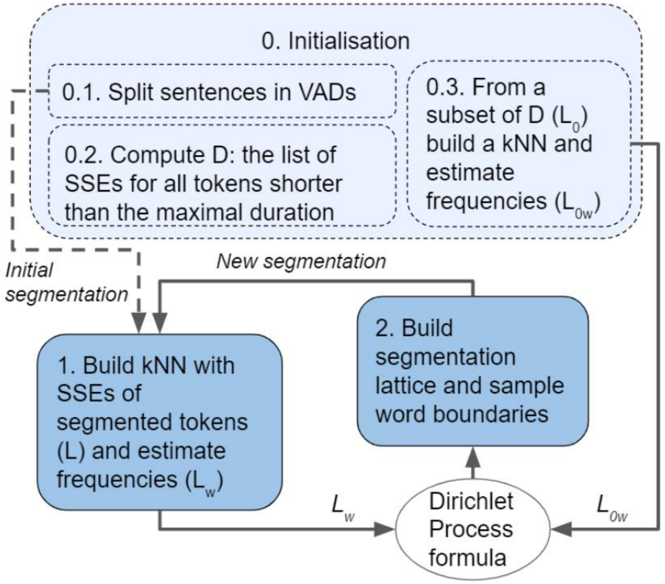
- Unsegmented phonemes: phonetic transcription

- Raw speech

- **Evaluation**

The average of the indicator function score(word) > score(nonword) over the test set of pairs (word, nonword).

Unit level	**Model**
Word	Accumulator LSTM
Unsegmented phonemes	DP-Parse DP-Parse + LSTM Clustered CPC + LSTM
Raw speech	DP-Parse DP-Parse + LSTM Clustered CPC + LSTM



0. Initialisation
0.1. Split sentences in VADs
0.2. Compute D: the list of SSEs for all tokens shorter than the maximal duration
0.3. From a subset of D (L_0) build a kNN and estimate frequencies (L_{0w})
Initial segmentation
New segmentation
1. Build kNN with SSEs of segmented tokens (L) and estimate frequencies (L_w)
2. Build segmentation lattice and sample word boundaries
L_w
Dirichlet Process formula
L_{0w}

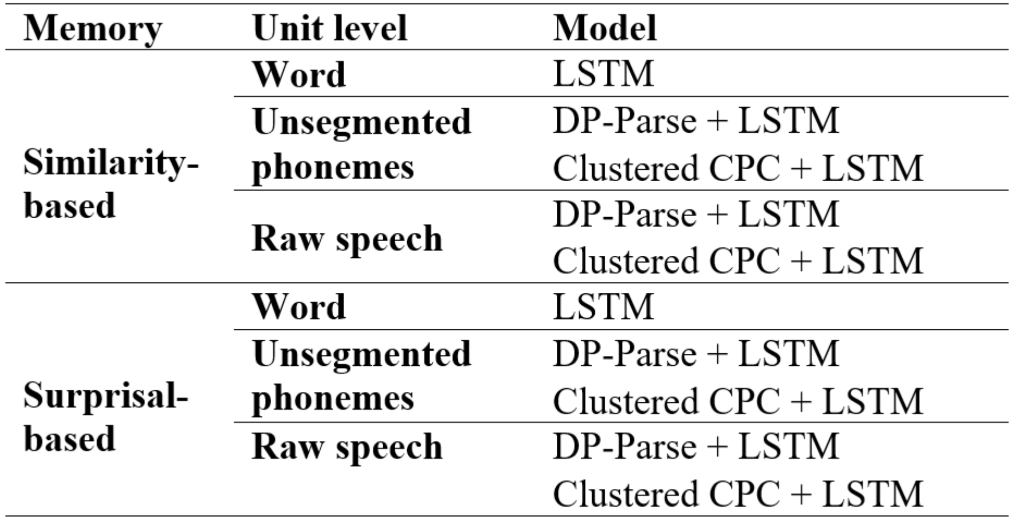
Integrating memory mechanism

- Episodic memory

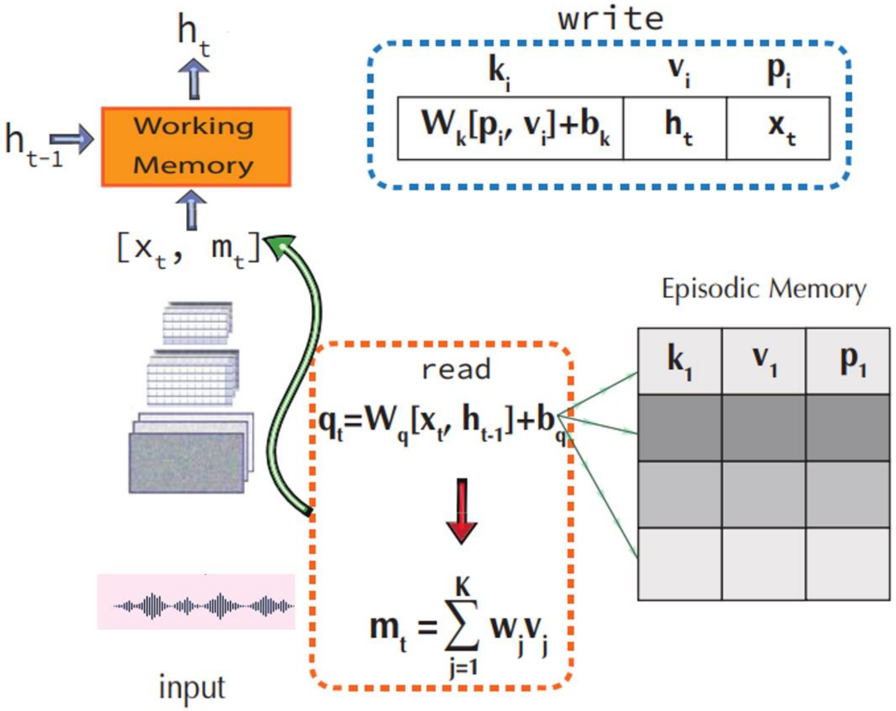
- Add the episodic memory module to the LSTM

- Selective mechanism

Similarity-based v.s. Surprisal-based



Memory	Unit level	Model
Similarity-based	Word	LSTM
	Unsegmented phonemes	DP-Parse + LSTM Clustered CPC + LSTM
	Raw speech	DP-Parse + LSTM Clustered CPC + LSTM
Surprisal-based	Word	LSTM
	Unsegmented phonemes	DP-Parse + LSTM Clustered CPC + LSTM
	Raw speech	DP-Parse + LSTM Clustered CPC + LSTM



Integrating memory mechanism

- Long-term memory

Q: Online v.s. offline knowledge distillation?

Looking forward to your suggestions & comments!