# Learning Word Embeddings from Speech

**Yu-An Chung and James Glass**
Computer Science and Artificial Intelligence Laboratory
Massachusetts Institute of Technology
Cambridge, MA 02139
{andyyuan,glass}@mit.edu

## Abstract

In this paper, we propose a novel deep neural network architecture, Sequence-to-Sequence Audio2Vec, for unsupervised learning of fixed-length vector representations of audio segments excised from a speech corpus, where the vectors contain semantic information pertaining to the segments, and are close to other vectors in the embedding space if their corresponding segments are semantically similar. The design of the proposed model is based on the RNN Encoder-Decoder framework, and borrows the methodology of continuous skip-grams for training. The learned vector representations are evaluated on 13 widely used word similarity benchmarks, and achieved competitive results to that of GloVe. The biggest advantage of the proposed model is its capability of extracting semantic information of audio segments taken directly from raw speech, without relying on any other modalities such as text or images, which are challenging and expensive to collect and annotate.

## 1 Introduction

Natural language processing (NLP) techniques such as GloVe [Pennington et al., 2014] and word2vec [Mikolov et al., 2013] transform words into fixed dimensional vectors. The vectors are obtained by unsupervised learning from co-occurrences information in the text, and contain semantic information about the word which are useful for many NLP tasks. Given the observation that humans learn to speak before they can read or write, one might wonder that since machines can learn semantics from raw text, might they also be able to learn the semantics of a spoken language from raw speech as well?

Previous research has explored the concept of learning vector representations from speech [He et al., 2017, Kamper et al., 2016, Chung et al., 2016, Settle and Livescu, 2016, Bengio and Heigold, 2014, Levin et al., 2013]. These approaches were based on notions of acoustic-phonetic similarity, rather than *semantic*, so that different instances of the same underlying word would map to the same point in the embedding space. Our work uses a very different skip-gram formulation to focus on the semantics of *neighboring* acoustic regions, rather than acoustic segment associated with the word itself.

Recent research by Harwath and Glass [2017], Harwath et al. [2016], Harwath and Glass [2015] has presented a deep neural network model capable of rudimentary spoken language acquisition using raw speech training data paired with contextually relevant images. Using this contextual grounding, the model learned a latent semantic audio-visual embedding space. In this paper, we propose a deep neural network architecture capable of learning fixed-length vector representations of audio segments from *raw* speech without any other modalities, such that the vector representations contain semantic information of underlying words. The proposed model, called Sequence-to-Sequence Audio2Vec, integrates an RNN Encoder-Decoder framework with the concept of continuous skip-grams, and can handle arbitrary length speech segments. The resulting vector representations contain information pertaining to the meaning of the underlying spoken words such that semantically similar words produce vector representations that are nearby in the embedding space.

31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA.

# 2 Proposed Approach

Our goal is to learn a fixed-length vector representation of an audio segment that is represented by a variable-length sequence of acoustic features such as Mel-Frequency Cepstral Coefficients (MFCCs), $\mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_T)$, where $\mathbf{x}_t$ is the acoustic feature at time $t$ and $T$ is the length of the sequence. We desire that this fixed-length vector representation is able to describe the semantics of the original audio segment to some degree. Below we first review the RNN Encoder-Decoder framework in Section 2.1, followed by formally proposing the Sequence-to-Sequence Audio2Vec model in Section 2.2.

## 2.1 RNN Encoder-Decoder Framework

Recurrent neural networks (RNNs) are neural networks whose hidden neurons form a directed cycle. Given a sequence $\mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_T)$, an RNN updates its hidden state $\mathbf{h}_t$ according to the current input $\mathbf{x}_t$ and the previous $\mathbf{h}_{t-1}$. The hidden state $\mathbf{h}_t$ acts as a form of internal memory at time $t$ that enables the network to capture dynamic temporal information, and also allows the network to process variable length sequences. Unfortunately, in practice RNNs do not seem to learn long-term dependencies well, so Long Short-Term Memory (LSTM) networks [Hochreiter and Schmidhuber, 1997], an advanced version of the vanilla RNN, have been widely used to conquer such difficulties.

An RNN Encoder-Decoder consists of an Encoder RNN and a Decoder RNN [Sutskever et al., 2014, Cho et al., 2014]. The Encoder reads the input sequence $\mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_T)$ sequentially, and the hidden state $\mathbf{h}_t$ of the RNN is updated accordingly. After the last symbol $\mathbf{x}_T$ is processed, the corresponding hidden state $\mathbf{h}_T$ is interpreted as the learned representation of the entire input sequence. Subsequently, by initializing its hidden state using $\mathbf{h}_T$, the Decoder generates an output sequence $\mathbf{y} = (\mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_{T'})$ sequentially, where $T$ and $T'$ can be different, or, in other words, the sequence lengths of $\mathbf{x}$ and $\mathbf{y}$ can be different. Such a sequence-to-sequence framework does not constrain the input or target sequences, and has been successfully applied to a wide range of challenging tasks such as machine translation [Sutskever et al., 2014, Cho et al., 2014], video caption generation [Venugopalan et al., 2015], abstract meaning representation parsing and generation [Konstas et al., 2017], and acquisition of acoustic word embeddings [Chung et al., 2016].

## 2.2 Sequence-to-Sequence Audio2Vec

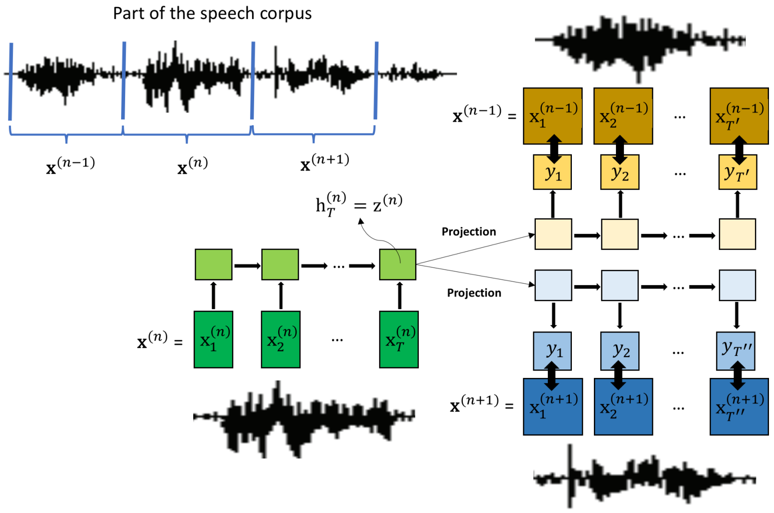

Figure 1: The Seq2seq Audio2vec model consists of an Encoder RNN and a Decoder RNN. The Encoder first takes an audio segment $\mathbf{x}^{(n)} = (\mathbf{x}_1^{(n)}, \mathbf{x}_2^{(n)}, ..., \mathbf{x}_T^{(n)})$ as input and encodes it into a vector representation of fixed dimensionality $\mathbf{z}^{(n)}$. The Decoder then maps $\mathbf{z}^{(n)}$ to several audio segments $\mathbf{x}^{(i)}, i \in \{n-k, ..., n-1\} \bigcup \{n+1, ..., n+k\}$ within in a certain range $k$ (in this example, $k = 1$). To successfully decode nearby audio segments, the encoded vector representation $\mathbf{z}^{(n)}$ should contain semantic information about the current audio segment $\mathbf{x}^{(n)}$.

Figure 1 depicts the structure of the proposed Sequence-to-Sequence Audio2Vec model (Seq2seq Audio2vec), which integrates the RNN Encoder-Decoder framework with a continuous skip-gram for unsupervised learning of audio segment representations that contain semantic information.

The idea of Seq2seq Audio2vec is simple: for each audio segment $\mathbf{x}^{(n)}$ in a speech corpus, the model is trained to predict the audio segments $\{\mathbf{x}^{(n-k)}, ..., \mathbf{x}^{(n-1)}, \mathbf{x}^{(n+1)}, ..., \mathbf{x}^{(n+k)}\}$ within a certain range $k$ before and after $\mathbf{x}^{(n)}$. By applying such a methodology, the audio segments of semantically similar spoken words are mapped to nearby points in the embedding space produced by the encoder. Figure 1 is an instance of Seq2seq Audio2vec setting $k = 1$.

The details of the proposed Seq2seq Audio2vec are as follows. Seq2seq Audio2vec consists of an Encoder RNN and a Decoder RNN. Given the $n$-th audio segment in any speech corpus, represented as a sequence of acoustic features $\mathbf{x}^{(n)} = (\mathbf{x}_1^{(n)}, \mathbf{x}_2^{(n)}, ..., \mathbf{x}_T^{(n)})$ of any length $T$, the Encoder RNN reads each acoustic feature $\mathbf{x}_t^{(n)}$ sequentially and updates the hidden state $\mathbf{h}_t^{(n)}$ accordingly. After the last acoustic feature $\mathbf{x}_T^{(n)}$ has been read and processed, the hidden state $\mathbf{h}_T^{(n)}$ of the Encoder RNN is viewed as the *learned representation* $\mathbf{z}^{(n)}$ of the current audio segment $\mathbf{x}^{(n)}$. The Decoder RNN now takes over the process. It first initializes its hidden state with $\mathbf{h}_T^{(n)}$, then for each audio segment $\mathbf{x}^{(i)}, i \in \{n-k, ..., n-1\} \bigcup \{n+1, ..., n+k\}$ within a certain range $k$ before and after $\mathbf{x}^{(n)}$, the Decoder RNN generates another sequence $\mathbf{y}^{i} = (\mathbf{y}_1^{(i)}, \mathbf{y}_2^{(i)}, ..., \mathbf{y}_{T'}^{(i)})$. The target of the output sequence $\mathbf{y}^{(i)}$ is set to be the corresponding audio segment $\mathbf{x}^{(i)}$, that is, the Decoder RNN attempts to predict *all* of the nearby audio segments at the same time. Note that it is the same Decoder RNN that generates all the output audio segments, and the audio segments can have different lengths. To successfully *decode* the nearby audio segments, the learned representation $\mathbf{z}^{(n)}$ should contain sufficiently useful information about the semantics of the current audio segment $\mathbf{x}^{(n)}$. The model is trained by minimizing the general mean squared error $\sum_{i \in \{n-k,...,n-1\} \bigcup \{n+1,...,n+k\}} \left\| \mathbf{x}^{(i)} - \mathbf{y}^{(i)} \right\|^2$.

# 3 Experiments

## 3.1 Experimental Setup

We use LibriSpeech [Panayotov et al., 2015], a large corpus of read English speech, as the data for experimentation. The corpus contains about 500 hours of broadband speech produced by 1252 speakers. Acoustic features consisted of 13 dimensional MFCCs produced every 10ms. The corpus was segmented according to word boundaries obtained by forced alignment with respect to the reference transcriptions, resulting in a large set of audio segments $\{\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, ..., \mathbf{x}^{(|C|)}\}$, where $|C|$ denotes the total number of audio segments (and words) in the corpus.

The Seq2seq Audio2vec model was implemented with PyTorch. The Encoder consists of 3-layers of LSTMs using 300 hidden units (so the dimensionality of the learned vector representations was 300), and the Decoder was a single-layer LSTM model with 300 hidden units. The model was trained by stochastic gradient descent without momentum, with a fixed learning rate of $1e-3$ and 500 epochs. We set $k$ to 5, meaning that during training, the model took the current audio segment $\mathbf{x}^{(n)}$ as input and attempted to predict the audio segments of the five preceding and following word segments.

After training the model, the Decoder RNN was no longer needed and could be discarded. Each audio segment $\mathbf{x}^{(n)}$ in the corpus was processed by the Encoder RNN, and encoded as a vector representation $\mathbf{z}^{(n)}$ of 300 dimensions. The vector representations representing the audio segments of the same word were then averaged to obtain a single 300-dim vector.

## 3.2 Evaluation and Results

We evaluated the vector representations learned by the proposed Seq2seq Audio2vec model on 13 different benchmarks [Faruqui and Dyer, 2014] that have been widely used to measure word similarity. They are: **WS-353** [Yang and Powers, 2006], **WS-353-REL** [Agirre et al., 2009], **WS-353-SIM**, **MC-30** [Miller and Charles, 1991], **RG-65** [Rubenstein and Goodenough, 1965], **Rare-Word** [Luong et al., 2013], **MEN** [Bruni et al., 2012], **MTurk-287** [Radinsky et al., 2011], **MTurk-771** [Halawi et al., 2012], **YP-130** [Yang and Powers, 2006], **SimLex-999** [Hill et al., 2015], **Verb-143** [Baker et al., 2014], and **SimVerb-3500** [Gerz et al., 2016]. These 13 benchmarks contain different numbers of pairs of English words that have been assigned similarity ratings by humans, and each of them tries to evaluate the word vectors in terms of different aspects. For example, **RG-65** and **MC-30** focus on nouns, **YC-130** and **SimVerb-3500** focus on verbs, and **Rare-Word** focuses on rare-words. We compared the vector representations learned by Seq2seq Audio2vec with GloVe trained on Wikipedia 2014. The similarity between a given pair of words was calculated by computing the cosine similarity between their corresponding vector representations. We then reported the Spearman's rank correlation coefficient $\rho$ between the rankings produced by each model against the human rankings [Myers and Well, 1995]. The results were displayed in Table 1 From Table 1, we can see that the performance of the vector representations learned by Seq2seq Audio2vec is competitive to the performance of GloVe word vectors on most of the word similarity tasks. This demonstrates that our proposed Seq2seq Audio2vec is capable of capturing semantic information from raw speech and representing it in a fixed-length vector representation, although the scores of our model were consistently lower than that obtained by GloVe. Aside from the differences due to the speech and text training data, we believe the reason for this difference is due to the inherent variability in speech production. Unlike textual representations, every instance of any spoken word ever uttered is different, due to vocal tract differences across speakers, speaking styles, contextual differences, and environmental conditions, to name but a few of the major influences on a speech recording. Clearly, one of the challenges for learning semantics directly from raw speech is to derive a more robust mechanism to address these issues. To us, what is more impressive is that many of the test scores are close.

Table 1: The Spearman's rank correlation coefficient $\rho$ between the rankings produced by each model against the human rankings. #(word pairs) is the number of word pairs in the dataset, and #(not found) is the number of word pairs whose vector representations could not be found.

| No. | Dataset | #(word pairs) | Seq2seq Audio2vec | | GloVe Wikipedia 2014 | |
|---|---|---|---|---|---|---|
| | | | #(not found) | $\rho$ | #(not found) | $\rho$ |
| 1 | WS-353 | 353 | 21 | 0.5324 | 0 | 0.6054 |
| 2 | WS-353-REL | 252 | 12 | 0.4959 | 0 | 0.5725 |
| 3 | WS-353-SIM | 203 | 7 | 0.5842 | 0 | 0.6638 |
| 4 | MC-30 | 30 | 0 | 0.6647 | 0 | 0.7026 |
| 5 | RG-65 | 65 | 0 | 0.7274 | 0 | 0.7662 |
| 6 | Rare-Word | 2034 | 783 | 0.3158 | 252 | 0.4118 |
| 7 | MEN | 3000 | 122 | 0.6877 | 0 | 0.7375 |
| 8 | MTurk-287 | 287 | 13 | 0.5647 | 0 | 0.6332 |
| 9 | MTurk-771 | 771 | 22 | 0.6010 | 0 | 0.6501 |
| 10 | YP-130 | 130 | 0 | 0.5173 | 0 | 0.5613 |
| 11 | SimLex-999 | 999 | 0 | 0.2985 | 0 | 0.3705 |
| 12 | Verb-143 | 144 | 0 | 0.2877 | 0 | 0.3051 |
| 13 | SimVerb-3500 | 3500 | 126 | 0.2023 | 2 | 0.2267 |

Using word similarity tasks as the only way to measure the quality of word vectors is not perfect and can sometimes lead to incorrect inferences [Faruqui et al., 2016, Schnabel et al., 2015]. In this preliminary study, we used these word similarity benchmarks to validate the effectiveness of the proposed model for learning meaningful vector representations from speech. In the future, we will evaluate the vector representations learned by our model on other downstream NLP tasks. It is also true, that some supervision was incorporated into the learning by using forced alignment segmentations as the basis for audio segments. In the future, it would be interesting to explore less supervised segmentations to learn word boundaries [Kamper et al., 2017b,a, 2015].

## 4 Conclusion and Future Work

In this paper, we proposed a Seq2seq Audio2vec model for unsupervised learning of audio segment representations. The vector representations generated by the model were evaluated on 13 commonly used word similarity benchmarks and were compared to those produced by GloVe from text data. To the best of our knowledge, this is the first work that attempts to learn fixed-length vector representations that contain semantic information directly, and only from raw speech. In the future, we will evaluate the vector representations on other tasks to examine their usefulness for speech and language processing.

## References

E. Agirre, E. Alfonseca, K. Hall, J. Kravalova, M. Paşca, and A. Soroa. A study on similarity and relatedness using distributional and wordnet-based approaches. In *NAACL HLT*, 2009.

S. Baker, R. Reichart, and A. Korhonen. An unsupervised model for instance level subcategorization acquisition. In *EMNLP*, 2014.

S. Bengio and G. Heigold. Word embeddings for speech recognition. In *INTERSPEECH*, 2014.

E. Bruni, G. Boleda, M. Baroni, and N.-K. Tran. Distributional semantics in technicolor. In *ACL*, 2012.

K. Cho, B. van Merriënboer, Ç. Gülçehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio. Learning phrase representations using rnn encoder–decoder for statistical machine translation. In *EMNLP*, 2014.

Y.-A. Chung, C.-C. Wu, C.-H. Shen, H.-Y. Lee, and L.-S. Lee. Audio word2vec: Unsupervised learning of audio segment representations using sequence-to-sequence autoencoder. In *INTERSPEECH*, 2016.

M. Faruqui and C. Dyer. Community evaluation and exchange of word vectors at wordvectors.org. In *ACL: System Demonstrations*, 2014.

M. Faruqui, Y. Tsvetkov, P. Rastogi, and C. Dyer. Problems with evaluation of word embeddings using word similarity tasks. In *RepEval*, 2016.

D. Gerz, I. Vulić, F. Hill, R. Reichart, and A. Korhonen. Simverb-3500: A large-scale evaluation set of verb similarity. In *EMNLP*, 2016.

G. Halawi, G. Dror, E. Gabrilovich, and Y. Koren. Large-scale learning of word relatedness with constraints. In *KDD*, 2012.

D. Harwath and J. Glass. Deep multimodal semantic embeddings for speech and images. In *ASRU*, 2015.

D. Harwath and J. Glass. Learning word-like units from joint audio-visual analysis. In *ACL*, 2017.

D. Harwath, A. Torralba, and J. Glass. Unsupervised learning of spoken language with visual context. In *NIPS*, 2016.

W. He, W. Wang, and K. Livescu. Multi-view recurrent neural acoustic word embeddings. In *ICLR*, 2017.

F. Hill, R. Reichart, and A. Korhonen. Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*, 41(4):665–695, 2015.

S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.

H. Kamper, A. Jansen, and S. Goldwater. Fully unsupervised small-vocabulary speech recognition using a segmental bayesian model. In *INTERSPEECH*, 2015.

H. Kamper, W. Wang, and K. Livescu. Deep convolutional acoustic word embeddings using word-pair side information. In *ICASSP*, 2016.

H. Kamper, A. Jansen, and S. Goldwater. A segmental framework for fully-unsupervised large-vocabulary speech recognition. *Computer Speech and Language*, 46:154–174, 2017a.

H. Kamper, K. Livescu, and S. Goldwater. An embedded segmental k-means model for unsupervised segmentation and clustering of speech. In *ASRU*, 2017b.

I. Konstas, S. Iyer, M. Yatskar, Y. Choi, and L. Zettlemoyer. Neural amr: Sequence-to-sequence models for parsing and generation. In *ACL*, 2017.

K. Levin, K. Henry, A. Jansen, and K. Livescu. Fixed-dimensional acoustic embeddings of variable-length segments in low-resource settings. In *ASRU*, 2013.

M.-T. Luong, R. Socher, and C. D. Manning. Better word representations with recursive neural networks for morphology. In *CoNLL*, 2013.

T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *NIPS*, 2013.

G. A. Miller and W. G. Charles. Contextual correlates of semantic similarity. *Language and Cognitive Processes*, 6(1):1–28, 1991.

J. L. Myers and A. D. Well. *Research Design and Statistical Analysis*. Routledge, 1 edition, 6 1995.

V. Panayotov, G. Chen, D. Povey, and S. Khudanpur. Librispeech: an asr corpus based on public domain audio books. In *ICASSP*, 2015.

J. Pennington, R. Socher, and C. Manning. Glove: Global vectors for word representation. In *EMNLP*, 2014.

K. Radinsky, E. Agichtein, E. Gabrilovich, and S. Markovitch. A word at a time: computing word relatedness using temporal semantic analysis. In *WWW*, 2011.

H. Rubenstein and J. B. Goodenough. Contextual correlates of synonymy. *Communications of the ACM*, 8(10):627–633, 1965.

T. Schnabel, I. Labutov, D. Mimno, and T. Joachims. Evaluation methods for unsupervised word embeddings. In *EMNLP*, 2015.

S. Settle and K. Livescu. Discriminative acoustic word embeddings: Recurrent neural network-based approaches. In *SLT*, 2016.

I. Sutskever, O. Vinyals, and Q. V. Le. Sequence to sequence learning with neural networks. In *NIPS*, 2014.

S. Venugopalan, M. Rohrbach, J. Donahue, R. Mooney, T. Darrell, and K. Saenko. Sequence to sequence-video to text. In *CVPR*, 2015.

D. Yang and D. M. Powers. Verb similarity on the taxonomy of wordnet. In *GWC*, 2006.