End-to-End Accented Mandarin Recognition with Hybrid CTC/Attention Architecture (Post-print)

Authors: Yang Wei, Hu Yan

Date: 2020-09-28T00:00:00+00:00

Abstract

To address the problem of multi-accent recognition in Mandarin speech recognition tasks, we propose a hybrid end-to-end model combining connectionist temporal classification (CTC) and MultiHead attention, along with multi-objective training and joint decoding methods. Experimental analysis reveals that with decreasing CTC weight and deeper encoder layers in the hybrid architecture, the hybrid model demonstrates better learning capability on accented datasets. Additionally, training a network with an encoder-decoder architecture reaching a depth of 48 layers yields a model that outperforms all previous end-to-end models, achieving a 5.6% character error rate and a 26.2% sentence error rate on the 200-hour accented dataset open-sourced by Datatang. Experiments demonstrate that the end-to-end model proposed in this paper surpasses the recognition accuracy of general end-to-end models and exhibits certain advancement in solving accented Mandarin recognition.

Full Text

Preamble

Vol. 38 No. 3
Application Research of Computers
ChinaXiv Partner Journal

Hybrid CTC/Attention Architecture for End-to-End Multi-Accent Mandarin Speech Recognition

Yang Wei, Hu Yan†
(School of Computer Science & Technology, Wuhan University of Technology, Wuhan, Hubei 430000, China)

Abstract: To address the challenge of multi-accent Mandarin speech recognition, this paper proposes a hybrid end-to-end model that combines Connectionist Temporal Classification (CTC) with Multi-Head Attention, employing multi-objective training and joint decoding. Experimental analysis reveals that as the CTC weight decreases in the hybrid architecture and encoder depth increases, the model demonstrates superior learning capability on accented datasets. We trained an extremely deep encoder-decoder network with up to 48 layers, which outperforms all previous end-to-end approaches on the Aidatatang 200-hour open-source multi-accent dataset, achieving a Character Error Rate (CER) of 5.6% and Sentence Error Rate (SER) of 26.2%. Our experiments demonstrate that the proposed end-to-end model surpasses conventional end-to-end models in recognition accuracy and offers advanced performance for accented Mandarin speech recognition.

Keywords: accent; hybrid CTC/attention end-to-end model; multi-head attention; connectionist temporal classification; speech recognition

0 Introduction

With the rapid advancement of artificial intelligence, speech recognition has become a standard feature of smart devices and a crucial modality for human-computer interaction. Accent remains a persistent challenge in speech recognition technology. For Mandarin Chinese, the prevalence of regional dialects means that most speakers' Mandarin is heavily influenced by local dialectal patterns. Research indicates that over 79.6% of Mandarin speakers have accents, with 44% exhibiting strong accents.

Early approaches to accent-related recognition problems involved dictionary adaptation, which essentially expanded pronunciation dictionaries based on dialectal phonetic habits. However, this method increased lexical confusability and yielded limited improvements. Consequently, research focus shifted to acoustic modeling. The most straightforward approach involved building separate acoustic models for each accent, but this required substantial dialect-specific training data to achieve high accuracy, followed by model selection via phonetic decision trees. The ideal solution is a unified adaptive acoustic model capable of accurately recognizing diverse accents.

Numerous studies have explored adaptive acoustic modeling for accent problems. Initially, researchers proposed Maximum Likelihood Linear Regression (MLLR) on Gaussian Mixture Density Hidden Markov Models (GMM-HMM). Subsequent work combined MLLR with Maximum A Posteriori (MAP) estimation, achieving improvements for Shanghai-accented Mandarin recognition. With the rise of deep neural networks in automatic speech recognition (ASR), researchers added accent-specific 判别 layers trained with Kullback-Leibler divergence (KLD), yielding significant improvements for non-native English speech recognition on Deep Neural Networks (DNN). Later work integrated i-vector

speaker-specific features with accent-specific 判别 layers, substantially improving accented Mandarin recognition. Other researchers proposed a convolutional neural network-based speaker speech segmentation model under fused features, providing valuable insights for Mandarin and Sichuan dialect speech recognition scenarios.

Recent studies show that Long Short-Term Memory Recurrent Neural Networks (LSTM-RNN) can achieve performance comparable to state-of-the-art DNN-HMM systems. Researchers have extensively investigated LSTM-RNN-based speech recognition models, significantly improving overall Mandarin ASR performance from various perspectives. Building on these successes, accented speech recognition research has employed DNN layers as accent-dependent feature filters to extract accent-specific features, using Bidirectional LSTM-RNN (BLSTM-RNN) to train adaptive acoustic models, achieving moderate success in multi-accent Mandarin recognition. However, these methods require separate classification and training for different dialects, making it difficult to achieve high accuracy for dialects with limited data. Moreover, such models still follow traditional modular approaches requiring multi-stage independent training (acoustic model, pronunciation dictionary, language model) and extensive expert knowledge for hyperparameter configuration.

In contrast, end-to-end models can directly transcribe speech features to text without intermediate components, establishing a direct mapping from speech to text that potentially optimizes all parts of the final task. These models can be trained using the same framework across different languages. End-to-end models have demonstrated performance comparable to state-of-the-art traditional models across many speech recognition tasks. Three primary architectures exist: CTC-based models leverage Markov assumptions to effectively solve dynamic sequence alignment; attention-based models solve acoustic frame-to-label alignment through an attention mechanism; and hybrid CTC/attention models combine both approaches through joint training and decoding, achieving superior performance to either method alone. To address end-to-end models' robustness issues and information loss from fixed-length speech frames, researchers proposed ResNet-BLSTM networks that effectively reduce character error rates.

This paper proposes a hybrid end-to-end architecture combining Multi-Head Attention encoder-decoder models with CTC, using joint training and decoding. We investigate the impact of deep networks on speech recognition by training a very deep encoder-decoder network, while also examining training processes on accented datasets with varying dropout rates in shallow encoder-decoder networks.

---

1 Hybrid CTC/Attention Model

As shown in Figure 1 [Figure 1: see original paper], we propose a deep encoder-decoder network based on Multi-Head Attention and Connectionist Temporal

Classification. In this hybrid architecture, CTC and Multi-Head Attention are jointly trained and scored, with recognition accuracy further improved by deepening the encoder and decoder networks.

Figure 1 Deep encoder-decoder architecture based on multi-head attention and connectionist temporal classification

For end-to-end speech recognition tasks, the goal is to compute the probability of all output label sequences y through a single network given input sequence x, where $U \leq T$ and $y \in L$ (the finite character set), outputting the label sequence with maximum probability:

$$y^* = \arg\max_y P(y|x)$$

1.1 Connectionist Temporal Classification (CTC)

CTC training defines intermediate label sequences π that allow repeated labels and insert a blank label $\langle b \rangle$ as a separator, i.e., $\pi \in L \cup \{\langle b \rangle\}$. Assuming y' is the expanded version of y with separators inserted, e.g., $y = (n, i, h, a, o)$, $y' = (\langle b \rangle, n, \langle b \rangle, i, \langle b \rangle, h, \langle b \rangle, a, \langle b \rangle, o, \langle b \rangle)$. A many-to-one mapping $B : {L'}^{\leq T} \to L^{\leq T}$ is defined, where ${L'}^{\leq T}$ is the set of possible intermediate label sequences. The output label probability can be obtained as:

$$P(y|x) = \sum_{\pi \in B^{-1}(y)} P(\pi|x)$$

As shown in Figure 2 [Figure 2: see original paper], for CTC training, at each time step t of input sequence x, a corresponding output π_t is computed. Assuming conditional independence between output sequences at each time step:

$$P(\pi|x) = \prod_{t=1}^{T} P(\pi_t|x) \approx \prod_{t=1}^{T} P(\pi_t|x), \quad \forall \pi \in {L'}^{\leq T}$$

From equations (1)-(3), the CTC loss function is defined as the negative log probability:

$$L_{ctc} = -\ln P(y|x) = -\ln \sum_{\pi \in B^{-1}(y)} P(\pi|x) = -\ln \sum_{\pi \in B^{-1}(y)} \prod_{t=1}^{T} P(\pi_t|x)$$

In equation (4), S represents the mapping between input sequence x and output labels, and $q_t(\pi)$ represents the probability of label π at time step t. The computation requires enumerating all label sequences across all time steps, needing N^T iterations where N is the total number of labels—computationally prohibitive.

The CTC algorithm borrows forward-backward algorithms from HMMs to optimize computation speed:

$$P(y|x) = \sum_{u=1}^{2|y|+1} \frac{\alpha_t(u)\beta_t(u)}{q_t(\pi'_u)}$$

In equation (5), $\alpha_t(u)$ represents the forward probability of the u-th label at time step t, and $\beta_t(u)$ represents the backward probability.

1.2 Multi-Head Attention Encoder-Decoder

Our encoder-decoder model [20,21] is inspired by Google' s Transformer [26,27], employing a multi-layer encoder-decoder architecture based on Multi-Head Attention with dropout layers before each layer. Each encoder layer consists of a Multi-Head Attention layer and a fully connected feed-forward network; each decoder layer comprises a masked Multi-Head Attention layer (preventing attention to future information), an attention layer over encoder inputs, and a three-layer feed-forward network.

As shown in Figure 3 [Figure 3: see original paper], the encoder computes an intermediate output z from input sequence x, while the decoder takes sequence z as input and updates one label from output sequence y at each time step t. During label updating, the model uses previous labels as input to update the next output label:

$$z = \text{Encoder}(x) = (z_1, \dots, z_T)$$

$$y_t = \text{Decoder}(z, y_{t-1})$$

Based on maximum likelihood estimation, the loss function maximizes the conditional probability of the output sequence given the input sequence:

$$L_{att} = -\ln P(y|x) = -\sum_{t=1}^{U} \ln P(y_t|y_{t-1}, \dots, y_1, z)$$

From equations (6)-(7), the decoder input depends on the same sequence z, which provides the final time step' s hidden vector computed from sequence x. However, in natural language sequences, each label output typically depends more on nearby temporal features, motivating the introduction of attention mechanisms in encoder-decoder models.

The attention mechanism outputs a result c_t at each time step, transforming the fixed intermediate sequence z into a dynamic representation that changes based

on current output. Multi-Head Attention first applies linear transformations to initialize three weight matrices Q, K, V:

$$Q, K, V = \text{Linear}(X)$$

Similarity between matrices Q and K is computed via dot product:

$$f(Q, K) = \frac{QK^T}{\sqrt{d_k}}$$

For decoder inputs, only similarities between current and previous time steps can be computed, masking future information by filling with zeros to convert the matrix into a lower triangular form. Softmax normalization is applied, and weighted averages are computed based on attention distributions:

$$\text{Attention}(Q, K, V) = \text{softmax}(f(Q, K))V$$

To prevent overfitting, dropout is applied. Following Multi-Head Attention principles, the above computation is repeated n times, with final results concatenated and linearly transformed:

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, \dots, \text{head}_n)W^O$$

After Multi-Head Attention computation, each encoder and decoder layer processes through a fully connected feed-forward network with two linear transformations and an activation function:

$$\text{FFN}(x) = \text{dropout}(\max(0, xW_1 + b_1)W_2 + b_2)$$

This yields sequence scores and attention distribution maps.

1.3 Multi-Objective Learning and Joint Decoding

Our hybrid model, inspired by [17,22], jointly trains CTC and Multi-Head Attention models with a shared encoder. Multi-objective learning (MTL) [22] leverages CTC' s forward-backward algorithm for forced monotonic alignment between speech and labels, accelerating alignment. Compared to pure CTC models, the character-level attention objective complements the sequence-level CTC objective, improving CTC accuracy. Using cross-entropy criteria, the combined loss enhances robustness:

$$L_{MOL} = \lambda \ln P_{ctc}(y|x) + (1 - \lambda) \ln P_{att}(y|x)$$

where $\lambda \in [0, 1]$ is an interpolation weight.

Beyond multi-objective training, joint decoding is crucial. Hybrid architectures typically employ beam search, computing scores for each hypothesis by combining CTC and attention scores. Using dynamic programming, each time step extends previous paths to obtain local probabilities g, determined by prior paths. Through current time step label probabilities c, all possible hypotheses h are obtained. The joint score is computed via equation (15), with paths extended at each time step. When the end character is predicted, sequences meeting minimum threshold requirements are added to final candidates, and the highest-scoring sequence is selected. We use multi-path beam search, extending only the top k probability nodes at each step, preventing exponential growth and significantly reducing exhaustive search complexity.

---

2 Experiments

2.1 Experimental Models

Our end-to-end model is a hybrid CTC/attention architecture. Traditional modular baselines include Gaussian Mixture Model-Hidden Markov Model (GMM-HMM), Time-Delay Neural Network (TDNN), and Chain models. End-to-end baselines include CTC/location-based attention, CTC/content-based attention, and pure Multi-Head Attention models.

2.2 Data

Experiments use the Aidatatang 200-hour open-source multi-accent Mandarin dataset, containing 300,000 colloquial sentences from 6,408 speakers (2,999 male, 3,301 female) across 34 provincial regions including Guangdong, Fujian, and Shandong. Traditional modular models were implemented on Kaldi [28], while end-to-end experiments used the ESPnet toolkit [29]. All experiments employed 16 kHz sampling frequency, 80-dimensional FBANK features with 25 ms frames and 10 ms frame shift.

2.3 Network Structure and Parameters

Our network uses 4-head Multi-Head Attention with 256-dimensional input attention, 256-dimensional feed-forward input features, 2,048-dimensional hidden features, and 2,048 bidirectional LSTM units. Joint training parameter λ is 0.3, dropout rate is 0.1, and label smoothing is 0.1. Beam search width is 10, language model weight is 0.1, and training runs for 50 epochs.

2.4 Evaluation Metrics

We evaluate on Aidatatang' s development and test sets using Character Error Rate (CER) and Sentence Error Rate (SER). CER is the percentage of inser-

tions (I), substitutions (S), and deletions (D) required to match the recognized sequence to the reference, divided by total reference characters. SER considers a sentence incorrect if any word is misrecognized:

$$\text{CER} = \frac{S + D + I}{N} \times 100\%$$

$$\text{SER} = \frac{\text{Number of incorrect sentences}}{\text{Total sentences}} \times 100\%$$

2.5 Comparative Experiments

We implemented three traditional models as baselines: GMM-HMM, TDNN, and Kaldi' s Chain model—the top-performing models on the Aidatatang multi-accent dataset, using Kaldi' s aidatatang recipe.

GMM-HMM: After feature extraction, researchers found Gaussian mixture distributions suitable for fitting speech feature sequences when ignoring temporal dependencies. GMM was integrated into HMM to model state-based output distributions.

TDNN: TDNN models long-term temporal dependencies using subsampling to reduce computation, training faster than RNNs while leveraging i-Vector features for speaker and environment adaptation, showing good performance on accented speech.

Chain Model: A DNN-HMM variant borrowing CTC's blank symbol to absorb uncertain boundaries, using correct sequence log-probability as the objective function. It computes numerator and denominator probabilities during training, backpropagating their difference, and effectively improves decoding speed.

Multi-Head Attention Model [20,21]: Based on self-attention mechanisms operating within sequences, learning speech features in the encoder and label features in the decoder, offering better performance than standard attention for sequence problems.

LSTM-RNN-CTC Model [30]: Combines LSTM' s strong sequence modeling with discriminative training for different regional Mandarin accents, greatly improving accent recognition capability.

---

3 Results

3.1 Comparison Between Traditional and End-to-End Models

Table 1 compares traditional modular ASR systems with common end-to-end models on multi-accent Mandarin recognition. The MTL parameter represents the attention score weight in joint training. The hybrid architecture uses a

12-layer encoder and 6-layer decoder. Compared to pure Multi-Head Attention, the hybrid model achieves 10.0%-10.9% absolute CER reduction and 6.3%-7.1% SER reduction. In pure attention models, most errors occur in heavily accented utterances, indicating severe limitations in fitting pronunciation variations across regions. Compared to traditional modular models, the hybrid end-to-end model shows strong performance, typically outperforming TDNN and GMM-HMM, and slightly trailing the state-of-the-art Chain model. However, end-to-end models offer convenient holistic optimization.

Notably, as MTL weight increases (0.0, 0.1, 0.3), recognition accuracy improves. As shown in Figure 4 [Figure 4: see original paper], when CTC loss weight dominates MTL training, performance degrades with severe overfitting. This likely stems from CTC' s conditional independence assumption, which cannot effectively leverage accent similarity in multi-accent datasets, hindering convergence. However, in hybrid models, CTC still improves accuracy and accelerates convergence. Under proper MTL weighting, CTC' s monotonic forced alignment significantly boosts Multi-Head Attention training and speeds convergence.

3.2 Study on Model Depth and Dropout Rate

To further improve multi-accent Mandarin recognition, we deepened encoder and decoder networks and gradually increased dropout rates (0.0, 0.1, 0.5) in shallow networks.

Table 2 shows that our deep Multi-Head Attention and CTC hybrid model substantially outperforms traditional GMM-HMM and TDNN models. Compared to LSTM-RNN-CTC and standard Multi-Head Attention, it achieves both higher accuracy and faster training. As depth increases, CER reaches 5.6%, matching the best Chain model without requiring a lexicon, while remaining simple and easy to optimize.

As shown in Figure 5 [Figure 5: see original paper], deeper encoders learn more representative audio features. Figure 6 [Figure 6: see original paper] demonstrates that deeper decoders strengthen label correlations, significantly improving silence detection at sentence endings and reducing deletion errors compared to shallow networks.

---

4 Conclusion

This paper proposes a hybrid CTC and Multi-Head Attention encoder-decoder architecture for accented Mandarin speech recognition. Unlike traditional modular models requiring extensive manual resources, dialect-specific corpora, and complex adaptation, our single model surpasses conventional methods and enables convenient holistic optimization, achieving state-of-the-art performance. In our deep encoder-decoder model, increased encoder depth better learns audio representations, significantly improving accented Mandarin recognition. De-

coder depth yields smaller overall gains, while increased dropout in shallow networks provides minor improvements. Future work will explore improved dropout methods for deeper networks and address slow training in very deep networks.

References

[1] Huang C, Chen T, Chang E. Accent issues in large vocabulary continuous speech recognition [J]. International Journal of Speech Technology, 2004, 7 (2-3): 141-153.

[2] ZHANG D. xin (National Leading Group Office of the Teaching of Chinese to Foreigners, Beijing 100083) [J]. My Point of View on the Standard of Newborn Words, 2000, 5.

[3] Ding G H. Phonetic confusion analysis and robust phone set generation for Shanghai-accented Mandarin speech recognition [C]// Ninth Annual Conference of the International Speech Communication Association. 2008.

[4] Elfeky M, Bastani M, Velez X, et al. Towards acoustic model unification across dialects [C]// 2016 IEEE Spoken Language Technology Workshop (SLT). IEEE, 2016: 624-628.

[5] Wang Z, Schultz T, Waibel A. Comparison of acoustic model adaptation techniques on non-native speech [C]// 2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP' 03). IEEE, 2003, 1: I-I.

[6] Zheng Y, Sproat R, Gu L, et al. Accent detection and speech recognition for Shanghai-accented mandarin [C]// Ninth European Conference on Speech Communication and Technology. 2005.

[7] Huang Y, Yu D, Liu C, et al. Multi-accent deep neural network acoustic model with accent-specific top layer using the KLD-regularized model adaptation [C]// Fifteenth Annual Conference of the International Speech Communication Association. 2014.

[8] DeMarco A, Cox S J. Native accent classification via i-vectors and speaker compensation fusion [C]// Interspeech. 2013: 1472-1476.

[9] Chen M, Yang Z, Liang J, et al. Improving deep neural networks based multi-accent mandarin speech recognition using i-vectors and accent-specific top layer [C]// Sixteenth Annual Conference of the International Speech Communication Association. 2015.

[10] Zhang P. Research on adaptive recognition of different accent dialogues based on convolutional neural networks [D]. Chongqing University, 2018.

[11] Sak H, Senior A, Beaufays F. Long short-term memory based recurrent neural network architectures for large vocabulary speech recognition [J]. arXiv preprint arXiv: 1402. 1128, 2014.

[12] Yi J, Ni H, Wen Z, et al. Improving blstm rnn based mandarin speech recognition using accent dependent bottleneck features [C]// 2016 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA). IEEE, 2016: 1-5.

[13] Graves A, Jaitly N. Towards end-to-end speech recognition with recurrent neural networks [C]// International conference on machine learning. 2014: 1764-1772.

[14] Miao Y, Gowayyed M, Metze F. EESEN: End-to-end speech recognition using deep RNN models and WFST-based decoding [C]// 2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU). IEEE, 2015: 167-174.

[15] Chorowski J, Bahdanau D, Cho K, et al. End-to-end continuous speech recognition using attention-based recurrent nn: First results [J]. arXiv preprint arXiv: 1412. 1602, 2014.

[16] Bahdanau D, Chorowski J, Serdyuk D, et al. End-to-end attention-based large vocabulary speech recognition [C]// 2016 IEEE international conference on acoustics, speech and signal processing (ICASSP). IEEE, 2016: 4945-4949.

[17] Lu L, Zhang X, Renais S. On training the recurrent neural network encoder-decoder for large vocabulary end-to-end speech recognition [C]// 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2016: 5060-5064.

[18] Graves A, Fernández S, Gomez F, et al. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks [C]// Proceedings of the 23rd international conference on Machine learning. ACM, 2006: 369-376.

[19] Cho K, Van Merriënboer B, Gulcehre C, et al. Learning phrase representations using RNN encoder-decoder for statistical machine translation [J]. arXiv preprint arXiv: 1406. 1078, 2014.

[20] Sutskever I, Vinyals O, Le Q V. Sequence to sequence learning with neural networks [C]// Advances in neural information processing systems. 2014: 3104-3112.

[21] Kim S, Hori T, Watanabe S. Joint CTC-attention based end-to-end speech recognition using multi-task learning [C]// 2017 IEEE international conference on acoustics, speech and signal processing (ICASSP). IEEE, 2017: 4835-4839.

[22] Watanabe S, Hori T, Kim S, et al. Hybrid CTC/attention architecture for end-to-end speech recognition [J]. IEEE Journal of Selected Topics in Signal Processing, 2017, 11 (8): 1240-1253.

[23] Chorowski J K, Bahdanau D, Serdyuk D, et al. Attention-based models for speech recognition [C]// Advances in neural information processing systems. 2015: 577-585.

[24] Chorowski J, Jaitly N. Towards better decoding and language model integration in sequence to sequence models [J]. arXiv preprint arXiv: 1612. 02695, 2016.

[25] Hu Zhangfang, Xyu Xuan, Fu Yanqin, et al. End-to-end speech recognition based on ResNet-BLSTM [J/OL]. Computer Engineering and Applications: 1-8 [2020-03-26]. http://kns. cnki. net/kcms/detail/11. 2127. TP. 20191014. 1706. 010. html.

[26] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need [C]// Advances in neural information processing systems. 2017: 5998-6008.

[27] Pham N Q, Nguyen T S, Niehues J, et al. Very Deep Self-Attention Networks for End-to-End Speech Recognition [J]. arXiv preprint arXiv: 1904. 13377, 2019.

[28] Watanabe S, Hori T, Karita S, et al. Espnet: End-to-end speech processing toolkit [J]. arXiv preprint arXiv: 1804. 00015, 2018.

[29] Povey D, Ghoshal A, Boulianne G, et al. The Kaldi speech recognition toolkit [C]// IEEE 2011 workshop on automatic speech recognition and understanding. IEEE Signal Processing Society, 2011 (CONF).

[30] Yi J, Wen Z, Tao J, et al. CTC Regularized Model Adaptation for Improving LSTM RNN Based Multi-Accent Mandarin Speech Recognition [J]. Journal of Signal Processing Systems, 2018, 90 (7): 985-993.

Note: Figure translations are in progress. See original paper for figures.