END-TO-END CODE-SWITCHING ASR FOR LOW-RESOURCED LANGUAGE PAIRS

Xianghu Yue [1,2], Grandee Lee [2], Emre Yılmaz [2], Fang Deng [1], Haizhou Li [2]

[1]Beijing Institute of Technology, Beijing, China
[2]National University of Singapore, Singapore
{xianghu.yue, grandee.lee}@u.nus.edu, fangdeng@bit.edu.cn, {emre, haizhou.li}@nus.edu.sg

ABSTRACT

Despite the significant progress in end-to-end (E2E) automatic speech recognition (ASR), E2E ASR for low resourced code-switching (CS) speech has not been well studied. In this work, we describe an E2E ASR pipeline for the recognition of CS speech in which a low-resourced language is mixed with a high resourced language. Low-resourcedness in acoustic data hinders the performance of E2E ASR systems more severely than the conventional ASR systems. To mitigate this problem in the transcription of archives with code-switching Frisian-Dutch speech, we integrate a designated decoding scheme and perform rescoring with neural network-based language models to enable better utilization of the available textual resources. We first incorporate a multi-graph decoding approach which creates parallel search spaces for each monolingual and mixed recognition tasks to maximize the utilization of the textual resources from each language. Further, language model rescoring is performed using a recurrent neural network pre-trained with cross-lingual embedding and further adapted with the limited amount of in-domain CS text. The ASR experiments demonstrate the effectiveness of the described techniques in improving the recognition performance of an E2E CS ASR system in a low-resourced scenario.

Index Terms— Code-switching, end-to-end ASR, language modeling, multi-graph, under-resourced languages

1. INTRODUCTION

As multilingualism is becoming more common in today's globalized world [1], there has been increasing interest in code-switching (CS) automatic speech recognition (ASR) [2]. Code-switching refers to the phenomenon where two languages are spoken in contact within one utterance [3]. Code-switching, such as Mandarin-English [4], Spanish-English [5] and Hindi-English [6], is commonly practiced in multi-lingual societies.

Traditionally, an ASR system consists of several components including acoustic model, pronunciation and language model that are separately trained and optimized with different objectives, thus building an ASR system needs specialized expertise in the field. Various end-to-end (E2E) ASR approaches are emerging quickly because of its simplicity compared to the traditional ASR architecture. An E2E system predicts phones or characters directly from acoustic information without predefined alignment. Some notable architectures include connectionist temporal classification (CTC) [7], attention based encoder-decoder networks [8, 9], and recurrent neural network (RNN) transducers [10]. More recently, hybrid E2E systems have been successfully implemented and applied to common ASR benchmarks [11]. These E2E models have been successfully used in monolingual and multilingual ASR systems by achieving promising results on various benchmarks [12–16].

E2E ASR approaches enable lexicon-free recognition which is a key advantage over traditional hybrid hidden Markov model/deep neural networks (HMM/DNN) approaches in low-resourced settings, since there are many low-resourced languages without an available pronunciation lexicon. However, there is very limited work done for recognizing CS speech using E2E techniques, especially for low-resourced language pairs. This is mainly due to the fact that low-resourcedness in acoustic data hinders the performance of E2E CS ASR more severely than the conventional ASR system. Hiroshi [17] built an encoder-decoder based E2E ASR system that can recognize the mixed-language speech. However, the work relies on training data that is generated from monolingual datasets, rather than natural code-switching speech. Kim [18] and Toshniwa [19] both used encoder-decoder model to build multilingual E2E ASR, but their systems cannot deal with CS scenario. Li [20] incorporate a frame-level language identification (LID) model to linearly adjust the posteriors of an E2E CTC model for the high-resourced Mandarin-English language pair.

In this paper, we integrate a designated decoding scheme and a code-switch language model (LM) rescoring scheme to mitigate this problem in our recognition scenario, namely transcripts of archives with CS Frisian-Dutch speech in which Frisian is a low-resourced language and Dutch is a high-resourced language. The code-switch LM [21] is a recurrent neural network (RNN) that is trained with cross-lingual embedding and adapted to maximize the use of the available textual resources. The decoding scheme provides a

new multi-graph back-end for E2E CS ASR in which parallel search spaces are employed for monolingual and mixed recognition subtasks. The code-switch RNN LM can both preserve the cross-lingual correspondence derived from larger monolingual textual resources and leverage the low-resourced language on the high-resourced language at the same time.

The rest of this paper is organized as follows. Section 2 introduces the E2E CTC acoustic model. The incorporated multi-graph decoding strategy and CS RNN LM rescoring are described in section 3 and 4 respectively. We describe the experimental setup in section 5 and then present and discuss the results provided by the described E2E ASR pipeline in section 6.

2. END-TO-END CTC ACOUSTIC MODEL

Unlike in the traditional hybrid HMM-DNN system, an E2E CTC acoustic model is not trained using frame-level labels with respect to the cross-entropy (CE) criterion. Instead, a CTC model learns the alignments automatically between speech frames and their label sequences, i.e., phone sequences, by adopting the CTC objective. It predicts the conditional probability of the label sequence by summing over the joint probabilities of the corresponding set of CTC symbol sequences. The CTC framework has the output independent assumption that CTC symbols are conditionally independent at each frame, which may be more desirable for dealing with CS speech (though less accurate in general) as the current output does not explicitly depend on previous outputs [20]. The conditional probability of the whole label sequence is:

$$P(\mathbf{z}|\mathbf{x}) = \sum_{\pi \in \mathcal{B}^{-1}(\mathbf{z})} P(\pi|\mathbf{x}) = \sum_{\pi:\pi \in Z', \mathcal{B}(\pi_{1:T})=\mathbf{z}} \prod_{t=1}^{T} y_{\pi_t}^t \quad (1)$$

where $\mathbf{z} = (z_1, \ldots, z_u, \ldots, z_U)$ denotes a phone label sequence containing U phones, $z \in Z$ and Z is the phone set. $\mathbf{x} = (x_1, \ldots, x_t, \ldots, x_T)$ denotes a sequence of T speech frames, with t being the frame index. The length of z is constrained to be no greater than the length of the utterance, i.e., $U \leq T$. $\pi_{1:T} = (\pi_1, \ldots, \pi_t, \ldots, \pi_T)$ is an output symbol sequence at frame level, named CTC *path*. Each output symbol $\pi \in Z^{'}$ and $Z^{'} = Z \cup blank$. $blank$ is a special label in the CTC framework, which maps frames and labels to the same length. $\mathcal{B}$ is a multiple-to-one mapping with first removing the repeated labels and then all $blank$ symbols from the paths. $y_{\pi_t}^t$ is the posterior probability of output symbol π_t at time t. Equation (1) can be efficiently evaluated and differentiated using forward-backward algorithm [22]. Given training utterances, the acoustic model networks are trained to minimize the CTC objective function:

$$\mathcal{L} = -\sum_{k=1}^{Q} \ln(P(\mathbf{z_k}|\mathbf{x_k})) \quad (2)$$

where k is the index of training utterances and Q is the total number.

3. MULTI-GRAPH DECODING STRATEGY

In modern ASR architectures, weighted finite-state transducers (WFST) are used to integrate different knowledge sources and perform search space optimization to achieve the best search efficiency using highly-optimized FST libraries such as OpenFST [23, 24]. In E2E CTC ASR framework [25], individual components, containing CTC labels, lexicons, and N-gram language models, are encoded into three individual WFSTs and then composed into a comprehensive search graph that encodes the mapping from a CTC symbol sequence emitted from the speech frames to a sequence of words. The search space is represented as $\mathbf{T} \circ \mathbf{L} \circ \mathbf{G}$ in the Eesen toolkit [25], where $\mathbf{T}$ is a *token* WFST that maps a sequence of frame-level symbols to a single lexicon unit, $\mathbf{L}$ is a lexicon WFST that encodes the mapping from sequences of lexicon units to words, and $\mathbf{G}$ is a grammar WFST that encodes the word sequences information in N-gram language model. Thus, using WFST-based decoding framework, we can incorporate different word-level language model efficiently to make full use of the available textual resources and overcome the imbalance in acoustic data between low-resourced and high-resourced language in our CS scenario.

In our previous work [26], Yilmaz et al. proposed a multi-graph decoding strategy which creates parallel search spaces for each monolingual and bilingual recognition tasks for the conventional CS ASR system. This strategy can be easily extended to E2E CTC ASR system to address the above-mentioned data imbalance problem. For the multi-graph decoding strategy, we use the union operation to create a larger graph with parallel bilingual and monolingual (Frisian and Dutch) subgraphs. The parallel graphs used during decoding are characterized by the incorporated language model component, as they share the same token ($\mathbf{T}$) and lexicon ($\mathbf{L}$) components. This approach has been shown to outperform standard LM interpolation [26], that makes effective use of the text resources of the high-resourced language by creating three different search spaces with an identical acoustic model (AM). Monolingual and code-mixed utterances are decoded using best-matching subgraph, yielding improved monolingual recognition performance on the high-resourced language without any accuracy loss on the code-mixed utterances.

4. CS RNN LANGUAGE MODELING

In language modeling, we face data sparsity both in terms of availability of CS corpus, and scarcity of CS occurrences in the corpus. To address these problems, we propose a two-step approach to language modeling. Firstly, in terms of data augmentation, we boost the size of CS corpus by synthetically

generating CS text using a well-trained long short-term memory (LSTM) language model. Similar techniques are also proposed in [27, 28]. However, in [27] a sentence level aligned parallel corpus is available, thus synthetic CS data can be generated based on word or phrase alignment between the parallel sentences and guided by linguistic rules. Unlike [27], we lack a parallel corpus, thus we cannot explicitly establish the word-level cross-lingual correspondence between the two languages. This motivates the second step of our language model, i.e., to find the cross-lingual mapping of the monolingual word embeddings using an unsupervised self-learning method proposed by [29]. The method finds the mapping functions W_M, W_N that maximize the cosine similarity between the monolingual embeddings of source language M and target language N, based on an iteratively learned dictionary D:

$$\arg \max_{W_M, W_N} \sum_{i,j \in D} (M_i W_M) \cdot (N_j W_N) \tag{3}$$

i, j are paired entries in the dictionary that represent a translation pair and M_i, N_j are the respective monolingual embeddings. Since the transformation matrices and embeddings are length normalized, cosine similarity is optimized. Thus, the method explicitly aligns the word based on the monolingual distributional property and projects both monolingual embedding into the same embedding space. Resultant word embeddings of the related words in both languages are grouped together and at the same time, monolingual syntactic information is preserved [21].

$$y_k = LSTM(w_k) \tag{4}$$

$$p_k = \frac{e^{y_k}}{\sum_{j=1}^{V} e^{y_j}} \tag{5}$$

$$Loss = -\frac{1}{Q-1} \sum_{k=1}^{Q-1} Y_{k+1} \ln(p_k) \tag{6}$$

This pre-trained cross-lingual embedding is used to initialize our neural language model and the embedding layer is fixed during training. The output y_k from LSTM with the current word embedding w_k is passed through a softmax function Eq. (5) to form a distribution p_k over the total vocabulary V, which represents the next word probability. The loss function is the cross-entropy between the true target Y_{k+1} and p_k in Eq. (6), where Q is the number of words in the corpus. By freezing the embedding layer, we aim to preserve the cross-lingual correspondence derived from larger monolingual corpora and let the low-resourced language leverage on the resource rich language.

5. EXPERIMENTAL SETUP

5.1. Datasets

The experiments are conducted on the low-resourced Frisian-Dutch CS corpus from the FAME! project, this project aims to develop a spoken document retrieval system for the disclosure of the archives of Omrop Fryslân (Frisian Broadcast) covering a large time span and a wide variety of topics which contain monolingual Dutch and Frisian speech as well as code-mixed Frisian-Dutch speech. Further details can be found in [30]. It is worth mentioning that proposed approaches can also be applied to other low-resourced language pairs and scenarios with more than two languages as in [31].

Table 1. Acoustic data composition used for CTC AM training (in hours)

Traning data	Annot.	Frisian	Dutch	Total
(1) FAME	Manual	8.5	3.0	11.5
(2) Frisian Broad.	Auto.	125.5		125.5
(3) CGN-NL	Manual	-	442.5	442.5

The training data used in the experiments are summarized in Table 1. Both monolingual and CS data is used for acoustic model training, since monolingual acoustic data augmentation has been shown to improve the CS ASR on both monolingual and code-mixed test utterances [32]. The manually annotated CS data is from the FAME corpus containing 8.5 hours and 3 hours of orthographically transcribed speech from Frisian (fy) and Dutch (nl) speakers respectively. The 'Frisian Broadcast' data containing 125.5 hours of automatically transcribed speech data extracted from the target broadcast archive. Monolingual Dutch data comprises 442.5 hours Dutch component of the Spoken Dutch Corpus (CGN) [33] that contains diverse speech materials including conversations, interviews, lectures, debates, read speech and broadcast news. The development and test sets consist of 1 hour of speech from Frisian speakers and 20 minutes of speech from Dutch speakers each. The sampling frequency of all speech data is 16 kHz.

5.1.1. Text data

Bilingual text corpus (107M words) consisting of generated CS text (61M words), monolingual Frisian text (37M words) and monolingual Dutch text (9M words) are used for training the baseline CS LM. The transcripts of the FAME training data is the only source of CS text containing 140k words and textual data augmentation techniques described in [32] have been applied to increase the amount of CS text. The Frisian text is extracted from monolingual resources such as Frisian novels, news and Wikipedia articles. The Dutch text is extracted from the transcripts of the CGN speech corpus. We use the larger monolingual subset (300M words) of the NLCOW text corpus[1] together with Dutch text (9M words) which is used in baseline CS LM to train larger Dutch LM and create larger monolingual Dutch graph.

[1] http://corporafromtheweb.org

5.2. Implementation details

All the recognition experiments are performed in the Eesen E2E CTC ASR toolkit [25]. The 3-fold data augmentation [34] is applied to the in-domain acoustic training data, i.e., (1) and (2) in Table 1. The acoustic model is a 6-layer bidirectional LSTM with 640 hidden units trained without predefined alignment. The 40-dimensional filterbank features with their first and second-order derivatives are stacked using 3 contiguous frames to form 360-dimensional spliced features as inputs. The features are normalized via mean subtraction and variance normalization on a per-speaker basis. The learning rates starts at 0.00004 and remains unchanged until the drop of label error rate on validation set between two consecutive epochs falls below 0.5%. From then on, the learning rate is halved at the subsequent epochs. The conventional ASR system is trained using the Kaldi ASR toolkit [35]. A context-dependent Gaussian mixture model-hidden Markov model (GMM-HMM) system is firstly trained using MFCC including the deltas and deltas-deltas to obtain the alignments. Then these alignments are used for training a TDNN-LSTM acoustic model (1 standard, 6 time-delay and 3 LSTM layers) with LF-MMI [36] criterion using 40-dimensional MFCC as features combined with i-vectors for speaker adaptation.

The language models used in the first pass ASR decoding are standard bilingual 3-grams with interpolated Kneser-Ney smoothing. The baseline RNN LM with gated recurrent units (GRU) has 400 hidden units and is trained using noise contrastive estimation[2] for lattice rescoring. The CS RNN LM with the same architecture is adapted to the CS transcripts to reduce the mismatch. The adaptation is performed at the last 5 epochs while following the overall learning rate decay of 0.8. In summary, we have 7 LMs: (1) baseline CS LM (cs) trained on the bilingual text (107M), (2) baseline monolingual Frisian LM (fy) trained on monolingual Frisian text (37M), (3) baseline monolingual Dutch LM (nl) trained on monolingual Dutch text (9M), (4) larger monolingual Dutch LM (nl++) trained on 309M words, (5) interpolated LM (interp-nl++) with the interpolation between cs LM and nl++ LM, whose interpolation weight yields the lowest perplexity on the development set, (6) baseline RNN LM trained on the corresponding bilingual text (107M) using 1 layer LSTM with 400 hidden units, (7) CS RNN LM trained using the similar parameters. The RNN LM weight for rescoring is 0.75. The first five LMs are used in the conventional signal-graph E2E ASR systems for comparison with the corresponding multi-graph decoding systems using the same amount monolingual and bilingual text. The perplexities of the baseline CS and the Dutch LMs on the monolingual Dutch component of the development and test set are shown in Table 2, the perplexities of two RNN LMs on development and test set show that CS RNN LM has a lower perplexity than its baseline in Table 3.

[2]https://github.com/yandex/faster-rnnlm

Table 2. Perplexities obtained on the Dutch component of the development and test set using different LMs

LM	Total # words	Dev.	Test
Baseline CS LM	107M	188	197
interp-nl++ LM	416M	176	182
Baseline NL LM	9M	150	151
nl++ LM	309M	123	119

Table 3. Perplexities obtained on the different components of development and test transcripts using different LMs

	Dev.			Test		
	fy	nl	cs	fy	nl	cs
3-gram LM	158	191	272	138	189	227
Base. RNN LM	205	187	330	177	177	283
CS RNN LM	183	164	296	159	156	257

Table 4. WER (%) obtained on the monolingual utterances in the development and test set of the FAME Corpus

		Dev.		Test	
		fy	nl	fy	nl
# of Frisian words		9190	0	10753	0
# of Dutch words		0	4569	0	3475
ASR System	Graph				
Baseline CS ASR	cs	32.9	33.7	30.6	29.0
fy	fy	32.5	-	30.8	-
nl	nl	-	33.6	-	27.9
nl++	nl++	-	**30.1**	-	**25.9**

5.3. ASR experiments

Four sets of ASR experiments are conducted to evaluate the performance of the proposed method. Firstly, the ASR performance of the baseline single-graph ASR systems using cs and interp-nl++ LMs are presented. Secondly, the results provided by the bi-graph systems using the cs graph together with one of the monolingual graphs, namely fy, nl and nl++, are presented. Thirdly, tri-graph decoding systems with varying monolingual graphs are evaluated.

After finalizing the multi-graph decoding experiments, we present the RNN LM rescoring experiment performed to evaluate the performance of CS RNN LM on CS speech compared to a baseline RNN LM. For the rescoring of the multi-graph systems, graph identification tags are used to identify the graph used for the hypothesized ASR output and then the rescoring is performed with the corresponding RNN LM. The CS RNN LMs are trained on the same text data with the N-gram used in decoding. The monolingual Frisian and Dutch RNN LMs are trained on Frisian text corpora (fy, 37M) and the largest Dutch text corpora (nl++, 309M) respectively using the same parameters as the baseline and CS RNN LMs. The recognition results are reported separately for Frisian

Table 5. WER (%) obtained on the development and test set of the FAME Corpus

			Dev.				Test				Total
			fy	nl	fy-nl	all	fy	nl	fy-nl	all	
# of Frisian words			9190	0	2381	11 571	10 753	0	1798	12 551	24 122
# of Dutch words			0	4569	533	5102	0	3475	306	3781	8883
ASR System	Graph(s)	Rescoring									
Kaldi CS ASR	cs	No	26.3	27.6	36.8	28.4	25.1	24.4	39.3	26.7	27.6
Single-graph systems											
Base. E2E CS ASR	cs	No	32.9	33.7	42.6	34.9	30.6	29.0	42.4	31.8	33.4
Base. E2E CS ASR	cs	Yes	31.6	32.8	42.1	33.9	29.6	**27.9**	**40.7**	30.7	32.3
Base. E2E CS ASR	cs	CS-RNN	30.4	**31.2**	**41.0**	32.5	29.0	28.6	41.2	30.6	**31.6**
interp-nl++	cs-nl++	No	32.6	32.3	42.3	34.3	30.7	28.7	42.6	31.8	33.1
interp-nl++	cs-nl++	Yes	31.3	32.5	41.5	33.4	29.9	28.2	41.0	31.0	32.2
Multi-graph systems											
union-fy	cs, fy	No	32.7	33.0	42.3	34.5	30.7	28.6	42.7	31.9	33.2
union-nl	cs, nl	No	32.7	32.5	42.8	34.4	30.6	28.0	42.6	31.6	33.0
union-nl++	cs, nl++	No	32.8	30.1	42.5	33.8	30.6	26.7	42.5	31.4	32.6
union-nl++	cs, nl++	Yes	31.9	**28.4**	**42.1**	32.8	29.7	**23.6**	**42.4**	30.1	**31.5**
union-fy-nl	cs, fy, nl	No	32.9	32.4	42.9	34.6	30.8	28.1	42.8	31.8	33.2
union-fy-nl++	cs, fy, nl++	No	32.9	30.1	42.8	33.9	30.8	25.6	43.1	31.3	32.5
union-fy-nl++	cs, fy, nl++	Yes	32.3	28.2	41.7	32.8	30.2	23.1	41.3	30.2	31.6
union-fy-nl++	cs, fy, nl++	CS-RNN	32.3	**28.2**	**40.5**	32.6	30.2	**23.1**	**40.5**	30.1	**31.4**

only (fy), Dutch only (nl) and code-mixed (fy-nl) utterances. The overall performance is also reported to use as an overall performance indicator. The recognition performance of the ASR system is quantified using the word error rate (WER).

6. RESULTS AND DISCUSSION

The recognition results obtained by using only monolingual graphs on the corresponding monolingual utterances are presented in Table 4. The ASR system using only Frisian (fy) graph gives similar recognition performance to the baseline CS system on monolingual Frisian utterances, which indicates that the latter CS system has the ability to recognize monolingual Frisian speech as well as a monolingual Frisian ASR system. For monolingual Dutch utterances, the performance by using only Dutch (nl) graph is slightly better than baseline CS system on the test set with a WER of 27.9% compared to 29.0%. Using the largest monolingual Dutch graphs nl++ yields a WER of 25.9% on the Dutch utterances respectively, revealing that the performance of the baseline CS graph can be improved by using larger monolingual Dutch graph in a multi-graph decoding framework.

The ASR results obtained using multi-graph decoding strategy and the CS RNN LM rescoring are presented in Table 5. The number of Frisian and Dutch words in each component of development and test sets are presented in the upper panel. Then two baseline results using single-graph systems (cs and interp-nl++) are shown in the middle panel. The results provided by an equivalent Kaldi [35] ASR system with conventional architecture is also given as a reference. Compared to the baseline E2E CS ASR system, using the interpolated larger Dutch LM brings marginal improvements from 33.7% (29.0%) to 32.3% (28.7%) on the development (test) set. This indicates that using interpolated larger LM in single graph is ineffective in improving the accuracy on monolingual utterances.

Finally, the ASR results provided by the multi-graph E2E ASR systems are presented in the bottom panel. According to these results, using an additional monolingual Frisian graph during the multi-graph decoding (union-fy and union-fy-nl) does not improve the ASR performance on the fy utterances, which is consistent with the previous results reported in [26]. Including the largest monolingual Dutch graph in the union-fy-nl++ system improves the ASR accuracy on nl utterances with a WER of 30.1% (25.6%), yielding a 10.7% (11.7%) relative WER reduction.

For RNN LM rescoring, CS RNN LM provides absolute overall 0.7% WER reduction from 32.3% to 31.6% over the baseline RNN LM in single-graph systems and 1.2% (0.8%) WER reduction on fy-nl utterances for the union-fy-nl++ system perhaps due to the fact that the CS RNN LM could preserve more cross-lingual information. The Dutch RNN LM (trained on 309M Dutch text corpora) provides the best WER of 28.2% (23.1%) on monolingual Dutch utterances, while

the Frisian RNN LM (trained on 37M Frisian text) and the baseline RNN LM (trained on 107M bilingual text) give limited improvements on the corresponding subsets. Finally, the WER of E2E CTC ASR system is significantly reduced to 31.4%.

7. CONCLUSION

In this paper, we propose an E2E CTC ASR pipeline for a CS scenario in which a low-resourced language is mixed with a high-resourced language. We first incorporate a multi-graph decoding strategy by creating parallel search spaces for monolingual and code-switching recognition tasks. Moreover, we perform language model rescoring using a recurrent neural network pre-trained with cross-lingual embedding and then adapted with the limited amount of in-domain code-switching text. For evaluating the effectiveness of the proposed pipeline, ASR experiments are conducted on the Frisian-Dutch CS speech, in which the target Frisian language is low-resourced with limited acoustic and textual resources while Dutch language is high-resourced. The experimental results demonstrate that the multi-graph decoding approach can improve monolingual Dutch recognition performance of an E2E CS ASR system without degradation in the CS performance. The adapted recurrent neural network language model further improves the performance on CS speech. Finally, the proposed pipeline gives 16.3% (20.3%) relative WER reduction on monolingual Dutch speech and absolute 2.1% (1.9%) WER reduction on code-switching speech.

8. ACKNOWLEDGEMENTS

This research is supported by the National Research Foundation Singapore under its AI Singapore Programme (Award Number: AISG-100E-2018-006). This research is also supported by the Agency for Science, Technology and Research (A*STAR) under its AME Programmatic Funding Scheme (Project #A18A2b0046). This research is partially supported by the Key Program of National Natural Science Foundation of China (No.61933002) and the National Key Research and Development Program of China (No.2018YFB1309300).

9. REFERENCES

[1] Colin Baker, "Foundations of bilingual education and bilingualism," *Multilingual matters*, vol. 79, 2011.

[2] Sunayana Sitaram, Khyathi Raghavi Chandu, Sai Krishna Rallabandi, and Alan W Black, "A survey of code-switched speech and language processing," arXiv preprint arXiv:1904.00784, 2019.

[3] Peter Auer, "Code-switching in conversation: Language, interaction and identity," *Journal of Linguistics*, vol. 37, pp. 627–649, 2001.

[4] Dau-Cheng Lyu, Tien-Ping Tan, Eng-Siong Chng, and Haizhou Li, "Seame: a mandarin-english code-switching speech corpus in south-east asia," in *Eleventh Annual Conference of the International Speech Communication Association (INTERSPEECH)*. ISCA, 2010, pp. 1986–1989.

[5] Alfredo Ardila, "Spanglish: an anglicized spanish dialect," *Hispanic Journal of Behavioral Sciences*, vol. 27, pp. 60–81, 2005.

[6] Anik Dey and Pascale Fung, "A hindi-english code-switching corpus," in *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC)*. ELRA, 2014, pp. 2410–2413.

[7] Alex Graves, "Sequence transduction with recurrent neural networks," in *arXiv preprint arXiv:1211.3711*, 2012.

[8] Jan Chorowski, Dzmitry Bahdanau, Dmitriy Serdyuk, Kyunghyun Cho, and Yoshua Bengio, "Attention-based models for speech recognition," in *Proceedings of the 28th International Conference on Neural Information Processing Systems (NIPS)*. ACL, 2015, pp. 577–585.

[9] 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," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. ACL, 2014, pp. 1724–1734.

[10] Alex Graves and Navdeep Jaitly, "Towards end-to-end speech recognition with recurrent neural networks," in *Proceedings of the 31st International Conference on Machine Learning (ICML)*. ACM, 2014, pp. 1764–1772.

[11] Shinji Watanabe, Takaaki Hori, Suyoun Kim, John R. Hershey, and Tomoki Hayashi, "Hybrid CTC/attention architecture for end-to-end speech recognition," *IEEE Journal of Selected Topics in Signal Processing*, vol. 11, no. 8, pp. 1240–1253, Dec 2017.

[12] Eric Battenberg, Jitong Chen, Rewon Child, Adam Coates, Yashesh Gaur, Yi Ci Li, Hairong Liu, Sanjeev Satheesh, David Seetapun, Anuroop Sriram, and Zhenyao Zhu, "Exploring neural transducers for end-to-end speech recognition," in *Automatic Speech Recognition and Understanding (ASRU), 2017 IEEE Workshop on*. IEEE, 2017, pp. 206–213.

[13] William Chan, Navdeep Jaitly, Quoc V. 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.

[14] William Chan, Navdeep Jaitly, Quoc V. Le, and Oriol Vinyals, "Neural speech recognizer: Acoustic-toword lstm model for large vocabulary speech recognition," in *Eighteenth Annual Conference of the International Speech Communication Association (INTERSPEECH)*. ISCA, 2017, pp. 3707–3711.

[15] Jinyu Li, Guoli Ye, Amiya Das, Rui Zhao, and Yifan Gong, "Advancing acoustic-to-word ctc model," in *Acoustics, Speech and Signal Processing (ICASSP), 2018 IEEE International Conference on*. IEEE, 2018, pp. 5794–5798.

[16] Changhao Shan, Chao Weng, Guangsen Wang, Dan Su, Min Luo, Dong Yu, and Lei Xie, "Investigating end-to-end speech recognition for mandarin-english code-switching," in *Acoustics, Speech and Signal Processing (ICASSP), 2019 IEEE International Conference on*. IEEE, 2019, pp. 6056–6060.

[17] Hiroshi Seki, Shinji Watanabe, Takaaki Hori, Jonathan Le Roux, and John R. Hershey, "An end-to-end language-tracking speech recognizer for mixed-language speech," in *Acoustics, Speech and Signal Processing (ICASSP), 2018 IEEE International Conference on*. IEEE, 2018, pp. 4919–4923.

[18] Suyoun Kim and Michael L. Seltzer, "Towards language-universal end-to end speech recognition," in *Acoustics, Speech and Signal Processing (ICASSP), 2018 IEEE International Conference on*. IEEE, 2018, pp. 4914–4918.

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

[20] Ke Li, Jinyu Li, Guoli Ye, Rui Zhao, and Yifan Gong, "Towards code-switching ASR for end-to-end CTC models," in *Acoustics, Speech and Signal Processing (ICASSP), 2019 IEEE International Conference on*. IEEE, 2019, pp. 6076–6080.

[21] Grandee Lee and Haizhou Li, "Word and class common space embedding for code-switch language modelling," in *Acoustics, Speech and Signal Processing (ICASSP), 2019 IEEE International Conference on*. IEEE, 2019, pp. 6086–6090.

[22] Alex Graves, Santiago Fernandez, Faustino Gomez, and Jurgen Schmidhuber, "Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks," in *Proceedings of the 23rd International Conference on Machine Learning (ICML)*, 2006, pp. 369–376.

[23] Mehryar Mohri, Fernando C. N. Pereira, and Michael Riley, "Speech recognition with weighted finite-state transducers," *Computer Speech & Language*, vol. 16, pp. 69–88, 2002.

[24] Cyril Allauzen, Michael Riley, Wojciech Skut Johan Schalkwyk, and Mehryar Mohri, "Openfst: A general and efficient weighted finite-state transducer library," in *Implementation and Application of Automata*. Springer, 2007, pp. 11–23.

[25] Yajie Miao, Mohammad Gowayyed, and Florian Metze, "Eesen: End-to-end speech recognition using deep rnn models and wfst-based decoding," in *Automatic Speech Recognition and Understanding (ASRU), 2017 IEEE Workshop on*. IEEE, 2017, pp. 167–174.

[26] Emre Yılmaz, Samuel Cohen, Xianghu Yue, David van Leeuwen, and Haizhou Li, "Multi-graph decoding for code-switching ASR," in *Twentieth Annual Conference of the International Speech Communication Association (INTERSPEECH)*. ISCA, 2019, pp. 3750–3754.

[27] Grandee Lee, Xianghu Yue, and Haizhou Li, "Linguistically motivated parallel data augmentation for code-switch language modeling," in *Twentieth Annual Conference of the International Speech Communication Association (INTERSPEECH)*. ISCA, 2019, pp. 3730–3734.

[28] Emre Yılmaz, Henk van den Heuvel, and David van Leeuwen, "Code-switching detection with data-augmented acoustic and language models," in *the Sixth International Workshop on Spoken Language Technology for Under-resourced Languages (SLTU)*. Procedia Computer Science, 2018, pp. 127–131.

[29] Artetxe Mikel, Labaka Gorka, and Agirre Eneko, "A robust self-learning method for fully unsupervised cross-lingual mappings of word embeddings," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*. ACL, 2018, pp. 789–798.

[30] Emre Yılmaz, Maaike Andringa, Sigrid Kingma, Jelske Dijkstra, Frits Van der Kuip, Hans Van de Velde, Frederik Kampstra, Jouke Algra, Henk van den Heuvel, and David A. van Leeuwen, "A longitudinal bilingual frisian-dutch radio broadcast database designed for code-switching research," in *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC)*. ELRA, 2016, pp. 4666–4669.

[31] Emre Yılmaz, Astik Biswas, Febe de Wet, Ewald Van der Westhuizen, and Thomas Niesler, "Building a unified code-switching ASR system for South African languages," in *Nineteenth Annual Conference of the International Speech Communication Association (INTERSPEECH)*. ISCA, 2018, pp. 1923–1927.

[32] Emre Yılmaz, Henk van den Heuvel, and David van Leeuwen, "Acoustic and textual data augmentation for improved ASR of code-switching speech," in *Nineteenth Annual Conference of the International Speech Communication Association (INTERSPEECH)*. ISCA, 2018, pp. 1933–1937.

[33] Nelleke Oostdijk, "The spoken dutch corpus: Overview and first evaluation," in *Proceedings of the Second International Conference on Language Resources and Evaluation (LREC)*. ELRA, 2000, pp. 886–894.

[34] Tom Ko, Vijayaditya Peddinti, Daniel Povey, and Sanjeev Khudanpur, "Audio augmentation for speech recognition," in *Sixteenth Annual Conference of the International Speech Communication Association (INTERSPEECH)*. ISCA, 2015, pp. 3586–3589.

[35] Daniel Povey, Arnab Ghoshal, Gilles Boulianne, Nagendra Goel, Mirko Hannemann, Yanmin Qian, Petr Schwarz, and Georg Stemmer, "The Kaldi speech recognition toolkit," in *Automatic Speech Recognition and Understanding (ASRU), 2011 IEEE Workshop on*, 2011, pp. 1–4.

[36] Daniel Povey, Vijayaditya Peddinti, Daniel Galvez, Pegah Ghahremani, Vimal Manohar, Xingyu Na, Yiming Wang, and Sanjeev Khudanpur, "Purely sequence-trained neural networks for asr based on lattice-free mmi," in *Seventeenth Annual Conference of the International Speech Communication Association (INTERSPEECH)*. ISCA, 2016, pp. 2751–2755.