LANGUAGE TRANSFER OF AUDIO WORD2VEC: LEARNING AUDIO SEGMENT REPRESENTATIONS WITHOUT TARGET LANGUAGE DATA

Chia-Hao Shen[1], *Janet Y. Sung*[2], *Hung-Yi Lee*[3]

National Taiwan University, Electrical Engineering Department, {r04921047,hungyilee}@ntu.edu.tw[1,3]
Harvard University, Design Engineering Department, jsung@mde.harvard.edu[2]

ABSTRACT

Audio Word2Vec offers vector representations of fixed dimensionality for variable-length audio segments using Sequence-to-sequence Autoencoder (SA). These vector representations are shown to describe the sequential phonetic structures of the audio segments to a good degree, with real world applications such as spoken term detection (STD). This paper examines the capability of language transfer of Audio Word2Vec. We train SA from one language (source language) and use it to extract the vector representation of the audio segments of another language (target language). We found that SA can still catch the phonetic structure from the audio segments of the target language if the source and target languages are similar. In STD, we obtain the vector representations from the SA learned from a large amount of source language data, and found them surpass the representations from naive encoder and SA directly learned from a small amount of target language data. The result shows that it is possible to learn Audio Word2Vec model from high-resource languages and use it on low-resource languages. This further expands the usability of Audio Word2Vec.

Index Terms— Audio Word2Vec, Spoken Term Detection, Seq2Seq, Autoencoder, Language Transfer

1. INTRODUCTION

Embedding audio word segments into fixed-length vectors has many applications in speech processing such as speaker identification, audio emotion classification, and spoken term detection (STD) [1–3]. In these application, audio segments are represented in fixed-length vectors instead of the original segments in variable lengths in order to reduce the effort for indexing, accelerate the speed of calculation, and improve the efficiency for the retrieval task [4, 5].

Existing works have shown the possibility to transform audio word segments into fixed dimensional vectors by deep learning [6, 7]. In [3], the authors used annotated data to train a LSTM network which located same words pair closer to each other. Human annotated data is required in the supervised setting [6–8]. To reduce the annotation effort, [9] proposed a LSTM-based sequence-to-sequence autoencoder, namely Audio Word2Vec, which used the last state of the RNN encoder as the representation of the audio segment.

Although deep learning approaches have produced satisfactory results, the data-hungry nature of the deep model makes it hard to produce the same performance with low-resource data. Both supervised and unsupervised approaches assume that a large amount of audio data of the target language is available. A question arises whether it is possible to transfer the Audio Word2Vec model learned from a high-resource language into a model targeted at a low-resource language. While this problem is not yet to be fully examined in Audio Word2Vec, works in neural machine translation (NMT) successfully transfer the model learned on high-resource languages to low-resource languages [10]. As for audio, all languages are uttered by human with common acoustic patterns, implying that the knowledge obtained from one spoken language can be transferred onto other languages.

This paper verifies that sequence-to-sequence autoencoder is not only able to transform audio word segments into fixed-length vectors, the model is also transferable to the languages it has never heard before. We also demonstrate its promising applications with a spoken term detection (STD) experiment. In this experiment, even without tunning with partial low-resource language segments, the autoencoder can still produce high-quality vector representations.

2. AUDIO WORD2VEC

The goal for Audio Word2Vec model is to extract the phonetic patterns in acoustic feature sequences such as Mel-scale Frequency Cepstral Coefficients(MFCCs). Given a sequence $\mathbf{x} = (x_1, x_2, ..., x_T)$ where x_t is the acoustic feature at time t, and T is the length, Audio Word2Vec transforms the features into fixed-length vector $\mathbf{z} \in \mathbb{R}^d$ with dimension d based on the phonetic structure.

Figure 1 depicts the structure of Sequence-to-sequence Autoencoder (SA), which integrates the RNN Encoder-Decoder [11] framework with Autoencoder [12] for unsupervised learning of audio segment representations. SA consists of an RNN Encoder (the left part of Figure 1) and an RNN Decoder (the right part). The RNN Encoder reads each acoustic feature x_t sequentially and the hidden state $\mathbf{h}_t$ is updated accordingly. After the last acoustic feature x_T has been read and processed, the hidden state $\mathbf{h}_T$ of the Encoder RNN is viewed as the *learned representation* $\mathbf{z}$ of the input

sequence (the purple block in Figure 1). The Decoder RNN takes $\mathbf{h}_T$ as the initial state of the RNN cell, and generates an output y_1. Instead of taking y_1 as the input of the next time step, a zero vector is fed in as input to generate y_2, and so on. This structure is called the historyless decoder, a weaken decoder to obtain better representation [13]. The RNN Encoder and Decoder are jointly trained by minimizing the reconstruction error, measured by the general mean squared error $\sum_{t=1}^{T} \|x_t - y_t\|^2$. Because the input sequence is taken as the learning target, the training process does not need any labeled data. The fixed-length vector representation $\mathbf{z}$ will be a meaningful representation for the input audio segment $\mathbf{x}$ because the whole input sequence $\mathbf{x}$ can be reconstructed from $\mathbf{z}$ by the RNN Decoder. While RNN is capable of capturing dynamic temporal information, it does not seem to learn long-term dependencies due to the vanishing gradient problem [14]. To better model long-term dependencies, LSTM [15] and GRU [16] were proposed and produced amazing results.

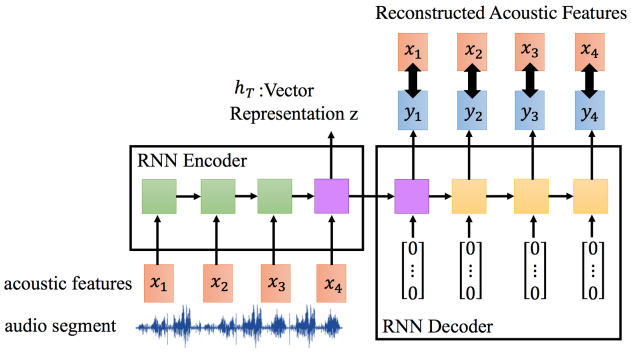


Fig. 1: Sequence-to-sequence Autoencoder (SA).

3. LANGUAGE TRANSFER

In the study of linguistic, scholars define a set of universal phonetic rules which describe how sounds are commonly organized across different languages. Actually, in real life, we often find languages sharing similar phonemes especially the ones spoken in nearby regions. These facts imply that when switching target languages, we do not need to learn the new audio pattern from scratch due to the transferability in spoken languages. Language transfer has shown to be helpful in STD [17]. In this paper, we focus on studying the capability of transfer learning of Audio Word2Vec.

We first train an SA using the high-resource source language, as shown in the upper part of Fig. 2, and then the encoder is used to transform the audio segment of a low-resource target language. It is also possible to fine-tune the parameters of SA with the target language. In the following experiments, we found that in some cases the performance of the encoder without fine-tuning with the low-resource target language can be as good as the one with fine-tuning.

4. APPLICATION: SPOKEN TERM DETECTION

The audio segment representation $\mathbf{z}$ learned in the last section can be applied in many possible scenarios. Here in the

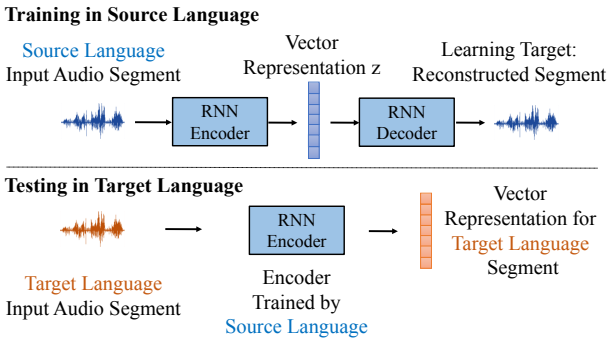


Fig. 2: Language Transfer Mechanism.

preliminary tests we consider the query-by-example spoken term detection (QbE-STD), whose target is to locate the occurrence regions of the input spoken query term in a large spoken archive without speech recognition. Figure 3 shows how the representation $\mathbf{z}$ proposed here can be easily used in this task. This approach is inspired from the previous work [4], but completely different in the ways to represent the audio segments. In the upper half of Figure 3, the audio archive are segmented based on word boundaries into variable-length sequences, and then the system exploits the trained RNN encoder in Figure 1 to encode these audio segments into fixed-length vectors. All these are done off-line. In the lower left corner of Figure 3, when a spoken query is entered, the input spoken query is similarly encoded by the same RNN encoder into a vector. The system then returns a list of audio segments in the archive ranked according to the cosine similarities evaluated between the vector representation of the query and those of all segments in the archive. Note that the computation requirements for the online process here are extremely low.

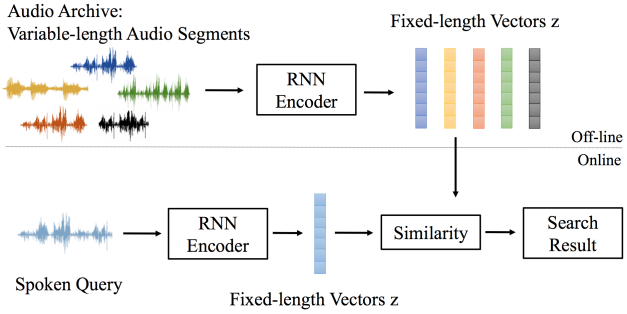


Fig. 3: Spoken Term Detection Application.

5. EXPERIMENTAL SETUP

5.1. Dataset

Two corpora, LibriSpeech [18] and GlobalPhone [19], across five languages were used in the experiment as shown in table1. We used the English LibriSpeech corpus as the high-resource source language and the GlobalPhone corpus, which consists of French(FRE), German(GER), Czech(CZE), and Spanish(ESP), as the low-resource target languages. 39-dim MFCCs were used as the acoustic features with a sequence limit of 50 frames. All datasets were segmented according to the word boundaries obtained by forced alignment with respect to the reference transcriptions. Although the oracle

word boundaries were used here for the QbE-STD in the preliminary tests, the comparison in the following experiment was fair since all approaches used the same segmentation.

Table 1: Number of segments used for training or fine-tuning, STD database, and STD query in each corpus. For GlobalPhone, the amount shown is retrieved from each language.

Corpus	Training or Fine-Tuning	Database	Query
LibriSpeech	22 Million	250000	1000
GlobalPhone	2000	20000	1000

5.2. Models

Both the proposed model (SA) and baseline model (NE) were implemented with Tensorflow. The network structure and the hyper parameters were set as below:

- Both RNN Encoder and Decoder consisted one hidden layer of GRU cells [16]. The number of units in the layer would be discussed in the experiment.
- The networks were trained by SGD with gradient clipping. The initial learning rate was in range of [0.01, 1] and decayed with a factor of 0.95 every 500 batches.

The high-resource language and low-resource languages were trained using the same setting. Naive encoder (NE) is used as the baseline approach. In this encoder, the input acoustic feature sequence of the 39-dimension MFCC, $\mathbf{x}$, was divided into m partitions with roughly equal length T/m. Then, we averaged each partition and concatenating the average vectors sequentially into a vector representation of dimensionality $39 \times m$. Although NE is simple, similar approaches have achieved successful results in STD [20].

6. EXPERIMENTS

6.1. Analysis on Dimension of Audio Word2Vector

We first experimented on the primary SA model in the source language (English). The results are shown in Table 2. Besides, comparing SA and NE, we examined the influence of the dimension of Audio Word2Vector on the mean average precision (MAP). We also compared the MAP results on large testing database (250K segments) and small database (20K).

In Table 2, we varied the dimension of Audio Word2Vector as 100, 200, 400, 600, 800 and 1000. To match up the dimensionality with SA, we tested NE with dimensionality 117, 234, 390, 585, 819, 1014 ($m = 3, 6, 10, 15, 21, 26$). We denoted the terms NE_d and SA_d where d is the dimensionality.

SA gets higher MAP values than NE no matter the vector dimension and the size of database. Also, both SA and NE greatly surpass Dynamic Time Warping(DTW) [21] approach. The highest MAP score SA can achieve is 0.881 (SA_{800} on small database), while the highest score of the NE model is 0.490 (NE_{234} on small database). The MAP scores of the two models both drop in the large database. For example, NE_{234} drops from 0.490 to 0.158, decaying by 68%, and the performance of SA_{800} drops from 0.881 to 0.317, decaying by 64%. As shown in Table 2, larger dimensionality does not imply better performance in QbE-STD. The MAP scores gradually improve until reaching the dimensionality of 400 in SA and 234 in NE, and start to decrease as the dimension increases. In the rest of the experiments, we would use 400 GRU units in the SA hidden layer, and set $NE = NE_{234}$ ($m = 6$).

Table 2: Retrieval Performance in MAP. Dim is the dimension of the vector representation.Small DB is the small database with 20000 examples, Large DB is the large database with 250000 examples

Dim	AE	100	200	400	800	1000
	NE	117	234	390	819	1014
Small DB	DTW	0.173				
	NE	0.390	0.490	0.484	0.351	0.325
	AE	0.731	0.685	0.737	**0.881**	0.713
Large DB	NE	0.100	0.158	0.169	0.092	0.091
	AE	0.234	0.307	**0.400**	0.317	0.233

6.2. Analysis of Language Transfer

We trained the Audio Word2Vec model by SA from the source language, English, and applied it on different target languages, French (FRE), German (GER), Czech (CZE), and Spanish (ESP). We computed the average cosine similarity of the vector representations for each pair of the audio segments in the retrieval database of the target languages (20K segments for each language), and compare it with the phoneme sequence edit distance (PSED) [9]. The average and variance of the cosine similarity for groups of pairs clustered by the phoneme sequence edit distances (PSED) between the two words are shown in Table 3. We also provide the results from the English retrieval database (250K segments), where the segments were not seen by the model in training procedure.

In Table 3, the cosine similarities of the segment pairs get smaller as the edit distances increase, and the trend is observed in all languages. The score differences from PSED=n to PSED=$n+1$, $n = 0, 1, 2, 3$, is obvious. This means that SA learned from English can successfully encode the sequential phonetic structures into fixed-length vector for the target languages to some good extend even though *it has never seen any audio data of the target languages*.

In the source language, English, the variances of the five edit distance groups are fixed at 0.030, meaning that the cosine similarity in each group is centralized. However, the variances of the groups in the target languages vary. In French and German, the variance grows from 0.030 to 0.060 as the edit distance increases from 0 to 4. For Czech/Spanish, the variance starts at a larger value of 0.040/0.050 and increases to 0.050/0.073. We suspect the fluctuating variance is due to the similarity between languages [22].

6.3. Visualization

To further investigate the performance of SA, we visualize the vector representation of two sets of word pairs differing by only one phoneme from French and German as below:

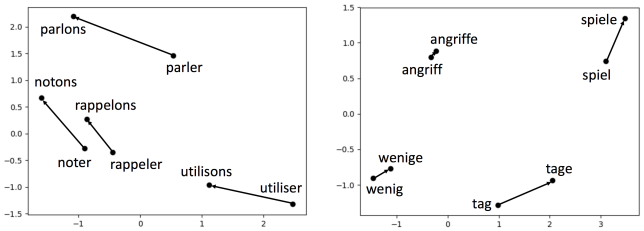
1. French Word Pairs: (parler, parlons), (noter,notons), (rappeler, rappelons), (utiliser, utilisons)

Table 3: The average(μ)/variance(σ^2) of the cosine similarity between vector representations for all segment pairs in the target languages testing set, clustered by the phoneme sequence edit distances (PSED).

		ENG	GER	FRE	CZE	ESP
PSED=0	μ	0.297	0.432	0.419	0.378	0.348
	σ^2	0.033	0.033	0.037	0.042	0.056
PSED=1	μ	0.164	0.328	0.306	0.211	0.168
	σ^2	0.039	0.046	0.048	0.050	0.073
PSED=2	μ	0.055	0.18	0.028	0.061	0.028
	σ^2	0.039	0.056	0.055	0.048	0.064
PSED=3	μ	0.009	0.077	0.017	0.021	0.017
	σ^2	0.039	0.056	0.058	0.042	0.053
PSED>3	μ	-0.013	-0.012	0.005	0.012	0.005
	σ^2	0.037	0.039	0.042	0.043	0.050

2. German Word Pairs: (tag, tage), (spiel, spiele), (wenig, wenige), (angriff, angriffe)

To show the vector representations in Fig. 4, we first obtained the mean value of representations for different audio segments of a specific word, denoted by δ(word). Then the average representation δ was projected from 400-dimensional to 2-dimensional using PCA [23]. The result of the difference vector from each word pair, e.g. δ(parlons) - δ(parler), is shown. Although the representations for French and German word audio segments were extracted from the model trained by English audio word segments, the direction and magnitude of the different vectors are still coherent. In term of magnitude, words differ in more phonemes had larger distance between the two vectors. As for direction, changing from a specific phoneme to the other in different word pairs shifted the original word vector in the same direction. The magnitude property is shown previously in Table 3. As for the direction property, in Fig. 4a, δ(parlons) - δ(parler) shifted at the same direction as δ(utilison) - δ(utiliser); and δ(tage) - δ(tag) shifted at the same direction as δ(wenige) - δ(wenig) in Fig. 4b.



(a) French: last phoneme changes from 'er' to 'ons'. (b) German: ending with 'e' or not.

Fig. 4: Difference between average vectors for word pairs differing by one edit distance in (a) French and (b) German.

6.4. Language Transferring on STD

Besides analyzing the cosine similarity of the learned representations, we also apply them to the QbE-STD task. Here we compare the retrieval performance in MAP of SA with different levels of accessibility to the low-resource target language along with two baseline models, NE and SA trained purely by the target languages. For the four target languages, the total available amount of audio word segments in the training set were 2 thousands for each language. In Table 4, we took different partitions of the target language training sets to fine tune the SA pretrained by the source languages. The amount of audio word segments in these partitions are: 1K, 2K and without fine-tuning.

From Table 4, SA trained by source language generally outperforms the SA trained by the limited amount of target language ("SA No Transfer"), proving that with enough audio segments, SA can identify and encode universal phonetic structure. Comparing with NE, SA surpasses NE in German and French even without fine-tuning, whereas in Czech, SA also achieves better score than NE with fine-tuning. However, in Spanish, SA achieved a MAP score of 0.13 with fine-tuning, slightly lower than 0.17 obtained by NE. Back to Table 3, the gap between phoneme sequence edit distances 2 and 3 in Spanish is smaller than other languages. Also, as discussed earlier in Section 6.2, the variance in Spanish is also bigger. The smaller gap and bigger variance together indicate that the model is weaker on Spanish at identifying audio segments of different words and thus affects the MAP performance in Spanish.

Table 4: The retrieval performance of NE, SA trained by the target language only (denoted as SA No Transfer), and SA of the source language tuning with different amounts of data.

		FRE	GER	CZE	ESP
NE		0.22	0.18	0.09	**0.17**
SA No Transfer		0.03	0.01	0.00	0.00
SA	No Fine-Tuning	**0.26**	0.24	0.06	0.04
	1K	0.24	0.20	0.09	0.13
	2K	**0.26**	**0.25**	**0.10**	0.12

7. CONCLUSION AND FUTURE WORK

In this paper, we verify the capability of language transfer of Audio Word2Vec using Sequence-to-sequence Autoencoer (SA). We demonstrate that SA can learn the sequential phonetic structure commonly appearing in human language and thus make it possible to apply an Audio Word2Vec model learned from high-resource language to low-resource languages. The capability of language transfer in Audio Word2Vec is beneficial to many real world applications, for example, the query-by-example STD shown in this work. For the future work, we are examining the performance of the transferred system in other application scenarios, and exploring the performance of Audio Word2Vec under automatic segmentation.

8. REFERENCES

[1] Najim Dehak, Reda Dehak, Patrick Kenny, Niko Brummer, Pierre Ouellet, and Pierre Dumouchel, "Support vector machines versus fast scoring in the low-dimensional total variability space for speaker verification," in *INTERSPEECH*, 2009.

[2] A. Norouzian, A. Jansen, R. Rose, and S. Thomas, "Exploiting discriminative point process models for spoken term detection," in *INTERSPEECH*, 2012.

[3] Shane Settle, Keith Levin, Herman Kamper, and Karen Livescu, "Query-by-example search with discriminative neural acoustic word embeddings," *arXiv preprint arXiv:1706.03818*, 2017.

[4] Keith Levin, Aren Jansen, and Benjamin Van Durme, "Segmental acoustic indexing for zero resource keyword search," in *ICASSP*, 2015.

[5] Herman Kamper, Weiran Wang, and Karen Livescu, "Deep convolutional acoustic word embeddings using word-pair side information," in *ICASSP*, 2016.

[6] Herman Kamper, Weiran Wang, and Karen Livescu, "Deep convolutional acoustic word embeddings using word-pair side information," in *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*. IEEE, 2016, pp. 4950–4954.

[7] Wanjia He, Weiran Wang, and Karen Livescu, "Multi-view recurrent neural acoustic word embeddings," *arXiv preprint arXiv:1611.04496*, 2016.

[8] Guoguo Chen, Carolina Parada, and Tara N Sainath, "Query-by-example keyword spotting using long short-term memory networks," in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*. IEEE, 2015, pp. 5236–5240.

[9] Yu-An Chung, Chao-Chung Wu, Chia-Hao Shen, Hung-Yi Lee, and Lin-Shan Lee, "Audio word2vec: Unsupervised learning of audio segment representations using sequence-to-sequence autoencoder," in *INTERSPEECH*, 2016, pp. 765–769.

[10] Barret Zoph, Deniz Yuret, Jonathan May, and Kevin Knight, "Transfer learning for low-resource neural machine translation," *arXiv preprint arXiv:1604.02201*, 2016.

[11] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," *arXiv preprint arXiv:1406.1078*, 2014.

[12] Geoffrey E Hinton and Ruslan R Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.

[13] Stanislau Semeniuta, Aliaksei Severyn, and Erhardt Barth, "A hybrid convolutional variational autoencoder for text generation," *arXiv preprint arXiv:1702.02390*, 2017.

[14] Yoshua Bengio, Patrice Simard, and Paolo Frasconi, "Learning long-term dependencies with gradient descent is difficult," *Neural Networks, IEEE Transactions on*, vol. 5, no. 2, pp. 157–166, 1994.

[15] Sepp Hochreiter and Jürgen Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[16] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *arXiv preprint arXiv:1412.3555*, 2014.

[17] Haipeng Wang, Tan Lee, Cheung-Chi Leung, Bin Ma, and Haizhou Li, "Using parallel tokenizers with DTW matrix combination for low-resource spoken term detection," in *ICASSP*, 2013.

[18] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur, "Librispeech: an asr corpus based on public domain audio books," in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*. IEEE, 2015, pp. 5206–5210.

[19] Tanja Schultz, "Globalphone: a multilingual speech and text database developed at karlsruhe university.," in *INTERSPEECH*, 2002.

[20] Hung-Yi Lee and Lin-Shan Lee, "Enhanced spoken term detection using support vector machines and weighted pseudo examples," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 21, no. 6, pp. 1272–1284, 2013.

[21] Eamonn Keogh and Chotirat Ann Ratanamahatana, "Exact indexing of dynamic time warping," *Knowledge and Information Systems*, vol. 7, no. 3, pp. 358–386, Mar 2005.

[22] M. Paul Lewis, Ed., *Ethnologue: Languages of the World*, SIL International, Dallas, TX, USA, sixteenth edition, 2009.

[23] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean, "Distributed representations of words and phrases and their compositionality," in *NIPS*, 2013.