Sinhala and Tamil Speech Intent Identification From English Phoneme Based ASR

Yohan Karunanayake, Uthayasanker Thayasivam, Surangika Ranathunga
Department of Computer Science and Engineering
University of Moratuwa
Katubedda 10400, Sri Lnanka
{*yohan.13, rtuthaya, surangika*}*@cse.mrt.ac.lk*

***Abstract*—Today we can find many use cases for content-based speech classification. These include speech topic identification and spoken command recognition. Automatic Speech Recognition (ASR) sits underneath all of these applications to convert speech into textual format. However, creating an ASR system for a language is a resource-consuming task. Even though there are more than 6000 languages, all of these speech-related applications are limited to the most well-known languages such as English, because of the availability of data. There is some past research that looked into classifying speech while addressing the data scarcity. However, all of these methods have their own limitations. In this paper, we present an English language phoneme based speech intent classification methodology for Sinhala and Tamil languages. We use a pre-trained English ASR model to generate phoneme probability features and use them to identify intents of utterances expressed in Sinhala and Tamil, for which a rather small speech dataset is available . The experiment results show that the proposed method can have more than 80% accuracy for a 0.5-hour limited speech dataset in both languages.**

***Keywords*-Low-Resource; Speech Intent Classification; Sinhala; Tamil;**

I. Introduction

Automatic Speech Recognition (ASR) has evolved widely, and recent research shows human-level performance in some tasks [1]. Popular application areas of ASR include intent identification of spoken commands, topic identification of speech, and call center call classification [2]–[4]. For example, in a smart speaker use case, when a user issues a command such as "Play some music", it is capable of identifying the intent of the given free-form command as a request to turn on the music. Here, the requirement is to identify speech, based on the spoken content. In the currently available topic or intent identification systems, this is enabled by cascading an ASR system and a Natural Language Understanding (NLU) system or a text classification model [2], [3].

The task of the ASR system is to convert a sequence of acoustic features into the most likely sequence of words. Normally Mel-frequency Cepstral Coefficients (MFCC) of the speech signals are used as input features. Earlier ASR models incorporated an acoustic model, a pronunciation lexicon that maps phones into words, and a language model to rank the likelihood of words [5]. Today we can see the use of deep neural network-based end-to-end ASR models [1], [6]. Advantage of these models is that they are capable of modeling all the acoustic, pronunciation lexicon and language into a single model.

The output of the ASR module is the input for the NLU model. Then, the NLU model outputs semantic labels for a given text sequence, and it is trained with labeled data using supervised learning. Further, there is research that uses either an n-best list of the ASR output or the intermediate features of the ASR. This is to eliminate the errors made by using the single best output of the ASR [4], [7], [8].

Still, the ASR based applications are only available for most widely used languages, but not for low-resource languages (LRLs). Here, languages that lack electronic resources for speech and/or language processing are referred to as LRLs [9]. Because of the data scarcity, it is difficult to create robust ASR systems for LRLs. Normally more than 1000 hours of transcribed speech data is required to train an accurate ASR model [9]. Besides, the accuracy of the ASR model is a very important factor for the above described cascaded system for speech intent identification. Errors made by the ASR component can propagate into the NLU system and can result in false outputs [7]. Hence, this limits the development of speech intent identification systems for LRLs.

Recent research has presented some end-to-end speech intent identification models for languages such as English by utilizing transfer learning [4], [10]. Our previous work [11] presented the successful use of this transfer learning technique for LRLs using character probability values and shows good results. Further, there have been approaches to predict the intent using different features [2], [12].

In the past, researchers have suggested methods to create ASR for LRLs using phoneme annotated speech datasets [13]. Phonemes represent the perceptually distinct units of sound in a language, and it is closer to the sound representation compared to characters. Inspired by this and our previous work [11], we are presenting a phoneme-based domain-specific speech intent identification methodology in this paper. Experimental results show that phoneme based features perform better compared to the previous work [11]. Using Sinhala and Tamil speech data, we were able to reach an overall classification accuracy of 80% using not more than 500 speech samples.

978-1-7281-5014-7/19/$31.00

II. Related Work

The major issue in low-resource speech intent identification is the limited amount of annotated speech data. This restricts the development of robust ASR systems, and without ASR we cannot have speech intent identification. One obvious way to solve this is by compiling a sufficiently large speech corpus in the targeted language. However, this is a time and resource-consuming task [9]. Currently, we can observe few approaches that try to address this. One method is focused on improving the low-resource ASR, while another focuses on speech intent identification using different features other than ASR text output. This includes intermediate features of the ASR models.

In this first method, researchers have focused on developing ASR systems optimized for languages that have smaller speech corpora. One successful approach for this is adapting or retraining of an ASR system trained on a high resource language [9]. Another way is to use multiple smaller speech datasets and training ASR models with multitask-learning [9].

Some research focuses on topic identification of the speech. Here, the topics are similar to intents, but they vary in a broad range and tries to represent the whole subject presented in the spoken content. Work of Wiesner et al. [13] presents such topic classification methodology. They use a low-resource ASR development method and a multilingual speech corpus with universal phones annotations [14]. It has provided promising results when there is very limited training data. The text output of this ASR is used on a classifier model that can identify the corresponding topic. There are 11 different topics and some of them are "Evacuation", "Food Supply", "Urgent Rescue", "Medical Assistance" and, "Shelter". In this method, we need to have a phone-annotated multilingual speech corpus to train the ASR. Further, having a good understanding of the targeted LRL phonology is mandatory.

An ASR is trained to output the most probable word sequence for a given acoustic feature sequence [4]. Hence, when we use ASR generated text as input for others, the 1-best output of the ASR becomes input for the rest of the components. Here, it is difficult to ensure that the best output of the ASR is always a correct one. He et al. [8] and Yaman et al. [7] explored this in their works and proposed some approaches to overcome the issues by using the n-best list of the ASR output and joint optimization techniques. This still has the overhead of ASR development for the targetted language.

As mentioned above, in the second approach, features generated from the respective speech queries are directly used for intent/topic classification instead the final text output of the ASR. We can identify few prominent techniques in past literature. Liu et al. [2] and Wiesner et al. [13], used features such as phone-like units discovered via acoustic unit discovery (AUD) [15], [16], or word-like units discovered via unsupervised term discovery (UTD) [17]. These are unsupervised feature extraction methods and do not require speech data with transcripts or annotations. However, these unsupervised methods require more data to identify better feature representations and more computational power to process data. In contrast to this, Buddhika et al. [12] presented a low-resource speech intent classifier that uses MFCC features directly. They use classifier models such as Support Vector Machines (SVM), and Convolution Neural Networks (CNN) to identify intents from MFCC features. This approach achieved a 74% classification accuracy for a 10 hour domain specific Sinhala dataset.

Chen et al. [4] presented an intent identification method for the English language queries using intermediate features of a pre-trained English ASR model. Here, they used character probability values generated by the ASR as features for a CNN based intent classification model, and obtained good results for call center call classification. Lugosch et al. [10] presented another such similar work and showed good results using a 14.7-hour dataset while utilizing the pre-training strategy. In their work, they identify not only the intent, but also the slot values such as action, object, and location mentioned in the speech query. In this way, we do not need to worry about 1-best output of the ASR, and can optimize jointly. However, in both of these works, an ASR trained on a large English corpus is used to identify intent on the same language.

Utilizing this pre-training strategy, our previous work [11] demonstrated a successful method for speech intent identification for LRLs. In this work, we used a pre-trained English model to identify the intent of low-resource Tamil and Sinhala speech commands. This method could reach to an overall accuracy of 80% using 1 hour of speech data containing 1000 samples.

In summery, previous research has tried to tackle low-resource speech intent identification via either developing low-resource ASR or using different features generated from the speech query. The latter approach can eliminate the issue of having an accurate ASR, since they rely on different input features other than the 1-best output of an ASR. Because of this reason, this method is much more suitable for low-resource scenarios.

III. Methodology

In our previous work, we used ASR generated character probabilities as features to identify intent. Characters are more language-specific and try to follow the syntactic and semantic rules of a particular language. Compared to characters, phonemes try to represent perceptually distinct units of sound in a specified language. Hence phonemes have more ability to represent sounds than the characters. In this work, we try to exploit this.

In Section II, we highlighted the benefit of the pre-training strategy. In this work, we use a pre-trained ASR model of a source language to generate phoneme based features for another language. Then we can use these features to identify the intent in the LRL. In machine learning paradigm, this is known as transfer learning, where we try to reuse a model trained on one task in another similar or related task [18]. Here, we use an ASR

model trained to convert high resource speech into text, in low resource speech intent identification. For a better understanding of the methodology, first, we introduce the character related features and then phoneme features.

A. Character Probabilities

In Section I, we mentioned that there are well-performing all neural end-to-end ASR models [1], [6]. Such an ASR model is trained to predict the most probable word sequence for a given sequence of acoustic features. To enable predicting words, most of the end-to-end ASR models produce character probability values in a given time step. Hence, when we input a sequence of acoustic features $x^{(i)}$, the model converts this into a sequence of character probabilities $y^{(i)}$, with $\hat{y}_t = \mathbb{P}(c_t|x)$ where c_t represents the possible character set including other special characters-(to represent spaces, silents) in the training language.

B. Phoneme Probabilities

In this work, our focus is to use phoneme probability values rather than character probabilities and examine the effectiveness. ASR systems that output a phoneme sequence for a given audio sequence are not so common. Most of the ASR systems are trained on characters, which makes it easy to predict probable word sequence. Lugosch et al. [10], presented an ASR system that uses phonemes as intermediate targets. This model outputs a sequence of phoneme probabilities $p^{(i)}$, $\hat{p}_t = \mathbb{P}(p_t|x)$ when given a sequence of acoustic features $x^{(i)}$ where p_t represents the possible phoneme set in the training language. Hence we use this model to generate phoneme based features.

C. Feature Classification Model

To identify a fixed set of intents, previous research has used a classifier model on features generated from audio recordings as described in Section II. They have experimented with models such as Support Vector Machines (SVM), Feed-forward Neural Networks (FFN), and Convolutional Neural Networks (CNN). Our previous work showed that SVM and CNN based classifiers work better with character probability features and highlighted the superior performance of CNN [11]. Further, there were two different CNN types: 1D CNNs and 2D CNNs which was experimented. In 1D type, the convolution and pooling operations are done along only on one dimension, while in 2D this happens along on two directions.

The above-described character/phoneme probability features are two dimensional and one dimension is used to represent time steps while other dimension represents different characters or phonemes. Visualization of these features is presented in Figure 2. For SVM models, these features need to be converted into a series format to provide as inputs. Further, when we use 1D CNN models, convolution and pooling operation are done along the time axis. For 2D CNN, those operations are done along both axes.

In summary, our speech intent identification methodology is as follows. First, we train a phoneme based ASR using high resource language data. Then this ASR is used to generate features. Using these features we train different classifier models to identify the intent of the speech. Figure 1 shows the complete arrangement of the system with the pre-trained ASR and the CNN classifier.

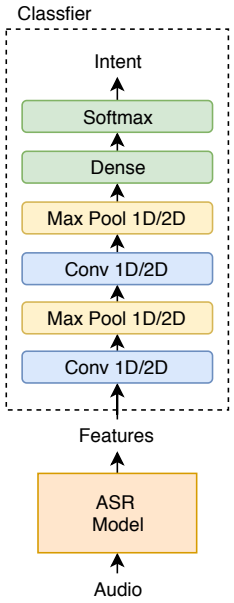


Figure 1. Arrangement of the final model

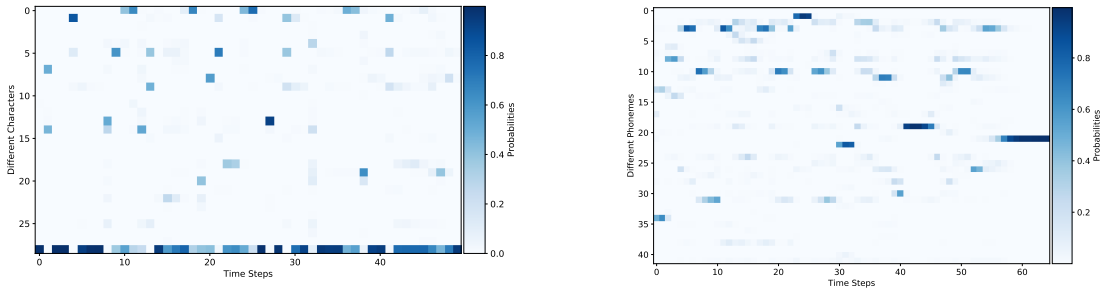
IV. Dataset

To evaluate the performance of the proposed approach, we use the same dataset used in our previous work [11]. It includes Sinhala and Tamil speech data related to the banking domain with intent labels. This Sinhala and Tamil data has been collected respectively from 215 and 40 students. Datasets contain both male and female voices. All the contributors were in the age between 20 to 25 years. Further all these audio clips have been collected through mobile phones via crowd sourcing. Because of that we can expect domestic noises in the audio. In the Sinhala dataset, all the queries are expressed in Sinhala, However, Tamil queries contain some code-mixed speech with English terms. Table I shows the statistics of the dataset. In the table, 'I' represents the number of infections, i.e. different ways of expressing intents. 'S' represents the number of samples.

V. Experiment

Training an ASR model on a large dataset requires high computational power. Hence we used already trained openly available ASR models for the experiments. To get the character probability features, we used the DeepSpeech (DS)[1] model [1]. This DS model has been trained on the Common Voice American English corpus, and reports a 11% word error rate on the LibriSpeech clean test corpus. To get phoneme based probability values, we adapted

[1]DeepSpeech Model
https://github.com/mozilla/DeepSpeech



(a) Character Feature

(b) Phoneme Feature

Figure 2. Visualization of features

Table I
DETAILS OF THE DATASETS (I-INFLECTIONS, S-NUMBER OF SAMPLES)

Intent	Sinhala		Tamil	
	I	S	I	S
1. Request Acc. balance	8	1712	7	101
2. Money deposit	7	1306	7	75
3. Money withdraw	8	1548	5	62
4. Bill payments	5	1004	4	46
5. Money transfer	7	1271	4	49
6. Credit card payments	4	795	4	67
Total	39	7624	31	400
Unique words	32		46	
Size in hours	7.5		0.5	

the pre-trained ASR model of lugosch et at. [10][2]. This model was trained on the LibriSpeech 690 hour English corpus [10].

Then we extracted the probability features using the ASR models and fed them into the classifier models for training. For character probability features, character set of $\{a, b, c, .., z, space, apostrophe, blank\}$ is used since it is trained on English speech. The phoneme set had a set of 42 symbols that includes the ARPAbet English phoneme set (39 phonemes), and 3 non-speech annotations [10].

Since there is a limited amount of data, we employed 5-fold cross-validation to measure the overall classification accuracy. Models are trained up to the maximum accuracy without getting over-fitted into the training data set. Additionally, we used the Bayesian optimization-based algorithm for hyper-parameter tuning [19]. The optimization algorithm is employed with 500 iterations to select the suitable hyper-parameters, which improves the overall accuracy. This was very significant for the CNN model parameters such as the number of filters and kernel sizes. For the SVM models, a linear kernel is used after experimenting with several different kernels types (Polynomial, Radial Basis Function (RBF)). Table II shows the final overall classification accuracy of different classifier models and a comparison between previous work.

Further, we evaluated the overall accuracy change with respect to the number of available training samples. To do that we drew a random data sample with a particular size, and performed 5-fold cross-validation. We did this for 20 times to get the average accuracy for a given sample size. This task was performed on the Sinhala dataset since it contained more than 5000 samples. Figure 3 summarizes these results. In Figure 3, connected lines represent the overall accuracy change in the Sinhala data. Points on the vertical line represent the Tamil dataset accuracy values with 400 data samples.

Table III presents the most probable character/phoneme sequences for the two selected sentences in Sinhala and Tamil languages. For better understanding, we present 39 phonemes with their IPA (International Phonetic Alphabet) notation.

VI. DISCUSSION

Overall results presented in Table II, and the graph in Figure 3 emphasize that phoneme probability features are more effective for speech intent identification compared to character features. For Sinhala and Tamil datasets, the proposed method achieves an overall accuracy of 97.38% and 81.70%, respectively. Further, these values indicate the usefulness of phoneme probability features despite the targeted LRL. According to the Figure 3, having 500 is enough to reach up to more than 80% accuracy. It needs more than 1000 data samples to achieve similar results using character probability features [11].

For comparison purposes, we marked the Tamil dataset results in the graph presented in Figure 3. Disconnected dots on the dashed-line represent the Tamil dataset experiment results for a 400 data sample. There we can see that the accuracy for the Tamil dataset also lies close to the Sinhala dataset trend line. However, when we examined closely, we can find one exception - 2D CNN shows a similar accuracy for both character and phoneme features in Tamil.

In the Tamil dataset, 61% of the sentences contain code-mixed speech queries with at least 1 English word. This can be a reason for such a higher result. In contrast to this, we cannot identify such anomaly results with phoneme features or 1D CNN character feature results. Further, this can happen because of having proper hyper-parameters for the 2D CNN character model. Because of the limited

[2]Phoneme Based Model
https://github.com/lorenlugosch/pretrain_speech_model

Table II
SUMMARY OF RESULTS

Features	DS Character Prob [11]			Phoneme Prob		
Classifier	SVM	1D CNN	2D CNN	SVM	1D CNN	2D CNN
Accuracy Sinhala	70.04%	93.16%	92.09%	78.21	**97.31%**	94.16%
Accuracy Tamil	23.77%	37.57%	76.30%	49.83	**81.70%**	76.28%

Table III
MOST PROBABLE CHARACTERS AND PHONEMES FOR A GIVEN UTTERANCE

Language	Spoken Utterance	Character Sequence	Phoneme Sequence
Sinhala	මෙය කියද (ēēṣaya kīyada)	shhe as shhakin ane she a shakkkingg in shheeshhaki the shhe heki shhe shhaky the	ʃʃʃʃʃieɪɛɪɛɪʃʃʃiiɪkkkiiiiiʌnðʌʌllllɹ ʃʃtʃʃieɪɛɪɛɪʃʃʃɪɪɪkkkiiiiiʌnnnn kaɪ ð kkk ʃʃʃʃʃiiiɪʃʃʃiiikkkiiiiiððʌʌ ssssʃieɪɛɪɛɪtʃtʃtʃðnʌʌkkkjɪjjjuuuunnn ʃʃʃʃɛɪɛɪʃʃʃɛɪɛɪɛɪkkiiiɪnnʌʌtt
Tamil	காசு இன்னொரு account கு மாத்த வேணும் (Kācu iṉṉoru account ku mātta vēṇum)	causi nodid gon o coman theworom care nnowi con ocomat_thean casly an moda cound the mot thhe ra cary nnodicont ccommatevernnam casi nnoo goncamatowerrn	kkɑɑzssiiŋnnoʊttɪɪddgɔɔandʌkkʌwwʌʌttðʌʌwɜ-ɜ-nʌʌm kkkkɑɑɑɑθiiidnnoʊoiʌtkɔɔndʌkʌmɑɑtbʌlʌlɪɪnʌʌ lll kkkæɑsleɪɪɪnnɹɜ-ɜ-ðʌʌvfɑɹgk ðɪnnmɑɑatkkððʌlwɔɔɔɹnni kkkɑɑssiidnnoʊtkkɑɑnæʌkʌʌmmɑɑttɪɪlɜ-ɜ-ɜ-ggʌʌmm kkɑɑɑɑsssiinmwaɾaɾlaɾʌttkkwʌnnnn kʌʌmɑɑtðʌlɜ-ɜ-nn

data, it is difficult to identify the effect of code mixing. However, in general phoneme probability features give better results compared to character probability features regardless of the language.

We can identify some common patterns when we closely examine the most probable character or phoneme sequences presented in the Table III. Most of the time these patterns do not occur sequentially, there are some other symbols in between them. The intent identification model trained on probability features tries to identify those hidden patterns. These sequential patterns can be affected by the language model of the high resource training language. Here, the ASR models used to generate features are based on the Recurrent Neural Networks (RNNs). and they are capable of language modeling [20], [21]. Therefore the ASR models try to predict character or phoneme sequences as observed in the training language (English in this case).

This effect is more visible in character probability outputs in the Table III. Here, first few characters with corresponding sounds have been detected. Sometimes it has predicted the English words with a similar sound. However, when it comes to the middle and end of a sequence, it is difficult to find any patterns and all look random. This is quite different for the phoneme sequence. If we inspect the generated symbol sequence for the Tamil sentence, even in the middle we can find some patterns. Hence, phoneme-based features gives a better representation compared to character-based features for a given audio. Consequently, we can observe higher results.

The next visible effect on results is the performance difference of the 1D and 2D CNN models. When there is limited data, 2D CNN outperforms 1D CNN in character probability features. This is changed when training data size increases. With phoneme features, 1D CNN always outperforms 2D CNN. However, when we examine the feature visualization for intensity points (most probable symbol), these values change quite rapidly in the character feature map compared to phonemes. Additionally, we were able to identify visible patterns inside the generated phoneme sequences. Hence 1D CNN can perform better

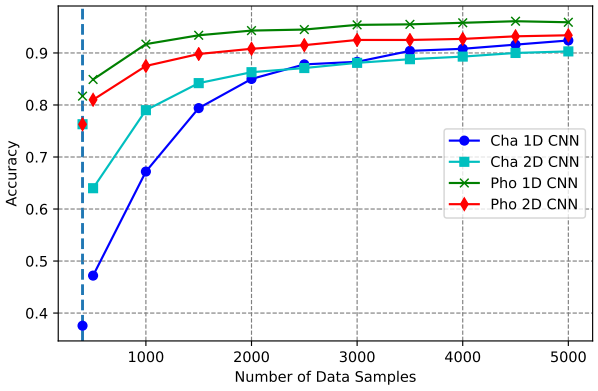


Figure 3. Overall accuracy change with the samples size (Connected dots - Sinhala, Disconnected dots - Tamil

with phoneme features. There are more rapid distortions in character features. Hence, the 2D CNN may be more useful for character probability features.

VII. CONCLUSION AND FUTURE WORK

In this paper, we presented an English phoneme based feature generation and intent identification method for Sinhala and Tamil speech data using a pre-trained English ASR model. We can identify the expressing intent of speech queries more accurately using these features. To evaluate the effectiveness of the proposed method, we used two datasets from the two different languages in the banking domain, which have a limited number of data samples. Experiment results show that phoneme base probability features are more effective compared to character features in low resource scenarios. Additionally, we observed that 1D CNN models perform better compared to 2D CNNs in classifying phoneme based features. The proposed method can reach to an 80% accuracy even with a dataset that has 0.5 hours of speech data.

In the future, we hope to extend this work using different language datasets to examine the effectiveness and the generalizability across languages, and the effect of having code-mixed speech.

ACKNOWLEDGMENT

This research was funded by a Senate Research Committee (SRC) Grant of University of Moratuwa.

REFERENCES

[1] D. Amodei, S. Ananthanarayanan, R. Anubhai, J. Bai, E. Battenberg, C. Case, J. Casper, B. Catanzaro, Q. Cheng, G. 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.

[2] C. Liu, J. Trmal, M. Wiesner, C. Harman, and S. Khudanpur, "Topic identification for speech without asr," in *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, vol. 2017-August, 2017, pp. 2501–2505.

[3] A. Ram, R. Prasad, C. Khatri, A. Venkatesh, R. Gabriel, Q. Liu, J. Nunn, B. Hedayatnia, M. Cheng, A. Nagar *et al.*, "Conversational ai: The science behind the alexa prize," *arXiv preprint arXiv:1801.03604*, 2018.

[4] Y.-P. Chen, R. Price, and S. Bangalore, "Spoken language understanding without speech recognition," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 6189–6193.

[5] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz *et al.*, "The kaldi speech recognition toolkit," IEEE Signal Processing Society, Tech. Rep., 2011.

[6] A. Hannun, C. Case, J. Casper, B. Catanzaro, G. Diamos, E. Elsen, R. Prenger, S. Satheesh, S. Sengupta, A. Coates *et al.*, "Deep speech: Scaling up end-to-end speech recognition," *arXiv preprint arXiv:1412.5567*, 2014.

[7] S. Yaman, L. Deng, D. Yu, Y.-Y. Wang, and A. Acero, "An integrative and discriminative technique for spoken utterance classification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 16, no. 6, pp. 1207–1214, 2008.

[8] X. He and L. Deng, "Speech-centric information processing: An optimization-oriented approach," *Proceedings of the IEEE*, vol. 101, no. 5, pp. 1116–1135, 2013.

[9] L. Besacier, E. Barnard, A. Karpov, and T. Schultz, "Automatic speech recognition for under-resourced languages: A survey," *Speech Communication*, vol. 56, pp. 85–100, 2014.

[10] L. Lugosch, M. Ravanelli, P. Ignoto, V. S. Tomar, and Y. Bengio, "Speech model pre-training for end-to-end spoken language understanding," *arXiv preprint arXiv:1904.03670*, 2019.

[11] Y. Karunanayake, U. Thayasivam, and S. Ranathunga, "Transfer learning based free-form speech command classification for low-resource languages," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, 2019, pp. 288–294.

[12] D. Buddhika, R. Liyadipita, S. Nadeeshan, H. Witharana, S. Javasena, and U. Thayasivam, "Domain specific intent classification of sinhala speech data," in *2018 International Conference on Asian Language Processing (IALP)*. IEEE, 2018, pp. 197–202.

[13] M. Wiesner, C. Liu, L. Ondel, C. Harman, V. Manohar, J. Trmal, Z. Huang, N. Dehak, and S. Khudanpur, "Automatic speech recognition and topic identification for almost-zero-resource languages," in *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, vol. 2018-September, 2018, pp. 2052–2056.

[14] K. M. Knill, M. J. Gales, A. Ragni, and S. P. Rath, "Language independent and unsupervised acoustic models for speech recognition and keyword spotting," in *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, 2014, pp. 16–20.

[15] L. Ondel, L. Burget, and J. Černocký, "Variational inference for acoustic unit discovery," *Procedia Computer Science*, vol. 81, pp. 80–86, 2016.

[16] C. Liu, J. Yang, M. Sun, S. Kesiraju, A. Rott, L. Ondel, P. Ghahremani, N. Dehak, L. Burget, and S. Khudanpur, "An empirical evaluation of zero resource acoustic unit discovery," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 5305–5309.

[17] A. Jansen and B. Van Durme, "Efficient spoken term discovery using randomized algorithms," in *2011 IEEE Workshop on Automatic Speech Recognition & Understanding*. IEEE, 2011, pp. 401–406.

[18] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on knowledge and data engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.

[19] J. Bergstra, B. Komer, C. Eliasmith, D. Yamins, and D. D. Cox, "Hyperopt: a python library for model selection and hyperparameter optimization," *Computational Science & Discovery*, vol. 8, no. 1, p. 014008, 2015.

[20] T. Mikolov, M. Karafiát, L. Burget, J. Černockỳ, and S. Khudanpur, "Recurrent neural network based language model," in *Eleventh annual conference of the international speech communication association*, 2010.

[21] A. Graves and N. Jaitly, "Towards end-to-end speech recognition with recurrent neural networks," in *International conference on machine learning*, 2014, pp. 1764–1772.