Vocapia-LIMSI System for 2020 Shared Task on Code-switched Spoken Language Identification

Claude Barras[1], *Viet-Bac Le*[1], *Jean-Luc Gauvain* [1,2]

[1]Vocapia Research, Orsay, France
[2]Université Paris-Saclay, CNRS, LIMSI, Orsay, France

barras@vocapia.com, levb@vocapia.com, gauvain@limsi.fr

Abstract

This paper describes the systems submitted by Vocapia Research and LIMSI for the shared task on Code-switched Spoken Language Identification, organized in the conjunction with the First Workshop on Speech Technologies for Code-switching in Multilingual Communities 2020. Our primary system combines an acoustic approach based on i-vector modeling of audio segments with a phonotactic approach that focuses on sequences of language-independent phone units. Both modeling approaches provided comparable performance, and a gain was obtained by a simple linear combination of their scores, showing their complementarity. One of our submissions obtained first rank for all combinations of tasks and language pairs. For the utterance-level detection task (task A), an F-measure of 76.0% was obtained with our combined system for which the average accuracy on the development set was 83.3%. For the frame-level detection task, the average accuracy was 81.2% on the development set and 78.7% on the evaluation set. However, a detailed analysis reveals a very high rejection of the 200ms code-switched frames, which comprise only 12% of the corpus. This shows that a more precise modeling of code-switched segments is needed for an accurate segmentation.

Index Terms: language identification, code-switching, phonotactic model

1. Introduction

Code-switching is very usual in multilingual communities. In formal situations, a speaker may choose one language according to the situation. For such speech data, automatic language identification can be performed at the speaker turn or document level before further content processing. However, in more spontaneous cases, short code-switched segments may occur in the middle of a sentence. This type of code-switching is much harder to detect and adversely affects the speech transcription, since words in the alternative language will be missing from the vocabulary of the speech-to-text system. Although code-switching has been studied in the linguistic community for many years, it has recently started attracting growing interest in the speech technology domain with the collection of several code-switching corpora for Cantonese-English [1], Mandarin-English [2], Frisian-Dutch [3], Hindi-English and Spanish-English [4], South African languages [5], Egyptian Arabic-English [6] or CanVEC Vietnamese-English [7]. This interested has also resulted in special sessions at Interspeech conferences since 2017, covering various language pairs (e.g., Mandarin-English [8], Hindi-English [9], isiZulu-English [10], English-Spanish [11], French-Algerian Arabic [12] or Frisian-Dutch [13, 14]) and addressing linguistic analysis, speech synthesis [15], code-switching detection [14], language modeling [11, 16] or automatic transcription [17, 18, 19, 20].

In this paper, we describe the systems submitted by Vocapia Research and LIMSI laboratory for the shared task on Code-switched Spoken Language Identification (LID), which was organized in conjunction with the First Workshop on Speech Technologies for Code-switching in Multilingual Communities 2020[1]. Our system combines an acoustic approach based on the i-vector modeling of audio segments and a phonotactic approach focusing on sequences of language-independent phonetic units. The outputs of the two component systems were also submitted individually to obtain contrastive results.

The next section summarizes the two evaluation tasks and experimental conditions. Section 3 describes the component i-vector and phonotactic systems. Their performance on the development and evaluation data is presented in Section 4, before a conclusion.

2. Task description

A short summary of the tasks and evaluation plan is given here along with some characteristics of the corpus. A complete description is available from the link in footnote 1.

2.1. Subtasks

Two subtasks were proposed: (Task A) utterance-level identification of monolingual vs. code-switched utterances; (Task B) frame-level language identification in code-switched utterances, where frames are 200ms contiguous audio segments. For each task, three language pairs were considered with one primary Indian language among Gujarati, Telugu and Tamil and English as a possible code-switching target. The primary language was known a priori.

2.2. Evaluation metric

The primary evaluation metric for the evaluation was the accuracy rate, defined as the ratio of correctly predicted samples over the total number of samples. Task A is a sentence-level binary classification task (code-switched vs. non code-switched utterance), while task B requires a frame-level labeling with 3 classes (silence, primary language or English). There is no specific cost function, so the prior distribution of the classes is the factor governing the relative weight of each error type.

In this paper, we also report three other metrics: recall and precision rates expressed as the number of correctly detected code-switched samples (utterances for task A or frames for task B) over the number of expected or hypothesized samples, respectively; the F-measure is defined as the harmonic mean of

[1] https://www.microsoft.com/en-us/research/event/workshop-on-speech-technologies-for-code-switching-2020/

the recall and precision; and the false positive and false negative rates expressed as the number of false positive (resp. negative) hypothesis over the total number of positive (resp. negative) code-switched samples.

The evaluation plan also proposed an EER metric defined, for task A, as the total number of false rejects and false accepts divided by twice the total number of sentences. Being redundant with the accuracy rate, this metric is not reported in the paper.

2.3. Corpus

The training and evaluation data is composed of sets of sentences with durations ranging from 2 and 20 seconds, and an average duration of 6.8 sec. For task A, about 8000 training sentences and 1000 development sentences without code-switching were provided for each target language (Gujarati, Tamil and Telugu) along with a similar number of code-switched sentences. The detail is provided in Table 1.

Table 1: *Number of sentences without or with code-switching (resp. no CS and CS) in the training and dev. sets for task A.*

	Training		Development	
Language	no CS	CS	no CS	CS
Gujarati	8161	8619	1012	1079
Tamil	8965	8978	1123	1135
Telugu	8766	8225	1088	1047

For task B, only code-switched sentences were provided with the corresponding 200ms frame-level annotation: 8000 sentences for training (about 15 hours) and 1000 sentences for development (about 2 hours) per language. On average, 21% of the frames are labelled as silence, 12% as English and the remaining 67% as either Gujarati, Tamil or Telugu. Table 2 shows the detail for the training and development sets.

Table 2: *Cumulated duration (hh:mm) of primary language, English code-switched and silent frames (resp. P, CS and SIL) in the training and development sets for task B.*

	Training			Development		
Language	P	CS	SIL	P	CS	SIL
Gujarati	10:39	2:08	3:24	1:20	0:16	0:25
Tamil	10:52	1:47	3:32	1:21	0:15	0:26
Telugu	10:53	1:50	3:27	1:21	0:14	0:25

Note that the additional monolingual datasets provided for the three languages were not used to develop the submitted systems described in this paper.

3. Submitted systems

In this section, we describe the two types of systems developed for the detection of code-switched utterances or frames and used for the evaluation: one based on the acoustic (i-vector) approach and other on a phonotactic approach. In addition we also submitted results obtained by combining the outputs of these two.

3.1. i-vector acoustic modeling

The i-vector framework [21] has been successfully applied in Speaker Verification [22, 23] and Language Identification [24] tasks. The i-vector system characterizes the language of an utterance with vectors obtained by projecting the speech data onto a *total variability space*. The approach is generally formulated as follows: $S = m + Tw$ where w is called an *i-vector*, T is a matrix representing the *total variability space*; and m and S represent Gaussian supervectors (GSV) obtained from a language-independent and a language dependent model respectively. The language-independent model is also called *Universal Background Model* (UBM).

In our implementation, the input features of the i-vector language identification system are 40 dimensional phonetic bottleneck features. For each frame, a 32 ms window and a 10 ms offset are used to extract 32-band Mel scale spectrogram concatenated with log-pitch, delta-log-pitch and voicing probability. Then, TRAP-DCT features [25] are estimate on 100 ms windows (11 frames), retaining the first 6 coefficients including the DC component [26]. The resulting TRAP-DCT features with 210 dimensions (35x6) are input to a bottle-neck DNN that has 3 hidden layers with 2000 units and 1 bottle-neck hidden layer with 400 units. Each hidden layer is followed by a non-linearity p-norm unit [27] which reduces the dimension of the layer to 200 and 40, respectively. The phonetic bottle-neck DNN was trained on about 1000 hours of English Broadcast Data. The bottle-neck features are extracted without cepstral mean or variance normalization (CMVN).

The full covariance GMM with 2048 components, the UBM, and an i-vector extractor are estimated using the training data using the Kaldi toolkit [28]. A 600-dimension i-vector is extracted for each training utterance or segment. The i-vector length is normalized to unity [23]. A language-specific i-vector is obtained by averaging the normalized i-vectors for each training utterance [23].

During the test phase, an i-vector is extracted for each test utterance or segment, and is processed to compensate for session variability. Different techniques can be used to compute the test utterance scores. Multi-class logistic regression (MLR) [29] is used in this work. The use of probabilistic linear discriminant analysis (PLDA) [30] such as applied in [22, 23], was explored for the NIST 2015 Language Recognition Evaluation (LRE15) [31] but since the MLR method provided better results on Broadcast data on an internal LID dataset it was adopted here. The MLR model is estimated on all training utterances/segments using an expectation-maximization algorithm.

For Task A, and for each of the 3 Indian languages, an i-vector is extracted for each training utterance (code-switched or non-code-switched) and then a logistic regression (LR) classifier (positive and negative code-switched classes) is estimated on these i-vectors. In the test phase, an i-vector is extracted for each test utterence and scored using the LR model. No voice activity detection (VAD) is performed on the training and test sentences before acoustic features extraction in this task.

For task B, all audio files of the training, developement and evaluation data are analyzed using 600ms-long overlapping windows with a 200ms step (a frame). The label of the segment was associated to the frame at the center of the window, with e.g. the [0-600ms] window corresponds to the frame [200-400ms]. For each of 3 the Indian languages, an MLR classifier is estimated on all training i-vectors (one/frame) for each class (silence, native language and English). In the test phase, an i-vector is extracted for each test frame and scored using the MLR model. For this task, the use of an explicit i-vector class for silence trained on the target dataset significantly improved the silence frame detection performance over using either GMM or DNN pretrained VAD models.

3.2. Phonotactic identification

The phonotactic approach to automatic language identification relies on the idea that the phonetic sequence in an audio sample is characteristic of the language used. It has been shown over time to perform very competitively compared to purely acoustic approaches, from phone-based acoustic decoding [32] to parallel phone recognizers [33] and phone lattices, as presented to LRE15 [31]; RNN-based phonotactic models were also shown as very efficient [34]. Similar to [31], phone decoders using phonetic models from several languages are used to decode the training data and to estimate phone n-gram statistics on the resulting phone lattices for each target class; then, given a new utterance, expectation of its phonetic log-likelihood is computed according to each target models, resulting in a set of posterior scores. The implementation relies on the VoxSigma Software Suite, a commercial product from Vocapia.

For task A and for each target language, a pair of phonotactic models was estimated separately using either the positives code-switched sentences or the negative non-code-switched sentences. The development/evaluation utterances were then scored against the matching language pair. Given the global balance between positive and negative samples in the data set, a 0.5 decision threshold was chosen, i.e. the class with the highest posterior score was selected.

To train models for task B, the frames labeled as silence in the reference annotation were discarded and a pair of models (one for English and one for the primary language) were trained for each target language on the associated training audio segments. For development and evaluation, the frame-level VAD from the i-vector module, as described at the end of previous sub-section, was first applied and frames automatically labelled as silence were discarded prior to further processing. The remaining frames were scored against both the code-switched and non code-switched models. Similar to the i-vector system, each frame was extended to a 600ms audio segment centered around it for training or scoring. As explained in Section 2, the distribution of code-switched and native speech frames is unequal, so a subset of the training data was kept apart and used for to optimize the decision threshold for the code-switched class.

3.3. Systems combination

Since i-vector and phonotactic modeling capture different, and potentially complementary, information, the two models were combined for the primary submission. For both tasks, a linear combination of the posterior scores of the phonotactic and the i-vector models was used, using weights optimized on the development data.

Table 3: *Task A: accuracy (%) by language on the development / on the evaluation data sets for the phonotactic, i-vector and combined systems. Accuracy of the organizer's baseline system is also given on the dev set. Best score on evaluation data in bold.*

Language	baseline	phonotactic	i-vector	combined
Gujarati	76.8	79.1 / **75.4**	84.1 / 57.3	84.3 / 68.8
Tamil	71.2	77.4 / 76.6	79.1 / 76.4	82.2 / **79.8**
Telugu	74.0	79.0 / 77.1	78.8 / 78.3	81.5 / **79.4**
Average	74.0	78.5 / 76.4	80.6 / 70.7	83.3 / 76.0

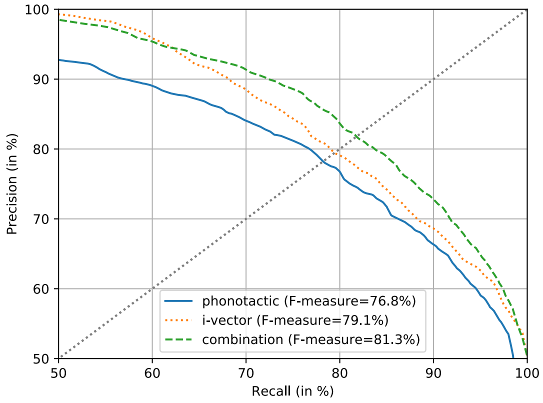

Figure 1: *Task A: recall and precision of monolinugal vs. code-switched utterance detection on the dev set, for the phonotactic, i-vector and combined systems averaged across all languages. F-measure is given for the chosen operating point.*

4. Experiments and results

Figure 1 shows the recall and precision for the code-switched utterance detection (Task A) on the development data set, as computed globally across the three languages for the combined system and the individual phonotactic and i-vector systems. The combination weight was set to 0.4 for the phonotactic scores and 0.6 for the i-vector scores. We can see that the i-vector system generally performs better than the phonotactic system, and that their combination provides a gain for almost all operating points. This is confirmed by looking at the detailed accuracy by language in Table3. On the development data, the gap between the phonotactic and i-vector models is especially high for Gujarati (79.1 vs 84.1%), but even in this case the combination is slightly positive. The performance difference is less for Tamil (and even slightly better for the phonotactic models for Telugu), where their combination provides more than a 2% absolute gain in accuracy. On average for the three languages, the combined system achieves an accuracy of 83.3% on the development set, i.e. an error rate of 16.7%.

On the evaluation dataset, there is a very specific and dramatic degradation of performance for the i-vector system on Gujarati, dropping from 84.1% on the development set to 57.3% on the evaluation set, which carries over to a lesser extent in the combined system. This resulted from a shift of the Gujarati i-vector score distribution on the evaluation set compared to the development set (a 22% relative increase of the average score), which was not observed for the other languages. It is interesting that the phonotactic system proved to be much more robust, with a more limited reduction from 79.1% to 75.4%. For Tamil and Telugu, the performance on the evaluation data was slightly reduced compared with the development data, while keeping a 1-3% absolute gain in accuracy due to system combination. Overall, the combined system has an accuracy of 76.0%, i.e. an error rate of 24%.

For Task B, Figure 2 shows the balance between the false positive and false negative rates of code-switched frame detection as a function of the decision threshold. The axes are scaled by their standard normal deviates, and non-speech frames were excluded for the figure. The phonotactic and i-vector systems show similar behaviors and their combination brings an im-

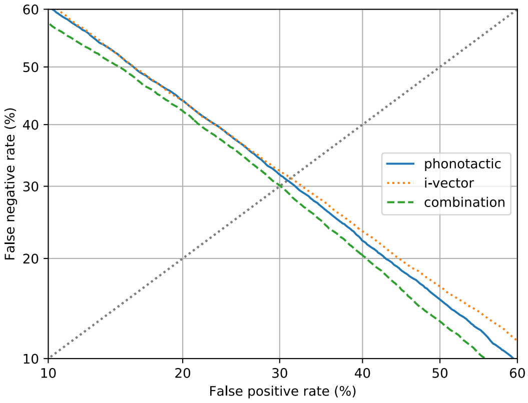

Figure 2: *Task B: false positive and false negative rates of code-switched frame detection computed on the development set and cumulated on all language pairs, for the phonotactic, i-vector and combined systems. Non-speech frames are excluded.*

provement across all operating points. The discrimination between code-switched and non code-switched frames appears to be a difficult task with an equal error rate above 30%. The actual accuracy rates shown in Table 4 confirm this behavior both on the development and evaluation data sets. The combined system accuracy rates averaged across the 3 languages are 81.2% and 78.7%, respectively.

Given the relatively low prior of the code-switched samples at about 12%, the decision thresholds of the systems were optimized according to the evaluation primary metric towards a very low false positive (or false alarm) rate for code-switching and thus a very high negative (or miss) rate. The confusion matrix in Table 5 between the three target classes (silence, code-switched English or primary language) of the combined system, cumulated for all languages on the development set, shows that only 3016 code-switched frames out of 13332, i.e. 22.6%, were correctly labelled, leaving room for improvement. Conversely, identification was correct for 68,089 out of 72,771 speech frames in the primary language, i.e. 93.5%.

One important factor impacting code-switching detection should be the length of the segments; furthermore, code-switched segment shorter than the 600ms analysis window will provide a very sparse information to the phonotactic or i-vector modelling. We show on Figure 3 the histogram of code-switched segments according to their duration: 56% of the segments last only 200 or 400ms. We also show the accuracy of our combined system on the development set cumulated for all languages, in the situation where code-switching would account for half of the spoken content, corresponding to the equal-error-rate configuration. As expected, the accuracy increases with duration, raising from 63% for 200ms segments to 74% for 1.2sec segments.

5. Conclusion

The shared task on Code-switched Spoken Language Identification allowed us to compare different approaches for the utterance-level and frame-level detection of code-switched speech, thanks to the annotated corpora provided in three language pairs. For each combination of task and language pair, one of our systems was ranked first among the submitted systems. In general, both the phonotactic and i-vector acoustic modeling obtained comparable performances, and a simple linear combination brought a further improvement showing their complementarity.

Table 4: *Task B: accuracy (%) by language on the development / evaluation data sets for the phonotactic, i-vector and combined systems. Accuracy of the organizer's baseline system is also given on the dev set. Best score on evaluation in bold.*

Language	baseline	phonotactic	i-vector	combined
Gujarati	76.7	79.9 / 76.9	80.0 / 76.9	80.5 / **77.7**
Tamil	76.5	79.5 / 77.5	80.8 / 78.6	81.2 / **78.8**
Telugu	77.6	80.0 / 78.9	81.4 / 78.9	81.8 / **79.6**
Average	76.9	79.8 / 77.8	80.7 / 78.1	81.2 / 78.7

Table 5: *Task B: frame-level confusion matrix for the combined system on the development set; Columns for reference, rows for hypothesis. SIL stands for non-speech, CS for code-switched English, P for primary language ie. Gujarati, Tamil or Telugu.*

hyp \ ref	SIL	CS	P
SIL	14,097 (69.7%)	248 (1.9%)	1,797 (2.5%)
CS	445 (2.2%)	3,016 (22.6%)	2,885 (4.0%)
P	5,675 (28.1%)	10,068 (75.5%)	68,089 (93.5%)

For the utterance-level detection task, an F-measure of about 80% was obtained. Compared to the provided baseline system with a 74% accuracy average across the three languages, our combined system had an 83.3% accuracy on the development data. Seen conversely, the error rate was reduced from 26% to 16.7%. On the evaluation set, its performance was lower, 76.0%, caused by a distribution shift of the i-vector scores for one of the language pairs. The phonotactic scores were less affected by this distribution mismatch and the phonotactic modeling appears to be more robust than the i-vector approach.

For the frame-level annotation task, the accuracy appears to be of the same order of magnitude, with an average of 81.2% on the development set and 78.7% on the evaluation set. However, a detailed analysis revealed a very high rejection of the code-switched frames, which amount to only 12% of the corpus. This shows that a more precise modeling of the code-switched segments is needed for an accurate segmentation. For this aim, following metrics more specifically fitted to the code-switching detection task as e.g. the ones proposed by Guzmán et al. [35], would certainly be beneficial.

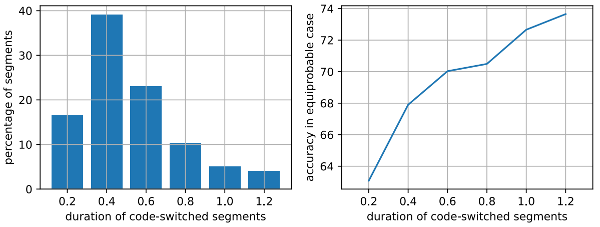

Figure 3: *Task B: histogram of code-switched segments as a function of their duration (left) and accuracy in code-switching frame detection in a equiprobable setting (right), cumulated over all languages for the combined system on the dev set.*

6. References

[1] J. Y. C. Chan, P. C. Ching, and T. Lee, "Development of a Cantonese-English Code-Mixing Speech Corpus," in *Interspeech 2005*, Lisbon, Portugal, Sep. 2005, p. 4.

[2] D.-C. Lyu, T.-P. Tan, E. S. Chng, and H. Li, "SEAME: A Mandarin-English Code-Switching Speech Corpus in South-East Asia," in *Interspeech 2010*, Makuhari, Chiba, Japan, Sep. 2010, p. 4.

[3] E. Yilmaz, M. Andringa, S. Kingma, J. Dijkstra, F. V. der Kuip, H. V. de Velde, F. Kampstra, J. Algra, H. van den Heuvel, and D. 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 2016)*. Paris, France: ELRA, May 2016.

[4] V. Ramanarayanan and D. Suendermann-Oeft, "Jee haan, I'd like both, por favor: Elicitation of a Code-Switched Corpus of Hindi–English and Spanish–English Human–Machine Dialog," in *Interspeech 2017*. ISCA, Aug. 2017, pp. 47–51.

[5] E. V. der westhuizen and T. Niesler, "A First South African Corpus of Multilingual Code-switched Soap Opera Speech," in *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*. Miyazaki, Japan: ELRA, May 2018.

[6] I. Hamed, M. Elmahdy, and S. Abdennadher, "Collection and Analysis of Code-switch Egyptian Arabic-English Speech Corpus," in *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*. Miyazaki, Japan: ELRA, May 2018.

[7] L. Nguyen and C. Bryant, "CanVEC - the Canberra Vietnamese-English Code-switching Natural Speech Corpus," in *Proceedings of The 12th Language Resources and Evaluation Conference*. Marseille, France: ELRA, May 2020, pp. 4121–4129.

[8] P. Guo, H. Xu, L. Xie, and E. S. Chng, "Study of Semi-supervised Approaches to Improving English-Mandarin Code-Switching Speech Recognition," in *Interspeech 2018*. ISCA, Sep. 2018, pp. 1928–1932.

[9] S. Ganji and R. Sinha, "A Novel Approach for Effective Recognition of the Code-Switched Data on Monolingual Language Model," in *Interspeech 2018*. ISCA, Sep. 2018, pp. 1953–1957.

[10] E. van der Westhuizen and T. Niesler, "Synthesising isiZulu-English Code-Switch Bigrams Using Word Embeddings," in *Interspeech 2017*. ISCA, Aug. 2017, pp. 72–76.

[11] V. Soto and J. Hirschberg, "Improving Code-Switched Language Modeling Performance Using Cognate Features," in *Interspeech 2019*. ISCA, Sep. 2019, pp. 3725–3729.

[12] D. Amazouz, M. Adda-Decker, and L. Lamel, "Addressing Code-Switching in French/Algerian Arabic Speech," in *Interspeech 2017*. ISCA, Aug. 2017, pp. 62–66.

[13] E. Yılmaz, H. van den Heuvel, and D. V. Leeuwen, "Exploiting Untranscribed Broadcast Data for Improved Code-Switching Detection," in *Interspeech 2017*. ISCA, Aug. 2017, pp. 42–46.

[14] Q. Wang, E. Yılmaz, A. Derinel, and H. Li, "Code-Switching Detection Using ASR-Generated Language Posteriors," in *Interspeech 2019*. ISCA, Sep. 2019, pp. 3740–3744.

[15] S. Rallabandi and A. W. Black, "Variational Attention Using Articulatory Priors for Generating Code Mixed Speech Using Monolingual Corpora," in *Interspeech 2019*. ISCA, Sep. 2019, pp. 3735–3739.

[16] G. Lee, X. Yue, and H. Li, "Linguistically Motivated Parallel Data Augmentation for Code-Switch Language Modeling," in *Interspeech 2019*. ISCA, Sep. 2019, pp. 3730–3734.

[17] B. M. L. Srivastava and S. Sitaram, "Homophone Identification and Merging for Code-switched Speech Recognition," in *Interspeech 2018*. ISCA, Sep. 2018, pp. 1943–1947.

[18] E. Yılmaz, S. Cohen, X. Yue, D. A. van Leeuwen, and H. Li, "Multi-Graph Decoding for Code-Switching ASR," in *Interspeech 2019*. ISCA, Sep. 2019, pp. 3750–3754.

[19] A. Biswas, E. Yılmaz, F. de Wet, E. van der Westhuizen, and T. Niesler, "Semi-Supervised Acoustic Model Training for Five-Lingual Code-Switched ASR," in *Interspeech 2019*. ISCA, Sep. 2019, pp. 3745–3749.

[20] H. Seki, T. Hori, S. Watanabe, J. L. Roux, and J. R. Hershey, "End-to-End Multilingual Multi-Speaker Speech Recognition," in *Interspeech 2019*. ISCA, Sep. 2019, pp. 3755–3759.

[21] D. Najim, K. Patrick, D. Reda, D. Pierre, and O. Pierre, "Front-End Factor Analysis for Speaker Verification," *IEEE Trans. on Acoustics, Speech and Signal Process.*, vol. 19, no. 4, pp. 788–798, 2011.

[22] P. Kenny, "Bayesian speaker verification with heavy-tailed priors." in *Proc. Odyssey*, 2010, p. 14.

[23] D. Garcia-Romero and C. Y. Espy-Wilson, "Analysis of i-vector length normalization in speaker recognition systems." in *Proc. Interspeech*, 2011, pp. 249–252.

[24] D. Martınez, O. Plchot, L. Burget, O. Glembek, and P. Matejka, "Language recognition in ivectors space," *Proc. Interspeech*, pp. 861–864, 2011.

[25] H. Hermansky and S. Sharma, "Traps – classifiers of temporal patterns." in *Proc. ICSLP*, 1998.

[26] P. Schwarz, P. Matejka, and J. Cernocky, "Towards lower error rates in phoneme recognition." in *TSD*, Sep 2004, pp. 456–472.

[27] D. P. X. Zhang, J. Trmal and S. Khudanpur, "Improving deep neural network acoustic models using generalized maxout networks." in *Proc. IEEE ICASSP*, Adelaide, May 2014.

[28] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, J. Silovsky, G. Stemmer, and K. Vesely, "The Kaldi Speech Recognition Toolkit," in *IEEE Workshop on Automatic Speech Recognition and Understanding*, Dec. 2011.

[29] D. van Leeuwen and N. Brummer, "Channel-dependent gmm and multi-class logistic regression models for language recognition," in *Proc. Odyssey*, 2006.

[30] S. J. Prince and J. H. Elder, "Probabilistic linear discriminant analysis for inferences about identity," in *IEEE 11th Int. Conf. on Computer Vision. ICCV 2007.*, 2007, pp. 1–8.

[31] G. Gelly, J.-L. Gauvain, L. Lamel, A. Laurent, V. B. Le, and A. Messaoudi, "Language Recognition for Dialects and Closely Related Languages," in *Proc. Odyssey*, Jun. 2016, pp. 124–131.

[32] L. F. Lamel and J.-L. Gauvain, "Language identification using phone-based acoustic likelihoods," in *Proc. IEEE ICASSP*, Adelaide, Apr. 1994.

[33] M. A. Zissman, "Comparison of four approaches to automatic language identification of telephone speech," *IEEE Trans. on Speech and Audio Process.*, vol. 4, no. 1, pp. 31–44, 1996.

[34] B. M. L. Srivastava, H. Vydana, A. K. Vuppala, and M. Shrivastava, "Significance of neural phonotactic models for large-scale spoken language identification," in *2017 International Joint Conference on Neural Networks (IJCNN)*. Anchorage, AK, USA: IEEE, May 2017, pp. 2144–2151.

[35] G. Guzmán, J. Ricard, J. Serigos, B. E. Bullock, and A. J. Toribio, "Metrics for Modeling Code-Switching Across Corpora," in *Interspeech 2017*. ISCA, Aug. 2017, pp. 67–71.