Multi-view recurrent neural acoustic word embeddings with phonetic embeddings

Lynn Zheng[1], *Valerie Zhao*[1]

[1]University of Chicago

ruolinzheng@uchicago.edu, vzhao@uchicago.edu

Abstract

By mapping acoustic speech signals to vector representations, acoustic word embeddings can be useful for tasks like query-by-example search and whole-word speech recognition. Prior work developed a multi-view recurrent neural network, which learned acoustic embeddings and character embeddings together to leverage the correlation between how a word sounds and its orthographic representation [1]. In this work, we make a first step towards extending this approach, by investigating whether the model performance can improve by jointly learning acoustic and *phonetic* embeddings, and/or with different objectives that may rely on cost-sensitive margins. We found that phonetic embeddings indeed improved performance, but cost-sensitive margins had mixed results. Given these initial findings, we describe potential future directions.

Index Terms: acoustic word embeddings, speech recognition

1. Introduction and Background

Acoustic word embeddings can help address tasks that traditional word embedding could not. While word embeddings map word semantics to vectors representations, acoustic word embeddings instead consider acoustic signals that correspond to words. This property makes them potentially useful for whole-word speech recognition as well as query-by-example for low-resource languages.

Recent work developed a multi-view approach that leverages the similarity between how a word sounds and its character sequence [1]. This work presented a recurrent neural network that learns pairs of acoustic and character embeddings together. The models utilize contrastive objectives that attempt to separate, by some margin, the distance between a matching pair of acoustic and character embeddings (where both correspond to the same word) from the distance between a mismatching pair (where one of the embeddings does not correspond to the word). The performance of this model on word discrimination tasks—designed to approximate query-by-example and spoken term detection tasks—exceeded previous state-of-the-art performance.

A number of interesting questions arise from this approach. First, in addition to the text and acoustic segment of a word being correlated, its *phonetic* sequence is likely to correlate with the audio segment as well. Repeating this approach on acoustic and phonetic embeddings should therefore also achieve high (if not higher) performance, especially since the phonetic sequence may mirror the audio segment. Second, the original work trained with multiple contrastive objectives that mostly relied on a fixed margin. The work proposed a cost-sensitive margin (see Section 2) as well, but did not utilize it for most objectives. Finally, [1] calculated cost-sensitive margins with Levenshtein edit distance between positive and negative instances of character sequences (in other words, the character sequence that matches the word and a sequence that does not). It may be possible to improve upon this method when learning phonetic embeddings, as some phones are similar enough to preserve semantics when used in place of each other, while others are too distinct to do so.

In this work, we take a first step towards answering these questions. We first investigated whether it is possible to improve performance by jointly learning acoustic and *phonetic* embeddings. We found that for the same objective, models trained on acoustic and phonetic sequences indeed outperformed those trained on acoustic and character sequences. Next, for both types of embedding pairings (acoustic with character or phonetic), we explored whether objectives with cost-sensitive margins based on Levenshtein edit distance performed better than objectives with fixed margins. We observed that this was the case for acoustic word discrimination tasks, but not for cross-view word discrimination. Finally, for phonetic embeddings, we evaluated whether objectives with cost-sensitive margins would benefit from weighted phone substitution costs instead of Levenshtein edit distance. Our results did not show performance improvement in this area, and we discuss potential reasons for this.

2. Approach

Our approach mirrors that of [1], in that we used bidirectional long short-term memory networks (BLSTMs) to learn acoustic embeddings and character (or phonetic) embeddings together. We will largely delegate description of the approach to that body of work, but highlight and justify the objectives we chose.

We chose a subset of objectives from [1] for our experiments. Based on the terminology from the original work, the objectives we chose are i) $obj^0 + obj^2$, which is the combination of the two objectives below; and ii) obj^0 alone:

$$\min_{f,g} obj^0 := \frac{1}{N}\sum_{i}^{N} \max(0, m + dis(f(x_i^+), g(c_i^+)) - dis(f(x_i^+), g(c_i^-)))$$

$$\min_{f,g} obj^2 := \frac{1}{N}\sum_{i}^{N} \max(0, m + dis(f(x_i^+), g(c_i^+)) - dis(f(x_i^-), g(c_i^+)))$$

Both obj^0 and obj^2 attempts to separate the (cosine) distance between matching pairs of embeddings from mismatching pairs by some margin m. The first objective (obj^0) constructs a mismatching pair with a positive (correct) utterance of the word (x_i^+) and a negative (incorrect) subword unit label sequence (characters or phones) of the word (c_i^-). The second

objective (obj^2) constructs a mismatching pair between a negative utterance of the word (x_i^-) and a positive subword unit label sequence of the word (c_i^+). Therefore, we chose to study the combination of $obj^0 + obj^2$ because it thoroughly increases the distance between mismatching utterance and subword unit labels.[1] Indeed, [1] found that it (with fixed margins) yielded the best performance for acoustic-word embeddings out of the objectives tested. We hypothesize that it would also yield the best performance for acoustic-phonetic embeddings.

With both $obj^0 + obj^2$ and obj^0 objectives, we explored whether they perform better with fixed margins, or with the cost-sensitive margin below proposed by [1]:

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

In particular, [1] showed that obj^0 performed better with cost-sensitive margins for character embeddings, therefore this pattern may extend to phone embeddings as well. We chose obj^0 to test this hypothesis.

For phonetic embeddings, we also compared performance between cost-sensitive margins that rely on two different edit distances: one based on Levenshtein edit distance, the other based on phone-dependent substitution costs [3]. The former would give each of the three operations (insertion, deletion, and substitution) an equal unit cost. The latter would instead assign higher cost to substitution between two very dissimilar phones. For instance, we defined the substitution of `AH` with `AW` as having cost 3 but the substitution of `AH` with `UW` as having cost 8. This weighted substitution cost combined with our previous definitions of model objectives should place heavier penalty on mismatching phone sequence labels.

3. Methodology

To conduct our experiments, we extended the open-source, acoustically grounded word embedding (AGWE) codebase from [2],[2] which had adapted the code from [1]. The AGWE codebase supported both character and phonetic embeddings for the model. Our extension includes implementation of cost-sensitive margins, and is available to the public.[3]

3.1. Data

We trained and tested with data from the Switchboard English conversational speech corpus [4], using the default experimental setup available from the AGWE codebase.

3.2. Model Details and Hyperparameter Tuning

We will give a brief overview of the model, as it is similar to that of [2]. We used a 6-layer BLSTM for the acoustic view model and a 2-layer BLSTM for the character or phonetic view model. Both models contained 512 units per direction per layer. For training models with fixed-margin objectives, we tuned the initial learning rate over $\{0.0001, 0.001\}$ and the margin over $\{0.4, 0.5\}$ with a dropout rate of 0.4. These were the values that [1] highlighted as ones that tend to lead to best performance. We trained the model until convergence, which we observed to be no longer than about 150 epochs. For cost-sensitive margins, we planned to follow suggestions from [1] to tune the maximum margin m_{max} over $\{0.5, 0.6, 0.7\}$ and the threshold t_{max} over $\{7, 9, 11, 13\}$. Due to time and resource constraints we were not able to perform hyperparameter tuning here. For cost-sensitive margins with weighted phone substitution costs, we were also unable to train for more than 10 epochs.

Our experiments proceeded as follows. First, we trained the models for acoustic-word embeddings with fixed margins in each objective ($obj^0 + obj^2$ or obj^0), and repeated for acoustic-phonetic embeddings. Next, we repeated the step above with cost-sensitive margins. For phonetic embeddings, we adapted the Levenshtein edit distance to phones. Lastly, we trained the model with acoustic-phonetic embeddings again, this time using the cost-sensitive margin with different phone substitution costs. In all instances, we trained the models and tested them with a development set.

Like [1], our performance metric is the average precision (AP) of the model on acoustic and cross-view word discrimination tasks. As described in that work, the acoustic word discrimination task approximates model performance for query-by-search tasks. Given two acoustic speech segments, the model attempts to determine whether they correspond to the same or different words. The cross-view word discrimination task approximates spoken term detection. Given an acoustic speech segment and a written word, the model attempts to determine whether the speech segment corresponds to the written word.

4. Results

Table 1 lists the results of model performance varied by type of embeddings, objectives, and margins. Regardless of type of embeddings, models performed better with $obj^0 + obj^2$ than with obj^0 alone for both acoustic and cross-view word discrimination. This mirrors findings for character embeddings from [1]. Models trained with acoustic and phonetic embeddings performed better than their acoustic and text counterparts. Models typically performed slightly better with Levenshtein edit distance than fixed margins on acoustic word discrimination. However, they were worse at cross-view word discrimination. Cost-sensitive margins with weighted phonetic substitution costs did not achieve high performance.

Our best model is the combination of $obj^0 + obj^2$ with fixed margins trained on phones, even when comparing to the baseline fixed-margin character-embeddings condition, which (with a different architecture) had performed best in [1]. We show the test set AP results for these two conditions in Table 2. While we could have conducted a more thorough grid search on tuning this model, it was already able to surpass results reported from [1] in acoustic word discrimination. Note that we conducted our own experiment on the baseline condition because our model differed from that of [1].

Figure 1 visualizes the progression of the models' train set AP for acoustic word discrimination, with AP evaluated after every epoch. APs generally stabilized around or before 150 epochs, and $obj^0 + obj^2$ surpassed performance of obj^0 with fixed margins. Although they were not tuned, objectives with Levenshtein edit distance performed as well, if not better, than fixed margins (Figure 2). Figure 3 visualizes some phonetic sequence embeddings in 2-dimension, for words that end in "-ly", "-ing", or "-tion". While words with "-ing" suffixes are somewhat distributed, we see close clusters for "-ly" and for "-tion", indicating that similar words are embedded close together.

[1] In actual training, we used the k most mismatching pairs rather than the most mismatching one, and relied on the mean of their cosine distances. This detail comes from the codebase we adapted, which is from [2].

[2] https://github.com/ankitapasad/agwe-recipe

[3] https://github.com/RuolinZheng08/phonetic-acoustic-word-embeddings

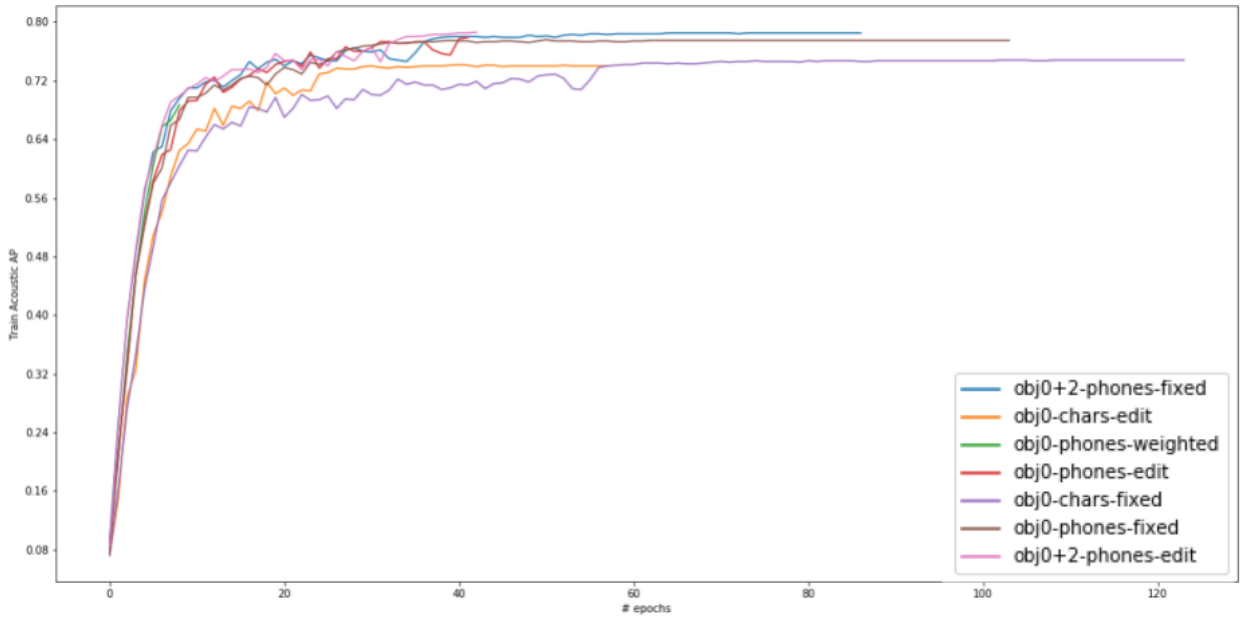

Figure 1: *Train set AP for our model on acoustic word discrimination, varied by objective, margin, and type of embeddings.*

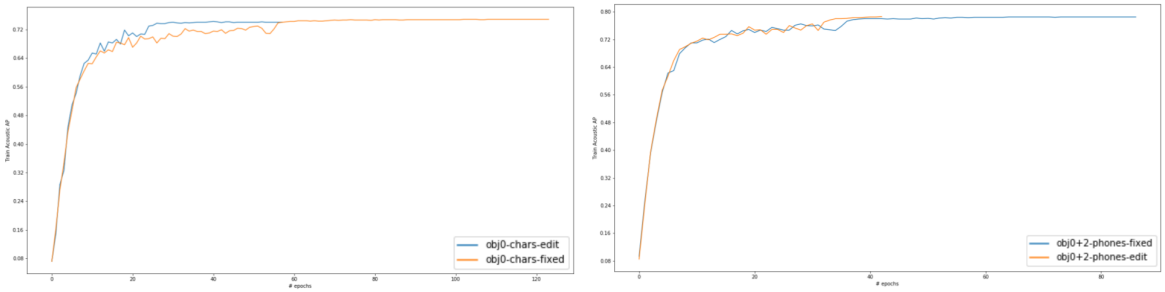

Figure 2: *Train set AP for our model on acoustic word discrimination, comparing fixed margins with edit distance-based margins.*

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[a]	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[b]	obj^0	0.687	0.641

[a]Due to constraints in computing resource, we were not able to tune the hyperparameters for the edit distance models.

[b]Due to constraints in computing resource, this model was only trained for 10 epochs.

Table 2: *Final test set AP for the multi-view LSTM approach with character vs. phonetic embeddings.*

Type of Embeddings	Test AP (acoustic)	Test AP (cross-view)
Character[a]	0.742	0.658
Phonetic	**0.847**	**0.786**

[a]We did not include the values reported in He et al. [1] as baseline above, but rather conducted our own experiment, as our architecture differed slightly from theirs.

5. Conclusion and Future Work

With recent work [1] on multi-view BLSTMs learning acoustic word embeddings, we examined whether joint learning with phonetic embeddings and/or different objectives could improve performance. We found that phonetic embeddings achieved higher performance than character embeddings, and that cost-sensitive margins with Levenshtein edit distance performed better overall at acoustic word discrimination than fixed margins. However, we note that we were not able to complete a comprehensive set of experiments and parameters tuning due to time and resource constraints. We propose the following future work.

We expect a lot of improvements from model hyperparameter tuning. For cost-sensitive models, in addition to tun-

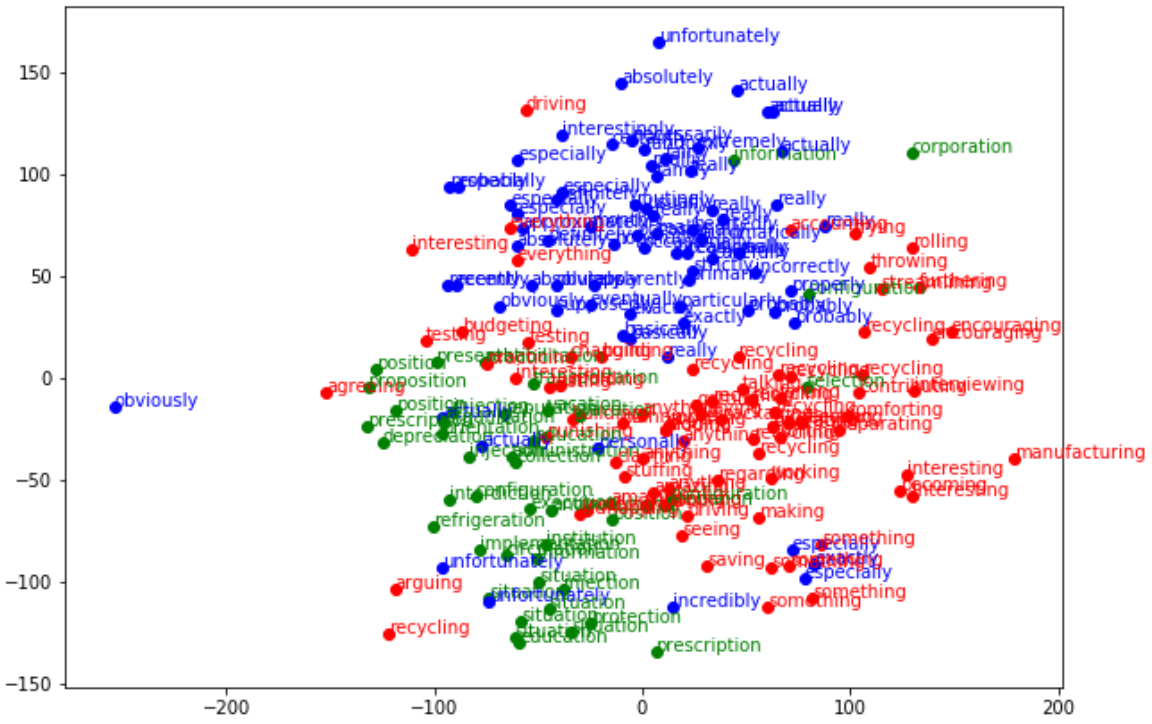

Figure 3: *t-SNE visualization of phonetic sequence embeddings for words ending in "-ly" (blue), "-ing" (red), and "-tion" (green).*

ing RNN hyperparameters like learning rate, we can also tune t_max, m_max with a simple grid search has proved effective in past works. In particular, [1] suggested tuning the maximum margin m_{max} over $\{0.5, 0.6, 0.7\}$ and the threshold t_{max} over $\{7, 9, 11, 13\}$.

Time and computing resources limited our ability to train edit-distance-based models to full convergence and to tune their hyperparameters. In particular, computing edit distance with weighted substitution costs was computationally heavy because it cannot use an efficient bit arithmetic optimized algorithm that an open-source edit distance library like `editdistance`[4] uses. We can leverage time-space tradeoff to speed up training: Instead of computing edit distances on the fly when evaluating different phone pairs, we can precompute them and store them in a lookup table. This is feasible for a vocabulary of medium size. For instance, for a vocabulary of size 9000 (like our train vocabulary), storing each distance as an `int8` requires a total of less than 1 GB of storage; however, for vocabulary of size 30000, this will require 7 GB.

6. References

[1] W. He, W. Wang, and K. Livescu, "Multi-view recurrent neural acoustic word embeddings," in *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017. [Online]. Available: https://openreview.net/forum?id=rJxDkvqee

[2] S. Settle, K. Audhkhasi, K. Livescu, and M. Picheny, "Acoustically grounded word embeddings for improved acoustics-to-word speech recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2019, Brighton, United Kingdom, May 12-17, 2019*. IEEE, 2019, pp. 5641–5645. [Online]. Available: https://doi.org/10.1109/ICASSP.2019.8682903

[3] P. Jyothi and K. Livescu, "Revisiting word neighborhoods for speech recognition," in *Proceedings of the 2014 Joint Meeting of SIGMORPHON and SIGFSM*. Baltimore, Maryland: Association for Computational Linguistics, Jun. 2014, pp. 1–9. [Online]. Available: https://www.aclweb.org/anthology/W14-2801

[4] J. J. Godfrey, E. C. Holliman, and J. McDaniel, "Switchboard: Telephone speech corpus for research and development," in *Proceedings of the 1992 IEEE International Conference on Acoustics, Speech and Signal Processing - Volume 1*, ser. ICASSP'92. USA: IEEE Computer Society, 1992, p. 517–520.

[4] https://github.com/roy-ht/editdistance