IndicSUPERB: A Speech Processing Universal Performance Benchmark for Indian languages

Tahir Javed[1,2], **Kaushal Bhogale**[1,2], **Abhigyan Raman**[2]
Pratyush Kumar[2,3], **Anoop Kunchukuttan**[2,3], **Mitesh M. Khapra**[1,2]
[1]Indian Institute of Technology Madras
[2]AI4Bharat
[3]Microsoft
{tahir, cs22d006}@cse.iitm.ac.in, ramanabhigyan@gmail.com
pratyush@cse.iitm.ac.in, ankunchu@microsoft.com, miteshk@cse.iitm.ac.in

Abstract

A cornerstone in AI research has been the creation and adoption of standardized training and test datasets to earmark the progress of state-of-the-art models. A particularly successful example is the GLUE dataset for training and evaluating Natural Language Understanding (NLU) models for English. The large body of research around self-supervised BERT-based language models revolved around performance improvements on NLU tasks in GLUE. The success of large self-supervised models such as wav2vec2 enable creation of speech models with relative ease to access unlabelled data. These models can then be evaluated on SLU tasks, such as the SUPERB benchmark. In this work, we extend this to Indic languages by releasing the IndicSUPERB benchmark. Specifically, we make the following three contributions. (i) We collect Kathbath containing 1,684 hours of labelled speech data across 12 Indian languages from 1,218 contributors located in 203 districts in India. (ii) Using Kathbath, we create benchmarks across 6 speech tasks: Automatic Speech Recognition, Speaker Verification, Speaker Identification (mono/multi), Language Identification, Query By Example, and Keyword Spotting for 12 languages. (iii) On the released benchmarks, we train and evaluate different self-supervised models alongside a commonly used baseline FBANK. We show that language-specific fine-tuned models are more accurate than baseline on most of the tasks, including a large gap of 76% for Language Identification task. However, for speaker identification, self-supervised models trained on large datasets demonstrate an advantage. We hope IndicSUPERB contributes to the progress of developing speech language understanding models for Indian languages.

Introduction

Over the past few years, several works have shown that self-supervised learning (SSL) is very effective for natural language processing, vision and speech tasks (Devlin et al. 2019; Newell and Deng 2020; Yang et al. 2021a). The key idea is to pre-train a large scale model on easily available unlabelled data and then adapt the same model for a wide variety of tasks by fine-tuning on smaller amounts of task specific data. Given the quantity and diversity of the pre-training data, such models learn to encode general purpose representations which can then be specialised for a wide variety of tasks. To evaluate the effectiveness of SSL, several diverse benchmarks such as GLUE (Wang et al. 2018), XTREME (Hu et al. 2020), SUPERB (Yang et al. 2021a), *etc.* have been created. Creating such benchmarks, while expensive and time consuming, has become very crucial for driving progress. In particular, languages for which such benchmarks do not exist often do not receive their fair share of representation in academic fora (Joshi et al. 2020).

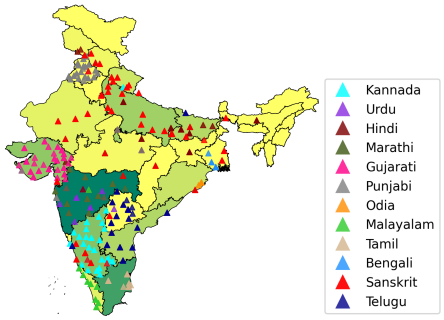

Figure 1: Distribution of Speakers across 22 Indian States

A case in point is that of languages from the Indian subcontinent. India has 22 constitutionally recognised languages with a collective speaker base of over 1 Billion speakers. On one hand, there is an acute need for building speech and language understanding models for Indian languages, while on the other hand, there is an acute shortage of good benchmarks for training and evaluating such models. Recently released benchmarks such as IndicGLUE (Kakwani et al. 2020) and FLORES-200 (NLLB Team et al. 2022) have alleviated the situation to a certain extent for natural language understanding and machine translation. However, for Speech Language Understanding (SLU), to the best of our knowledge, such a diverse benchmark is only available for English (Yang et al. 2021a). In this work, we address this gap and build a SLU benchmark for 12 Indian languages covering 6 speech tasks: Automatic Speech Recognition, Speaker Verification, Speaker Identification, Language Identification, Query By Example, and Keyword Spotting.

Our first main contribution is to build a large dataset for automatic speech recognition containing read speech across

12 Indian languages from 1,218 speakers spanning 203 districts (see Figure 1). This is a one of its kind open-source effort resulting in a very large dataset containing 1,684 hours of labelled speech recognition data. The data was collected with the help of paid speakers using a *maker-checker* flow. Every spoken utterance was verified by a human to check that (i) there was no objectionable content in the audio, (ii) the recorded audio was faithful to the source text, and (iii) the recorded audio was clearly audible, *i.e.*, it had appropriate volume and no background noise. For each of the 12 languages, we provided at least 100 hours of training data which will help in improving ASR models. Note that for 2 of the 12 languages, no ASR training data is publicly available. Further, for 5 languages, our data helps in at least doubling the amount of training data in existing datasets. We refer to this dataset as Kathbath[1].

Next, using Kathbath, we create a benchmark for four SLU tasks, *viz.* automatic speech recognition, speaker verification, speaker identification and language identification. For the latter three tasks, we use the speaker and language information associated with each utterance to create training and evaluation sets with appropriate target labels. Further, to allow a robust evaluation of speech models, we create multiple test conditions, *viz.*, (i) a test set containing speakers seen during training, (ii) a test set containing novel speakers which are not seen during training, (iii) a test set containing noisy data, *i.e.*, data which was rejected during the verification stage, and (iv) gender balanced test sets to evaluate the effect of having a higher number of female speakers in the training data. For speaker identification, we also create a language agnostic test set by pooling speakers from all the 12 languages. Apart from these four tasks, we also create a benchmark for query-by-example, wherein we ask speakers to utter a specific query and then collect all utterances containing this query. Lastly, we create a benchmark for keyword spotting which involves classifying an utterance into a predefined set of keywords. We refer to the above benchmark containing 6 tasks for 12 languages as IndicSUPERB.

We believe that using IndicSUPERB it would be possible to answer several research questions which have previously not been addressed in the context of Indian languages. In this work itself, we address a few such questions: (i) How effective are SSL models as feature encoders for Indian languages? (ii) Are language-family specific SSL models better than those pretrained on a larger set of languages? (iii) How does the performance generalise to unknown speakers? (iv) How robust are existing models to gender-specific biases in the training data? (v) How robust are existing models to noise in the utterances? Based on our experiments we observe that SSL models are good feature encoders with models pretrained using data from a diverse set of speakers being more suitable for speaker verification tasks. We also observe that while for some of the tasks, these models generalise well to unknown speakers in the test set and are reasonably robust to noise, they do get affected by specific biases (such as gender bias) in the training data. All the code, datasets and models developed as a part of this work have been made publicly available[2] and we hope that they will help in furthering research on speech technology for Indian languages.

	ours	mucs	msr	csc	iisc	csd	kdc	cv	isd	htc	vac	ith	imc	smc
ASR	☑	☑	☑	☑	☑	☑	☑	☑	☑	☑	☑	☑	☑	☑
SID	☑	x	x	x	x	x	x	x	x	x	x	x	x	x
LID	☑	x	x	x	x	x	x	x	x	x	x	x	x	x
ASV	☑	x	x	x	x	x	x	x	x	x	x	x	x	x
QbE	☑	x	x	x	x	x	x	x	x	x	x	x	x	x
KS	☑	x	x	x	x	x	x	x	x	x	x	x	x	x
#L	12	6	3	3	2	6	1	6	7	2	1	1	1	1
#A	140	67	40	206	214	3	3	60	2	141	56	2k	109	1

Table 1: Comparison of IndicSUPERB (ours) with other datasets for Indian languages. (#L = no. of languages supported, #A = avg. # hours in training data for ASR)

Related Work

As mentioned earlier, there are many datasets available for English, such as Librispeech (Panayotov et al. 2015) and Common Voice(Ardila et al. 2020) for ASR, VoxCeleb1 (Nagrani, Chung, and Zisserman 2017) and VoxCeleb2 (Chung, Nagrani, and Zisserman 2018) for speaker recognition. More recently, SUPERB (Yang et al. 2021b) and SUPERB-SG (Tsai et al. 2022) have been released and contain various speech language understanding and synthesis tasks. In contrast, there are very few datasets for Indian languages as summarised in Table 1. Most of these datasets such as MUCS (Diwan et al. 2021), MSR (Srivastava et al. 2018), Gramvaani(Bhanushali et al. 2022), Crowdsourced Speech Corpora (CSC) (Kjartansson et al. 2018a), IISC-MILE Corpus (Ayyavu, Pilar, and G 2022), Crowdsourced Multispeaker Speech Dataset (CSD) (He et al. 2020), Kashmiri Data Corpus (KDC)[3], Common Voice, IIIT-H Indic Speech Databases (ISD) (Prahallad et al. 2012), Hindi-Tamil ASR Challenge[4] (HTC), Vāksañcayaḥ (VAC)(Adiga et al. 2021), IIIT-H Telugu Corpus(Ganesh et al. 2021) (ITH), IITB Marathi Corpus (IMC) (Abraham et al. 2020a) and SMC Malayalam Corpus [5] contain only ASR data. Further, most of them support very few languages. One interesting effort worth mentioning is the LDCIL repository, which contains datasets for 22 Indian languages with all the meta information about speakers, language, gender and hence can be used to create a complete benchmark covering all the tasks. However, this data is not freely available and is shared selectively after a thorough vetting process to ensure non-commercial usage. To the best of our knowledge, ours is the first effort for creating a robust benchmark for a diverse set of SLU tasks across 12 Indian languages.

[1] In Kashmiri, Kathbath means talks and conversations

[2] https://github.com/AI4Bharat/indicSUPERB

[3] https://www.openslr.org/122/

[4] https://sites.google.com/view/indian-language-asrchallenge

[5] https://blog.smc.org.in/malayalam-speech-corpus/

Data Collection Methodology

The first step towards creating IndicSUPERB was to collect voice samples and their transcriptions from a diverse set of speakers for multiple Indian languages. Once such voice samples were collected, we could use the meta-data associated with them (such as, speaker/language information) to create data for different tasks as mentioned earlier. We now describe the various choices that we made for collecting these voice samples which form the core of our dataset.

Languages: Ideally, we would have liked to create IndicSUPERB for all the 22 constitutionally recognised languages in India, but based on budget restrictions we had to choose 12 languages. We selected these languages to have a fair representation from North, South, East and West of India. The specific languages that we selected were Kannada, Malayalam, Tamil, and Telugu from South India, Gujarati and Marathi from West India, Bengali and Odia from East India, Hindi and Punjabi from North India, and Sanskrit and Urdu which are spoken in different parts of the country.

Type of Data: While collecting speech recognition data, there are three main options that one could consider. The first option is to collect read-speech wherein participants are shown a piece of text and are asked to speak it out. The second option is to ask participants to speak on a topic without a script and then transcribe the data. The third option is to curate existing audio content from the web (such as YouTube) and transcribe it. In line with similar efforts for low resource languages (such as the Mozilla Common Voice project) we decided to collect read speech data which is cheaper. The other advantage is that it is easier for speakers to participate as opposed to collecting unscripted extempore data where many participants may not be able to speak more than a few words. Further, unlike YouTube data where there are issues in finding the meta-data associated with the speaker (age, geography, *etc.*), here the participants can explicitly provide such information. Such meta-data is important for measuring the diversity of the data as well as for creating data for downstream tasks, such as, speaker identification.

Selection of Text: We used IndicCorp (Kakwani et al. 2020) for collecting sentences which can then be read by speakers. IndicCorp is the largest publicly available collection of monolingual corpora for Indian languages collected from a diverse set of India-specific sources on the web (such as news articles, government websites, *etc*). We collected $\sim$100K sentences for each of the 12 languages from IndicCorp while restricting the character set to alphanumeric only and the sentence length to 8-15 words. This was done to ensure that the sentences were clean and small enough to be spoken fluently by the annotators. The size of the vocabulary derived from these sentences is summarized in Table 2.

Tool for Data Collection and Verification: Our data collection started during the COVID-19 pandemic making it difficult for participants to physically assemble at one location. We thus needed a tool which allowed us to distribute and track work remotely to a large number of participants who were geographically spread out. To do so, we used Microsoft's open source crowdsourcing platform called Karya which was used in the past to collect voice data (Abraham et al. 2020b). Karya is available as an Android application which can run offline and sync with the backend server at periodic intervals to store the collected data. For every language, we uploaded 100K sentences into Karya which could be accessed in batches of 100. The distribution of tasks was managed with the help of access codes. In particular, Karya generated an access code for each batch of 100 sentences. These access codes were shared with participants who could then log in to the tool and work on the corresponding tasks. Once a task is completed, Karya initiates a verification workflow wherein the collected data is verified by a human.

Ensuring Quality Control: To ensure that the recorded data is of high quality, it is essential that (i) the text data does not contain any objectionable content (ii) the audio samples are accurate, *i.e.*, they contain the same content as in the text data (iii) the audio samples are clearly audible with no background ambient noise. To ensure this, we took the following steps. First, during the recording stage, the participants were strictly instructed to skip any sentences which contained objectionable content. In addition, each recorded sample was also passed through a verification stage, where the verifiers were again asked to discard any recordings which contained objectionable content even if they were accurate and audible. Apart from eliminating all objectionable content, the verifiers were asked to score each recorded sample on 3 parameters: *accuracy*, *volume*, *quality*. Each score ranges from 0 to 2. A recorded sample would be given a score of 2 on accuracy only if it exactly matched the source sentence. Similarly, a score of 2 for volume would be given only if the recorded sample was clearly audible. Lastly, a score of 2 on quality would be given only if the recorded sample had no background noise. We accepted only those samples which had a score of 2 for all the 3 parameters.

During the initial phases of data collection, we observed that nearly 60% of the data was getting discarded during the verification stage. On further analysis, we found that the main reasons for rejection were (i) inaccurate reading of the source text (ii) presence of background noise. To avoid inaccurate recordings, we requested the annotators to practice by reading the sentence aloud a couple of times before actually recording it. We also asked them to skip a sentence if they were unsure about the pronunciation of certain words in the sentences or found certain words to be difficult to pronounce. To avoid background noise, we requested participants to record sentences during the night when noise from vehicles, household chatter, *etc.* was minimal. With these instructions, we found a significant reduction in the rejection rate. Table 2 summarises the total amount of clean and noisy data that was collected as a part of the process.

Ensuring Diversity: To ensure diversity in the collected data, we ensured that the participants came from different districts, age groups and genders. In total, we collected data from 1,218 speakers spanning 203 districts across 22 states in India. For most languages, we ensured that the collected data had a much higher proportion of female speakers as female participants are often poorly represented in many AI datasets. However, as explained later, while creating test sets we ensured that there were equal number of male and female speakers as well as equal duration of audio from male and female speakers. This would allow a more systematic study

	bn	gu	hi	kn	ml	mr	or	pa	sa	ta	te	ur
CD	116	129	150	166	147	185	112	137	116	185	155	87
ND	39	65	48	65	16	87	28	37	76	96	75	77
VS	0.1	1.1	0.5	1.8	2.7	1.3	0.9	0.6	3.0	1.7	1.5	0.4

Table 2: No. of hours of audio data and vocabulary size (VS - in million) in Kathbath. (CD=Clean data, ND=Noisy data)

of the effect of bias in training data on unbiased test sets. Table 3 shows the total number of male and female speakers in each language. Figure 1 shows the different states from which the data was collected. Please refer to the supplementary material for the age-wise distribution of speakers.

Consent of Participants: For every language, we had a local coordinator from a data collection agency who clearly explained the nature of the activity to the participant. The local coordinator ensured that each participant understood that his/her voice samples would be publicly released and used for building AI models and other research purposes. The terms and conditions and the nature of activity were also clearly shown when the participants logged into Karya. The recording would be enabled only after the participants read the instructions and provided their consent. No Personal Identifiable Information was collected from the participants.

Cost of Data Collection: The cost was INR 1500 per hour (approximately 25 USD per hour) including logistics costs. Each participant was paid between INR 500 - 1000 for 1 hour of recording (approximately 6 to 12 USD per hour).

IndicSUPERB

The above process resulted in a total of 1,684 hours of read speech across 12 languages with a total of 0.9M utterances. Each utterance in the data is unique, *i.e.*, a given sentence is only spoken by one speaker and never repeated again. We standardized the audio data by converting it into 16Khz WAV with mono channel. We now explain the procedure for creating the IndicSUPERB benchmark from this data.

Train, Validation and Test Splits

Our goal was to provide training as well as evaluation data for various SLU tasks. Further, we wanted that the benchmark should support different conditions, *e.g.*, (i) evaluation for speakers existing in the training data (ii) evaluation for speakers not existing in the training data (iii) evaluation on noisy data. To enable this, we divided the data into training set and multiple validation and test sets as explained below.

Test-Unknown: For each of the 12 languages, we first set aside audio recordings corresponding to 10 male and 10 female speakers contributing a total of 3 hours (1.5h each). We then removed all the instances of these speakers from the rest of the data. This ensured that these speakers are only seen at test time and are unseen during training and validation.

Test-Known: For creating a test set containing known speakers, for each of the 12 languages, we again take 10 male and 10 female speakers and sample audio recordings for them containing a total of 5 hours (2.5h each). Unlike before, we do not remove other instances of these speakers from the rest of the data. Hence, some data for these speakers is also seen during training.

Validation: We create this using the same procedure as used for creating the test-known set. Note that some of the speakers in test-known are also present in the validation data.

Training Data: We use the remaining data as training data. We reiterate that some of the speakers in the validation and test-known splits are present in the training data but none of the speakers in the test-unknown split are present in the training data. Further, since each utterance is unique, no sentence in the training data is present in any of the other splits.

Next, we create some noisy test sets which would help in evaluating the robustness of models. For this, we consider the data that was rejected during the verification stage and extract all recordings which had a rating of at least 1 for all the 3 parameters (accuracy, volume, quality). We refer to this as the noisy set and create the following splits from it.

Noisy-Test-Unknown: A total of 3 hours of noisy data from 10 male and 10 female speakers which are not seen in the training data described above.

Noisy-Test-Known: A total of 5 hours of noisy data from 10 male and 10 female speakers which are also seen in the training data described above.

Table 3 summarises the statistics of all the above splits.

Speech Language Understanding Tasks

We now described the different tasks covered in IndicSUPERB and the method used for creating training and evaluation data for these tasks.

Automatic Speech Recognition (ASR) ASR is the task of transcribing a given audio utterance into text. For training and evaluating an ASR system we need an audio utterance paired with its text transcript. Since the collected data already contains this alignment by design, no extra processing was needed to create training or validation data for ASR. We simply use the aligned audio-transcript pairs in the splits described earlier for training, validation and testing.

Language Identification (LID) In LID, the task is to take the raw audio as input and classify it into one of the given n languages ($n = 12$ in this case). To create training data for this task, we simply pooled the audio recordings from all the languages in the ASR training data described earlier and assigned the known language ID as the target label. We thus get labeled data for an n-class classification problem. We similarly create the multiple test and validation sets from the corresponding splits of ASR.

Speaker Identification (SID) SID is the task of identifying the speaker in an audio clip. In other words, the task is to take the raw audio as input and classify it into one of the given k speaker IDs. We consider two scenarios here. In the first type, we identify speakers within a given language. We refer to this as SID-mono. To create training data for this task, we pooled the audio recordings from all the speakers for a given language in the ASR training data described earlier and assigned the known speaker ID as the target label. We thus get labeled training data for a k-class classification problem for each of the 12 languages (the value of k is different for different languages). We, similarly create

	bn	gu	hi	kn	ml	mr	or	pa	sa	ta	te	ur
Train												
#M	10	34	48	43	7	72	4	55	85	106	43	26
#F	18	25	53	16	10	51	22	67	100	32	41	21
Mh	24	28	42	56	46	67	49	20	31	80	52	22
Fh	79	88	95	97	89	105	49	104	72	92	90	52
Validation												
#M	9	10	10	10	6	10	4	10	10	10	10	10
#F	10	10	10	10	10	10	10	10	10	10	10	10
Mh	2.5	2.5	2.5	2.5	2.5	2.5	2.5	2.5	2.5	2.5	2.5	2.5
Fh	2.5	2.5	2.5	2.5	2.5	2.5	2.5	2.5	2.5	2.5	2.5	2.5
Test-Known												
#M	10	10	10	10	7	10	4	10	10	10	10	10
#F	10	10	10	10	10	10	10	10	10	10	10	10
Mh	2.5	2.5	2.5	2.5	2.5	2.5	2.5	2.5	2.5	2.5	2.5	2.5
Fh	2.5	2.5	2.5	2.5	2.5	2.5	2.5	2.5	2.5	2.5	2.5	2.5
Test-Unknown												
#M	8	10	10	10	5	10	6	10	10	10	10	10
#F	10	10	10	10	10	10	10	10	10	10	10	10
Mh	1.7	1.5	1.5	1.5	1.5	1.6	1.5	1.5	1.5	1.5	1.4	1.5
Fh	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.5	1.7	1.7	1.6
Noisy-Test-Known												
#M	10	10	10	10	6	10	4	10	10	10	10	10
#F	10	10	10	10	10	10	10	10	10	10	10	10
Mh	2.5	2.5	2.5	2.5	1.7	2.5	2.5	1.3	2.5	2.5	2.5	2.5
Fh	2.5	2.5	2.5	2.5	2.5	2.5	2.5	2.5	2.5	2.5	2.5	2.5
Noisy-Test-Unknown												
#M	8	10	10	10	5	10	6	10	10	10	10	10
#F	10	10	10	10	10	10	10	10	10	10	10	10
Mh	0.8	1.6	0.4	0.9	1.0	1.1	0.7	0.4	0.8	0.7	0.9	0.4
Fh	1.4	1.6	0.9	3.0	0.3	1.6	0.5	0.2	0.6	2.0	1.1	0.9

Table 3: Language wise distribution of speakers and audio duration across different splits.(#M/F= No. of male/female speakers, (M/F)h = Male/female duration in hours)

the (noisy-)test-known and (noisy-)validation data for SID-mono from the corresponding splits for ASR. Note that since this is a k-class classification problem, there cannot be unknown speakers/classes at test time.

Next, we consider a multilingual setup wherein we pooled the audio recordings from all the speakers from all the languages in the ASR training data described earlier and assigned the known speaker ID as the target label. We thus get labeled training data for a m-class classification problem (where m is the sum of the number of speakers across all languages). We refer to this as SID-multi. We similarly create the multiple test and validation splits for SID-multi from the corresponding ASR splits.

Automatic Speaker Verification (ASV) In ASV, the input contains a pair of utterances and the task is to identify if the two utterances belong to the same speaker or not (the identity of the speaker does not matter). For this, we follow Yang et al. (2021b) and train a speaker identification model using the SID training data described above. We then compute speaker embeddings using this model and compute the cosine similarity between the embeddings of two utterances to decide if they belong to the same speaker. To create the test and validations sets for this binary classification task we make 50,000 pairs of utterances from the corresponding split of the ASR data such that 50% of them belong to the same speaker and 50% belong to different speakers.

Query by Example (QbE) In QbE, the task is to fetch all the audio clips which contain the spoken query word. Unlike text based retrieval, here the query is also an audio file and the collection also contains audio files. The ASR read speech data that we had collected contained only sentences and no queries. Hence, for this task, we did a separate data collection where we first curated 50 popular entity names in each language (names of states, cities, celebrities, *etc.*). For every query, we mined 20 sentences from publicly available sources, such as, IndicCorp and Wikipedia resulting in a total of approximately 1K sentences for each language (please see supplementary material for the exact number of utterances). Once we had the queries and the corresponding sentence containing these query words, we set up volunteers to read out the queries and the sentences. For every language, we involved a total of 20 volunteers, 10 for speaking out the queries and 10 for speaking out the sentences. There was an equal proportion of male and female speakers. Around 40% of the speakers were from the 20-30 year age groups and 20% each from the 15-20, 30-45 and 45+ age groups. Note that since this is a retrieval task there is no training involved.

Keyword Spotting (KS) KS is the task of classifying an utterance into a predefined set of keywords. The keywords are usually commands such as up, down, open, close, *etc*. We use the command and control words from the LDCIL dataset for all the 12 languages except Sanskrit. We took top 50 frequent words in every language from the command and control category. These keywords were spoken by multiple speakers and we used these utterances to create the train, valid, test splits. Please see supplementary material for the the number of utterances in train, valid and test splits.

Experimental Setup

We now describe the experimental setup that we used.

ASR: Recent work on Indian languages has shown that wav2vec2 based models perform well on a wide variety of ASR benchmarks (Javed et al. 2022). Following this, we evaluate two wav2vec2 based models, *viz.*, IndicWav2Vec and XLS-R (Babu et al. 2021). The former is pre-trained using 17K hours of raw audio data across 40 Indian languages while the latter is trained on half a million hours of raw audio data from 128 languages including 11 of our 12 Indian languages. We take the publicly available pretrained models and fine-tune them using the training split from Kathbath to build a separate acoustic model for each of the 12 languages. We restricted the output vocabulary of the model to characters only. In addition to the acoustic model, we also trained a 6-gram KenLM language model for each language using all the sentences from IndicCorp (ranging from 8M to 87M for different languages). During decoding, we combine the scores of the acoustic model and the language model. We use a beam size of 128 and set the LM weight and word score

	msr			mucs						oslr	
	gu	**ta**	**te**	**gu**	**hi**	**mr**	**or**	**ta**	**te**	**bn**	**avg**
All	14.3	**17.8**	14.1	17.8	12.9	14.1	**23.4**	19.5	16.1	11.7	16.2
+KB	**11.2**	18.6	**13.0**	**12.2**	**10.1**	**13.0**	24.3	**19.1**	**13.4**	**11.5**	**14.7**

Table 4: Comparisons of models trained using all existing data (All) and existing data+Kathbath on 3 benchmarks.

parameter to 2 and -1 respectively. For all our experiments we use WER (word error rate) to measure the performance.

Other Tasks: For the remaining tasks, we compare the features extracted from log mel filterbanks (FBANK), with the representations from SSL models, *viz.*, IndicWav2Vec and XLS-R. We use the s3prl(Yang et al. 2021b) framework which allows us to evaluate SSL models by extracting representation from different layers of the pretrained model. For the three classification tasks, *viz.*, **LID, SID, KS**, we take the extracted representations, mean pool them and train a linear classifier on top of them using the cross-entropy loss function. For these 3 tasks, we use accuracy as the metric.

For **ASV**, we use the same setup as Yang et al. (2021b) and train the X-Vector (Snyder et al. 2018) on the extracted representations. We use the cosine similarity between the x-vectors to measure the similarity between a pair of utterances. We use equal error rate (EER) as the evaluation metric. EER is the location on a ROC curve where the false acceptance and false rejection rates are equal.

For **QbE**, we run DTW (Giorgino 2009) to compute the similarity score for each query document pair using the exponential cosine distance function as suggested in (Yang et al. 2021b). We explore representations from all the layers of the pretrained models on the validation set. We finally report results on the test set by taking the best layer based on the validation set. Following standard practice, we use maximum term weighted value (MTWV) as the evaluation metric which balances misses and false alarms.

Results and Discussions

Based on the results of our experiments as summarised in Tables 4 to 8, we try to answer the following questions:

Does Kathbath help in improving ASR for Indian languages? To assess the usefulness of Kathbath, we compare the performance of two IndicWav2Vec models. The first model is trained on all existing data from publicly available benchmarks, *i.e.*, MUCS (Diwan et al. 2021), MSR (Srivastava et al. 2018) and a subset of OpenSLR (Kjartansson et al. 2018b) obtained from Shetty and Umesh (2021). This training data contained 40 hours each for Gujarati, Tamil and Telugu and 95 hours each for Hindi, Marathi and Odia and 70 hours from OpenSLR. The second model is trained on Kathbath in addition to the above existing data. We use the same 6-gram KenLM language model to decode the emissions for both the acoustic models. As is evident from the results in Table 4, adding Kathbath reduces the WER for most of the languages with an average improvement of 1.5%. Note that this is despite the fact that the data distribution in Kathbath is very different from the distribution of the data in these benchmarks. For example, while Kathbath contains formally written sentences from the News domain, the MUCS/MSR benchmarks contain code-mixed sentences.

	Test Clean		**Test Noisy**	
Model	**Known**	**Unknown**	**Known**	**Unknown**
Language Identification (LID) - Accuracy				
FBANK	27	14.10	26.15	13.37
IndicWav2Vec	**98.24**	**90.78**	**97.7**	**87.29**
XLS-R	94.38	79.96	92.97	74.50
Speaker Identification Monolingual (SID-Mono) - Accuracy				
FBANK	77.2	-	75.5	-
IndicWav2Vec	**95.6**	-	**95.2**	-
XLS-R	94.2	-	92.4	-
Speaker Identification Multilingual (SID-Multi) - Accuracy				
FBANK	36.79	-	36.23	-
IndicWav2Vec	**79.26**	-	**78.08**	-
XLS-R	70.71	-	69.22	-
Automatic Speaker Verification - EER				
FBANK	2.95	**11.24**	4.02	**11.24**
IndicWav2Vec	**2.08**	15.33	**2.11**	15.39
XLS-R	2.15	12.05	2.83	11.58

Table 5: Comparison of different feature representations for the tasks of LID, SID and ASV (lower EER is better).

	FBANK	IndicWav2Vec	XLS-R
KS (Accuracy)	21.49	96.89	**97.14**
QbE (MTWV)	0.001	**0.022**	0.012

Table 6: Comparison of different feature representations for the tasks of KS (accuracy) and QbE (MTWV) [Average scores across all languages.]

How effective are SSL models as feature encoders for Indian languages? We refer to Table 5 for a comparison of different feature representations for the 3 classification tasks, *viz.*, LID, SID and ASV. Due to lack of space, we only report the average performance across languages. Please refer to the supplementary material for detailed results per language. We observe that across all the evaluation conditions, the features extracted from IndicWav2Vec and XLS-R perform better than FBANK features. The only exception to this rule is the performance on the Test-Unknown and Noisy-Test-Unknown splits for the ASV task. Similarly, referring to Table 6 for QbE and KS, we observe that FBANK based features are easily outperformed by the features extracted from the two SSL models. We thus conclude that, similar to English, features extracted from SSL models are very effective for speech understanding tasks for Indian languages.

Are language-family specific SSL models better than those pre-trained on a larger set of languages? We now compare the features extracted from IndicWav2Vec which is pretrained using data for 40 Indian languages with XLS-R which is pretrained on a larger set of 128 languages. Referring to Tables 5 and 6, we observe that for LID, SID and

	bn	gu	hi	kn	ml	mr	or	pa	sa	ta	te	ur	avg
Clean Splits													
K	9.6	5.7	7.4	8.7	23.0	8.4	15.4	5.3	29.9	13.0	9.9	12.6	12.4
U	10.1	7.0	8.3	8.9	24.5	8.6	16.4	6.3	30.0	12.5	11.7	13.0	13.1
Noisy Splits													
K	11.1	7.2	9.3	9.1	24.0	9.3	17.5	6.7	34.5	16.1	12.5	14.4	14.3
U	12.8	10.1	9.2	13.6	25.9	10.1	18.2	10.5	34.4	15.0	16.4	17.7	16.2

Table 7: WER on different splits of Kathbath for ASR task. (K = Known splits, U = Unknown splits)

	gujarati	hindi	kannada
Male	8.44	10.89	8.4
Female	5.62	5.8	9.31

Table 8: Evaluation of gender bias in the performance (WER) of ASR models due to imbalanced training sets.

QbE, features extracted from IndicWav2Vec perform better than those from XLS-R. For ASV, XLS-R performs better than IndicWav2Vec on the splits containing unknown speakers. This is reasonable as XLS-R is trained on a much larger and diverse set of speakers which may help it generalise better to unknown speakers. Lastly, for the task of QbE there is only a small difference between the performance of the features extracted from the two models.

How does the performance generalise to unknown speakers? Once again referring to Table 5, we see that the performance for LID and ASV drops when the test splits contain unknown speakers. The drop in performance is significant for ASV task for all the models, with features from the two SSL models performing worse than FBANK. Also, as mentioned earlier, the XLS-R model which is trained on a much larger set of speakers performs better than the IndicWav2Vec model for the ASV task. Similarly, Table 7 shows the performance of IndicWav2Vec fine-tuned on Kathbath for the task of ASR. Once again, we see a drop in performance on the test split containing unknown speakers as compared to the split containing known speakers. While the gap is smaller for ASR, these results do suggest that while creating SLU benchmarks it is important to have test splits having novel speakers which are not seen during training.

How robust are existing models to noise in the utterances? Referring to Tables 5 and 7, we see a slight drop in the performance for all the tasks when evaluating on the noisy splits as compared to the corresponding clean splits. As expected, the gap between the most favourable condition (*i.e.*, clean splits with known speakers) and the most unfavourable condition (*i.e.*, noisy splits with unknown speakers) is quite high for most tasks. We thus believe that these multiple test conditions in IndicSUPERB will allow a more robust evaluation of speech models for Indian languages.

Are existing models robust to gender bias in the training data? As is evident from Table 3, the test and validation sets in IndicSUPERB are well balanced, *i.e.*, they have equal number of male and female speakers as well as equal duration from male and female speakers. However, while collecting data at scale it is hard to maintain this balance across languages. For example, our conscious choice to collect more data for female speakers was aided by the fact that we were able to find more female speakers easily despite the incentives being similar for male and female speakers. Similarly, for some languages we were able to find fewer participants who did bulk of the work as most participants were not willing to work for shorter duration. As a result, our training data has imbalances wherein for most languages we have more female speakers and/or more number of hours for female speakers. Using our balanced test sets, we can evaluate the effect of this imbalance in the training data. To do so, we consider the task of ASR and focus on 3 languages: (i) Gujarati which has fewer female speakers than male speakers (25 v/s 34) but the total duration for female speakers is 3 times more than that for male speakers (88 v/s 28) (ii) Hindi which has almost equal number of male and female speakers (48 v/s 53) but the total duration for female speakers is more than twice that for male speakers (95 v/s 42) and (iii) Kannada which has much fewer female speakers than male speakers (16 v/s 43) but the total duration for female speakers is almost twice that for male speakers (97 v/s 56). In Table 8 we separately present the results for the male and female speakers in the *test-unknown* set. Please refer to the supplementary material for gender specific results for the remaining languages. We observe that for Hindi and Gujarati where the number of female speakers as well as duration of female speakers is higher the performance is clearly better for female speakers. For Kannada, even though the duration is higher for females, the number of female speakers is much smaller and as a result there is a slight dip in the performance for female speakers as compared to male speakers. These results show that ASR models are indeed sensitive to bias in training data sizes across gender, especially as measured by diversity of speakers. Any data collection exercise will always have such biases due to easier availability of data from certain sections of the population and hence it is important to create balanced evaluation sets, as in IndicSUPERB, to evaluate the bias in models trained on such datasets.

Conclusion

In this work, we first present Kathbath, a large scale ASR dataset for 12 Indian languages. Using the meta information in Kathbath, we create a robust benchmark called IndicSUPERB for ASR, LID, SID and ASV containing different test conditions with known speakers, unknown speakers and noisy utterances. We also create a benchmark for query by example (a retrieval task) and keyword spotting. Through our experiments, we first show that the ASR training data in Kathbath helps in improving the performance on existing ASR benchmarks across languages. Next, we use IndicSUPERB, to evaluate the efficacy of existing SSL models and show that they serve as good feature encoders for a variety of SSL tasks. Lastly, we show that the different test conditions in IndicSUPERB are useful for evaluating (i) the capability of existing models to generalise to unknown speakers (ii) the robustness of existing models to noise and (iii) the robustness of existing models to bias in training data.

Acknowledgements

We would like to thank the Ministry of Electronics and Information Technology (MeitY[6]) of the Government of India and the Centre for Development of Advanced Computing (C-DAC[7]), Pune for generously supporting this work and providing us access to multiple GPU nodes on the Param Siddhi Supercomputer. We would like to thank the EkStep Foundation and Nilekani Philanthropies for their generous grant which went into hiring human resources as well as cloud resources needed for this work. We would like to thank DesiCrew for connecting us to native speakers for collecting data. We would like to thank Vivek Seshadri from Karya Inc. for helping setup the data collection infrastructure on the Karya platform. We would like to thank all the members of AI4Bharat team in helping create the Query by Example dataset.

References

Abraham, B.; Goel, D.; Siddarth, D.; Bali, K.; Chopra, M.; Choudhury, M.; Joshi, P.; Jyoti, P.; Sitaram, S.; and Seshadri, V. 2020a. Crowdsourcing Speech Data for Low-Resource Languages from Low-Income Workers. In *Proceedings of the 12th Language Resources and Evaluation Conference*, 2819–2826. Marseille, France: European Language Resources Association. ISBN 979-10-95546-34-4.

Abraham, B.; Goel, D.; Siddarth, D.; Bali, K.; Chopra, M.; Choudhury, M.; Joshi, P.; Jyoti, P.; Sitaram, S.; and Seshadri, V. 2020b. Crowdsourcing Speech Data for Low-Resource Languages from Low-Income Workers. In *Proceedings of the 12th Language Resources and Evaluation Conference*, 2819–2826. Marseille, France: European Language Resources Association. ISBN 979-10-95546-34-4.

Adiga, D.; Kumar, R.; Krishna, A.; Jyothi, P.; Ramakrishnan, G.; and Goyal, P. 2021. Automatic Speech Recognition in Sanskrit: A New Speech Corpus and Modelling Insights. In Zong, C.; Xia, F.; Li, W.; and Navigli, R., eds., *Findings of the Association for Computational Linguistics: ACL/IJCNLP 2021, Online Event, August 1-6, 2021*, volume ACL/IJCNLP 2021 of *Findings of ACL*, 5039–5050. Association for Computational Linguistics.

Ardila, R.; Branson, M.; Davis, K.; Henretty, M.; Kohler, M.; Meyer, J.; Morais, R.; Saunders, L.; Tyers, F. M.; and Weber, G. 2020. Common Voice: A Massively-Multilingual Speech Corpus. In *Proceedings of the 12th Conference on Language Resources and Evaluation (LREC 2020)*, 4211–4215.

Ayyavu, M.; Pilar, B.; and G, R. A. 2022. Subword Dictionary Learning and Segmentation Techniques for Automatic Speech Recognition in Tamil and Kannada. *CoRR*, abs/2207.13331.

Babu, A.; Wang, C.; Tjandra, A.; Lakhotia, K.; Xu, Q.; Goyal, N.; Singh, K.; von Platen, P.; Saraf, Y.; Pino, J.; et al. 2021. XLS-R: Self-supervised cross-lingual speech representation learning at scale. *arXiv preprint arXiv:2111.09296*.

[6]https://www.meity.gov.in/

[7]https://www.cdac.in/index.aspx?id=pune

Bhanushali, A.; Bridgman, G.; G, D.; Ghosh, P.; Kumar, P.; Kumar, S.; Raj Kolladath, A.; Ravi, N.; Seth, A.; Seth, A.; Singh, A.; Sukhadia, V.; S, U.; Udupa, S.; and Prasad, L. V. S. V. D. 2022. Gram Vaani ASR Challenge on spontaneous telephone speech recordings in regional variations of Hindi. In *Proc. Interspeech 2022*, 3548–3552.

Chung, J. S.; Nagrani, A.; and Zisserman, A. 2018. Voxceleb2: Deep speaker recognition. *arXiv preprint arXiv:1806.05622*.

Devlin, J.; Chang, M.; Lee, K.; and Toutanova, K. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Burstein, J.; Doran, C.; and Solorio, T., eds., *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, 4171–4186. Association for Computational Linguistics.

Diwan, A.; Vaideeswaran, R.; Shah, S.; Singh, A.; Raghavan, S.; Khare, S.; Unni, V.; Vyas, S.; Rajpuria, A.; Yarra, C.; Mittal, A.; Ghosh, P. K.; Jyothi, P.; Bali, K.; Seshadri, V.; Sitaram, S.; Bharadwaj, S.; Nanavati, J.; Nanavati, R.; Sankaranarayanan, K.; Seeram, T.; and Abraham, B. 2021. Multilingual and code-switching ASR challenges for low resource Indian languages. *Proceedings of Interspeech*.

Ganesh, M. S.; Vegesna, V. V. R.; Naroju, M. D.; Maity, S.; Yalla, P.; and Vuppala, A. K. 2021. CSTD-Telugu Corpus: Crowd-Sourced Approach for Large-Scale Speech data collection. *2021 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, 511–517.

Giorgino, T. 2009. Computing and visualizing dynamic time warping alignments in R: the dtw package. *Journal of statistical Software*, 31: 1–24.

He, F.; Chu, S.-H. C.; Kjartansson, O.; Rivera, C.; Katanova, A.; Gutkin, A.; Demirsahin, I.; Johny, C.; Jansche, M.; Sarin, S.; and Pipatsrisawat, K. 2020. Open-source Multi-speaker Speech Corpora for Building Gujarati, Kannada, Malayalam, Marathi, Tamil and Telugu Speech Synthesis Systems. In *Proceedings of the 12th Language Resources and Evaluation Conference*, 6494–6503. Marseille, France: European Language Resources Association. ISBN 979-10-95546-34-4.

Hu, J.; Ruder, S.; Siddhant, A.; Neubig, G.; Firat, O.; and Johnson, M. 2020. XTREME: A Massively Multilingual Multi-task Benchmark for Evaluating Cross-lingual Generalization. *CoRR*, abs/2003.11080.

Javed, T.; Doddapaneni, S.; Raman, A.; Bhogale, K. S.; Ramesh, G.; Kunchukuttan, A.; Kumar, P.; and Khapra, M. M. 2022. Towards building asr systems for the next billion users. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, 10813–10821.

Joshi, P.; Santy, S.; Budhiraja, A.; Bali, K.; and Choudhury, M. 2020. The State and Fate of Linguistic Diversity and Inclusion in the NLP World. In *Proceedings of the 58th Annual Meeting of the Association for Computational Lin-*

guistics, 6282–6293. Online: Association for Computational Linguistics.

Kakwani, D.; Kunchukuttan, A.; Golla, S.; N.C., G.; Bhattacharyya, A.; Khapra, M. M.; and Kumar, P. 2020. IndicNLPSuite: Monolingual Corpora, Evaluation Benchmarks and Pre-trained Multilingual Language Models for Indian Languages. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, 4948–4961. Online: Association for Computational Linguistics.

Kjartansson, O.; Sarin, S.; Pipatsrisawat, K.; Jansche, M.; and Ha, L. 2018a. Crowd-Sourced Speech Corpora for Javanese, Sundanese, Sinhala, Nepali, and Bangladeshi Bengali. In *SLTU*.

Kjartansson, O.; Sarin, S.; Pipatsrisawat, K.; Jansche, M.; and Ha, L. 2018b. Crowd-Sourced Speech Corpora for Javanese, Sundanese, Sinhala, Nepali, and Bangladeshi Bengali. In *Proc. The 6th Intl. Workshop on Spoken Language Technologies for Under-Resourced Languages (SLTU)*, 52–55. Gurugram, India.

Nagrani, A.; Chung, J. S.; and Zisserman, A. 2017. Voxceleb: a large-scale speaker identification dataset. *arXiv preprint arXiv:1706.08612*.

Newell, A.; and Deng, J. 2020. How Useful Is Self-Supervised Pretraining for Visual Tasks? In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, 7343–7352. Computer Vision Foundation / IEEE.

NLLB Team; Costa-jussà, M. R.; Cross, J.; Çelebi, O.; Elbayad, M.; Heafield, K.; Heffernan, K.; Kalbassi, E.; Lam, J.; Licht, D.; Maillard, J.; Sun, A.; Wang, S.; Wenzek, G.; Youngblood, A.; Akula, B.; Barrault, L.; Gonzalez, G. M.; Hansanti, P.; Hoffman, J.; Jarrett, S.; Sadagopan, K. R.; Rowe, D.; Spruit, S.; Tran, C.; Andrews, P.; Ayan, N. F.; Bhosale, S.; Edunov, S.; Fan, A.; Gao, C.; Goswami, V.; Guzmán, F.; Koehn, P.; Mourachko, A.; Ropers, C.; Saleem, S.; Schwenk, H.; and Wang, J. 2022. No Language Left Behind: Scaling Human-Centered Machine Translation.

Panayotov, V.; Chen, G.; Povey, D.; and Khudanpur, S. 2015. Librispeech: an asr corpus based on public domain audio books. In *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, 5206–5210. IEEE.

Prahallad, K.; Kumar, E. N.; Keri, V.; Rajendran, S.; and Black, A. W. 2012. The IIIT-H Indic speech databases. In *Thirteenth annual conference of the international speech communication association*.

Shetty, V. M.; and Umesh, S. 2021. Exploring the use of Common Label Set to Improve Speech Recognition of Low Resource Indian Languages. In *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 7228–7232.

Snyder, D.; Garcia-Romero, D.; Sell, G.; Povey, D.; and Khudanpur, S. 2018. X-Vectors: Robust DNN Embeddings for Speaker Recognition. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 5329–5333. IEEE Press.

Srivastava, B. M. L.; Sitaram, S.; Kumar Mehta, R.; Doss Mohan, K.; Matani, P.; Satpal, S.; Bali, K.; Srikanth, R.; and Nayak, N. 2018. Interspeech 2018 Low Resource Automatic Speech Recognition Challenge for Indian Languages. In *Proc. 6th Workshop on Spoken Language Technologies for Under-Resourced Languages (SLTU 2018)*, 11–14.

Tsai, H.-S.; Chang, H.-J.; Huang, W.-C.; Huang, Z.; Lakhotia, K.; Yang, S.-w.; Dong, S.; Liu, A. T.; Lai, C.-I. J.; Shi, J.; et al. 2022. Superb-sg: Enhanced speech processing universal performance benchmark for semantic and generative capabilities. *arXiv preprint arXiv:2203.06849*.

Wang, A.; Singh, A.; Michael, J.; Hill, F.; Levy, O.; and Bowman, S. R. 2018. GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding. In Linzen, T.; Chrupala, G.; and Alishahi, A., eds., *Proceedings of the Workshop: Analyzing and Interpreting Neural Networks for NLP, BlackboxNLP@EMNLP 2018, Brussels, Belgium, November 1, 2018*, 353–355. Association for Computational Linguistics.

Yang, S.; Chi, P.; Chuang, Y.; Lai, C. J.; Lakhotia, K.; Lin, Y. Y.; Liu, A. T.; Shi, J.; Chang, X.; Lin, G.; Huang, T.; Tseng, W.; Lee, K.; Liu, D.; Huang, Z.; Dong, S.; Li, S.; Watanabe, S.; Mohamed, A.; and Lee, H. 2021a. SUPERB: Speech Processing Universal PERformance Benchmark. In Hermansky, H.; Cernocký, H.; Burget, L.; Lamel, L.; Scharenborg, O.; and Motlícek, P., eds., *Interspeech 2021, 22nd Annual Conference of the International Speech Communication Association, Brno, Czechia, 30 August - 3 September 2021*, 1194–1198. ISCA.

Yang, S.-w.; Chi, P.-H.; Chuang, Y.-S.; Lai, C.-I. J.; Lakhotia, K.; Lin, Y. Y.; Liu, A. T.; Shi, J.; Chang, X.; Lin, G.-T.; et al. 2021b. Superb: Speech processing universal performance benchmark. *arXiv preprint arXiv:2105.01051*.