Survey: Automatic Speech Recognition For Indian Languages

Vineet Bhat, Pushpak Bhattacharyya

Indian Institute of Technology Bombay, India
vineetbhat2104@gmail.com, pb@cse.iitb.ac.in

Abstract

This survey paper provides a comprehensive overview of Automatic Speech Recognition (ASR), an area of research that has garnered significant attention due to its numerous real-world applications. The paper begins with a thorough exploration of the basic concepts and underlying theory of ASR, demystifying the fundamental principles of the field. It then delves into various types of end-to-end speech recognition systems, discussing their respective strengths and weaknesses, with a particular focus on Connectionist Temporal Classification (CTC), RNN-Transducers, and transformer-based models. The paper further highlights the unique challenges associated with ASR for Indian languages, a particularly diverse and less-resourced linguistic area, which poses interesting obstacles and opportunities for researchers. Key challenges discussed include the immense linguistic diversity and scarcity of labeled training data. Our aim with this survey is to provide readers with a holistic understanding of ASR, its progress, and potential future directions, particularly within the context of Indian languages.

1 Introduction

Automatic Speech Recognition (ASR) involves using algorithms and methodologies to convert speech signals into text using computers. While the problem of ASR has been extensively studied since the early 1960s, recent advancements in deep learning and processing power have significantly boosted research and industry developments, particularly in the early 2000s (Sonix). Speech recognition technologies find widespread use in various fields such as education, software development, utilities, luxury, and military. Consequently, both corporations and governments worldwide have invested substantial resources to create high-quality speech recognition systems. As the world moves towards automating tasks with AI, achieving near-human accuracy in ASR for all languages has become the ultimate goal for researchers globally. Therefore, studying this problem has become a critically important research area.

1.1 Problem Statement

Automatic Speech Recognition (ASR) is the task of converting speech signal into written text. More formally, given a spoken utterance represented as a sequence of acoustic feature vectors, $X = x_1, x_2, \ldots, x_T$, the goal of ASR is to find the most likely sequence of words, $W = w_1, w_2, \ldots, w_N$, that corresponds to X. This can be mathematically represented using the Bayes' theorem as follows:

$$\hat{W} = \arg\max_{W} P(W|X) \tag{1}$$

$$\hat{W} = \arg\max_{W} \frac{P(X|W)P(W)}{P(X)} \tag{2}$$

Here, $P(W|X)$ is the posterior probability, $P(X|W)$ is the likelihood, $P(W)$ is the prior probability, and $P(X)$ is the evidence. The most challenging part of ASR is accurately estimating the likelihood $P(X|W)$ which usually involves complex statistical modeling and machine learning techniques.

Word Error Rate (WER) is a commonly used to evaluate ASR systems. The WER is derived from the Levenshtein distance, working at the word level instead of the phoneme level. It can then be computed as:

$$WER = \frac{S + D + I}{N} = \frac{S + D + I}{S + D + C} \tag{3}$$

where S is the number of substitutions, D is the number of deletions, I is the number of insertions, C is the number of correct words, N is the number of words in the reference.

1.2 Motivation

- ASR research has gained momentum over the last few decades with large amounts of background papers and implementations providing

an excellent foundation for production and development work.

- Extensive data collection has been performed in widely spoken languages and this reduces the time and efforts needed to collect and annotate data before training machine learning models.
- Although such large amounts of research and data is available in the widely spoken languages, close to human accuracy has still not been achieved and as a result even the popular products such as Alexa and Siri have not been able to mimic human-like transcription.
- With a huge amount of resources spent in developing state-of-the-art research in widely spoken languages, ASR is yet to flourish in the regional languages across the world. With data being one of the biggest hurdles in solving this problem, innovative solutions such as Transfer Learning and Knowledge Sharing are yet to be explored to the full extent to deal with these problems and develop good systems in regional languages.

2 Background

Humans have always had extensive interest in speech technologies specifically for recognition. Speech recognition is often the first task in NLP systems and is a key module for downstream applications. For eg, a good speech recognition system paves way to a better machine translation system in Speech Translation. However, speech recognition did not receive much progress till the first half of the 20^{th} century. Starting from 1950s, corporations across the world spent a lot of money investing in recognition technologies, paving the way for high quality research and production.

The first speech recognizer was not tasked with identifying words but rather numbers[1]. In 1952, Bell labs created *Audrey* the first computer based digit recognizer capable of identifying single digits. Ten years later, Bell labs created *Shoebox* on a similiar framework but with the capability of identifying 16 isolated words in English. However, such efforts did not compensate for the large amount of resources being spent on the problem. In 1969, John Pierce, the director of Bell Technologies published *A Piercing Freeze* - a paper describing how the problem of ASR is a very difficult one and the amount of funding spent on this research will never yield successful results.

However, in 1970s, Carnegie Mellon University presented the *Harpy* speech recognition model capable of identifying 1000 words in English which is equivalent to a 3 year old's vocabulary (Lowerre and Reddy, 1976). In 1980s, one of the most revolutionary papers in speech technologies (Rabiner, 1989) was published which paved the way for including Hidden Markov Models in Automatic Speech Recognition. Using this technology, Dragon Inc designed "Dictate-MCA" a speech recognizer capable of typing documents through speaking[2]. On the onset of the 21st century, HMM based models had already taken over speech recognition with various production companies using custom datasets and domain specific knowledge to construct systems with close to 80% accuracies. In 2010s, deep learning gained momentum as advanced graphic cards were created and computing power of systems increased exponentially. The current path of ASR is towards exploring various deep learning frameworks to replace any statistical components of the traditional ASR pipeline in an attempt to completely get rid of manual engineering and domain specific performances.

Latest speech recognition systems rely heavily on deep learning based systems to work in an End-to-End manner. However, this was not the case before. Deep learning was initially introduced only to replace certain parts of the traditional ASR system. The use of deep feed-forward networks for acoustic modeling was introduced in 2009. For the next few years there were many research groups that developed various DL based systems but these only focused on replacing the acoustic modeling (Wang et al., 2019). In 2011, Microsoft Research Institute proposed a Hidden Markov model combined with context-based deep neural network called context-dependent (CD)-DNN-HMM (Dahl et al., 2012). Deep Speech (Hannun et al., 2014) was introduced in 2014 that revolutionized the ASR industry since this was the first known attempt to create an end-to-end speech recognition system. In 2017, Microsoft achieved the feat of creating the first ASR system capable of reaching human level accuracy for recognition (Xiong et al., 2018). Thereafter, majority of the research has been focused on improving the existing E2E systems as DL slowly overtook the

[1] https://letrario.pt/a-brief-history-of-asr-automatic-speech-recognition/

[2] https://en.wikipedia.org/wiki/DragonDictate

traditional HMM-GMM systems. In 2019, Facebook AI released their state-of-the-art wav2vec (Schneider et al., 2019) model for speech recognition followed by wav2vec 2.0 in 2020 (Baevski et al., 2020) which focuses on using self supervision and pre-training on raw audio before finetuning on supervised speech data. Using this approach, the authors were able to reduce the dependence on aligned corpora making it easier for researchers to use these models to create ASR models in any language with limited data. In 2020, Facebook also released the XLSR - wav2vec 2.0 model with capabilities of multilingual training to benefit development of speech recognition in low resource languages. Last year, OpenAI released the new Whisper[3] multilingual ASR and translation models capable of outputting high quality transcriptions in robust settings across domains.

2.1 ASR in Indian Languages

H.Tailor and B. Shah (2015) was the first to develop a speaker independent, continuous speech recognition system for Hindi. HMM tool kit was used by Choudhary et al. (2013) to develop a speaker independent isolated Hindi word speech recognizer for recognizing the ten digits in Hindi, using continuous HMM. Venkataramani et al.[4] worked on the development of an on-line *speech to text engine* for isolated word recognition on a vocabulary of 10 words (digits 0-9) which was implemented as a system on a programmable chip (SOPC). IBM research lab (Kumar et al., 2004), developed a large-vocabulary continuous speech recognition system for Hindi. They developed a Hindi Speech Recognition system which has been trained on 40 hours of audio data and has a trigram language model that is trained with 3 million words. For a vocabulary size of 65000 words, the system gives a word accuracy of 75% to 95%.

Gaurav et al. (2012) used Julius[5] to develop a domain specific speaker independent continuous speech recognizer for Hindi. They developed a Hindi isolated word speech recognizer using HTK on Linux platform which recognizes isolated words using acoustic word model. Vocabulary used for this system was just 30 words. HMM was used to train and recognize the speech. 39 MFCCs (12 Mel Cepstrum + Log energy + 1st and 2nd Order derivatives) features were extracted.

Aggarwal and Dave (2013) proposed an approach to speed up the statistical pattern classification by reducing the time consumed in the likelihood evaluation of feature vectors with the help of optimal number of Gaussian mixture components. They applied extended MFCC procedure by extracting 52 MFCC features (39 MFCC + 13 triple delta features) and then reducing them to 39 by using HLDA – Hetroscedastic linear discriminant analysis technique.

Ghai and Singh (2012) developed a connected Hindi digits recognition system using robust feature extraction techniques and HTK recognition engine[6]. In spite of creating just a baseline recognizer, the results were encouraging. Sivaraman and Samudravijaya (2011) made an attempt to compensate the mismatch between training and testing conditions with the help of unsupervised Speaker adaptation.

Indian languages speech recognition improved its accuracy with the advent of deep learning systems. The current state-of-the-art is set by the CLSRIL-23 system by Vakyansh (Gupta et al., 2022) which uses wav2vec 2.0's pretraining to self supervise thousands of hours of raw audio data in multiple Indian languages followed by finetuning for specific languages using aligned corpus to achieve extremely low WER of around 15%.

3 Challenges for Indian Languages ASR

One of the biggest challenges in Indian languages ASR is the lack of high quality and quantity of data. There have been some attempts to create a high quality corpus but with less investment, these do not match up with the standards of the other European languages. The problems can be categorized as follows -

- Limited data resources

  Majority of the Indian languages fall into the category of low resource languages. This means that they do not have enough speech data to generate high quality systems and often rely on transfer learning based approaches for developing good quality NLP systems.

- Multilingualism and Dialects

  India is a linguistically rich country. It is imperative to look at the linguistic diversity that

[3] https://cdn.openai.com/papers/whisper.pdf

[4] http://www.123seminarsonly.com/Seminar-Reports/013/48946526-speech-recognition.pdf

[5] https://github.com/julius-speech/julius

[6] https://htk.eng.cam.ac.uk/

resides in India. The number of living languages are *415*. The number of languages with more than *10 K* speakers is *121*. The number of languages with more than *100 K* speakers is *60*, while those with more than *1 M* speakers is *29*. Such wide variety of speakers and categorizations means that all the spoken languages have hundreds of regional dialects and hence the training corpus of any Indian language ASR system must take into account all this diversity in the samples.

4 End To End Speech Recognition

The breakthrough in ASR with the introduction of End-to-End (E2E) systems was revolutionary as creating excellent recognition systems became simpler with more emphasis on data rather than model engineering. There are various other advantages of E2E models over hybrid traditional models (Li, 2022).

- E2E systems optimize the entire network through a single objective function, as opposed to hybrid systems which optimize each component through a separate loss function. Optimizing through a single function guarantees global optimization of the entire pipeline
- E2E systems are less complicated than traditional systems and directly output characters/words. This makes it easier for engineers in various educational as well as business companies to create a SOTA speech recognition system with domain specific data
- Since E2E systems use a single network to map the input speech frames to output characters, they are more compact than traditional systems and can be deployed on servers easily

With all the above advantages, E2E systems have become the new SOTA for all ASR tasks. However, a majority of the commercially deployed systems still use traditional models because ASR accuracy is not the only metric for such services. A variety of practical factors such as latency, streaming, domain specific adaptation are also important which can be provided by engineered blocks of modules comprising of the traditional ASR pipeline. Such traditional models are optimized for production and hence are still used in many commercial devices.

With the development of large datasets for speech recognition, research began into constructing deep learning frameworks for Large Vocabulary Continuous Speech Recognition (LVCSR). Such deep learning models are End-to-End (E2E), ie, they try to map the input audio sequence directly to the corresponding character sequence, simultaneously learning the alignment and pronunciation modules. The functional structure of an E2E system is given in figure 1.

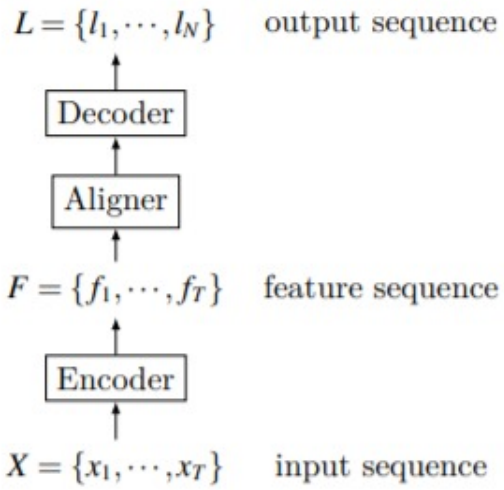

Figure 1: Structure of an end-to-end ASR model

The encoder maps the input audio sequence to a higher dimensional feature sequence which is often a fixed context vector. The aligner and the decoder tries to learn the alignments and the pronunciations of the phonemes together to output the character sequence *L*. Although the figure might look like a pipeline, the entire training and decoding process happens in an end to end fashion. The model takes input the raw audio files and outputs the characters of the hypothesis sentence. The E2E framework also solves the optimization problem as the model tries to reduce the loss function to attain global optimization of the entire pipeline.

4.1 Types of End-to-End ASR Models

Although E2E systems might seem easier to understand and train, they demand a large amount of annotated and aligned speech-text corpus. Additionally, the E2E systems still face the problem of label alignment to speech data. Depending on the various methods used to solve this alignment problem, the E2E systems are of 3 types: 1) CTC-Based Models, 2) RNN-Transducer and 3) Attention-Based models. These three methods are discussed in detail in this section.

4.1.1 Connectionist Temporal Classification

The Connectionist Temporal Classification (CTC) technique in ASR consists of mapping input speech frames to output text characters by introducing blank labels in output characters to compensate for the difference in length of speech frames and textual output. CTC, introduced in Graves et al. (2006), attempts to solve the data alignment problem as alignment between speech segments and audio is no longer needed.

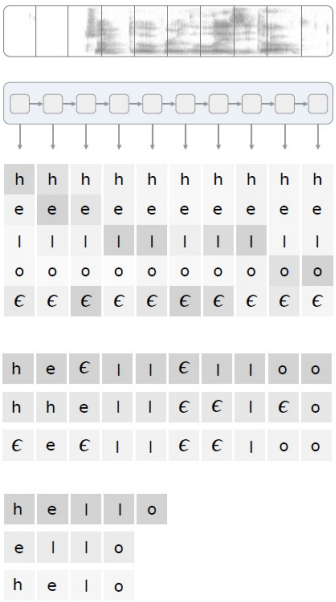

Figure 2: CTC in action; here ϵ is the blank symbol

An example of the working of a CTC based model is given in figure 2 Lets denote the input speech by **x** and output label sequence as **y**. The CTC loss function is defined as -

$$L_{CTC} = -ln(P(\mathbf{y}|\mathbf{x})) \tag{4}$$

where,

$$P(\mathbf{y}|\mathbf{x}) = \sum_{q \in B^{-1}(y)} P(q|x) \tag{5}$$

Here, $B^{-1}(y)$ is the set of all possible alignments of the output labels. With conditional independence, and T being the length of the speech sequence, this can be further split into each timestep as -

$$P(q|x) = \prod_{t=1}^{T} P(q_t|x) \tag{6}$$

The above conditional independence assumption is often criticized because it fails to capture the context dependent pronunciations which are a key part of every spoken language. This can be improved by using the attention mechanism which is the first step of language modelling across speech frames. After replacing the LSTM based networks, on which CTC was first developed, by Transformer based models, CTC has become a widely used technique with excellent accuracies (Higuchi et al., 2020). This is because the transformer based model provides a strong attention based mechanism whereas the CTC simplifies the decoding but also sums o ver all possible alignments. CTC also performs well on self supervision tasks (Chung and Glass, 2020), which has slowly become the hot topic of research in NLP. One such self supervision method will be discussed in the later sections.

4.1.2 RNN-Transducer Models

RNN Transducer (RNN-T) models provide an advanced learning framework where the previous character tokens and all the speech frames till the current time step are considered while decoding. Introduced in Graves (2012), RNN-Transducers remove the conditional independence assumption of the CTC models and provide a method for developing streaming ASR systems and have thus become a popular model for Industry (He et al., 2018).

As seen in figure 3, RNN-T consists of an encoder network, a prediction network and a joint network. The encoder network, as in earlier systems, generated a high level encoder representation h_t^{enc}. The prediction network produces another high level representation h_u^{pre} based on RNN-T's previous output label y_{u-1}.

The joint network, which is a feedforward network, combined both these features as -

$$z_{t,u} = \phi(Qh_t^{enc} + Vh_u^{pre} + b_z) \tag{7}$$

where ϕ is a non linear function (ReLU or TanH) and b_z is the bias vector. Q and V are weight matrices. $z_{t,u}$ is connected to the output layer as -

$$h_{t,u} = W_y z_{t,u} + b_y \tag{8}$$

where W_y is the weight matrix and b_y is the bias vector. Finally, we have the probability of output token k as -

$$P(y_u = k|x_{1:t}, y_{1:u-1}) = softmax(h_{t,u}^k) \tag{9}$$

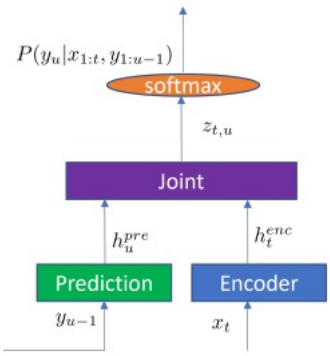

Figure 3: Simplified architecture of RNN Transducers

4.1.3 Attention Based Models

In attention based models (Vaswani et al., 2017), the encoder model maps the input audio signal to a sequence of vectors instead of one fixed vector. The decoder then assigns weights to these sequence of vectors while concatenating them to decode the higher dimensional features. In this way, the architecture models short range and long range dependencies. RNN based models always suffer from the problem of slow training due to dependencies on the previous time steps and faces problems in short and long term context understanding. Through the attention mechanism, at each time step, the previous and future time step features are taken into account while decoding the particular character and alignment.

An attention based model calculates the probability as -

$$P(y|x) = \prod_u P(y_u|x, y_{1:u-1}) \qquad (10)$$

where u is the output label index. The training objective is the same as that for CTC based models. Structure of an attention based ASR system is described in figure 4.

5 Modelling using E2E systems

In this section, we discuss two important neural network architectures that are popularly used to train End To End Speech Recognition systems.

5.1 wav2vec 2.0

In 2020, Facebook AI published the wav2vec 2.0 paper (Baevski et al., 2020) which revolutionized the performance of DL based ASR systems gradually becoming the gold standard for lowest WER across languages.

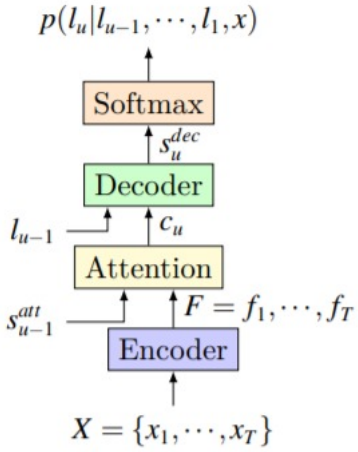

Figure 4: Structure of an attention based ASR model

wav2vec is a method to learn speech representations. The encoder network of wav2vec embeds raw audio into latent representations (f:X to Z) and a context vector combines multiple encoded representations into contextualised embeddings (g:Z to C). Self supervised learning is described in figure 5.

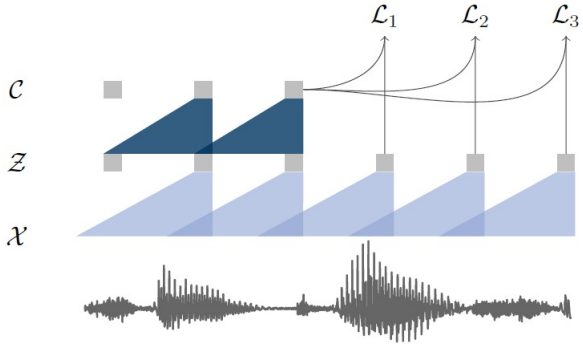

Figure 5: Self Supervised Speech Representation Learning

The paper discussed a method to pretrain a transformer based network on thousands of hours of unlabelled speech which learns context vectors. This follows the finetuning of the transformer model (which has a softmax layer on top of it) with minutes of transcribed speech resulting in excellent accuracy. This process is analogous to how humans learn to transcribe speech signals. Typically, babies need up to two years of listening to speech before they start speaking their first words. We can view this process in a similar manner, as babies undergo a pretraining phase where they are exposed to thousands of hours of speech data, enabling them to understand the contextual meaning

behind speech signals.

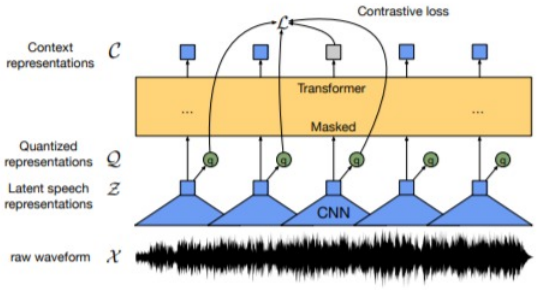

Figure 6: Joint pre-training of audio data in wav2vec 2.0

The contrastive loss is learnt by masking features at random time steps of the model. The task is to determine the missing features through the input audio as well as surrounding context features.

The wav2vec 2.0 paper was soon followed by Conneau et al. (2020) which paved the way for multilingual ASR systems. The authors of this paper pretrained the wav2vec 2.0 model on thousands of hours of unannotated speech in 50+ languages followed by finetuning this model for various low resource languages leading to extremely low WER systems establishing state-of-the-art results.

5.2 Whisper

OpenAI's Whisper model was introduced recently [7]. This model currently achieves the state-of-the-art performance in large scale speech recognition. The authors draw inspiration from the selective performance of pretraining models like wav2vec 2.0 and realize that the pretraining and finetuning strategy works well to create data centric ASR models but do not scale well to robust evaluations. In this paper, the authors use a transformer based encoder-decoder architecture and work with mel spectrogram inputs to predict the output character sequence. A major achievement of this paper is the collection of 680K hours of multilingual data from various sources such as YouTube videos and other open source websites. Such a vast and diverse corpus collection allows the powerful transformer model to learn complex features and achieve high level saturation in character prediction. The training data is compressed using byte pair encoding for English models and refit the vocabulary for multilingual speech recognition. The model further includes a multi task objective of speech recognition, speech translation and background sound detection.

The whisper model achieves a relative reduction in error rate across various domains of speech recognition datasets with an average reduction of 17% in WER compared to the wav2vec 2.0 baseline. Similar results are seen on multilingual speech recognition tasks.

[7] https://cdn.openai.com/papers/whisper.pdf

6 Summary

This paper encompasses a comprehensive exploration of the fundamental concepts and theories essential for conducting research in Automatic Speech Recognition (ASR). Our discussion begins by delving into the extensive background of ASR, spanning from the 1950s to the emergence of deep learning-based systems in the 21^{st} century. Furthermore, we delve into the mathematical foundations underlying significant models such as CTC and RNN-transducers. Moreover, we provide a detailed understanding of the challenges encountered in Indian languages, placing particular emphasis on the utilization of multilingual models to address the issue of limited data availability. Lastly, we examine and evaluate two cutting-edge and significant state-of-the-art systems, namely Wav2vec and Whisper, shedding light on their respective strengths and weaknesses.

References

R. K. Aggarwal and M. Dave. 2013. Performance evaluation of sequentially combined heterogeneous feature streams for hindi speech recognition system. *Telecommun. Syst.*, 52(3):1457–1466.

Alexei Baevski, Henry Zhou, Abdelrahman Mohamed, and Michael Auli. 2020. wav2vec 2.0: A Framework for Self-Supervised Learning of Speech Representations. *arXiv:2006.11477 [cs, eess]*. ArXiv: 2006.11477.

Annu Choudhary, Mr. R. S. Chauhan, and M. G. Gupta. 2013. Automatic speech recognition system for isolated & connected words of hindi language by using hidden markov model toolkit (htk).

Yu-An Chung and James Glass. 2020. Generative Pre-Training for Speech with Autoregressive Predictive Coding. *arXiv:1910.12607 [cs, eess]*. ArXiv: 1910.12607.

Alexis Conneau, Alexei Baevski, Ronan Collobert, Abdelrahman Mohamed, and Michael Auli. 2020. Unsupervised Cross-lingual Representation Learning for Speech Recognition. *arXiv:2006.13979 [cs, eess]*. ArXiv: 2006.13979.

G. E. Dahl, Dong Yu, Li Deng, and A. Acero. 2012. Context-Dependent Pre-Trained Deep Neural Networks for Large-Vocabulary Speech Recognition. *IEEE Transactions on Audio, Speech, and Language Processing*, 20(1):30–42.

Gaurav Gaurav, Devanesamoni Shakina Deiv, Gopal Krishna Sharma, and Mahua Bhattacharya. 2012. Development of Application Specific Continuous Speech Recognition System in Hindi. *Journal of Signal and Information Processing*, 03(03):394–401.

Wiqas Ghai and Navdeep Singh. 2012. Literature Review on Automatic Speech Recognition. *International Journal of Computer Applications*, 41(8):42–50.

Alex Graves. 2012. Sequence Transduction with Recurrent Neural Networks. *arXiv:1211.3711 [cs, stat]*. ArXiv: 1211.3711.

Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. 2006. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd international conference on Machine learning - ICML '06*, pages 369–376, Pittsburgh, Pennsylvania. ACM Press.

Anirudh Gupta, Rishabh Gaur, Ankur Dhuriya, Harveen Singh Chadha, Neeraj Chhimwal, Priyanshi Shah, and Vivek Raghavan. 2022. Effectiveness of text to speech pseudo labels for forced alignment and cross lingual pretrained models for low resource speech recognition. *arXiv:2203.16823 [cs, eess]*. ArXiv: 2203.16823.

Awni Hannun, Carl Case, Jared Casper, Bryan Catanzaro, Greg Diamos, Erich Elsen, Ryan Prenger, Sanjeev Satheesh, Shubho Sengupta, Adam Coates, and Andrew Y. Ng. 2014. Deep Speech: Scaling up end-to-end speech recognition. *arXiv:1412.5567 [cs]*. ArXiv: 1412.5567.

Yanzhang He, Tara N. Sainath, Rohit Prabhavalkar, Ian McGraw, Raziel Alvarez, Ding Zhao, David Rybach, Anjuli Kannan, Yonghui Wu, Ruoming Pang, Qiao Liang, Deepti Bhatia, Yuan Shangguan, Bo Li, Golan Pundak, Khe Chai Sim, Tom Bagby, Shuo-yiin Chang, Kanishka Rao, and Alexander Gruenstein. 2018. Streaming End-to-end Speech Recognition For Mobile Devices. *arXiv:1811.06621 [cs]*. ArXiv: 1811.06621.

Yosuke Higuchi, Shinji Watanabe, Nanxin Chen, Tetsuji Ogawa, and Tetsunori Kobayashi. 2020. Mask CTC: Non-Autoregressive End-to-End ASR with CTC and Mask Predict. In *Interspeech 2020*, pages 3655–3659. ISCA.

Jinal H.Tailor and Dipti B. Shah. 2015. Review on Speech Recognition System for Indian Languages. *International Journal of Computer Applications*, 119(2):15–18.

M. Kumar, N. Rajput, and A. Verma. 2004. A large-vocabulary continuous speech recognition system for Hindi. *IBM Journal of Research and Development*, 48(5.6):703–715.

Jinyu Li. 2022. Recent Advances in End-to-End Automatic Speech Recognition. *arXiv:2111.01690 [cs, eess]*. ArXiv: 2111.01690.

B. Lowerre and R. Reddy. 1976. The Harpy Speech Recognition System: performance with large vocabularies. *The Journal of the Acoustical Society of America*, 60(S1):S10–S11.

L.R. Rabiner. 1989. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286.

Steffen Schneider, Alexei Baevski, Ronan Collobert, and Michael Auli. 2019. wav2vec: Unsupervised Pre-Training for Speech Recognition. In *Interspeech 2019*, pages 3465–3469. ISCA.

Ganesh Sivaraman and K Samudravijaya. 2011. Hindi speech recognition and online speaker adaptation. *IJCA Proceedings on International Conference on Technology Systems and Management (ICTSM)*, (1):27–30. Full text available.

Sonix. A brief history of speech recognition.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention Is All You Need. *arXiv:1706.03762 [cs]*. ArXiv: 1706.03762.

Dong Wang, Xiaodong Wang, and Shaohe Lv. 2019. An Overview of End-to-End Automatic Speech Recognition. *Symmetry*, 11(8):1018.

W. Xiong, L. Wu, F. Alleva, J. Droppo, X. Huang, and A. Stolcke. 2018. The Microsoft 2017 Conversational Speech Recognition System. *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5934–5938. ArXiv: 1708.06073.