Reading group : Code Switching ASR

Dan Berrebbi

October 26th 2021

Presentation

- First year Masters' student at LTI-CMU
- Questions and Feedbacks are welcome anytime

Introduction

Definition

"Code switching occurs when a word or a phrase in one language subsitutes for a word or a phrase in a second language"

Je vais faire une présentation sur les End to End code switching models.

Who ? bilinguals (more than 60%)

Why code switching ?

- Bilinguals code switch because they do not know either language completely $\rightarrow$ code switch to compensate a lack of proficiency.
- Code-switch as a strategy in order to be better understood

Two important distinctions

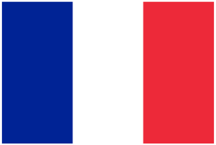
Code switching can be :

- extra sentential (T'es où? In the kitchen?)
- intra sentential (I need to take some vacaciones)
  - Insertional CS (only borrowed words → extend the vocab)
  - Alterational CS (grammar from the two languages, more challenging)

Bilinguals have :

- a preferred language (matrix language)
- a secondary language (embedded language)

Code Switching is part of our daily life (songs ...)



orz

It is a Japanese based emoticon of a man pounding his head on the floor.
The o is the head.
The r is the arms.
The z is the legs.
Used to symbolize the emotion of frustration.

Our puller is a complete n00b... he just pulled a soldier crawler and now we have three adds... orz...

Other properties

- spoken language
- 2 languages
  - Bilinguals code switch because they do not know either language completely $\rightarrow$ code switch to compensate a lack of proficiency.
  - Code-switch as a strategy in order to be better understood

What type of ASR architecture for code-switching ?

- GMM/DNN-HMM, hybrid ... $\rightarrow$ huge effort on lexicon, pronunciation dictionnaries ...
- End-to-End $\rightarrow$ need descent quantity of data

Dataset : SEAME corpus

Convertiational Bilingual (Mandarin-english) speech corpus.
Mandarin is the matrix language of speakers

Table 1. Data Statistics of SEAME [3]

Set	Speakers	Hours	Duration Ratio (%)		
			Man	En	CS
$train$	134	101.13	16	16	68
$test_{man}$	10	7.49	14	7	79
$test_{sge}$	10	3.93	6	41	53

Outline

1. LID task for code-switching problems
2. Data scarcity issue :
    - data augmentation methods
    - "monolingual" training

4. Lab papers and discussion

LID for code-switching

Detect the switching points

First idea that we have : detect the boundaries of languages

DETECTION OF LANGUAGE BOUNDARY IN CODE-SWITCHING UTTERANCES BY BI-PHONE PROBABILITIES

Joyce Y. C. CHAN, P. C. CHING*, Tan LEE*, Helen M. MENG***

*Department of Electronic Engineering
**Department of Systems Engineering and Engineering Management,
The Chinese University of Hong Kong, Hong Kong
{ycchan, pcching, tanlee}@ee.cuhk.edu.hk, hmmeng@se.cuhk.edu.hk

Dataset of cantonese with english words inserted, study led in 2004, score of 75%.

Detect switching points : a human approach

Sentences conatining code-switched words take longer to read and comprehend

Two mental lexicons : we need to determine which one is on and which one is off (Macnamara and Kushnir, 1971)

English : CC and CV whereas Mandarin : CV but no CC

Study (1996) : Chinese bilinguals' processing of English words $\rightarrow$ took longer to recognize English code-switched words containing CV than CC.

("Bilingual Language Mixing : Why do Bilinguals Code-Switch ?", Heredia and Altarriba)

Detect the switching points → use monolingual models

- detect switching points
- segment utterances
- apply monolingual/multilingual models on infered monolingual spans

Advantages : quite straightforward
Inconvenients : loosing context in particular in real world case of CS + error propagation

Chosen approach : Multi-task learning setup with (character/subword) Language Identification as a task (LID)

Principle

- baseline ASR model : $\mathcal{L}_{MODEL}$
- proposed model for CS : $\mathcal{L}_{MTL} = \mathcal{L}_{MODEL} + \mathcal{L}_{LID}$

Examples of LID usage

TOWARDS END-TO-END CODE-SWITCHING SPEECH RECOGNITION

Ne Luo, Dongwei Jiang*, Shuaijiang Zhao*, Caixia Gong*, Wei Zou, Xiangang Li*

AI Labs, Didi Chuxing, Beijing, China
{luone_i, jiangdongwei, zhaoshuaijiang, gongcaixia, zouwei, lixiangang}@didiglobal.com

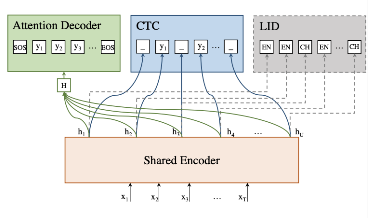


Table 4. MERs (%) on the development set (Dev) and test set (Test) of SEAME. λ_{LID} in the table represents the weight of LID loss in LID-MTL, while $\lambda_{Att} = 0.8$, $\lambda_{CTC} = 0.2 - \lambda_{LID}$.

Model	λ_{LID}	Dev	Test
Att + CTC	-	35.44	37.83
LID-Label	-	35.48	37.98
LID-MTL	0.05	34.45	37.03
LID-MTL	0.10	34.13	**36.48**
LID-MTL	0.20	35.43	37.82

Examples of LID usage

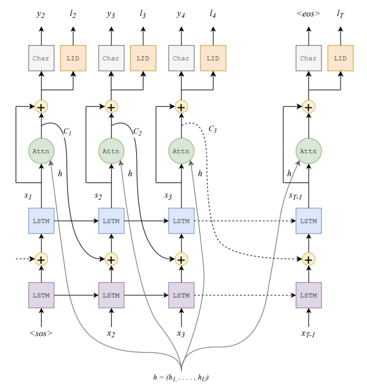


Figure 1: *Architecture of MTL-HARD decoder model. The first LSTM encodes the sub-sequence history and captures longer context, the second LSTM takes the character-level information and combines it with first context.*

Towards Context-Aware End-to-End Code-Switching Speech Recognition

Zimeng Qiu[†], *Yiyuan Li*[‡], *Xinjian Li*[‡], *Florian Metze*[‡], *William M. Campbell*[†]

† Amazon Alexa AI
‡ Carnegie Mellon University
{zimengqi, cmpw}@amazon.com {yiyuanli, xinjianl, fmetze}@andrew.cmu.edu

Model	**MER (%)**	**CER (%)**			
		All	CS	CN	EN
LAS	38.99	29.91	28.31	34.34	30.29
HARD	35.42	27.23	25.65	32.39	24.90
LAS + LID	34.93	27.06	25.82	31.68	25.22
HARD + LID	**34.50**	**26.56**	**25.54**	**30.44**	**24.78**

Data augmentation methods

Principle

Problem : End-to-end models needs large amounts of data

Proposed Solution : Generate artificial CS data

Extra-sentential augmentation

An End-to-End Language-Tracking Speech Recognizer for Mixed-Language Speech

Seki, H.; Watanabe, S.; Hori, T.; Le Roux, J.; Hershey, J.R.

Algorithm 1 Generation of code-switching corpus

$N_{\text{concat}} \Leftarrow$ maximum number of utterances to concatenate.
$N \Leftarrow$ number of languages.
$D \Leftarrow$ duration of the union of the original corpora.
$n_{\text{reuse}} \Leftarrow$ maximum number of times same utterance can be used.
for $i \leftarrow 1$ to N **do**
  $P(\text{lang}_i) = \frac{1}{2} \frac{\text{duration of lang}_i}{\sum_j \text{duration of lang}_j} + \frac{1}{2N}$
  $P(\text{utter}_{\text{lang}_i,k}) = \frac{1}{\text{number of utterances in lang}_i}$
end for
while duration(*generated corpus*) $\leq D$ **do**
  for $n_{\text{concat}} \leftarrow 1$ to N_{concat} **do**
    for $i \leftarrow 1$ to n_{concat} **do**
      Sample language lang_i and utterance $\text{utter}_{\text{lang}_i,k}$, resampling if $\text{utter}_{\text{lang}_i,k}$ already selected n_{reuse} times.
    end for
    Concatenate n_{concat} utterances.
    Add to *generated corpus*.
  end for
end while

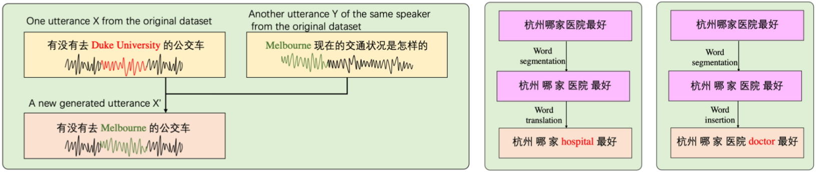
For real-world CS scenarios we need a method to produce intra-sentential CS corpus.

Intra-sentential augmentation : principle

DATA AUGMENTATION FOR END-TO-END CODE-SWITCHING SPEECH RECOGNITION

Chenpeng Du[1], *Hao Li*[1], *Yizhou Lu*[1], *Lan Wang*[2], *Yanmin Qian*[1]

[1] MoE Key Lab of Artificial Intelligence
SpeechLab, Department of Computer Science and Engineering
AI Institute, Shanghai Jiao Tong University, Shanghai, China
[2] CAS Key Laboratory of Human-Machine Intelligence-Synergy Systems,
Shenzhen Institutes of Advanced Technology



(a) Audio splicing (b) Word translation (c) Word insertion

Fig. 2. Augmented examples of the proposed code-switching data augmentation approaches

Intra-sentential augmentation : techniques involved

Baseline ASR : Transformer-based sequence-to-sequence model + ctc

$$L_{ASR} = -\alpha \cdot \log p_{s2s}(\mathbf{y};\mathbf{x}) - (1-\alpha) \cdot \log p_{ctc}(\mathbf{y};\mathbf{x})$$

TTS model : Fastspeech architecture, trained with original CS dataset

Dataset : ASRU 2019 CS ASR

Audio Splicing : GMM-HMM and Viterbi beam search for alignments

Intra-sentential augmentation : Experiments

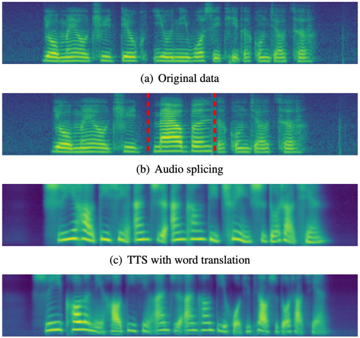
Table 1. WER (%) on Mandarin-English test dataset of baseline systems

Data Augmentation	**WER**		
	CN	**EN**	**TOTAL**
None	11.15	33.31	13.56
Speed Perturb	10.86	32.77	13.23
Monolingual TTS	11.13	31.61	13.35
SpecAug	9.60	30.18	11.84

Table 2. WER (%) on Mandarin-English test dataset of proposed methods

Data Augmentation	**WER**		
	CN	**EN**	**TOTAL**
None	11.15	33.31	13.56
Audio Splicing	10.75	31.74	13.02
+ SpecAug	9.23	28.81	11.36
Word translation with TTS	10.61	31.25	12.85
+ SpecAug	8.70	28.18	10.81
Word insertion with TTS	10.54	32.11	12.88
+ SpecAug	8.51	28.17	10.65
All three proposed + SpecAug	**8.29**	**26.74**	**10.29**

Intra-sentential augmentation : other evaluations



(a) Original data

(b) Audio splicing

(c) TTS with word translation

(d) TTS with word insertion

Fig. 3. Mel spectrograms of the augmented samples

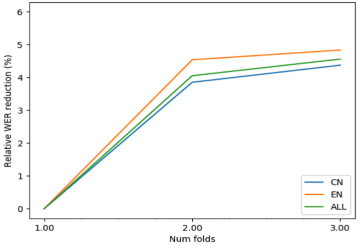


Fig. 4. Relation between WER relative reduction and the amount of augmented data using the proposed audio splicing approach.

Intra-sentential augmentation : discussion

- quite straightforward idea
- not SEAME, quite hard to compare (see next example)
- limitation : too small data , but they train a TTS with this data ...
- limitation : random insertion and splicing
- discussion ?

”Monolingual” methods

Principle

Problem : lack of CS data for End-to-End models

Solution : train on only on non CS utterances

New problem : when decoding intra-sentential CS, how to switch from one language to another ?

Solution : Forcing output token vectors to be close between different languages

Articles

Constrained Output Embeddings for End-to-End Code-Switching Speech Recognition with Only Monolingual Data

Yerbolat Khassanov[1], *Haihua Xu*[2], *Van Tung Pham*[1,2], *Zhiping Zeng*[2], *Eng Siong Chng*[1,2], *Chongjia Ni*[3] *and Bin Ma*[3]

[1]School of Computer Science and Engineering, Nanyang Technological University, Singapore
[2]Temasek Laboratories, Nanyang Technological University, Singapore
[3]Machine Intelligence Technology, Alibaba Group

{yerbolat002,haihuaxu,vantung001,zengzp,aseschng}@ntu.edu.sg,
{ni.chongjia,b.ma}@alibaba-inc.com

TRAINING CODE-SWITCHING LANGUAGE MODEL WITH MONOLINGUAL DATA

Shun-Po Chuang, Tzu-Wei Sung, Hung-yi Lee

National Taiwan University

Principle

Hypothesis : the difference between output token distributions of monolingual languages restricts the E2E CS ASR model from switching between languages.

Proposed approach : make the token distributions similar in the embedding space

Methods (1.1)

$$\mathcal{L}_{\text{MTL}} = \lambda \mathcal{L}_{\text{CTC}} + (1-\lambda)(\alpha \mathcal{L}_{\text{ATT}} + (1-\alpha)\mathcal{L}_{\text{constraint}})$$

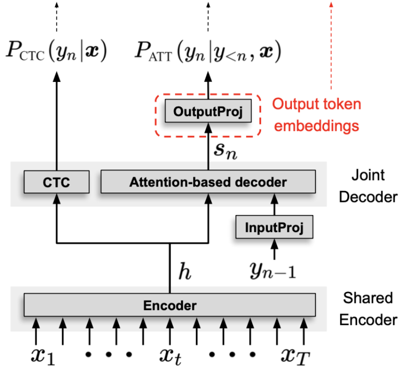


Figure 1: *Hybrid CTC/Attention end-to-end ASR architecture with constrained output token embeddings. The output token embeddings are learned by the parametric matrix of linear output projection layer (OutputProj).*

Methods (1.2)

Jensen-Shannon divergence. First, we assume that learned output token embeddings of monolingual language pair L_1 and L_2 follow a z-dimensional multivariate Gaussian distribution:

$$L_1 \sim Normal(\mu_1, \Sigma_1) \tag{10}$$

$$L_2 \sim Normal(\mu_2, \Sigma_2) \tag{11}$$

The JSD between these distributions is then computed as:

$$\begin{aligned}\mathcal{L}_{\text{JSD}} =& tr(\Sigma_1^{-1}\Sigma_2 + \Sigma_1\Sigma_2^{-1}) \\ &+ (\mu_1 - \mu_2)^T(\Sigma_1^{-1} + \Sigma_2^{-1})(\mu_1 - \mu_2) - 2z\end{aligned} \tag{12}$$

Lastly, we fuse the JSD constraint with the loss function of E2E-CS-ASR using Eq. (9) as follows:

$$\mathcal{L}_{\text{MTL}} = \lambda\mathcal{L}_{\text{CTC}} + (1-\lambda)(\alpha\mathcal{L}_{\text{ATT}} + (1-\alpha)\mathcal{L}_{\text{JSD}}) \tag{13}$$

where $\alpha \in [0, 1]$ controls the importance of the constraint.

Cosine distance. We first compute the centroid vectors C_1 and C_2 obtained by taking the mean of all output token embeddings of monolingual language pair L_1 and L_2, respectively. The cosine distance between two centroids is then computed as follows:

$$\mathcal{L}_{\text{CD}} = 1 - \frac{C_1 \cdot C_2}{\|C_1\| \, \|C_2\|} \tag{14}$$

The CD constraint is integrated into the loss function in a similar way as Eq. (13).

Methods (2.1)

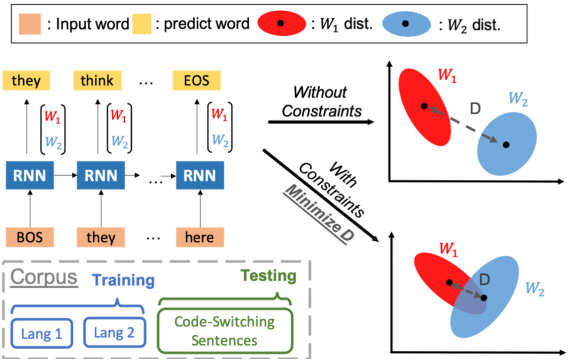


Fig. 1: Overview of proposed approach. Square brackets indicate concatenation operation between W_1 and W_2

Methods (2.2)

2.2.1. *Symmetric Kullback–Leibler Divergence*

Kullback–Leibler Divergence (KLD) is a well-known measurement on computing distance between distributions. With minimizing KLD between language distributions, the embedding space would be semantically overlapped. We assume both W_1 and W_2 follow a z-dimensional multivariate Gaussian distribution, that is,

$$W_1 \sim N(\mu_1, \Sigma_1), \quad W_2 \sim N(\mu_2, \Sigma_2)$$

where $\mu_1, \mu_2 \in \mathbb{R}^z$ and $\Sigma_1, \Sigma_2 \in \mathbb{R}^{z \times z}$ are the mean vector and co-variance matrix for W_1 and W_2 respectively. Based on the assumption of Gaussian distribution, we can easily compute KLD between W_1 and W_2. Due to the asymmetric characteristic of KLD, here we adopt symmetric form of KLD (SKLD), that is, we use the sum of KLD between W_1 and W_2 and the one between W_2 and W_1, yielding

$$\begin{aligned} L_{SKLD} = \frac{1}{2} \big[& tr(\Sigma_1^{-1}\Sigma_2 + \Sigma_2^{-1}\Sigma_1) \\ & + (\mu_1 - \mu_2)^T (\Sigma_1^{-1} + \Sigma_2^{-1})(\mu_1 - \mu_2) - 2z \big] . \end{aligned}$$

2.2.2. *Cosine Distance*

Cosine distance (CD) is a common measurement for semantic evaluation. With minimizing CD, we assume the semantic latent space of languages would be closer. Similar to SKLD, we can compute the mean vector μ_1 and μ_2 of W_1 and W_2 respectively, and CD between two mean vectors can be obtained as follows:

$$L_{CD} = 1 - \frac{\mu_1 \cdot \mu_2}{\|\mu_1\| \|\mu_2\|},$$

where $\|\cdot\|$ denotes the ℓ^2 norm. We hypothesize the latent representation of each word in $Lang1$ and $Lang2$ will distribute in the same semantic space and will overlap by minimizing SKLD or CD.

Results

Table 1: *SEAME dataset statistics after removing the CS utterances from the train set. 'Man' and 'Eng' refer to Mandarin and English languages, respectively*

	train		test_{man}	test_{eng}
	Man	Eng		
# tokens	∼216k	∼109k	∼96k	∼54k
# utterances	21,476	17,925	6,531	5,321
(# CS utterances)	(0)	(0)	(4,418)	(2,652)
Duration	15.8 hr	11.8 hr	7.5 hr	3.9 hr

final loss :

$$\mathcal{L}_{MTL} = \lambda\mathcal{L}_{CTC} + (1-\lambda)(\alpha\mathcal{L}_{ATT} + (1-\alpha)(\beta\mathcal{L}_{JSD} + (1-\beta)\mathcal{L}_{CD}))$$

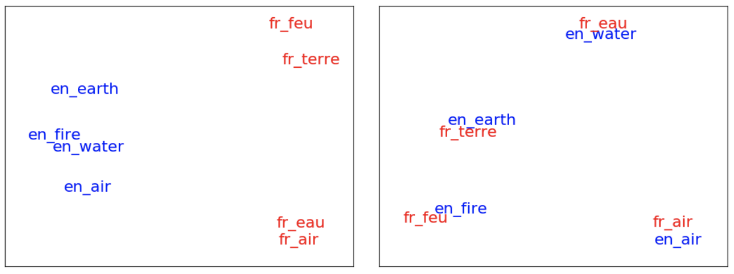
Table 2: *The MER (%) performance of different ASR models built using monolingual data. The test sets are further split into monolingual (mono) and code-switching (CS) utterances*

No.	Model	test_{man}			test_{eng}		
		mono utts.	CS utts.	all	mono utts.	CS utts.	all
1	Kaldi	-	-	39.1	-	-	45.2
2	Baseline	57.7	73.3	70.6	73.7	80.6	78.3
3	+ SP	39.4	56.0	53.2	54.2	65.9	62.2
4	+ BPE	38.1	51.8	49.5	52.9	61.4	58.9
5	+ CD	34.4	49.0	46.3	**47.2**	58.5	55.1
6	+ JSD	34.9	48.8	46.3	47.8	57.6	54.6
7	+ CD	**34.0**	**48.1**	**45.6**	**47.2**	**57.4**	**54.4**

Discussion

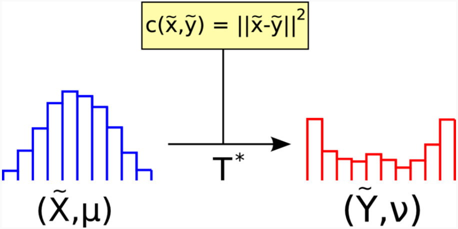
- limitation : score! Why did they use only SEAME ?
- Gaussian distribution (they mention MUSE embeddings)

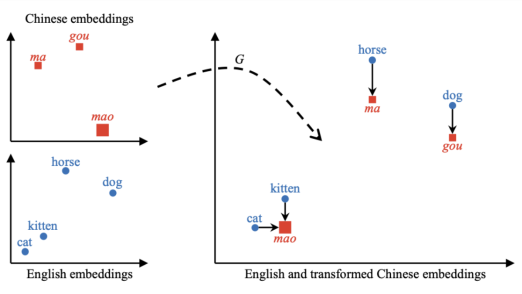
Procrustes' problem



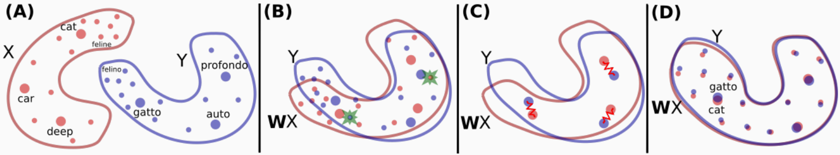
$$\min_{\mathbf{W}\in\mathbb{R}^{d\times d}} \|\mathbf{X}\mathbf{W} - \mathbf{Y}\|_2^2.$$

Optimal Transport - Earth moving distance - Wasserstein





Wasserstein-Procrustes



Further readings and papers from the lab

further readings ...

- methods with pronunciation
- Microsoft new paper ?

Papers on Code-switching from our lab

- Brian
- Sid (around 20% MER)
- Xinjian
- Shinji
- ...

Discussion

Discussion