A Better Phone Set for the TIMIT Dataset Discovered in Clustering of Listen, Attend and Spell

Anonymous Author(s)
Affiliation
Address
`email`

Abstract

Listen, Attend and Spell(LAS)[4] maps a sequence of acoustic spectra directly to a sequence of graphemes, with no explicit internal representation of phones. This paper asks whether LAS can be used as a scientific tool, to discover the phone set of a language whose phone set may be controversial or unknown. Phonemes have a precise linguistic definition, but phones may be defined in any manner that is convenient for speech technology: we propose that a practical phone set is one that can be inferred from speech following certain procedures, but that is also highly predictive of the word sequence. We demonstrate that such a phone set can be inferred by clustering the hidden node activation vectors of an LAS model during training, thus encouraging the model to learn a hidden representation characterized by acoustically compact clusters that are nevertheless predictive of the word sequence. We further define a metric for the quality of a phone set (sum of conditional entropy of the phone set given graphemes, and given acoustics), and demonstrate that according to this metric, the clustered-LAS phone set is better than the original TIMIT[5] phone set.

1 Introduction

Traditional automatic speech recognition(ASR) usually are composed of multiple components including an acoustic model, a language model, a pronunciation dictionary, and other possible elements. Recently, modern ASR models implemented based on neural network, such as connectionist temporal classification (CTC)[6] and LAS, accomplished directly speech-to-text with large successes. Since such models generally are not dependent on utilizing specific language models or pronunciation dictionaries, their simple architectures are popular with new researchers trying to enter the speech recognition community. The typical neural-network based models rely on the Recurrent Neural Networks(RNNs), and the key to success of utilizing such deep learning mechanism is to discover hidden representation of the training data.

In this work, we take a step further to explore the possibility of defining a new phone set for the TIMIT dataset using the LAS model by incorporating a clustering method to soft align acoustics and graphemes. In LAS model, the Listener takes the input acoustic signals and encodes the signals to a hidden nodes vector, and then the hidden nodes vector feeds into the Speller to generate transcripts. Since the hidden node vector represents the relationship between the words in transcripts and acoustic signals, we cluster these hidden nodes with corresponding trigraphs in the transcripts. In this way, we train the model to learn the underlying relationship between the trigraphs and acoustics.

The clustering pairs of hidden nodes and corresponding graphemes are the new defined phone set. We evaluate the new phone set by using an entropy utility function, the sum of the conditional entropy of different contexts given the phone set, and the contexts here refer to both graphemes and acoustics. The experiment reveals that the new phone set discovered by the experiment model better represents the TIMIT dataset under this metric.

Submitted to 32nd Conference on Neural Information Processing Systems (NIPS 2018). Do not distribute.

2 Related Work

2.1 Machine learning and attention in ASR

In recent years, machine learning has been extensively researched and applied to many aspects of studies. In the field of ASR, successful models such as CTC, LAS, and [8, 10], and other architectures take advantages of incorporating RNNs. Among these models, CTC and LAS don't require input segmentation and post-processed outputs. CTC generates the labels of sequence of data with RNNs based on the probability distribution given the input sequence during input time steps, whereas LAS neglects input time steps and generates output characters at each output time step using sequence-to-sequence attention mechanism given the transformed hidden nodes vector from input acoustics. Sequence-to-sequence attention mechanism are widely used, and studies such as [9, 2] demonstrate the successes of attention mechanism. These two essential mechanisms greatly benefit current ASR models.

2.2 Representations between acoustics and text in speech recognition models

Deep learning works if and only if it's able to find an accurate hidden representation of training data, thereby enabling the system to learn the relationship between the input signal and output words. For conventional ASR, phone recognition and phone segmentation are two important tasks since phone is the smallest temporal unit in speech and serves as an intermediate representation connecting speech and text. Hidden Markov Model(HMM) capture acoustic signal features and decompose vocabulary to context-independent phones [16]. Hybrid HMM-DNN systems use the DNN to compute phone likelihoods, and the HMM to compute phone alignment[11]. Belinkov and Glass [3] investigates the hidden representations of Deep Speech 2[1], and the study shows that the phonetic information loss gradually increases from the bottom layer to the top layer.

3 Model

3.1 Brief descriptions of LAS model

LAS is an end-to-end speech recognition model that generates the transcripts directly from input acoustic signals without the implementations of multiple submodules of traditional ASR. The basic LAS model includes two modules: a Listener and a Speller. The Listener composes a three layer of pyramidal Bidirectional Long Short Term Memory(pBLSTM)[7], which encodes the input acoustics signals and reduces the input time length to one-eighth of the original. The output of the Listener is represented by hidden nodes vectors $\mathbf{h}$, then the vectors are fed into the Speller as the input. The Speller is a sequence-to-sequence attention-based LSTM transducer. The attention mechanism of the transducer takes the hidden nodes vectors from the Listener and character distribution from previous step as input and generates a context vector for all attention probabilities of hidden nodes vectors for the current step. The context vector is then used to generate the output character at the current step.

3.2 Experiment LAS model

The Figure 1 shows the overall modified LAS model of the experiment. We are aiming at finding the hidden relationship between the input acoustic signals and output transcripts, so we introduce a clustering component in the original LAS model to encourage the Listener to learn a hidden representation in which frames are grouped into compact clusters. Specifically, for each character correctly inferenced by the LSTM transducer from the Speller, we cluster the corresponding maximally attended hidden nodes. The hidden nodes vectors are a cumulative nonlinear transformation of the Mel-Frequency Cepstral Coefficients(MFCCs), and are trained to optimally summarize whatever information about the MFCC is necessary for the Speller to correctly generate output characters. By clustering maximum attended hidden nodes, we force the system to learn groupings of speech frames that have similar hidden node vectors and are also connected to similar output character sequences.

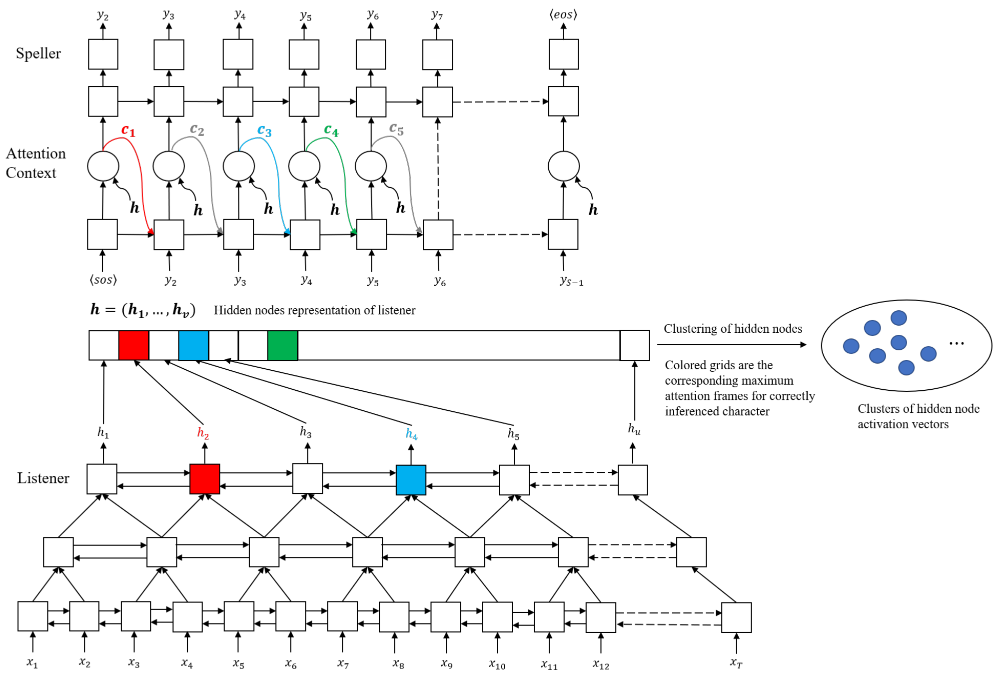

Figure 1: Modified architecture of Listen, attend and spell

The input of the clustering algorithm are the corresponding hidden nodes of the maximum attention frames generated by the AttentionContext vector for correctly inferenced character from the Speller. In this figure, y_2, y_4, and y_5 are correctly inferenced characters, and their corresponding context vectors of $\mathbf{c_1}$, $\mathbf{c_3}$, and $\mathbf{c_4}$ generate the attention vectors whose maximally attended input frames are h_2, h_4, and h_7 respectively. Thus, h_2, h_4, and h_7 are the input of the cluster.

3.3 Learning

The modified LAS system can be trained jointly for accurate character output, but also for optimally clustered internal hidden node vectors. The training criterion of the modified LAS model contains two parts: word loss and clustering loss. The loss function can be described as the following,

$$\varepsilon = \text{Edit Distance}(y, \hat{y}) + \sum_t ||\mathbf{h_t} - \mu_{\mathbf{k(t)}}||^{\mathbf{2}}$$

The first part of the training criterion of the system is the edit distance between the reference transcripts generated transcripts, and is the error measure used in the standard LAS algorithm; y and $\hat{y}$ refer to the reference character and generated character respectively. The second part is the squared distance between each hidden nodes with the centroids of their clusters for the duration of the input sentence. $\mathbf{h_t}$ is the hidden nodes vector of the input, $\mu_{k(t)}$ is the corresponding cluster of the hidden nodes vector. By minimizing this error function, we encourage the Listener to learn a hidden embedding, $\mathbf{h_t}$, that is useful in predicting the output character y_t, but that can also be clustered into compact phone-like clusters with centroids μ_k.

3.4 Clustering method

The clustering method in the modified LAS model has almost the same mechanism of k-means clustering except that the input varies after every step since the hidden nodes change for every batch during training stage. The objective of the clustering method is to minimize the clustering loss from the loss function.

The centroids of the clusters are randomly initialized with normal distribution. The hidden nodes are clustered and labeled for a certain number of iterations for every epoch. Then the centroids are updated and kept for the next epoch. After every epoch, the empty clusters that are never labeled for the past epoch will be deleted and replaced by splitting the largest labeled clusters by scaling the original centroids 0.01 and 0.99 of the original clustering centroids.

4 Experiment

4.1 Dataset descriptions

Two datasets are used to perform the experiment. English speech recognition training corpus of TED-LIUMv2(TEDLIUM)[15] is used to pre-train the LAS model. The TEDLIUM dataset was made from audio talks and transcripts from TED website. There are 1495 audio talks with aligned transcripts in the dataset.
The TIMIT dataset is used to train the experiment LAS model. The TIMIT dataset comes with its self-defined dictionary and phoneme alignment transcripts for the audio talks. During training, audio and transcripts of one female and one male are selected from each dialect region of the test dataset as development set, and the rest of the test dataset remains as test set. A new phone set is discovered for the TIMIT dataset and compared with the reference phone set.

4.2 Preprocessing of dataset

Both dataset are preprocessed using MFCCs algorithm. The raw acoustic signals in the dataset are framed by 10ms each, and the sampling rate of the input signals is 16000 Hz. The power spectrum is calculated for each frame by using periodogram estimate, the squared magnitude of Discrete Fourier Transform(DFT) of original acoustic signals. Forty filters in Mel-spaced filterbank are applied to the power spectrum, and log filterbank energies are computed by taking the log of the power spectrum. Then Discrete Cosine Transform(DCT) of these forty log filterbank energies give output of the cepstrum coefficients.

4.3 Experimental settings

The implementation of the basic LAS model is based on the toolkit eXtensible Neural Machine Translation(XNMT)[13] using Dynet framework[12]. The learning rate of the Adam optimizer is initialized to 0.01 and reduced to half of the original learning rate if WER of development set isn't improved after 3 epochs. Other parameters are consistent as indicated by the original paper. The hidden dimension of pLSTM is 512, which is the dimension of the hidden nodes vector. The Attender has hidden dimension 128. The dropout rate of the entire neural network is 0.3. The Speller uses a beam search with size 20 is used to infer test transcripts.
The experiment model is modified by introducing a new clustering module. The LAS model has been pre-trained for about 300 epochs. Starting with the pre-trained model, the experiment model is then trained for 100 epochs; the learning rate of the Adam optimizer remains at 0.03. The number of iterations for each clustering step is 20, and the dimension of each cluster centroid is the same as the dimension of the hidden nodes vector, which is 512 in this case. The number of clusters is 100, which is roughly twice as the number of English phonemes.

5 Results and discussions

5.1 Error measurements of experiment LAS model

Upon convergence, the pre-trained model of LAS has word error rate(WER) 16.72% and character error rate(CER) 8.46% on the test dataset. With the pre-trained LAS model, the experiment model has the final WER 26.99% and CER 10.67%. By the training criterion, clustering loss is 0.359 and maximum likelihood estimation loss is 0.948.

5.2 Comparisons of new discovered phone set and reference phone set

For the experimental LAS model, 100 clusters are used to discover a new phone set for TIMIT. For all generated transcripts of test set, every character in the transcripts is assigned to a cluster using the k-means cluster criterion, and the closest μ_k of the corresponding hidden node $\mathbf{h_t}$ is the one that each character assigned to. The top five most frequently assigned trigraphs for each cluster vote to determine the phone label of each cluster from phones in the TIMIT dataset. There are certainly some ambiguous cases when we try to identify the phones for one cluster. For example, the top five most frequently labeled trigraphs for one cluster are " wh", "ere", "whe", "wer", and " we". The cluster

Table 1: List of both phone sets discovered by experiment model and reference dictionary

phone categories	phone set of experiment model	phone set of reference dictionary
stops	b, d, g, k, p, t	b, d, g, k, p, t
affricates	ch, jh	ch, jh
fricatives	f, s, sh, th, v, z	**dh**, f, s, sh, th, v, z, **zh**
nasals	en, m, n, ng	**em**, en, **eng**, m, n, ng
semivowels and glides	el, hh, **hv**, r, w, l	el, hh, l, r, w, y
vowels	ae, ao, ax, axr, aw, ay, eh, er, ey, ih, ix, iy, ow, oy, uw	**aa**, ae, **ah**, ao, aw, ax, axr, ay, eh er, ey, ih, ix, iy, ow, oy, **uh**, uw
non-speech event	h#	h#

Table 2: Entropy of the distribution $P(\text{phones}|\text{graphemes})$ and $P(\text{phones}|\text{acoustics})$ for both experiment and reference phone sets

System	$H(\text{phones}|\text{graphemes})$	$H(\text{phones}|\text{acoustics})$
Experiment	0.0212	0.00117
Reference	0.0242	0.00136

certainly captures the similar pronunciations of the word "where", but the word may not be described using one single phoneme. In such cases, phone labels of clusters are edited by hand. For this cluster, we assign the phone label as "w". The final phone set discovered by the clusters of the experimental system contain 40 unique phones.
Since the reference transcripts of the TIMIT dataset contain the actual pronunciations of the phones, the phones of the transcripts are very different from the ones used in the TIMIT dictionary. In order to find a phone set that represents the reference transcripts, we utilize the function phonetisaurus-align in toolkit Phonetisaurus G2P[14] to generate the alignment between each character and the corresponding phone of the reference transcript. The stress markers of the TIMIT dictionary are eliminated. The reference phone set contain 46 unique phones.
The experiment and reference phone sets are as shown in the table1.

5.3 Entropy measurement

Entropy is commonly used to measure the randomness or disorder of a system. The output of the experiment model is evaluated by calculating the conditional entropy given different contexts for experiment and reference set of phones. The contexts include both graphemes and acoustics. The graphemes constitute of all possibilities of trigraphs in English, and the acoustics include all unique frames in the test dataset from TIMIT corpus. The conditional entropy is calculated as the following,

$$H(\text{phones}|\text{contexts}) = -\sum_{x \in \text{contexts}} p(x) \sum_{y \in \text{phones}} p(y|x) \log p(y|x)$$

Specifically, for calculating $H(\text{phones}|\text{graphemes})$, all possibilities of trigraphs are considered as the contexts of graphemes. The possibilities of trigraphs are calculated as all of length three permutations of 26 English letters and special tokens "’" and "-" appeared in both generated and reference transcripts.

$$p(x) = \frac{\text{number of occurrences for trigraph } x \text{ in transcripts}}{\text{total number of trigraphs in transcripts}}$$

$$p(y|x) = \frac{\text{number of occurrences for phone } y \text{ given trigraph } x \text{ in transcripts} + k}{\text{total number of trigraphs in transcripts} + k \times \text{number of phones in defined phone set}}$$

Similarly, for calculating $H(\text{phones}|\text{acoustics})$, unique acoustic frame is sorted out by measuring squared differences of all acoustic frames among each other in test dataset. Within tolerance of 1, similar acoustic frames are treated as the same frame in calculation. Thus, the prior acoustic distribution can be approximated by

$$p(x) \approx \frac{\text{number of occurrences of acoustic frame } x}{\text{total number of acoustic frames in test set}}$$

$$p(y|x) = \frac{\text{number of occurrences for phone } y \text{ given acoustic frame } x + k}{\text{total number of acoustic frames in test data} + k \times \text{number of phones in defined phone set}}$$

Laplace smoothing is applied for both conditional entropy calculations with the smoothing factor $k = 1$.

From Table2, the sum of the conditional entropy given both contexts of experiment phone set is less than that of the reference phone set, since $0.0212 + 0.00117 = 0.02237 < 0.0242 + 0.00136 = 0.02556$.

6 Conclusions

We defined a new phone set for the TIMIT dataset based on incorporating the clustering mechanism into the original LAS model. The learning criterion for the experiment model is composed of two parts: edit distance between generated transcripts and reference transcripts and squared distance between clustered hidden nodes and corresponding centroids of clusters. The learning criterion balances the learning objectives of the system – reducing the WER of generated transcripts meanwhile grouping the hidden vectors into compact clusters. The model is pre-trained with a larger dataset, TEDLIUM, and then trained on the TIMIT dataset for the experiment. The experiment result is evaluated by defining a utility function, the sum of the conditional entropy of graphemes given phones and the conditional entropy of acoustics given phones. We showed that the experiment phone set has both lower "grapheme entropy" and "acoustic entropy". Thus, we can claim that the phone set discovered by the experiment is better than the reference phone set in the TIMIT dataset based on this criterion.

Acknowledgments

We thank for everyone who provides insightful discussions with us for conducting research.

References

[1] D. Amodei, S. Ananthanarayanan, R. Anubhai, J. Bai, E. Battenberg, C. Case, J. Casper, B. Catanzaro, Q. Cheng, G. Chen, J. Chen, J. Chen, M. Chen, Z.and Chrzanowski, A. Coates, G. Diamos, K. Ding, N. Du, E. Elsen, J. Engel, W. Fang, L. Fan, C. Fougner, L. Gao, C. Gong, A. Hannun, T. Han, L. Johannes, B. Jiang, C. Ju, B. Jun, P. Legresley, L. Lin, J. Liu, Y. Liu, W. Li, X. Li, D. Ma, S. Narang, A. Ng, S. Ozair, Y. Peng, R. Prenger, S. Qian, Z. Quan, and Raima. Deep speech 2: End-to-end speech recognition in English and Mandarin. In *33rd International Conference on Machine Learning, ICML 2016*, number 33rd International Conference on Machine Learning, ICML 2016, pages 312–321, (1)Baidu Silicon Valley AI Lab, 2016.

[2] D. Bahdanau, J. Chorowski, D. Serdyuk, P. Brakel, and Y. Bengio. End-to-end attention-based large vocabulary speech recognition. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4945–4949, Mar 2016.

[3] Y. Belinkov and J. Glass. Analyzing hidden representations in end-to-end automatic speech recognition systems. In *Proceedings of Advances in Neural Information Processing Systems 30 (NIPS 2017)*, 2017.

[4] W. Chan, N. Jaitly, Q. V. Le, and O. Vinyals. Listen, attend and spell: A neural network for large vocabulary conversational speech recognition. In *2016 IEEE International Conference On Acoustics, Speech And Signal Processing (ICASSP)*, pages 4960–4964, May 2016.

[5] J. Garofolo, L. Lamel, W. Fisher, D. P. Jonathan Fiscus, N. Dahlgren, and V. Zue. TIMIT acoustic-phonetic continuous speech corpus LDC93S1, 1993. Web Download.

[6] A. Graves, S. Fernández, and F. Gomez. Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the International Conference on Machine Learning, ICML 2006*, pages 369–376, 2006.

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

[8] T. Hori, S. Watanabe, Y. Zhang, and W. Chan. Analyzing hidden representations in end-to-end automatic speech recognition systems. In *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH, 2017*, pages 949–953, 2017.

[9] S. Kim, T. Hori, and S. Watanabe. Joint ctc-attention based end-to-end speech recognition using multi-task learning. In *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, pages 4835–4839, 2017.

[10] Y. Miao, M. Gowayyed, and F. Metze. Eesen: End-to-end speech recognition using deep RNN models and wfst-based decoding. In *2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pages 167–174, Dec 2015.

[11] N. Morgan and H. Bourlard. Continuous speech recognition. In *IEEE Signal Processing Magazine IEEE Signal Process. Mag. Signal Processing Magazine, IEEE.*, volume 12, May 1995.

[12] G. Neubig, C. Dyer, Y. Goldberg, A. Matthews, W. Ammar, A. Anastasopoulos, M. Ballesteros, D. Chiang, D. Clothiaux, T. Cohn, K. Duh, M. Faruqui, C. Gan, D. Garrette, Y. Ji, L. Kong, A. Kuncoro, G. Kumar, C. Malaviya, P. Michel, Y. Oda, M. Richardson, N. Saphra, S. Swayamdipta, and P. Yin. Dynet: The dynamic neural network toolkit. *arXiv preprint arXiv:1701.03980*, 2017.

[13] G. Neubig, M. Sperber, X. Wang, M. Felix, A. Matthews, S. Padmanabhan, Y. Qi, D. S. Sachan, P. Arthur, P. Godard, J. Hewitt, R. Riad, and L. Wang. XNMT: The extensible neural machine translation toolkit. In *Conference of the Association for Machine Translation in the Americas (AMTA) Open Source Software Showcase*, Boston, March 2018.

[14] J. Novak, P. Dixon, N. Minematsu, K. Hirose, C. Hori, and H. Kashioka. Improving WFST-based G2P conversion with alignment constraints and RNNLM n-best rescoring. In *Interspeech*, 2012.

[15] A. Rousseau, P. Deléglise, and Y. Estève. Enhancing the TED-LIUM corpus with selected data for language modeling and more TED talks. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, May 2014.

[16] T. Vaich and A. Cohen. HMM phoneme recognition with supervised training and viterbi algorithm. In *Eighteenth Convention of Electrical and Electronics Engineers in Israel Electrical and Electronics Engineers in Israel, 1995.*, Mar 1995.