IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received February 27, 2022, accepted March 17, 2022, date of publication March 28, 2022, date of current version April 5, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3162854

Resources and Benchmarks for Keyword Search in Spoken Audio From Low-Resource Indian Languages

VIJAYA LAKSHMI V. NADIMPALLI[1], SANTOSH KESIRAJU[2], ROHITH BANKA[1], RASHMI KETHIREDDY[1], AND SURYAKANTH V. GANGASHETTY[3], (Member, IEEE)

[1]Speech Processing Laboratory, KCIS, International Institute of Information Technology, Hyderabad 500032, India
[2]Speech@FIT, Brno University of Technology, 612 00 Brno, Czechia
[3]Koneru Lakshamaiah Education Foundation, Vaddeswaram 522502, India

Corresponding authors: Vijaya Lakshmi V. Nadimpalli (nvv.lakshmi@research.iiit.ac.in) and Santosh Kesiraju (kesiraju@fit.vut.cz)

The work of Vijaya Lakshmi V. Nadimpalli was supported by the Department of Science and Technology, Government of India, under Women Scientist Scheme (DST WOS-A) under Grant SR/WOS-A/PM-95/2017. The work of Santosh Kesiraju was supported in part by the Czech National Science Foundation (GACR) Project NEUREM3 under Grant 19-26934X and in part by the Czech Ministry of Education, Youth and Sports for Computing on IT4I Supercomputer from the Large Infrastructures for Research, Experimental Development and Innovations Project e-Infrastructure under Grant CZLM2018140.

ABSTRACT This paper presents the resources and benchmarks developed for keyword search (KWS) in spoken audio from six low-resource Indian languages (from two families), namely Gujarati, Hindi, Marathi, Odia, Tamil, and Telugu. The current work on constructing keywords and building benchmark KWS systems is inspired by the popular IARPA Babel program and the subsequent works on low-resource KWS. The keywords are constructed by taking into account their properties i.e., occurrence, length, and average confusability; and their effects on the evaluation metric - the term-weighted value (TWV). We make use of freely available speech datasets, and reprocess them to create resources for KWS, thereby adding value to the existing speech resources. Four ASR-based KWS systems are built, and their performance is analyzed across the three keyword properties on all the six languages. The prepared keywords and other related resources to replicate our experiments are made available for the public. We believe that the analysis and guidelines provided in this paper will not only help the research community, but also practitioners and engineers to easily create KWS resources for newer languages, datasets, and scenarios.

INDEX TERMS Keyword search, low-resource languages, term-weighted value (TWV).

I. INTRODUCTION

Large amounts of publicly available datasets and resources (labels, annotations, open source software) have played a major role in advancing speech and language technologies. However, of more than 6500 spoken languages in the world, only a small number of languages fall under the high-resource category, while many come under the category of low/under resourced languages [1]. Different tasks under speech technologies require different kind of resources or manual annotations. However, certain tasks are relatively close to each other, and given the annotations for one (primary) task, semi-automatic methods can be employed to quickly build the required annotations and resources for the other related (secondary) task. For example, one requires speech recordings with corresponding textual transcriptions (optionally a lexicon) to build an automatic speech recognition system (ASR). By using semi-automatic methods, these resources can be extended for keyword search (KWS), a secondary but related task, and the required additional resources are a set of keywords (having specific properties).

Keyword search (KWS) or spoken term detection (STD) is the task of automatically searching, detecting, and retrieving a set of user-defined keywords (usually in text form) from a spoken audio corpus. The technologies developed for KWS have various applications including, but not limited to indexing, searching in multimedia archives [2], video lectures [3], or voice based human-computer interfaces [4].

The associate editor coordinating the review of this manuscript and approving it for publication was Chao Tong.

Most of the approaches for KWS rely on automatic speech recognition (ASR) systems. The ASR system is used to decode the speech utterances into lattices, and a KWS module performs the search within these lattices. Hence, the performance of KWS systems directly correspond to the word-error-rate (WER) of the underlying ASR system. It becomes especially challenging in low-resource scenarios (languages), where the amount of training data is relatively low and the WERs of ASR systems are relatively high [5]. In WER computation, every word is treated equally, whereas in the case of KWS, detecting rare words might be of higher interest than detecting frequent ones. Hence for evaluating a KWS, the most often used metric is term-weighted value (TWV), which is dependent on the occurrence of the keyword in a test set.

In KWS, the keywords are in text format, whereas in query-by-example spoken term detection (QbE-STD), the keywords or queries are in spoken form. The most widely used methods rely on template matching of speech features (eg: Gaussian or phone posteriorgrams) using dynamic time warping (DTW) algorithms [6]–[8], and similarity search based on end-to-end neural networks [9]. There exist hybrid approaches that use discriminatively trained models to extract speech features (eg: multilingual, articulatory bottleneck features) [10]–[13], which are then used in a template matching framework.

While there have been attempts to train ASR-free KWS systems [14], they do not yet perform as good as the traditional ASR-based KWS systems [15]. Moreover, any improvements and technological advances in ASR [16] could directly result in the improvement of KWS systems [17]. Research on low-resource KWS / STD gained importance during the past 15 years. The National Institute of Standards and Technology (NIST) initiated the spoken term detection (STD) evaluation (2006), and OpenKWS evaluations [18], [19] to facilitate research and development of speech technologies for retrieving information (keywords) from spoken data archives. The IARPA Babel program [20] (2012-2016) has played a major role in the progress of the low-resource ASR and KWS research. A majority of the techniques were released as recipes in the open source Kaldi toolkit [21]–[23], aiming to help a wider community of researchers and engineers.

On the other hand, MediaEval's spoken web search [24], [25], query-by-example spoken term detection [26]–[29], organized evaluations and provided spoken data in many Indo-European and African languages. The provided speech data does not contain transcriptions for speech or language labels, and is mainly focused on QbE-STD to help facilitate the research in unsupervised methods in low-resource scenarios.

There was a considerable amount of study and analysis of the KWS systems in low-resource languages from the IARPA Babel program [5], [23], [30], [31].

The work presented in this paper is inspired by [30], where the authors have analyzed the properties of keywords, such as occurrences, length (in terms of number of phonemes) and average confusability, with respect to the term-weighted value. The TWV is the most widely used metric for evaluating KWS systems [18], [19], [24], [32], [33], which is designed to be dependent on the occurrence of the keyword in the test set. More details on TWV and the properties of keywords are explained in Section III.

Even though several efforts have been made for KWS in low-resource languages, there is a lack of a carefully crafted set of keywords and benchmarks for datasets that are freely available to the public. Moreover there exists no recipe or guidelines on how to create a suitable set of candidate keywords from the existing speech resources. The aim of the current work is to create resources (keywords, training recipes) using existing speech datasets, and building benchmark KWS systems that would act as baseline systems for future research. For this, we used freely available datasets that were primarily proposed for research in ASR for low-resource languages. These datasets are reprocessed to make them suitable for training and testing KWS systems. The main contributions of the paper are summarized below:

1) Two existing speech datasets comprising six languages are analyzed, and reprocessed to make them suitable for KWS.
2) Using a semi-automatic procedure, keywords are prepared for all the six languages, while taking into account the three keyword properties and its effect on the term-weighted value, the KWS evaluation metric. Guidelines and analyses are provided that can help in rapidly creating keywords for newer languages, datasets and scenarios.
3) Four ASR-based KWS systems are built to benchmark the KWS results on six languages. These can act as baseline systems for future research works.
4) The keywords and other necessary resources (including Kaldi recipes) for replicating our experiments are made public.[1]

The rest of the paper is organised as follows: Section II gives an overview of existing datasets available for KWS, which serves as motivation for preparing keywords for low-resource languages. Section III explains the TWV evaluation metric for KWS and the properties of keywords that affect TWV. Section IV describes the data processing details and Section V presents four ASR-based KWS systems built using the open source Kaldi toolkit, followed by the keyword preparation methodology in Section VI. Experimental results and discussions are presented in Section VII, and conclusions are given in Section VIII.

II. AN OVERVIEW OF EXISTING DATASETS FOR KWS

This section presents an overview of existing datasets for KWS. The most popular one is from the IARPA Babel program which contains 25 languages, namely Amharic, Assamese, Bengali, Cantonese, Cebuano, Dholuo, Guarani, Haitian, Igbo, Javanese, Kazakh, Kurdish, Lao, Lithuanian,

[1] `https://github.com/skesiraju/indic-kws`

Mongolian, Pashto, Swahili, Tamil, Tagalog, Telugu, Tok Pisin, Turkish, Vietnamese, and Levantine Arabic. Each language pack contains about $40-80$ hours of training data (speech recordings with transcriptions) and approximately 10 hours of test data. There is also a 10 hour training set as part of a low-resource language pack. The number of keywords are in the range $300-3000$, depending on the language pack. At the time of writing this paper, the Babel language packs are not freely available, but one can obtain them from the Linguistic Data Consortium (LDC)[2] for a nominal price of $25 per pack.

There was little work done on KWS for Indian languages and there exists no free datasets or benchmarks that are comparable to that of IARPA Babel datasets. However, more recently the authors from [34] presented a work on 10 low-resource Indian languages, namely, Assamese, Bengali, Gujarati, Kannada, Malayalam, Marathi, Odia, Bodo, Manipuri, and Rajasthani. They used existing text-to-speech datasets[3] which contain about $4-51$ hours of audio data, depending on the language, from which 1000 utterances were selected for the test set. There are about 160 keywords per language; 100 keywords of length $1-10$ characters, 50 keywords of length $10-15$ characters, and 10 keywords of length $15-20$ characters. However, there is no further analysis on the occurrences of these keywords. Note that the TWV is directly dependent on the keyword occurrence.

The MediaEval's spoken web search (SWS) provided data in several low-resource languages: Hindi, Telugu, Gujarati, (Indian) English for SWS 2011, African languages (isiNdebele, Siswati, Tshivenda, and Xitsonga) from LWAZI [35] for SWS 2012. Albanian, Basque, Czech, non-native English, Isixhosa, Isizulu, Romanian, Sepedi and Setswana for SWS 2013. Slovak (apart from SWS 2013 languages) was added to MediaEval's Query-by-example search in speech (QUESST 2014). While these datasets cover a diverse set of low-resource languages, they do not have speech transcriptions or language labels and cannot be used directly for KWS.

III. EVALUATION METRIC FOR KWS

This section describes the term-weighted value (TWV), which is the most widely used metric for assessing the performance of KWS and STD systems. It was proposed by NIST in 2006 for spoken term detection evaluation [32]. It was later used in the IARPA Babel program, NIST OpenKWS evaluations [18], [19] and MediaEval SWS [24], [33], as the primary evaluation metric. The TWV is well studied [36] in the literature and there were attempts to train KWS systems that optimize directly for the TWV metric [37], [38].

The following notation is used to formally present the TWV metric. Let $\mathcal{T}$ be the set of terms (keywords), with $t \in \mathcal{T}$ be a given term (keyword). Let θ be the detection threshold. In order to compute TWV, a KWS system requires to provide a detection score, start and end time stamps for every hypothesized occurrence, also known as a detection. Every system detection is matched with the ground truth reference using a function that accounts for both the temporal overlap and the detection score. *Hit* corresponds to an instance where a correct detection is made, and *Miss* corresponds to an instance where the system failed to detect an actual occurrence of a keyword. *False alarm* (FA) corresponds to an instance where the model falsely hypothesized a keyword.

With the above definitions, the probabilities of miss (P_{Miss}) and FA (P_{FA}) at detection threshold θ for a single term t are calculated as

$$P_{\text{Miss}}(t,\theta) = 1 - \frac{N_{\text{correct}}(t,\theta)}{N_{\text{true}}(t)}, \tag{1}$$

$$P_{\text{FA}}(t,\theta) = \frac{N_{\text{FA}}(t,\theta)}{T_{\text{speech}} - N_{\text{true}}(t)}, \tag{2}$$

where $N_{\text{correct}}(t,\theta)$ corresponds to the number of correct detections of t recovered by the system with a detection score $\geq \theta$. $N_{\text{true}}(t)$ is the true occurrences of term t in the given test corpus, and $N_{\text{FA}}(t,\theta)$ represents the number of false alarms (incorrect detections) with a detection score $\geq \theta$. T_{speech} is the total amount of speech in test data in seconds,[4] which corresponds to the number of trials. This was rather an arbitrary choice, as acknowledged by NIST [18], since it is not possible to count the number of discrete number of trials from a continuous speech. Moreover, it is not necessary or required to provide a detection score for every keyword-utterance pair.

From (1) and (2), it can be seen that the cost of a miss depends on the number of true occurrences of a keyword ($N_{\text{true}}(t)$), whereas the cost of a FA is effectively a constant across all keywords (since $T_{\text{speech}} \gg N_{\text{true}}(t), \forall t \in \mathcal{T}$). A perfect hit adds value to the system, whereas a miss or a false alarm reduces the value of a system. Hence, TWV is one minus the average value lost by the system per term; which is a weighted linear combination of number of misses and false alarms.

$$\text{TWV}(\theta) = 1 - \frac{1}{|\mathcal{T}|}\sum_{t\in\mathcal{T}}\Big(P_{\text{Miss}}(t,\theta) + \beta\, P_{\text{FA}}(t,\theta)\Big) \tag{3}$$

$$\text{where } \beta = \frac{C}{V}\left(\frac{1}{p_t} - 1\right) \tag{4}$$

Here $p_t = 0.0001$ denotes the prior probability of a keyword, $\frac{C}{V} = 0.1$ represents the cost to value ratio. This results in the value of β to be 999.9. Again, these constants were defined by NIST for STD [32] and used in subsequent OpenKWS evaluations [18], [19]. Some of these choices seem arbitrary, but for the sake of simple interpretation, one can ignore the presence of p_t, C, V, and just view β as a scaling factor for P_{FA}. Alternative interpretation in terms of *effective prior* exists that combines all p_t, C, V into a single one [26]. Now, the range of TWV values of a system depends only on β,

[2] https://www.ldc.upenn.edu/

[3] https://www.iitm.ac.in/donlab/tts/voices.php

[4] This is calculated by force aligning the speech signal to the corresponding text transcriptions and considering only the duration of speech regions.

TABLE 1. Statistics of the datasets used for KWS experiments. The left half of the table presents the duration of audio in hours. The right half of the table gives the size of vocabulary and keyword sets. $\mathcal{V}$ represents the total vocabulary, with $\mathcal{V}_{\text{dev}}$ and $\mathcal{V}_{\text{test}}$ representing the vocabulary of words in the development and test sets respectively. $\mathcal{T}$ represents the set of keywords that occur in the test set, $\mathcal{T}_{\text{Train}} \subset \mathcal{T}$, and $\mathcal{T}_{\text{Dev}} \subset \mathcal{T}$ are set of keywords that appear in the training and development sets respectively. % Test exclusive refers to the percentage of keywords that appear only in the test set.

		Duration (in hours)			Vocabulary count			Keyword count			
Language	(ISO code)	Training	Dev	Test	$\lvert\mathcal{V}\rvert$	$\lvert\mathcal{V}_{\text{dev}}\rvert$	$\lvert\mathcal{V}_{\text{test}}\rvert$	$\lvert\mathcal{T}_{\text{train}}\rvert$	$\lvert\mathcal{T}_{\text{dev}}\rvert$	$\lvert\mathcal{T}\rvert$	(% Test exclusive)
Gujarati	(guj)	22.16	9.31	13.40	41178	15891	20174	544	428	711	(16.7)
Tamil	(tam)	23.90	10.38	10.73	53259	20428	19811	671	565	831	(14.7)
Telugu	(tel)	21.33	9.19	14.48	46331	16791	22683	649	536	811	(15.9)
Hindi	(hin)	70.81	2.25	2.05	6542	1633	1795	217	128	339	(29.8)
Marathi	(mar)	71.82	1.52	1.48	3395	732	872	89	34	181	(47.5)
Odia	(ori)	71.78	0.86	0.85	1644	507	513	57	121	190	(33.2)

which may not be suitable for every scenario. Moreover the systems are compared against a hard decision, rather than on the detection scores. In order to overcome these, the systems can be compared with the help of a detection error trade-off (DET) curve. The trade-off between P_{Miss} and P_{FA} (without the scaling factor β) for all possible values of θ can be seen on the DET curve. The actual TWV (ATWV) corresponds to the point on DET curve which is the TWV using the actual decisions made by a given system. The maximum TWV (MTWV) corresponds to the point on the DET curve where a value of θ yields the maximum TWV. The difference between ATWV and MTWV can show the score calibration issues or the loss in choosing a sub-optimal operating point. The KWS systems are often evaluated with the help of NIST F4DE (framework for detection evaluation) toolkit [39], which computes the TWV scores.

An alternative evaluation metric called the normalized cross-entropy cost (C_{nxe}) was proposed for MediaEval SWS 2013 [26], and was later used for MediaEval QUESST 2014 [28]. It relies on the likelihood ratio scores of *null* (i.e., the segment contains the query term t) and *alternative* (i.e., the segment does not contain query term t) hypotheses. This allows us to compare the systems based on their scores rather than on hard decisions. There are two main differences between TWV and C_{nxe}: (i) Unlike TWV, the C_{nxe} treats every detection equally regardless of the frequency of occurrence of the keyword, i.e., detecting rare and frequent words are weighted equally, (ii) two systems can be comparable only when they use the same segmentation and the same set of trials. In practice, a system provides score only for hypothesized occurrences, and not necessarily to every keyword-utterance pair. To address this issue, the organizers of MediaEval SWS, proposed to use the minimum score as the score for censored trials (the missing trials in a system's submission). This scheme might be suitable for evaluations where the onus lies on the organizers, but makes it impractical in our scenario. Hence, we use only TWV as our KWS evaluation metric throughout the paper.

While every keyword is treated equally for TWV, every detection is not, since P_{Miss} is dependent on the occurrence of keyword ($N_{\text{true}}(t)$) in the test set. This suggests that a good candidate set of keywords for evaluating any KWS system should contain many rare words. Rare words are likely to be more informative than frequent words. Apart from the keyword occurrence that explicitly influences TWV, there are two other keyword properties that have an influence on the TWV metric. These are keyword length and average confusability distance, as observed for various datasets (languages) from IARPA Babel program [30].

These three keyword properties are briefly explained below:

1) Keyword occurrence is simply the number of times a keyword has occurred in the test set. Detection of a rare keyword from the test set will enhance the ATWV score as compared to the detection of a common keyword.
2) Keyword length refers to the number of phonemes in the word as extracted from the lexicon. In general, a KWS system yields better detection performance for longer keywords than shorter ones. This is due to the availability of more acoustic information for longer keywords. Moreover, longer words are more likely to be rare.
3) The keyword confusability distance d_t for a keyword t is defined [30] as the average minimum Levenshtein distance for the keyword in every utterance $n = 1 \ldots N$,

$$d_t = \text{round}\left(\frac{1}{N}\sum_{n=1}^{N} \min\left(\underset{\forall i, t \neq w_{ni}}{\text{dist}}(t, w_{ni})\right)\right) \tag{5}$$

where round indicates rounding to nearest integer, N is the total number of utterances, w_{ni} is word i in utterance n, t is the keyword, and dist represents the Levenshtein distance (using phonemic transcriptions from lexicon) that is computed for all the words w_{ni} in a given utterance except when $t = w_{ni}$. The ATWV values will be low for more confusable keywords as compared to less confusable ones. Moreover, shorter words are more likely to be confusable than longer ones.

Based on the above properties, a good candidate set of keywords should ideally have the following trends with respect to the TWV metric: ATWV vs keyword occurrence shall be a decreasing curve, whereas ATWV vs keyword length and average confusability distance shall be increasing curves. The same was observed for several IARPA Babel languages [30]. The aim of this work is to create a set of keywords that follow the above described trends. More details on keyword preparation are described in Section VI.

IV. DATASET PREPARATION FOR KWS

This section describes the dataset preparation, which includes the strategy for splitting the data in training, development (dev), and test sets that are suitable for KWS. The data is prepared for six Indian languages namely, Gujarati (guj), Tamil (tam), Telugu (tel), Hindi (hin), Marathi (mar), and Odia (ori). The data for the first three languages (guj, tam, tel) are derived from the Interspeech 2018 special session for low resource Indian language ASR (IS 2018) [40], whereas the data for the latter three languages (hin, mar, ori) are derived from multilingual and code-switching ASR challenges for low resource Indian languages (MUCS 2021) [41], a special session from Interspeech 2021.

A. PROCESSING IS 2018 DATASET

The original data splits had 40 hours in the training set and approximately 5 hours each in the development (dev) and blind test sets respectively. The transcriptions of the blind test set were not released. The sampling frequency of the data is 16 kHz. As 5 hours is too short (for example, IARPA Babel languages had 10 hours of test data) and it would not be appropriate as test data for KWS, we made our own splits of the data. Moreover, with the original data splits, the best ASR systems achieved about 14% WER [42], which would not be an optimal choice for preparing keywords for a KWS system. Hence, we combined the original training and dev data (45 hours in total) and divided them into 3 splits in approximately 5 : 2 : 3 ratios for training, development, and test respectively, making sure that there are recordings of unseen speakers in the test data.[5] The details (duration in hours) of these news splits are given in Table 1, which are further used in all our experiments. The vocabulary sizes of the three languages are given in the upper half of Table 1.

The common phone set and lexicon that came with the data were used. Sequitur G2P [43] was trained to obtain phoneme sequences for a few missing Tamil words, which were then added to the lexicon.

B. PROCESSING MUCS 2021 DATASET

The data for Hindi, Marathi and Odia are taken from MUCS 2021 (sub-task1). The sampling frequency of the data is 8 kHz. The total duration of audio for each language is about 95 hours; however, the number of unique utterances is much lower. Several utterances are indeed repeated by multiple speakers. On average, each utterance is repeated 21 times in Hindi, 27 times in Marathi, and 70 times in Odia datasets.

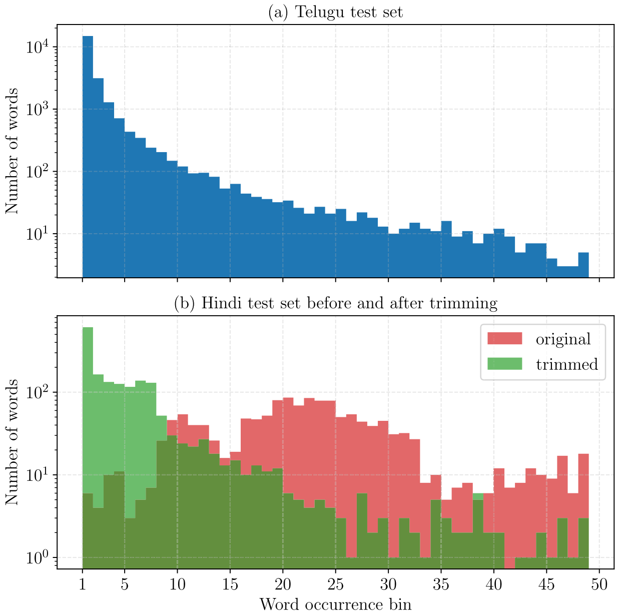


FIGURE 1. (a) Word occurrence histogram of the Telugu test set. (b) Word occurrence histogram of the Hindi test set before and after trimming.

This also resulted in a much smaller vocabulary, which posed a challenge in preparing keywords.

For each language, the data is pooled and then divided into 14 : 3 : 3 ratio for training, development (dev) and test, while spreading the lexical variability. This resulted in about 70 hr for training, 15 hr each for dev and test. The dev and test sets are further trimmed (i.e., several duplicate utterances are removed) in order to have more unique utterances. This is especially important, as it is desirable to have many rare words in the test split. Fig. 1 (a) shows histograms of word occurrences of the Telugu test set and Fig. 1 (b) presents histograms of word occurrences of the Hindi test set, before and after trimming. The duration (in hours) of audio data after trimming, for individual sets is presented in the bottom half of Table 1. The Table also presents vocabulary statistics. Note that Hindi, Marathi, and Odia datasets have much smaller vocabulary sizes as compared to Gujarati, Tamil, and Telugu datasets.

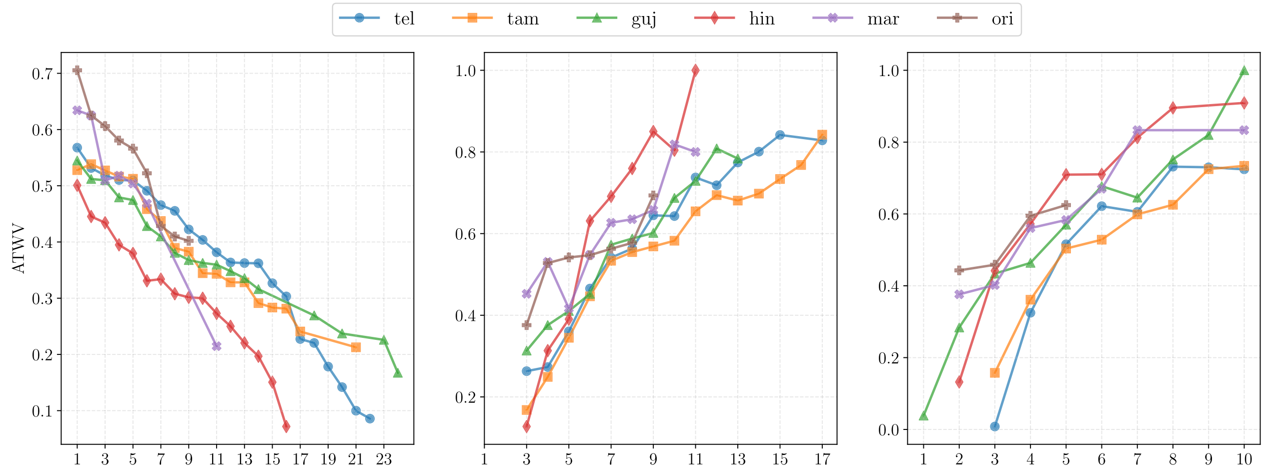
V. DESCRIPTION OF ASR BASED KWS SYSTEMS

Our KWS systems are based on hybrid ASR systems, trained using the Kaldi toolkit [21]. Generally, a Kaldi KWS system contains two parts: i) a large vocabulary continuous speech recognition (LVCSR) module that decodes the search collection and generates the corresponding lattices, and ii) a KWS module that builds an index for the lattices [44] and searches the keywords from the generated index [22]. A lattice is a representation of the alternative word-sequences that are sufficiently likely for a particular utterance.

Four systems are built relying on the standard recipes from the Kaldi toolkit.[6] Using a sliding window on the input

[5] Utterance ID to speaker ID mapping is obtained by listening to several recordings and identifying a pattern in the file name that matches with the speaker identity.

[6] `https://github.com/kaldi-asr/kaldi/tree/master/egs/babel/s5b`



(a) Keyword occurrence (b) Keyword length (c) Keyword confusability distance

FIGURE 2. The effect of three keyword properties on ATWV with DNN-sMBR KWS system, across all the six languages.

audio, 39 dimensional MFCC feature vectors ($\Delta + \Delta\Delta$) are extracted, which is followed by cepstral mean and variance normalization. The initial mono-phone system is trained on 1k short utterances to obtain accurate alignments. This is followed by an incremental GMM-based tri-phone system training with 1k, 5k, and 10k short utterances respectively. Without this incremental training, the WER of the systems are about 10% worse than those reported in Table 3. This is followed by a full tri-phone system training, and subsequently by LDA + MLLT[7] tri-phone system and speaker adaptive training of a GMM-HMM system. The resulting system is referred as **GMM** system in this paper. The alignments from this GMM system are used for the subsequent neural network based acoustic models.

The second system is based on a feed-forward (deep) neural network (DNN), that is trained to minimize the frame-level cross-entropy between predicted posterior probabilities and the true senones. At the input layer, the neural network takes 4 frames to the left and right as the context. The network has 5 layers with 300 hidden units each, and tanh non-linear activation functions. The network is trained for a maximum of 30 epochs with an initial and final learning rates 0.005 and 0.0005 respectively. This system is referred as **DNN**.

The third system uses the state-level minimum Bayes risk training criterion on top of the DNN acoustic model [46]. The model is trained to minimize the expected error of the state-labels corresponding to a given word sequence. The model is trained for 4 epochs. This system is denoted as **DNN-sMBR** in the paper. The DNN-sMBR system is used to create a candidate list of keywords, which will be explained in more detail in Section VI.

The fourth system is based on a time-delay neural network (TDNN) acoustic model [47], [48]. The initial layers operate on a narrower context, and the context widens as it goes deeper. This is enabled with the help of sub-sampling at each layer, which additionally reduces the number of computations.

TABLE 2. Contextual information at each layer of TDNN.

Layer	Input context with sub-sampling	Left and right context
1	[-2,2]	2, 2
2	{-1,0,1}	3, 3
3	{-1,0,1}	4, 4
4	{-3,0,3}	7, 7
5	{-6,0}	13, 7

Different configurations for the TDNN are investigated by varying temporal contexts, and hidden layers. The configuration that gave the best WER on the dev set is taken forward for KWS experiments. The corresponding TDNN has 5 hidden layers with 650 hidden units, and ReLU activation functions. The contextual information (including sub-sampling) at each layer is given in Table 2.

The input to the network is MFCC features concatenated with a 100 dimensional i-vector that is extracted per utterance [49]. The network is trained for 3 epochs with initial and final learning rates of 0.0015 and 0.00015 respectively. This system is referred as **TDNN**.

A 3-gram language model (LM) based on Kneser-Ney discounting is used for all the languages. The LM is trained only on transcriptions from the training set, and the hyper-parameters are tuned on the dev set to obtain lower perplexity. For Hindi, Marathi, and Odia, the LM is also trained on additional data from Wikipedia. There are no out-of-vocabulary words (OOVs) as the provided lexicon contained all the words.

VI. PREPARATION OF KEYWORDS

A semi-automatic paradigm is used to prepare the keywords that follow the three properties described in Section III.

[7] LDA: Linear discriminant analysis. MLLT: Maximum-likelihood linear transformations [45].

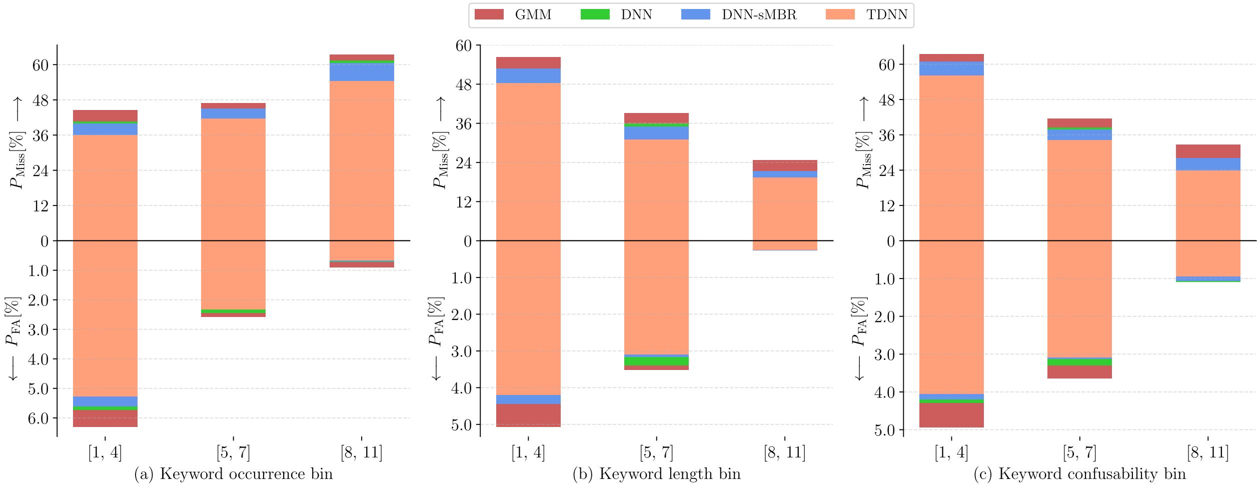


FIGURE 3. The P_{Miss} and P_{FA} for four different KWS systems on the Telugu (tel) test set with respect to the three keyword properties.

The DNN acoustic model and the ground truth test transcriptions are used to obtain accurate alignments between the test audio and the corresponding transcriptions. These act as ground truth alignments for evaluating KWS systems. For each language, T_{speech} from (2) is calculated from the speech regions after the force alignment.

For every unique word in the test set, the three mentioned properties can be inferred using the ground truth test transcriptions and the lexicon. However, to determine the difficulty of detecting the words, we need a reasonably well designed KWS system. For this, we rely on the DNN-sMBR based KWS. This KWS is used on all the unique words from the test set, which results in several hypothesized detections. Using the ground truth alignments, for every hypothesized detection one of the following decisions is made: hit, miss, false alarm, true negative. For the sake of simplicity, the detection threshold (θ) is fixed to 0.5. Combining these decisions with every word occurrence bin ranging from 1 to 24, keywords are randomly picked such that ATWV is a decreasing curve. For this to happen, more words (eg: >100 for guj, tam and tel) are picked in low word-occurrence bins and less words (eg: <30) in high occurrence bins. This results in around 700 - 800 keywords (eg: guj, tam, tel). It is important to note that rare words are actually rich in information, hence to evaluate a KWS system, it is preferable to have many rare words.

Once the keywords are selected based on occurrence, they are categorized (binned) based on word length i.e., number of phonemes (according to the lexicon). For each bin, the ATWV is computed. For most of the cases (bins), the ATWV increases with the keyword length. However, the increasing curve might break at few points (bins). For these bins, few words are either added or removed so that an overall increasing trend is preserved.

In a similar manner, the third property, i.e., keyword confusability distance follows an increasing curve with respect to ATWV, provided the above two properties are satisfied. At those few points where it breaks, few words are added or removed to maintain an overall increasing trend.

TABLE 3. Word error rates of the ASR systems on the new data splits. +wlm indicates the language model (LM) is trained on additional data from Wikipedia. The values in bold and underline indicate the first and second best systems respectively, for a given language.

	GMM		DNN		DNN-sMBR		TDNN	
Language	dev	test	dev	test	dev	test	dev	test
Gujarati	26.23	25.74	21.57	21.31	19.49	**19.45**	19.67	19.54
Tamil	28.28	34.12	24.24	29.28	22.83	27.38	21.28	**25.58**
Telugu	32.43	33.95	27.95	29.36	26.10	27.36	24.75	**26.00**
Hindi	38.49	38.96	29.07	29.81	27.23	26.94	21.02	**21.61**
+WLM	31.00	33.03	23.56	24.56	21.14	22.19	17.59	**18.46**
Marathi	57.26	62.68	54.06	**61.15**	60.39	65.46	56.59	63.59
+WLM	14.29	14.53	10.36	10.14	12.06	11.16	8.32	**7.77**
Odia	81.39	82.48	79.82	79.47	77.31	**76.40**	79.98	79.93
+WLM	31.10	34.29	27.19	31.23	26.83	27.86	23.64	**26.16**

Note that the curves need not to be strictly monotonous, but an overall increasing or decreasing trend is desired, depending on the respective keyword property. Although the keywords are selected based on the DNN-sMBR system, these trends can be seen for any KWS system. We also ensure that at least 15% of the selected keywords are exclusive to the test set. Table 1 presents the number of keywords obtained for each language. It also shows the percentage of keywords that appear only in the test set. Since we use the same constants in TWV metric as in IARPA Babel program, and NIST OpenKWS, we also aimed at selecting keywords that would result in ATWV scores around 0.5. This is mainly to be consistent with prior works [30], [31].

The Fig. 2 (a) illustrates that ATWV decreases with keyword occurrence while Fig. 2 (b) and Fig. 2 (c) depict that ATWV increases with keyword length and confusability distance across the six languages.

TABLE 4. ATWV (denoted by A) and MTWV (denoted by M) scores of the KWS systems with the respective sets of keywords prepared on the new data splits. +wlm indicates that language model (LM) is trained on additional data from Wikipedia. In every row (language), the values in bold and underline indicate the first and second best ATWV values on the test sets.

Language	GMM			DNN			DNN-sMBR			TDNN		
	dev (M)	test (A)	test (M)	dev (M)	test (A)	test (M)	dev (M)	test (A)	test (M)	dev (M)	test (A)	test (M)
Gujarati	0.516	0.441	0.447	0.572	0.498	0.498	0.578	0.489	0.492	0.594	**0.521**	0.524
Tamil	0.548	0.426	0.428	0.599	0.477	0.480	0.618	0.500	0.503	0.636	**0.531**	0.534
Telugu	0.456	0.447	0.449	0.522	0.489	0.490	0.533	0.495	0.503	0.557	**0.540**	0.547
Hindi	0.216	0.286	0.322	0.401	0.380	0.407	0.435	0.405	0.417	0.551	**0.485**	0.495
+WLM	0.375	0.398	0.427	0.474	0.455	0.474	0.507	0.505	0.543	0.606	**0.583**	0.583
Marathi	-	-	-	-	-	-	-	-	-	-	-	-
+WLM	0.677	0.503	0.507	0.719	0.571	0.579	0.644	0.553	0.571	0.792	**0.653**	0.671
Odia	-	-	-	-	-	-	-	-	-	-	-	-
+WLM	0.507	0.497	0.510	0.616	0.539	0.557	0.612	0.559	0.569	0.695	**0.621**	0.655

It reflects that longer keywords are more easily detected when compared to shorter keywords. More confusable words will have lower values of confusability distance and hence their corresponding ATWV values will be smaller. A higher value of confusability distance indicates that the word is less confusable and easily detectable resulting in higher value of ATWV, which is reflected in Fig. 2 (c). It can be observed that for languages Marathi, and Odia, the range of frequency class and word length are much shorter as they have a smaller vocabulary (see Table 1). This also made it challenging to prepare a good candidate set of keywords for them.

Note that for all the above analysis on ATWV, β is set to the default value of 999.9. Since β can influence ATWV, we illustrate in Fig. 3 the three keyword properties across all the four systems using only P_{Miss} and P_{FA} (without the scaling factor β).

It can be seen from Fig. 3 (a) that miss probability increases with keyword occurrence. On the other hand, Fig. 3 (b) and Fig. 3 (c) present the decreasing trend for miss and FA probabilities with respect to both keyword length and confusability distance across all the four KWS systems. Similar trends are observed for the remaining languages.

Although the keywords are selected based on the DNN-sMBR based KWS system, the desired trends are seen across all the KWS systems with regard to the three keyword properties.

VII. RESULTS AND DISCUSSION

This Section presents the ASR and KWS results on the re-processed datasets, using the systems described in Section V. We also empirically examine the relationship between WER and MTWV.

The word-error-rates (WERs) of the underlying ASR systems on the dev and test sets for all the systems are given in Table 3. The dev set is used to select an optimal language model weight (relative to the acoustic model weight) during decoding. In all the cases, DNN based systems have lower WER as compared to GMM based systems, and in most cases, both DNN-sMBR and TDNN systems have slightly lower WER than DNN systems. For languages Hindi, Marathi, and Odia, we also experimented with a Kneser-Ney 3-gram LM trained on additional text data from Wikipedia (indicated by +WLM in Table 3). This greatly reduced the WER, as can be seen from Table 3. These improvements suggest that the lexical variability in Marathi and Odia datasets is comparatively very small, and by adding an LM trained on external data, the ASR model can decode the utterances much better. We also tried the same for Gujarati, Tamil, and Telugu, but observed a significant degradation in perplexity scores on the respective dev sets. Hence we did not proceed with the decoding using Wikipedia LM. Moreover Gujarati, Tamil and Telugu datasets from IS 2018 are from news domain and Wikipedia text might not be an optimal choice for additional LM training.

Table 4 presents the MTWV and ATWV scores of four KWS systems across all the six languages. The LM weight and detection threshold (θ) that yielded maximum TWV on dev set are used for the test set. The LM weight is used during generating lattices, whereas the detection threshold is used to make hard decisions, in order to compute ATWV. From Table 4 it can be seen that DNN based KWS systems have higher ATWV scores as compared to GMM based systems, and in all cases the TDNN based KWS systems perform better than DNN-sMBR and DNN based KWS systems. For Marathi, and Odia languages, the KWS systems based on a smaller LM (i.e., LM trained only on training set) did not succeed to give a positive ATWV score, hence those values are left blank. It is possible that the selected keywords are too difficult for the KWS system to detect. But, by adding LM trained on Wikipedia text, decent ATWV scores are obtained for Marathi, and Odia. This can also be attributed to the size and variability of the vocabulary in these languages. The Table 4 also presents the MTWV scores on the test set. The difference between ATWV and MTWV scores show the loss in selecting a sub-optimal threshold. We can observe from Table 4 that these differences are mainly in second decimal, suggesting that our threshold selection is near optimal. This also suggests that the speaker/channel variations

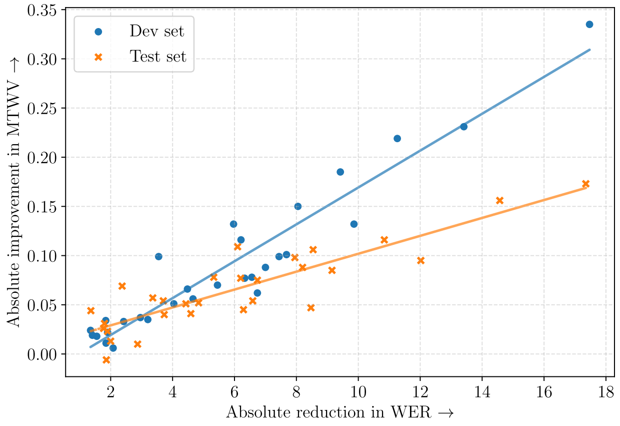


FIGURE 4. Illustration of absolute improvements in MTWV with respect to absolute reductions in word-error-rates (WER). The statistics are computed from Tables 3, and 4, for Gujarati, Tamil, Telugu and Hindi languages.

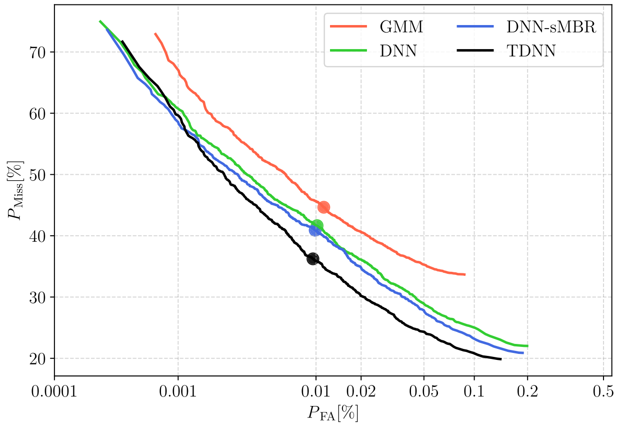


FIGURE 5. Detection-error-trade-off for all four KWS systems on Telugu. • indicates the P_{Miss} and P_{FA} that corresponds to ATWV.

and the difficulty of keywords in dev and test sets are consistent.

The performance of ASR-based KWS systems directly depends on the word-error-rates of the underlying ASR. Tables 3 and 4 show that reduction in WER has given an improvement in ATWV. To make this even more clear, we took the WER and ATWV scores for 4 languages (Gujarati, Tamil, Telugu and Hindi), and computed absolute reductions in WER with respect to every ASR system, and the corresponding changes in the ATWV. More specifically, we consider the absolute WER differences (from Table 3) between GMM based ASR system to the rest of the three systems, then with DNN based ASR to the remaining two, and finally between DNN-sMBR and TDNN based ASR systems. Similarly, we consider the absolute differences in MTWV (from Table 4) of these systems in the same order. These are illustrated in Fig. 4, and it can be seen that the absolute reductions in WER are correlated with the improvements in the ATWV. It is expected that the correlation is stronger for dev data than for test data, since we tuned the systems on dev data. For Fig. 4 we did not consider Marathi and Odia as they seem like outliers (due to limited lexical variability).

Fig. 5 presents the detection-error-trade-off (DET) curves of all the four KWS systems on the Telugu dataset. Although the probability of FA at the ATWV operating point is close across all the systems, the miss probability is much lower for TDNN based KWS system, as compared to others. Similar trends in DET curves are observed for other languages.

VIII. CONCLUSION

In this paper, we presented the idea of creating a suitable set of candidate keywords for keyword search from spoken audio, while using existing freely available speech datasets. We re-processed IS 2018 and MUCS 2021 datasets comprising six low-resource languages, and created keywords by taking into account the three properties (keyword occurrence, length and confusability distance), and their effect on the term-weighted value (the KWS evaluation metric). We trained four ASR-based KWS systems to benchmark KWS results on the created resources. We provided an in-depth analysis of keyword properties and their effect on TWV across all the languages and KWS systems.

While the current work focused on creating baseline systems, future research works would focus on multilingual KWS systems, and out-of-vocabulary keyword search. We believe that the analysis and guidelines provided in this paper will not only help the research community, but also practitioners and engineers to easily create KWS resources for newer languages, datasets, and scenarios.

ACKNOWLEDGMENT

Vijaya Lakshmi V. Nadimpalli acknowledges the support of Dr. Anil Kumar Vuppala, Faculty, IIIT Hyderabad; Sudarsana Reddy Kadiri; and Hemant Kumar Kathania for the useful discussions. The authors would like to thank the anonymous reviewers, whose feedback has greatly helped to improve the paper. This work was done while at the International Institute of Information Technology, Hyderabad for Rohit Banka. *(Vijaya Lakshmi V. Nadimpalli, Santosh Kesiraju, and Rohit Banka contributed equally to this work).*

REFERENCES

[1] P. Joshi, S. Santy, A. Budhiraja, K. Bali, and M. Choudhury, "The state and fate of linguistic diversity and inclusion in the NLP world," in *Proc. 58th Annu. Meeting Assoc. Comput. Linguistics*, 2020, pp. 6282–6293.

[2] A. Thambiratnam, "Acoustic keyword spotting in speech with applications to data mining," Ph.D. dissertation, Dept. QUT Sci. Eng., Queensland Univ. Technol., Brisbane, QLD, Australia, 2005.

[3] P. D. S. Karthik, M. S. Saranya, and H. A. Murthy, "A fast query-by-example spoken term detection for zero resource languages," in *Proc. Int. Conf. Signal Process. Commun. (SPCOM)*, Jun. 2016, pp. 1–5.

[4] A. H. Michaely, X. Zhang, G. Simko, C. Parada, and P. Aleksic, "Keyword spotting for Google assistant using contextual speech recognition," in *Proc. IEEE Autom. Speech Recognit. Understand. Workshop (ASRU)*, Dec. 2017, pp. 272–278.

[5] M. J. Gales, K. M. Knill, A. Ragni, and S. P. Rath, "Speech recognition and keyword spotting for low-resource languages: Babel project research at CUED," in *Spoken Language Technologies for Under-Resourced Languages*. ISCA, 2014, pp. 16–23.

[6] Y. Zhang and J. R. Glass, "Unsupervised spoken keyword spotting via segmental DTW on Gaussian posteriorgrams," in *Proc. IEEE Workshop Autom. Speech Recognit. Understand.*, Dec. 2009, pp. 398–403.

[7] G. Mantena, S. Achanta, and K. Prahallad, "Query-by-example spoken term detection using frequency domain linear prediction and non-segmental dynamic time warping," *IEEE Trans. Audio, Speech, Language Process.*, vol. 22, no. 5, pp. 946–955, May 2014.

[8] L.-S. Lee, J. Glass, H.-Y. Lee, and C.-A. Chan, "Spoken content retrieval—Beyond cascading speech recognition with text retrieval," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 23, no. 9, pp. 1389–1420, Sep. 2015.

[9] D. Ram, L. Miculicich, and H. Bourlard, "Neural network based end-to-end query by example spoken term detection," *IEEE/ACM Trans. Audio, Speech, Language Process.*, vol. 28, pp. 1416–1427, 2020.

[10] G. Mantena and K. Prahallad, "Use of articulatory bottle-neck features for query-by-example spoken term detection in low resource scenarios," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2014, pp. 7128–7132.

[11] S. Kesiraju, G. Mantena, and K. Prahallad, "IIIT-H system for MediaEval 2014 QUESST," in *Proc. Work. Notes MediaEval workshop*, vol. 1263, 2014, pp. 1–2.

[12] F. Grézl and M. Karafiát, "Bottle-neck feature extraction structures for multilingual training and porting," *Proc. Comput. Sci.*, vol. 81, pp. 144–151, Jan. 2016.

[13] D. Ram, L. Miculicich, and H. Bourlard, "Multilingual bottleneck features for query by example spoken term detection," in *Proc. IEEE Autom. Speech Recognit. Understand. Workshop (ASRU)*, Dec. 2019, pp. 621–628.

[14] K. Audhkhasi, A. Rosenberg, A. Sethy, B. Ramabhadran, and B. Kingsbury, "End-to-end ASR-free keyword search from speech," *IEEE J. Sel. Topics Signal Process.*, vol. 11, no. 8, pp. 1351–1359, Dec. 2017.

[15] Z. Zhao and W.-Q. Zhang, "End-to-end keyword search based on attention and energy scorer for low resource languages," in *Proc. Interspeech*, Oct. 2020, pp. 2587–2591.

[16] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, pp. 6–12.

[17] D. Seo, H.-S. Oh, and Y. Jung, "Wav2KWS: Transfer learning from speech representations for keyword spotting," *IEEE Access*, vol. 9, pp. 80682–80691, 2021.

[18] *NIST OpenKWS Evaluation 15*. Accessed: Mar. 29, 2022. [Online]. Available: https://www.nist.gov/system/files/documents/itl/iad/mig/KWS15-evalplan-%v05.pdf

[19] *NIST OpenKWS Evaluation 16*. Accessed: Mar. 29, 2022. [Online]. Available: https://www.nist.gov/itl/iad/mig/open-keyword-search-evaluation

[20] *IARPA Babel Program*. Accessed: Mar. 29, 2022. [Online]. Available: https://www.iarpa.gov/index.php/research-programs/babel

[21] 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 *Proc. Autom. Speech Recognit. Understand. Workshop (ASRU)*, 2011, pp. 1–4.

[22] G. Chen, S. Khudanpur, D. Povey, J. Trmal, D. Yarowsky, and O. Yilmaz, "Quantifying the value of pronunciation lexicons for keyword search in lowresource languages," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, May 2013, pp. 8560–8564.

[23] J. Trmal, M. Wiesner, V. Peddinti, X. Zhang, P. Ghahremani, Y. Wang, V. Manohar, H. Xu, D. Povey, and S. Khudanpur, "The kaldi OpenKWS system: Improving low resource keyword search," in *Proc. Interspeech*, Aug. 2017, pp. 3597–3601.

[24] F. Metze, N. Rajput, X. Anguera, M. Davel, G. Gravier, C. van Heerden, G. V. Mantena, A. Muscariello, K. Prahallad, I. Szoke, and J. Tejedor, "The spoken web search task at MediaEval 2011," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Mar. 2012, pp. 5165–5168.

[25] X. Anguera, F. Metze, A. Buzo, I. Szöke, and L. J. Rodríguez-Fuentes, "The spoken web search task," in *Proc. MediaEval Multimedia Benchmark Workshop*, vol. 1043, 2013, pp. 1–2.

[26] L. J. Rodriguez-Fuentes and M. Penagarikano, "Mediaeval 2013 spoken web search task: System performance measures," Dept. Electr. Electron., Univ. Basque Country, Biscay, Spain, Tech. Rep. TR-2013-1, 2013.

[27] L. J. Rodriguez-Fuentes, A. Varona, M. Penagarikano, G. Bordel, and M. Diez, "High-performance query-by-Example spoken term detection on the SWS 2013 evaluation," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2014, pp. 7819–7823.

[28] X. Anguera, L. J. Rodríguez-Fuentes, I. Szöke, A. Buzo, and F. Metze, "Query by example search on speech at MediaEval 2014," in *Proc. Work. Notes MediaEval Workshop*, vol. 1263, 2014, pp. 1–2.

[29] I. Szöke, L. J. Rodríguez-Fuentes, A. Buzo, X. Anguera, F. Metze, J. Proença, M. Lojka, and X. Xiong, "Query by example search on speech at MediaEval 2015," in *Proc. Work. Notes MediaEval workshop*, vol. 1436, 2015, pp. 1–3.

[30] W. Hartmann, D. Karakos, R. Hsiao, L. Zhang, T. Alumae, S. Tsakalidis, and R. Schwartz, "Analysis of keyword spotting performance across IARPA Babel languages," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Mar. 2017, pp. 5765–5769.

[31] M. J. Gales, K. M. Knill, and A. Ragni, "Low-resource speech recognition and keyword-spotting," in *Proc. Int. Conf. Speech Comput.*, Sep. 2017, pp. 3–19.

[32] J. Fiscus, J. Ajot, J. Garofolo, and G. Doddingtion, "Results of the 2006 spoken term detection evaluation," in *Proc. ACM SIGIR Workshop Searching Spontaneous Conversational*, vol. 7, 2007, pp. 51–55.

[33] F. Metze, X. Anguera, E. Barnard, M. Davel, and G. Gravier, "The spoken web search task at MediaEval 2012," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, May 2013, pp. 8121–8125.

[34] S. Shah, S. Guha, S. Khanuja, and S. Sitaram, "Cross-lingual and multilingual spoken term detection for low-resource Indian languages," 2020, *arXiv:2011.06226*.

[35] E. Barnard, M. Davel, and C. V. Heerden, "ASR corpus design for resource-scarce languages," in *Proc. Interspeech*, Sep. 2009, pp. 2847–2850.

[36] S. Wegmann, A. Faria, A. Janin, K. Riedhammer, and N. Morgan, "The TAO of ATWV: Probing the mysteries of keyword search performance," in *Proc. IEEE Workshop Autom. Speech Recognit. Understand.*, Dec. 2013, pp. 192–197.

[37] D. Karakos, R. Schwartz, S. Tsakalidis, L. Zhang, S. Ranjan, T. Ng, R. Hsiao, G. Saikumar, I. Bulyko, L. Nguyen, J. Makhoul, F. Grezl, M. Hannemann, M. Karafiat, I. Szoke, K. Vesely, L. Lamel, and V.-B. Le, "Score normalization and system combination for improved keyword spotting," in *Proc. IEEE Workshop Autom. Speech Recognit. Understand.*, Dec. 2013, pp. 210–215.

[38] K. Audhkhasi, A. Sethy, B. Ramabhadran, and S. S. Narayanan, "Semi-supervised term-weighted value rescoring for keyword search," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2014, pp. 7869–7873.

[39] M. Michel, D. Joy, J. Fiscus, V. Dreyvitser, V. Manohar, J. Ajot, and B. N. Barr. *Framework for Detection Evaluation (F4de)*. Accessed: Jan. 16, 2022. [Online]. Available: https://github.com/usnistgov/F4DE

[40] B. M. L. Srivastava, S. Sitaram, R. K. Mehta, K. D. Mohan, P. Matani, S. Satpal, K. Bali, R. Srikanth, and N. Nayak, "Interspeech 2018 low resource automatic speech recognition challenge for Indian languages," in *Proc. 6th Workshop Spoken Lang. Technol. Under-Resourced Lang. (SLTU)*, Aug. 2018, pp. 11–14.

[41] A. Diwan, R. Vaideeswaran, S. Shah, A. Singh, S. Raghavan, S. Khare, V. Unni, S. Vyas, A. Rajpuria, C. Yarra, A. Mittal, P. K. Ghosh, P. Jyothi, K. Bali, V. Seshadri, S. Sitaram, S. Bharadwaj, J. Nanavati, R. Nanavati, and K. Sankaranarayanan, "MUCS 2021: Multilingual and code-switching ASR challenges for low resource Indian languages," in *Proc. Interspeech*, Aug. 2021, pp. 2446–2450.

[42] B. Pulugundla, M. K. Baskar, S. Kesiraju, E. Egorova, M. Karafiát, L. Burget, and J. Ćernocký, "BUT system for low resource Indian language ASR," in *Proc. Interspeech*, Sep. 2018, pp. 3182–3186.

[43] M. Bisani and H. Ney, "Joint-sequence models for grapheme-to-phoneme conversion," *Speech Commun.*, vol. 50, no. 5, pp. 434–451, 2008.

[44] D. Can and M. Saraclar, "Lattice indexing for spoken term detection," *IEEE Trans. Audio, Speech, Language Process.*, vol. 19, no. 8, pp. 2338–2347, Nov. 2011.

[45] M. J. F. Gales, "Maximum likelihood linear transformations for HMM-based speech recognition," *Comput. Speech Lang.*, vol. 12, no. 2, pp. 75–98, Apr. 1998.

[46] K. Veselý, A. Ghoshal, L. Burget, and D. Povey, "Sequence-discriminative training of deep neural networks," in *Proc. Interspeech*, Aug. 2013, pp. 2345–2349.

[47] A. Waibel, "Consonant recognition by modular construction of large phonemic time-delay neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 1988, pp. 215–223.

[48] V. Peddinti, D. Povey, and S. Khudanpur, "A time delay neural network architecture for efficient modeling of long temporal contexts," in *Proc. Interspeech*, Sep. 2015, pp. 3214–3218.

[49] N. Dehak, P. J. Kenny, R. Dehak, D. Pierre, and O. Pierre, "Front-end factor analysis for speaker verification," *IEEE Trans. Audio, Speech, Language Process.*, vol. 19, no. 4, pp. 788–798, May 2011.

VIJAYA LAKSHMI V. NADIMPALLI received the M.Sc. degree from the University of Hyderabad, India, in 1984, and the Ph.D. degree in mathematics for the thesis, "Design Space Exploration of Nonlinear Systems" from ACRHEM, University of Hyderabad, in 2011. She received Junior and Senior Research Fellowships (JRF and SRF) funded by DRDO while pursuing the Ph.D. degree. She also actively participated and contributed significantly towards the execution and completion of a classified DRDO-funded project by HEMRL, Pune, India. She pursued this work as a Principal Investigator during the tenure of a project sanctioned by the Speech Processing Laboratory, Department of Science and Technology (DST, WOS-A), International Institute of Information Technology, Hyderabad, India. Her research interests include simulation and modeling, numerical methods, soft computing techniques, neural networks, and audio search. She was a recipient of National Merit Scholarship during post graduation.

SANTOSH KESIRAJU received the Ph.D. degree from the International Institute of Information Technology, Hyderabad, India, in 2021. He is currently a Postdoctoral Researcher with the Brno University of Technology, Czechia. His research interests include dialogue systems, automatic speech recognition, language modeling, multilingual representations, machine learning, and Bayesian methods.

ROHITH BANKA received the Bachelor of Technology degree in electronics and communication engineering (ECE) from Jawaharlal Nehru Technological University (JNTUH)—Sultanpur, Sangareddy, India, in 2018, and the Master of Science degree in information technology (IT) from the International Institute of Information Technology, Hyderabad, India, in 2020. He was worked as a Research Intern with the Speech Laboratory—KCIS, IIIT Hyderabad. He is currently working as a Machine Learning Engineer and a Google Certified TensorFlow Developer. His research interests include deep learning, computer vision, and natural language processing.

RASHMI KETHIREDDY received the Bachelor of Technology degree from the Kakatiya Institute of Technology and Science, Warangal, India, in 2011, with a specialization in information technology (IT), and the Master of Technology degree from Osmania University, Hyderabad, India, in 2017, with a specialization in computer science engineering. Then, she worked in IT services for a period of two years. She qualified for University Grant Commission National Eligibility Test (UGC-NET) and hence was awarded with Junior and Senior Research Fellowship. She is currently a Ph.D. Scholar with the International Institute of Information Technology, Hyderabad (IIIT-H). Her research interests include speech signal processing, acoustic analysis, machine learning, speech dialectal challenges, and speech dialect identification.

SURYAKANTH V. GANGASHETTY (Member, IEEE) received the Ph.D. degree in neural network models for recognition of consonant-vowel units of speech in multiple languages from IIT Madras, in 2005. He is currently a Faculty Member at KL University, Green Field, Vaddeswaram, Guntur, Andhra Pradesh, India. Before joining to KL University, he was worked as a Faculty Member at IIIT Hyderabad, Telangana, from 2006 to 2020. Previously, he was worked as a Senior Project Officer with the Speech and Vision Laboratory, IIT Madras. He was worked as a Faculty Member at BIET Davangere, Karnataka, from 1991 to 1999. He was also worked as a Visiting Research Scholar at OGI Portland (USA) for three months, in summer 2001. He has done his postdoctoral studies (PDF) with Carnegie Mellon University (CMU), Pittsburgh, PA, USA, from April 2007 to July 2008. He is the author of about 150 papers published in national and international journals, conferences, and edited volumes. He is a Life Member of the CSI, IE, IUPRAI, ASI, IETE, ORSI, and ISTE. He has reviewed papers for reputed journals and conferences. His research interests include speech processing, neural networks, machine learning, natural language processing, and artificial intelligence. He was the Local Organizing Chair for the INTERSPEECH-2018 Conference, Hyderabad, India, in September 2018.