TOWARDS END-TO-END AUTOMATIC CODE-SWITCHING SPEECH RECOGNITION

Genta Indra Winata, Andrea Madotto, Chien-Sheng Wu, Pascale Fung

Center for Artificial Intelligence Research (CAiRE)
Department of Electronic and Computer Engineering
Hong Kong University of Science and Technology, Clear Water Bay, Hong Kong
{giwinata, amadotto, cwuak}@connect.ust.hk, pascale@ece.ust.hk

ABSTRACT

Speech recognition in mixed language has difficulties to adapt end-to-end framework due to the lack of data and overlapping phone sets, for example in words such as "one" in English and "wàn" in Chinese. We propose a CTC-based end-to-end automatic speech recognition model for intra-sentential English-Mandarin code-switching. The model is trained by joint training on monolingual datasets, and fine-tuning with the mixed-language corpus. During the decoding process, we apply a beam search and combine CTC predictions and language model score. The proposed method is effective in leveraging monolingual corpus and detecting language transitions and it improves the CER by 5%.

Index Terms— code-switch, end-to-end speech recognition, transfer learning, joint training, bilingual

1. INTRODUCTION

Code-switching is the linguistic phenomenon of a person speaking or writing in one language and switches to another in the same sentence. It is very common among bilingual communities. With the advent of globalization, code-switching is becoming more common in predominantly monolingual societies as speakers use a second language in professional contexts. Speakers code-switch to empathize with each other, to express themselves better [1], and very often are not fully aware of using mixed codes in their language [2].

Code-switching poses a significant challenge for automatic speech recognition (ASR) systems even as the latter reach higher and higher performance within the new paradigm of neural network speech recognition [3, 4, 5]. The main reason lies in the unpredictability of points of code-switching in an utterance. Since speech recognition needs to be accomplished in real-time in most applications, it is not feasible to carry out language identification at each time step before transcribing the speech into text. Moreover, language identification of a single speaker within the same utterance is challenging as the speaker can carry over their pronunciation habits from the primary language to the foreign language in context. In the statistical ASR framework, acoustic modeling, pronunciation modeling and language modeling of code-switching speech are carried out separately and assumed to be independent of each other. The hypothesis $\hat{Y}$ is normally calculated as the following:

$$\hat{Y} = \underset{Y}{\operatorname{argmax}}\, P(X|Y)P(Y) \tag{1}$$

where X is the input signal, Y is the target sequence, $P(X|Y)$ is the acoustic observations and $P(Y)$ is the language model. The acoustic model consists of multiple Hidden Markov phoneme models, is trained from both languages bilingually, where the pronunciation of phones from both languages are mapped to each other. Each shared or mapped phoneme model is then trained from speech samples in both languages. Variations of this type of bilingual acoustic models abound, but they are not difficult to train.

Another challenge is how to train the code-switching language model, $P(Y)$. Reliable statistical language models are derived from large amounts of text. In the case of code-switching speech, the data is often not enough to be generalizable. Previous work attempted to solve this problem by either generating more code-switching data by allowing code-switch at every point in the utterance, or at the phrasal boundaries [6] or even by phrasal alignment of parallel data according to linguistic constraints [7]. In each case, the code-switching points are not learned automatically. The generalizability of language models derived from this data is therefore doubtful.

In this paper, we propose and investigate an entirely different approach of automatic code-switching speech recognition, using an end-to-end neural network framework to recognize speech from input spectrogram to output text, dispensing with the pipelined architecture of acoustic modeling followed by language modeling. We apply a joint-training in CTC-based speech recognition [4] and finetune the model on code-switching dataset to learn the language transitions between them. We show that our proposed approach learns how to distinguish signals from different languages and achieves better

This work is partially funded by ITS/319/16FP of the Innovation Technology Commission, HKUST 16214415 & 16248016 of Hong Kong Research Grants Council, and RDC 1718050-0 of EMOS.AI.

results compared to the training only with a code-switching corpus. We compare the results by adding different amounts of the mixed-language corpus and test the effectiveness of the joint-training. In the decoding step, we rescore our generated sequences with an n-gram language model.

2. RELATED WORK

Code-switching speech recognition: The prior study on code-switching ASR is to incorporate a tri-phone HMMs as an acoustic model with an equivalence constraint in the language model [6]. [8] combined recurrent neural networks and factored language models to rescore the n-best hypothesis. [7] proposed a lattice-based parsing to restrict the sequence paths to those permissible under the Functional Head Constraint. [9] applied different phone merging approaches and combination with discriminative training. An extensive study on Hindi-English phone set sharing and gains improvement in WER [10]. In another line of work, multi-task learning approaches in code-switching had been used for learning a shared representation of two or more different tasks on language modeling [11] and acoustic model [12].

End-to-end approaches: [3] presented the earliest implementation of CTC to end-to-end speech recognition. A sequence-to-sequence model with attention [13] that learned to transcribe speech utterances to characters has been introduced by [14] as Listen-Attend-Spell (LAS). A multi-lingual approach was proposed by [5] using Seq2Seq approach using a union of language-specific grapheme sets and train a grapheme-based sequence-to-sequence model jointly on data from all languages. [15] proposed to train a bilingual ASR for spontaneous Japanese and Chinese speech by using an end-to-end Seq2Seq model.

3. METHODOLOGY

In this section, we describe the joint training on our proposed end-to-end approach including the learning strategies. During the decoding stage, we add an external language model for rescoring. We denote our training sets, English monolingual dataset $\{(X_1^{en}, Y_1^{en}), ..., (X_n^{en}, Y_n^{en})\}$ and Mandarin Chinese monolingual dataset $\{(X_1^{zh}, Y_1^{zh}), ..., (X_n^{zh}, Y_n^{zh})\}$, and a code-switching dataset $\{(X_1^{cs}, Y_1^{cs}), ..., (X_n^{cs}, Y_n^{cs})\}$. The labels Y are graphemes and the character set is the concatenation of English and Simplified Chinese characters {`a-z, space, apostrophe,` 祥, 舌, `...,` 底}.

3.1. Connectionist Temporal Classification Model

A CTC network uses an error criterion that optimizes the prediction of transcription by aligning the input signals with the hypothesis. The loss function is defined as the negative log likelihood:

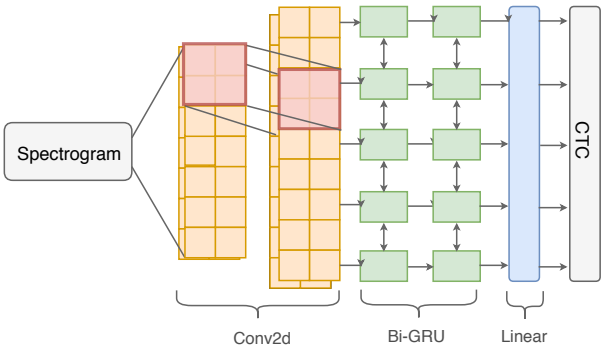

Fig. 1. Connectionist Temporal Classification Model

$$L_{CTC} = -\log P(I|X) = -\ln \sum_{\pi \in B^{-1}(Y)} P(\pi|X) \quad (2)$$

where we denote π as the CTC path, I as the transcription, and $B^{-1}(Y)$ as the mapping of all possible CTC paths π resulting from I. Comparing to other end-to-end models such as LAS, CTC is more stable in training; thus it is easier to converge. Our model consists of a multi-layer Convolutional Neural Network [16] to generate a rich input representation, followed by a multi-layer Recurrent Neural Network (RNN) with Gated Recurrent Unit (GRU) [17] to learn the temporal information of the audio frame sequences. We are using the CTC-based architecture similar to [4]. The input of the network is a sequence of log spectrograms and followed by a normalization step to regularize the parameters and avoid internal covariate shift. The CTC probabilities are used to find a better alignment between hypothesis and the input signals. To scale up the training process, batch normalization is employed to the recurrent layers to reduce the batch training time. The spectrograms are passed to a convolutional neural network (CNN) to encode the speech representation. The frame-wise posterior distribution $P(Y|X)$ is conditioned on the input X and calculated by applying a fully-connected layer and a softmax function.

$$P(Y|X) = \text{Softmax}(\text{Linear}(h)) \quad (3)$$

where h is the hidden state from the GRU. Next, it passes to a multi-layer bidirectional GRU.

3.2. Joint Training

We start the training as a joint training using monolingual dataset as a pretraining to learn the individual language. However, the CTC-based model can easily suffer catastrophic forgetting, where the model is not capable of remembering two distant languages such as English and Mandarin Chinese in separated supervision. It tends to keep one language and forget the other. Thus, we propose to train iteratively between two datasets; taking English and Chinese speeches in

Table 1. Data Statistics of SEAME Phase II [11].

	Train	Dev	Test
# Speakers	138	8	8
# Duration (hr)	100.58	5.56	5.25
# Utterances	78,815	4,764	3,933
# Tokens	1.2M	65K	60K
# Tokens Preprocessed	978K	53K	48K
Avg. segment	4.21	3.59	3.99
Avg. switches	2.94	3.12	3.07

Table 2. Data Statistics of Common Voice and HKUST

Dataset		# Duration (hr)	# Samples
Common Voice	**Train**	241.21	195,372
	Dev	4.99	4,065
	Test	4.94	3,986
HKUST	**Train**	168.88	873
	Dev	4.81	24

the batch. Thus, the model learns how to differentiate the characters. After the pretraining, we tune the model with code-switch data.

3.3. Language Modeling

To improve the quality of the decoded sequence, we train a 5-gram language model with Kneser Ney smoothing [18] on our code-switching training data using KenLM[1]. We use a prefix beam search with a beam width of w. We rescore the probability of the sequence $p_lm(Y)$ [4] and find the maximum score of $Q(Y)$:

$$Q(Y) = \log(P_{ctc}(Y|X)) + \alpha \log(p_{lm}(Y)) + \beta\, wc(Y) \quad (4)$$

where $wc(Y)$ is the word count of sequence Y, α controls the contribution of language model and β controls the number of word to be generated. In the beam search process, the decoder computes a score of each partial hypothesis and interpolates the result with the probability from the language model.

4. EXPERIMENT

In this section, we describe our datasets and code-switching ASR experiments with our proposed end-to-end system.

4.1. Corpus

We use speech data from SEAME Phase II (South East Asia Mandarin-English), a conversational Mandarin-English code-switching speech corpus consists of spontaneously spoken interviews and conversations [19]. We tokenize words using Stanford NLP toolkit [20] and follow the same preprocessing step as [11]. For the monolingual datasets, we use HKUST [21], a spontaneous Mandarin Chinese telephone speech recordings and Common Voice, an open accented English dataset collected by Mozilla[2]. Table 1 shows the statistics of SEAME Phase II dataset and Table 4 shows the statistics of Common Voice and HKUST datasets.

Table 3. Character Error Rate (CER %) for single dataset training, joint training, and rescoring with LM on SEAME Phase II.

Model	Dev	Test
SEAME Phase II		
- training (10% data, 10.13 hr)	49.81	46.23
- training (50% data, 50.41 hr)	38.08	32.10
- training (100% data, 100.58 hr)	36.18	29.82
+ LM (5-gram, α = 0.2)	35.77	29.14
Joint training		
- fine tuning (10% data, 10.13 hr)	38.44	43.86
- fine tuning (50% data, 50.41 hr)	34.24	27.97
- fine tuning (100% data, 100.58 hr)	32.06	25.54
+ LM (5-gram, α = 0.2)	**31.35**	**24.61**

4.2. Experimental Setup

We convert the inputs into normalized frame-wise spectrograms. We take 20 ms width with a stride of 20 ms. The audio is down-sampled to a single channel with a sample rate of 8 kHz. The CNN encoder is described as the following:

$$\text{Conv2d}(\text{in} = 1, \text{out} = 32, \text{filter} = 41 \times 11) \quad (5)$$

$$\text{Hardtanh}(\text{BatchNorm2d}(32)) \quad (6)$$

$$\text{Conv2d}(\text{in} = 32, \text{out} = 32, \text{filter} = 21 \times 11) \quad (7)$$

$$\text{Hardtanh}(\text{BatchNorm2d}(32)) \quad (8)$$

It is followed by a 4-layer bidirectional GRU with a hidden size of 400 and a fully connected layer with a hidden size of 400 is inserted afterward. In the joint-training, we combine both monolingual datasets and sorted by their audio length, and groups them in buckets. We shuffle the buckets and take 20 samples in a batch. Then, we take every batch for the training.

We start our training with different initial learning rates $\{1e-4, 3e-4\}$ and optimize our model by using Stochastic Gradient Descent (SGD) with momentum and Nesterov accelerated gradient [22]. In the sequence decoding stage, we take to run a prefix beam-search a beam size of 100 to find the best sequence. The best hyperparameters for rescoring are $\alpha = 0.2$ and $\beta = 1$. We first perform our fine-tuning experiment with smaller training sets to calculate the effectiveness of joint training. We take 10% (10.13 hr) and 50% (50.41 hr) from code-switching dataset. We also train our language model with 3-gram, 4-gram, and 5-gram models.

[1]The code can be found at https://github.com/kpu/kenlm

[2]The dataset is available at https://voice.mozilla.org/

Table 4. Generated Sequences from CTC 4-layer GRU

Model	Generated Sequence	CER %
reference	因为 我 的 friend 会 很 shy 我 的 friend 他 这 not really 很 shy 他 们 就是 要 人家 陪 他们 唱 因为 他们 觉得 一个人 唱 很 sian	-
baseline	为 我 的 friend wa 很 s 我 们 在 friend 他 这 not ra 能 s 他 就 要 人家 pa 他们 唱 因为 他们 觉得 一个 人 唱 很 闲	23.65%
+ fine-tuning	因 为 我 的 friend 会 很 shy 我 friend 他 做 not re 很 帅 他 们 就 要 人家 陪 他们 唱 因为 他们 觉得 一 个人 唱 很 鲜	15.05%
+ LM	因为 我 的 friend 会 很 shy 我 friend 他 not real s 他 们 就 要 人 家 陪 他们 唱 因为 他们 觉得 一 个 人 唱 很 s	11.82%
reference	then 你 做 什么 before what kind of job	-
baseline	你 做 什么 可是 what 他 要 dro	48.57%
+ fine-tuning	因 你 做 什 么 for what kid of job	22.85%
+ LM	你 做 什 么 for what kind of job	20.00%

5. RESULTS

Table 3 demonstrates speech recognition and language modeling results. The joint training with fine tuning improves the results 5% CER compared to training only with a code-switching corpus. Some generated characters are separated with excessive spaces. The joint training captures the sounds from monolingual corpus and transfer learns the information during the fine-tuning step. The baseline is not able to capture the word transition between Chinese and English. The joint training captures the sounds from the monolingual corpus and transfers learning to the code-switching sequences.

Joint training + Fine tuning: Joint training helps the model to learn different sounds the perspective of single language. It is also an excellent way to initialization of our model. However, from the training, we still suffer an issue, where it does not keep the information of both languages, and we need to solve this issue in the future work. As shown in Table 4, we can see that the results are getting better, it can generate a better mapping of similar sounds in English and Chinese to the corresponding characters. We test our model with smaller training data. According to our observation, by fine-tuning with only 50% code-switching data, we can achieve a comparable result to the whole training only with code-switching dataset.

Applying language model: The external language model constraints the decoder to generate more grammatical sentences. In general, language model effects positively to the decoder and achieves an additional 1% reduction by adding a 5-gram language model. From Table 4, it clearly shows that some misspelled words in the baseline are fixed.

Intra-sentential code-switching: The model we trained can predict some English words correctly between Chinese words testing on code-switching dataset. It can still predict the code-switching points, unlike the work by [5]. One of the possible reason is our CTC model is not constrained by the language information like the Seq2Seq-based model with language identifiers. In spite of that, the model is still predicting words with similar sound in the code-switching points such as "before" and "for".

6. CONCLUSION

We propose a new direction on automatic code-switching speech recognition by applying end-to-end approach. Our training method can be adapted to any languages pair. We evaluate our model on English-Mandarin corpus and achieve a significant gain through a combination of joint training and fine-tuning. It can handle code-switching transitions and recognize both English and Chinese characters. The rescoring using an external language model improves the decoding result and fixes the spelling mistakes. Our proposed model achieves a 5% reduction in CER and the joint-training procedure allows the model to learn distant languages. For future work, we are going to mitigate further the catastrophic forgetting in our joint training network, which degrades the performance of our bilingual model.

7. REFERENCES

[1] Rosamina Lowi, "Codeswitching: An examination of naturally occurring conversation," in *Proceedings of the 4th International Symposium on Bilingualism*. Cascadilla Press, pp Somerville, MA, 2005, pp. 1393–1406.

[2] Orit Shay, "To switch or not to switch: Code-switching in a multilingual country," *Procedia-Social and Behavioral Sciences*, vol. 209, pp. 462–469, 2015.

[3] Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *Proceedings of the 23rd international conference on Machine learning*. ACM, 2006, pp. 369–376.

[4] Dario Amodei, Sundaram Ananthanarayanan, Rishita Anubhai, Jingliang Bai, Eric Battenberg, Carl Case, Jared Casper, Bryan Catanzaro, Qiang Cheng, Guoliang Chen, et al., "Deep speech 2: End-to-end speech recognition in english and mandarin," in *International Conference on Machine Learning*, 2016, pp. 173–182.

[5] Shubham Toshniwal, Tara N. Sainath, Ron J. Weiss, Bo Li, Pedro J. Moreno, Eugene Weinstein, and Kanishka Rao, "Multilingual speech recognition with a single end-to-end model," *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4904–4908, 2018.

[6] Ying Li and Pascale Fung, "Code-switch language model with inversion constraints for mixed language speech recognition," *Proceedings of COLING 2012*, pp. 1671–1680, 2012.

[7] Ying Li and Pascale Fung, "Code switch language modeling with functional head constraint," *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 4913–4917, 2014.

[8] Heike Adel, Ngoc Thang Vu, Franziska Kraus, Tim Schlippe, Haizhou Li, and Tanja Schultz, "Recurrent neural network language modeling for code switching conversational speech," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*. IEEE, 2013, pp. 8411–8415.

[9] Ngoc Thang Vu, Dau-Cheng Lyu, Jochen Weiner, Dominic Telaar, Tim Schlippe, Fabian Blaicher, Eng-Siong Chng, Tanja Schultz, and Haizhou Li, "A first speech recognition system for mandarin-english code-switch conversational speech," in *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on*. IEEE, 2012, pp. 4889–4892.

[10] Sunit Sivasankaran, Brij Mohan Lal Srivastava, Sunayana Sitaram, Kalika Bali, and Monojit Choudhury, "Phone merging for code-switched speech recognition," in *Proceedings of the Third Workshop on Computational Approaches to Linguistic Code-Switching*, 2018, pp. 11–19.

[11] Genta Indra Winata, Andrea Madotto, Chien-Sheng Wu, and Pascale Fung, "Code-switching language modeling using syntax-aware multi-task learning," in *Proceedings of the Third Workshop on Computational Approaches to Linguistic Code-Switching*. 2018, pp. 62–67, Association for Computational Linguistics.

[12] Michael L Seltzer and Jasha Droppo, "Multi-task learning in deep neural networks for improved phoneme recognition," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*. IEEE, 2013, pp. 6965–6969.

[13] Thang Luong, Hieu Pham, and Christopher D Manning, "Effective approaches to attention-based neural machine translation," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, 2015, pp. 1412–1421.

[14] William Chan, Navdeep Jaitly, Quoc Le, and Oriol Vinyals, "Listen, attend and spell: A neural network for large vocabulary conversational speech recognition," in *Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on*. IEEE, 2016, pp. 4960–4964.

[15] Takaaki Hori, Shinji Watanabe, Yu Zhang, and William Chan, "Advances in joint ctc-attention based end-to-end speech recognition with a deep cnn encoder and rnn-lm," in *INTERSPEECH*, 2017.

[16] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[17] Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio, "Learning phrase representations using rnn encoder–decoder for statistical machine translation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014, pp. 1724–1734.

[18] Kenneth Heafield, Ivan Pouzyrevsky, Jonathan H Clark, and Philipp Koehn, "Scalable modified kneser-ney language model estimation," in *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 2013, vol. 2, pp. 690–696.

[19] Universiti Sains Malaysia Nanyang Technological University, "Mandarin-english code-switching in south-east asia ldc2015s04. web download. philadelphia: Linguistic data consortium," 2015.

[20] Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky, "The Stanford CoreNLP natural language processing toolkit," in *Association for Computational Linguistics (ACL) System Demonstrations*, 2014, pp. 55–60.

[21] Yi Liu, Pascale Fung, Yongsheng Yang, Christopher Cieri, Shudong Huang, and David Graff, "Hkust/mts: A very large scale mandarin telephone speech corpus," in *Chinese Spoken Language Processing*, pp. 724–735. Springer, 2006.

[22] Yurii E Nesterov, "A method for solving the convex programming problem with convergence rate o (1/kˆ 2)," in *Dokl. Akad. Nauk SSSR*, 1983, vol. 269, pp. 543–547.