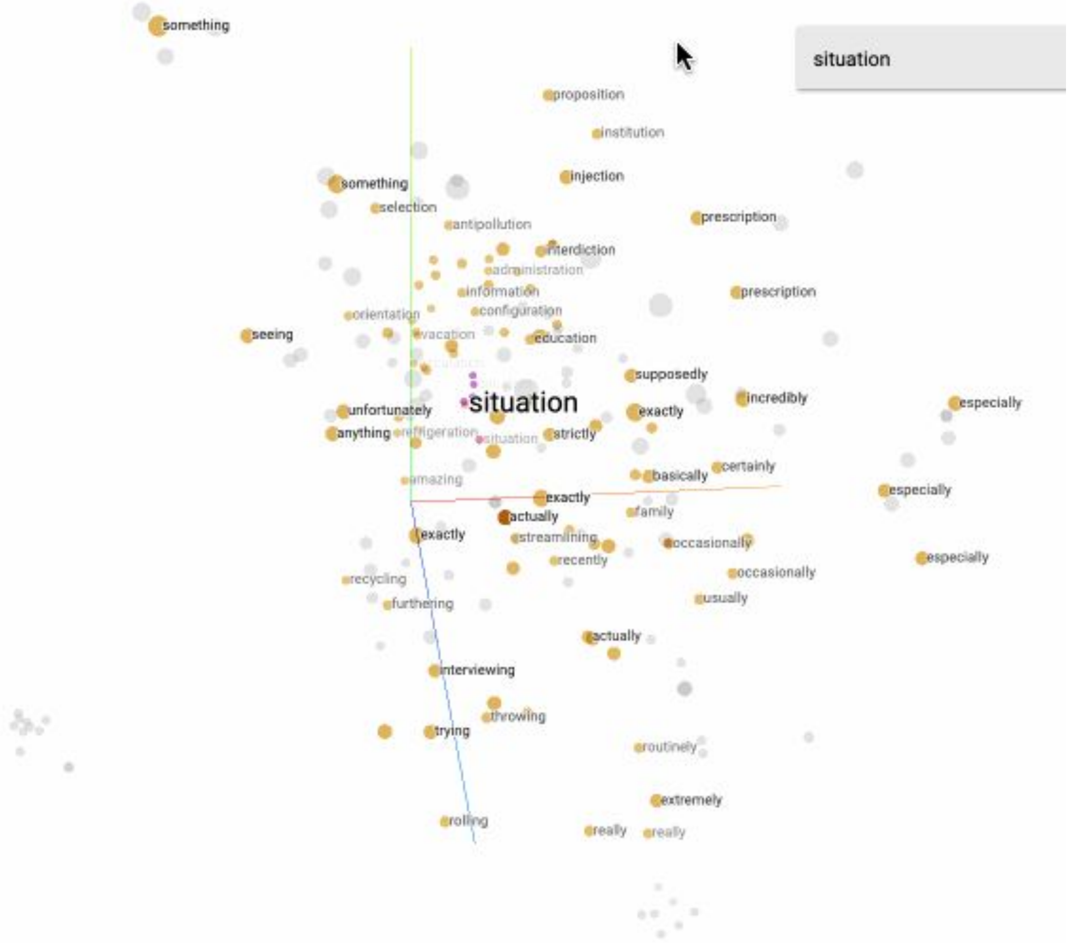
Multi-view Recurrent Neural Acoustic Word Embeddings with Phonetic Embeddings

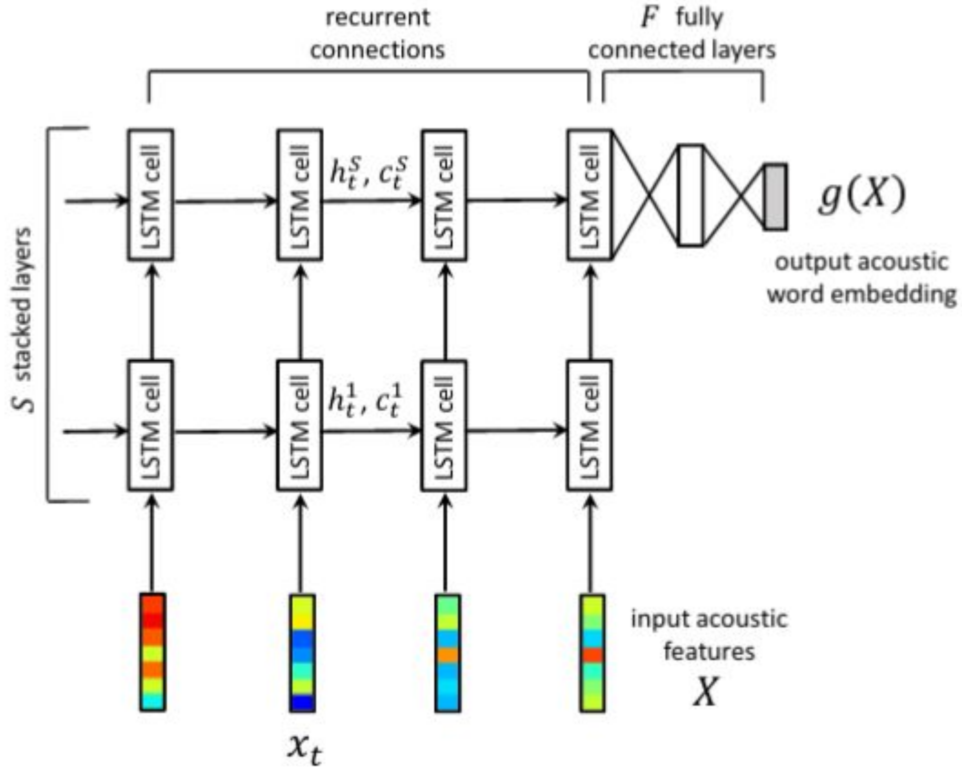


Valerie, Lynn
TTIC 31110
06/11/2020

Presentation Outline

- Acoustic Word Embeddings: Goals and Motivation
- Key Related Work: He et al, 2017
- Data and Experimental Setup
- Results and Figures
- Conclusions: Challenges, Limitations, Future Work

Acoustic Word Embeddings (AWE)



Key Related Work

Multi-view Recurrent Neural Acoustic Word Embeddings (He et al., 2017)

Idea: two views (character sequence and utterance), contrastive objectives

Future work they suggested: use phonetic sequence instead of character sequence

Method	**Test AP** (acoustic)	**Test AP** (cross-view)
MFCCs + DTW (Kamper et al., 2016)	0.214	
Correspondence autoencoder + DTW (Kamper et al., 2015)	0.469	
Phone posteriors + DTW (Carlin et al., 2011)	0.497	
Siamese CNN (Kamper et al., 2016)	0.549	
Siamese LSTM (Settle & Livescu, 2016)	0.671	
Our multi-view LSTM $\mathrm{obj}^0 + \mathrm{obj}^2$	**0.806**	**0.892**

Goals, Motivation, Research Questions

1. Performance with acoustic + phonetic embeddings vs. acoustic + text?
2. Performance with cost-sensitive margins vs. fixed margins in different objectives?
   a. (The paper's best results are reported on fixed margin+obj0+obj2, and they used cost-sensitive margins only for obj0)
3. Extend cost-sensitive margins to account for weighted phoneme substitution cost

Data and Experimental Setup

Dataset: Switchboard (same as in He et al.)

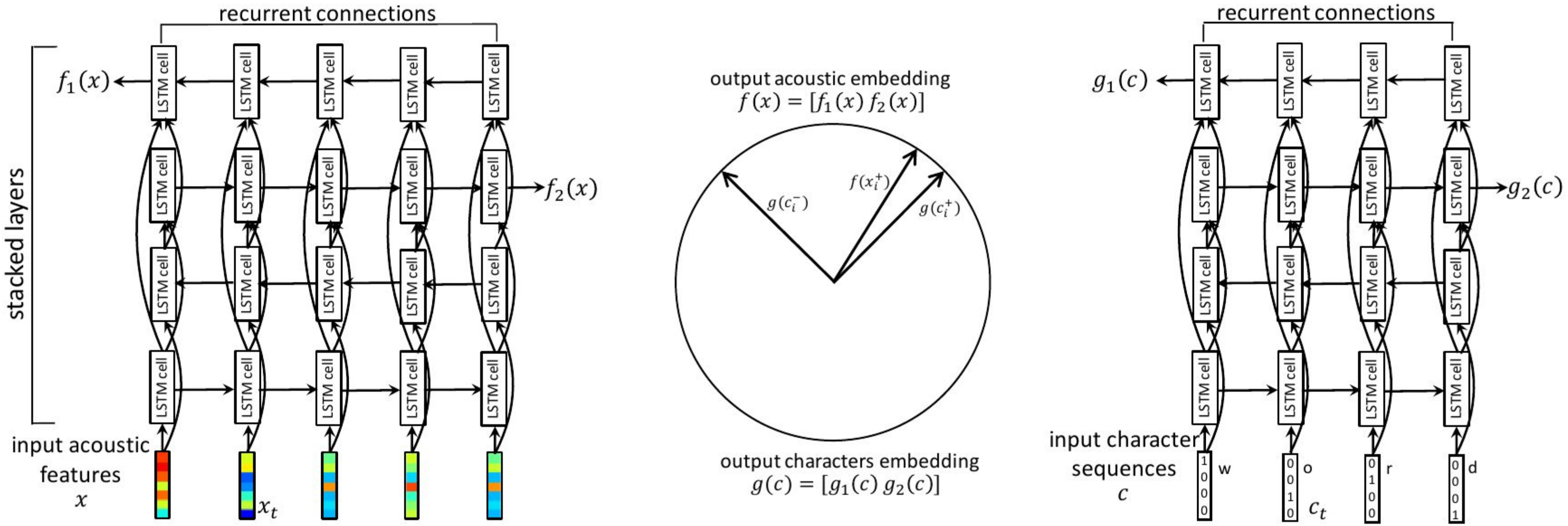


Figure 1: Illustration of our embedding architecture and contrastive multi-view approach.

Technical Details: Contrastive Objectives

$$\min_{f,g} \ \mathrm{obj}^0 := \frac{1}{N}\sum_{i}^{N} \max\left(0,\ m + dis\left(f(\mathbf{x}_i^+),\ g(\mathbf{c}_i^+)\right) - dis\left(f(\mathbf{x}_i^+),\ g(\mathbf{c}_i^-)\right)\right)$$

obj0: margin + dist(utterance, true label) - dist(utterance, most offending false label)

$$\min_{f,g} \ \mathrm{obj}^2 := \frac{1}{N}\sum_{i}^{N} \max\left(0,\ m + dis\left(f(\mathbf{x}_i^+),\ g(\mathbf{c}_i^+)\right) - dis\left(f(\mathbf{x}_i^-),\ g(\mathbf{c}_i^+)\right)\right)$$

obj2: margin + dist(utterance, true label) - dist(most offending utterance, true label)

Cost-sensitive margins

$$m(\mathbf{c}^+,\ \mathbf{c}^-) := m_{\max} \cdot \frac{\min\left(t_{\max},\ editdis(\mathbf{c}^+,\ \mathbf{c}^-)\right)}{t_{\max}}$$

Edit Distance, with hyperparameters max margin and max threshold

Technical Details: Edit Distance

- Levenshtein string edit distance applied to phones
  - Insertion, deletion, and substitution each has cost 1
- Use weighted phone substitution costs
  - Idea: **AH** is more different from **ER** than it is from **AW**
  - Substituting **AH** with **ER** should be more expensive than substituting **AH** with **AW**

		S	a	t	u	r	d	a	y
	0	1	2	3	4	5	6	7	8
S	1	0	1	2	3	4	5	6	7
u	2	1	1	2	2	3	4	5	6
n	3	2	2	2	3	3	4	5	6
d	4	3	3	3	3	4	3	4	5
a	5	4	3	4	4	4	4	3	4
y	6	5	4	4	5	5	5	4	**3**

OPERATION #1 - Delete "a"

OPERATION #2 - Delete "t"

OPERATION #3 - Transform "r" to "n"

3 operations in total

Results: Tabulated

Table 1: *Development set results.*

Subword	Margin	Objective	Dev AP (acoustic)	Dev AP (cross-view)
Chars	Fixed	obj^0	0.723	0.648
		$obj^0 + obj^2$	0.742	0.658
	Levenshtein	obj^0	0.741	0.615
Phones	Fixed	obj^0	0.775	0.730
		$obj^0 + obj^2$	0.785	0.738
	Levenshtein	obj^0	0.779	0.726
		$obj^0 + obj^2$	0.786	0.731
	Weighted	obj^0	0.687	0.641

Acoustic:
Given a pair of utterances, decide whether they are the same word or different words

Cross-view:
Given a pair: an utterance and a character/phone sequence, decide if the sequence is an example of the word

Note: Due to computing resource limitation, the edit distance models didn't fully converge and the weighted one has only been trained for 10 epochs

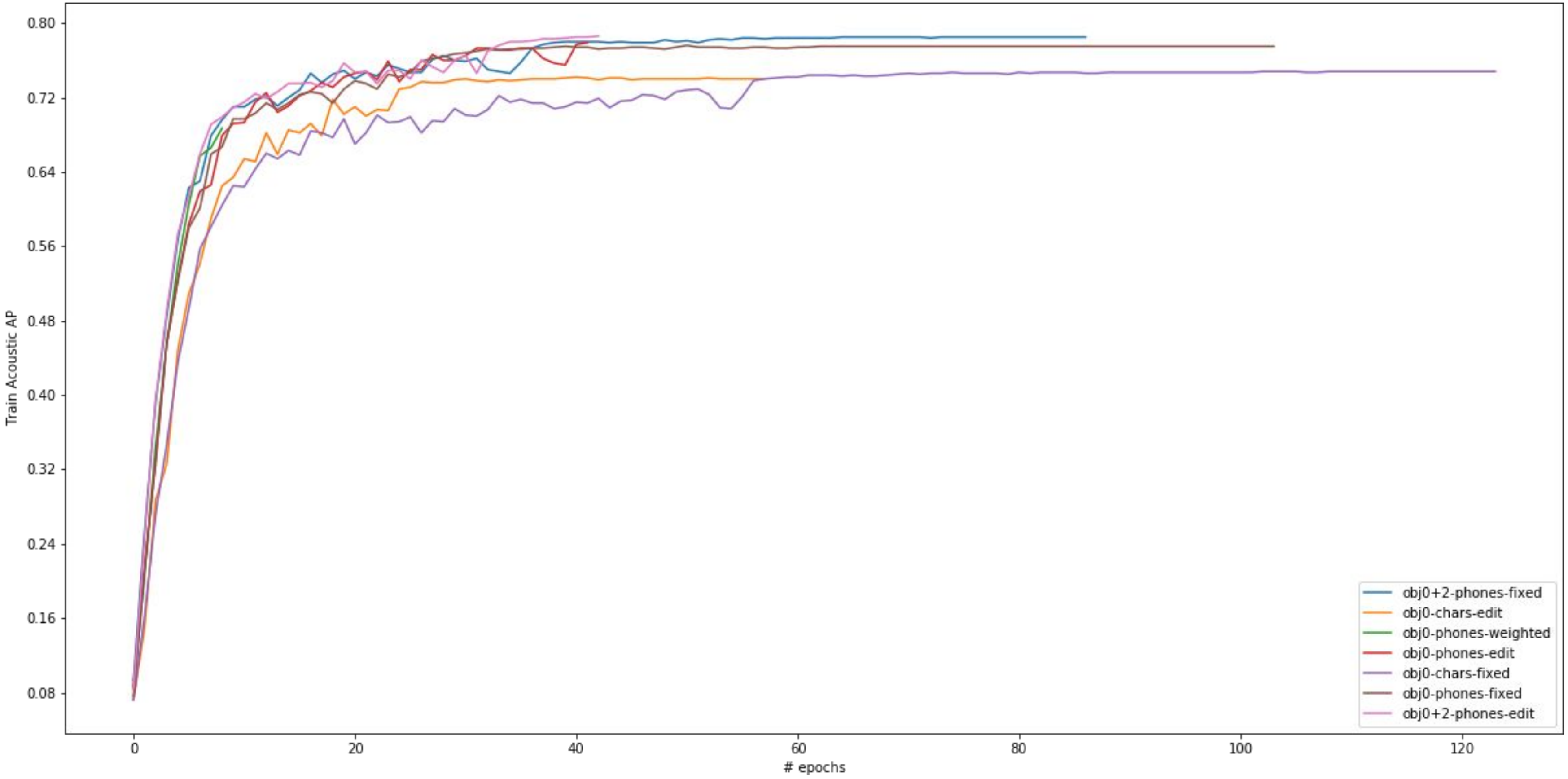
Results: Tabulated

Table 2: *Test set results.*

Subword	Margin	Objective	Test AP (acoustic)	Test AP (cross-view)
Chars	Fixed	obj^0	0.818	0.732
		$obj^0 + obj^2$	0.818	0.734
	Levenshtein	obj^0	0.806	0.689
Phones	Fixed	obj^0	0.836	0.775
		$obj^0 + obj^2$	0.847	0.786
	Levenshtein	obj^0	0.839	0.769
		$obj^0 + obj^2$	0.837	0.774
	Weighted	obj^0	0.755	0.673
Our multi-view LSTM		$\mathrm{obj}^0 + \mathrm{obj}^2$	**0.806**	**0.892**

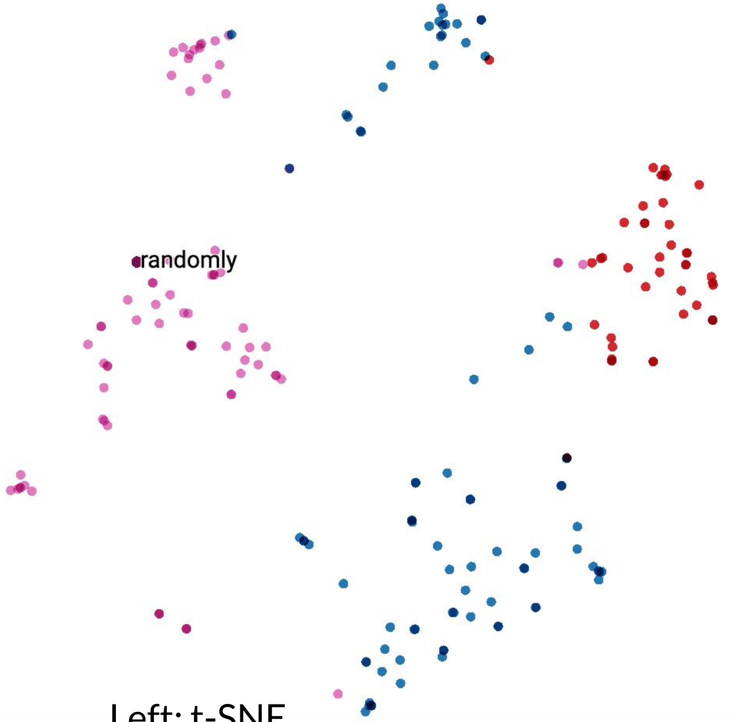
(From He et al.)

Results: Average Precision vs. Epoch



Results: t-SNE for Common Suffixes

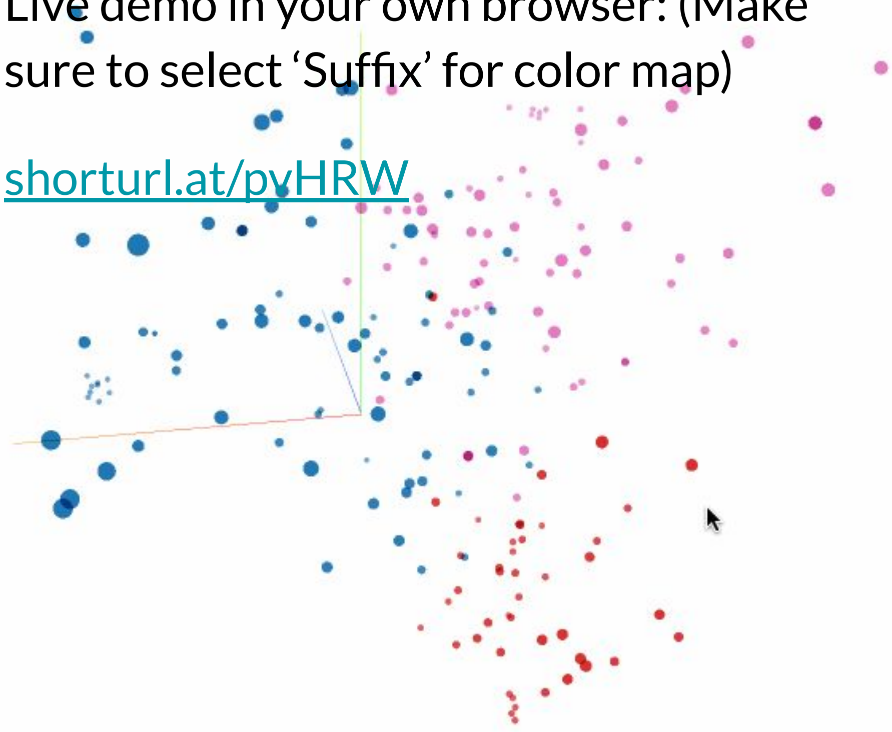
Pink: “ly” Red: “tion” Blue: “ing”



Left: t-SNE

Live demo in your own browser: (Make sure to select ‘Suffix’ for color map)

shorturl.at/pvHRW



Right: PCA

Conclusion: Challenges, Limitations & Future Works

- More hyperparameter tuning
    - For cost-sensitive, in addition to RNN hyperparameters, we can tune margin max and threshold max
- Speed up training
    - Especially for edit distance models
    - Solution: Instead of computing edit distances on the fly, precompute and store them in a lookup table
    - For a training vocab of 9000, this requires 9000*9000*8 bytes = ~1 GB

Q & A

Live demo in your own browser: (Make sure to select ‘Suffix’ for color map)

shorturl.at/pvHRW