Modèles Multimodaux pour l'Interaction Orale

Laurent Besacier

3 Sept 2025

Résumé : Ce cours explorera les avancées récentes des modèles multimodaux pour l'interaction orale. Après une introduction au traitement automatique de la parole et aux architectures de reconnaissance automatique de la parole (ASR), il présentera les encodeurs autosupervisés ainsi que les modèles de langue multimodaux combinant texte et parole. Nous terminerons par un aperçu des modèles de dialogue oral de type speech2speech.
[Notebook]

Speech facts

- Speech generally conveys a (linguistic) message (that can be reduced to a transcript)
- But not only (paralinguistics: speaker identity, speaker mood, speaker health condition, speaker accent, etc.)
- Variability at all levels (intra speaker, inter speaker, microphone, phone line, room acoustics, style)
- Speech is a continuous signal (no explicit word boundaries)
- May be decomposed into elementary units of sound (phonemes) that distinguish one word from another in a particular language (minimal pairs)
  - *kill* vs *kiss* - *pat* vs *bat*
  - phoneme set is language dependent
  - acoustic realization of the phoneme is dependent of its left and right neighbors (co-articulation)

(Main) Speech tasks

- Speech compression (solved)
- Speaker recognition (strong progresses over the last 10 years but still poor compared to other biometric modalities like fingerprint and iris)
- Text-to-speech synthesis (strong progresses over the last 2-3 years, do you know NotebookLLM [1] ?)
- **Speech-to-text and Speech-to-Speech (this talk)**
- Speech paralinguistics: detection of gender, age, deception, sincerity, nativeness, emotion, sleepiness, cognitive disorders, (drug or alcohol) intoxication, pathologies, etc.
- Main speech conference: *Interspeech* (core A, every year)

[1] `https://notebooklm.google.com/`

Speech-to-text

- Automatic Speech Recognition (ASR)
- Ideally we want to have a system that deals with: spontaneous speech, multi-speakers, unlimited output vocabulary, any acoustic condition
- But performances differ greatly for different contexts (read vs spontaneous speech ; small vs large vocabulary ; quiet vs noisy; native vs non-native speech)

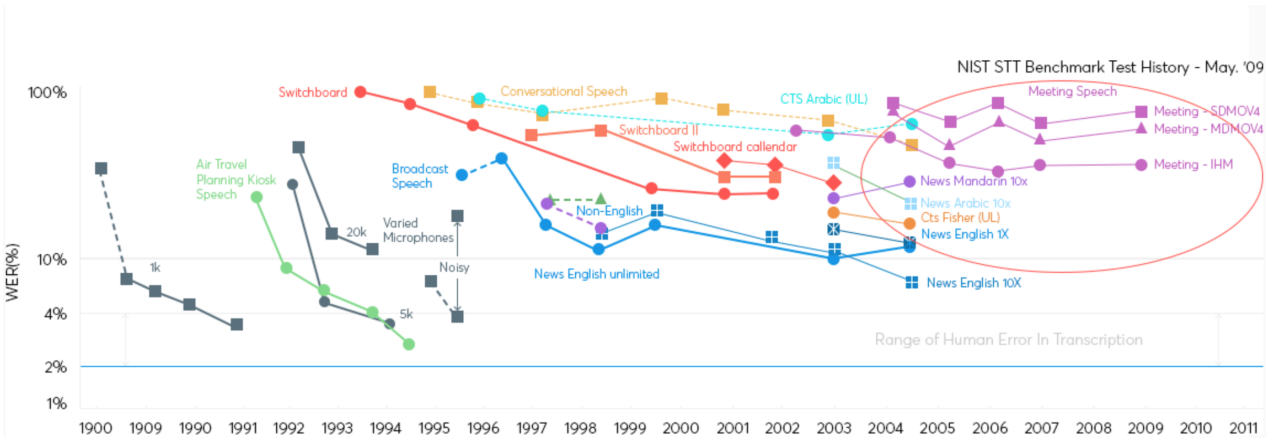


Figure: NIST ASR benchmark tests history ($<$ 2015)

Speech-to-text

- Automatic Speech Recognition (ASR)
- Ideally we want to have a system that deals with: spontaneous speech, multi-speakers, unlimited output vocabulary, any acoustic condition
- But performances differ greatly for different contexts (read vs spontaneous speech ; small vs large vocabulary ; quiet vs noisy; native vs non-native speech)

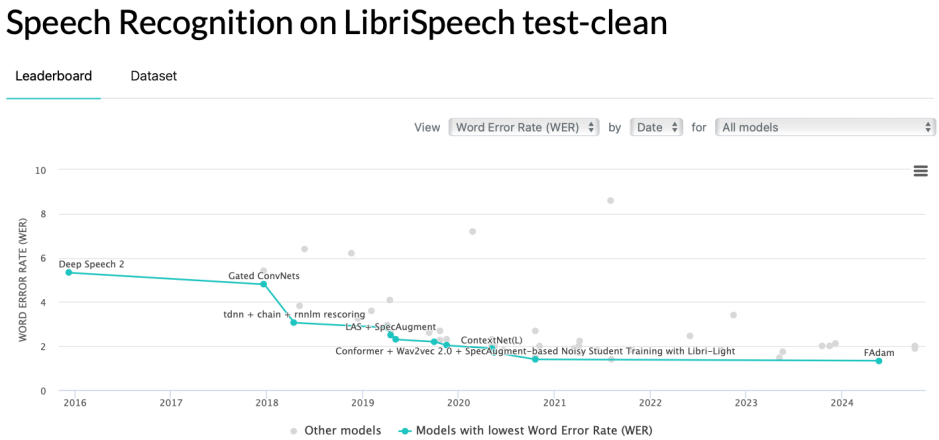


Figure: Librispeech ASR benchmark tests history ($>$ 2016)

ASR as a partial task in a larger system

- ASR for spoken language processing (speech understanding, speech translation, speech summarization, etc.)
- Not just a problem of noisy transcripts
- No sentence boundaries, punctuation, case
- Disfluencies in spontaneous speech: false starts, fillers, repaired utterances
  - btw, should we keep them or remove them ?
  - some speech tasks are ill defined (ex: speech translation)

$\Rightarrow$ End-to-end approaches from speech ?

Speech representations

- Handcrafted feature vectors
  - standard extraction on sliding windows of 20-30ms at a frame rate of 10ms
  - filterbanks (signal energy in different frequency bands)
  - cepstral coefficients (inverse Fourier transform of the logarithm of the estimated spectrum of a signal)
  - linear predictive coding (a sample is predicted as a weighted sum of preceding samples and weights are used as features)
  - prosodic features (pitch, energy)
- Raw waveform ($>$ 2015)
  - bypass handcrafted preprocessing
  - preprocessing become part of the acoustic modeling and training
  - introducing convolutional layers in the first stages of the NN pipeline

Speech representations

- Spectrograms ($<$ 1990 and $>$ 2015!)
  - time-frequency representation that is actually similar to sequence of filterbanks ...
  - ... but processed as an image

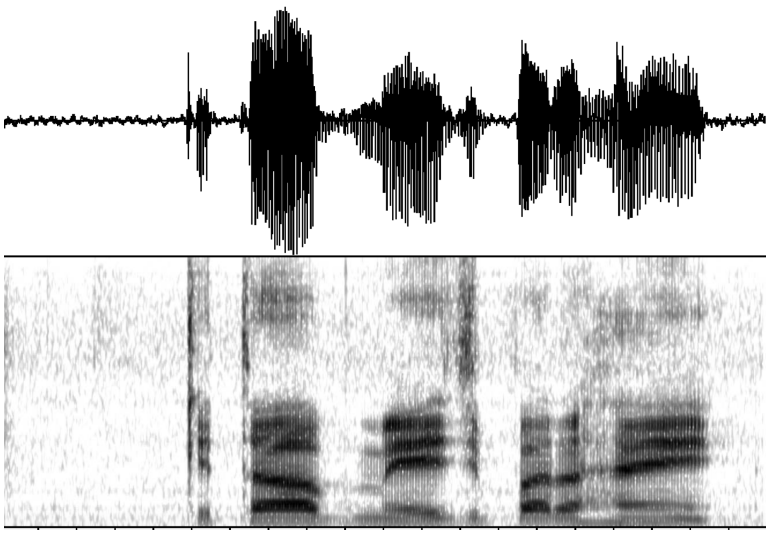


Figure: Speech signal (top) and spectrogram (bottom)

- Self-supervised learnt representations ($>$ 2020 we'll see that later today!)

Progresses over the years (truncated)

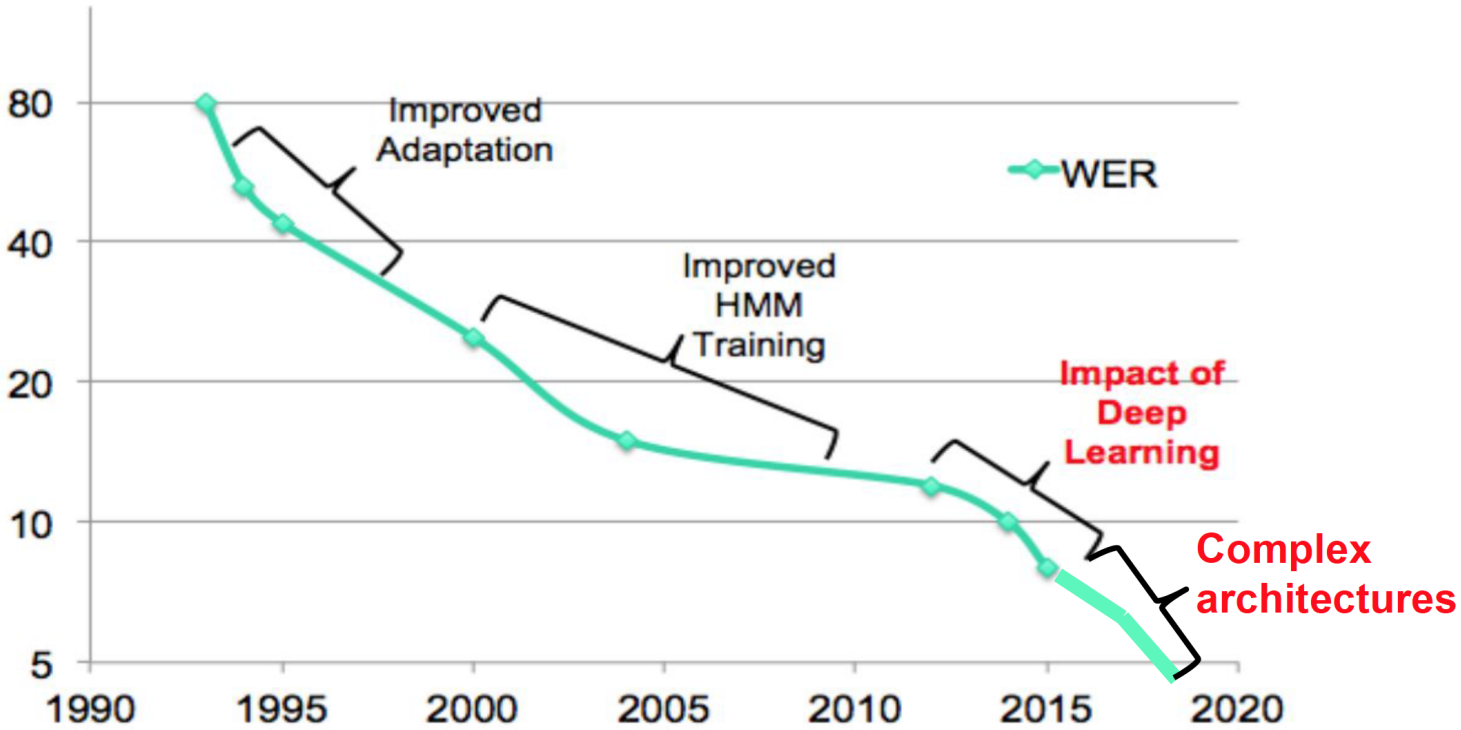


Figure: ASR Performance[2] on English Conversational Telephony (Switchboard)

[2] Image from Bhuvana Ramabhadran' s presentation at Interspeech 2018

Fundamental equation

x: observation (signal or features)
w: a word sequence

$$w^* = argmax_w p(w/x) = argmax_w p(x/w).p(w) \quad (1)$$

$p(x/w)$: acoustic model
$p(w)$: language model

Lexicons

- For acoustic modelling in large vocabulary speech recognition, we model phones instead of full words
- A pronunciation lexicon gives the decomposition of words into phonemes
- Adding a new word to the output vocabulary does not require retraining of the acoustic models
  - just add an entry to the pronunciation lexicon
  - *cat* /k a t/
- Hierarchical modelling of speech (signal/phones/words/utterance)

Hierarchical modelling of speech

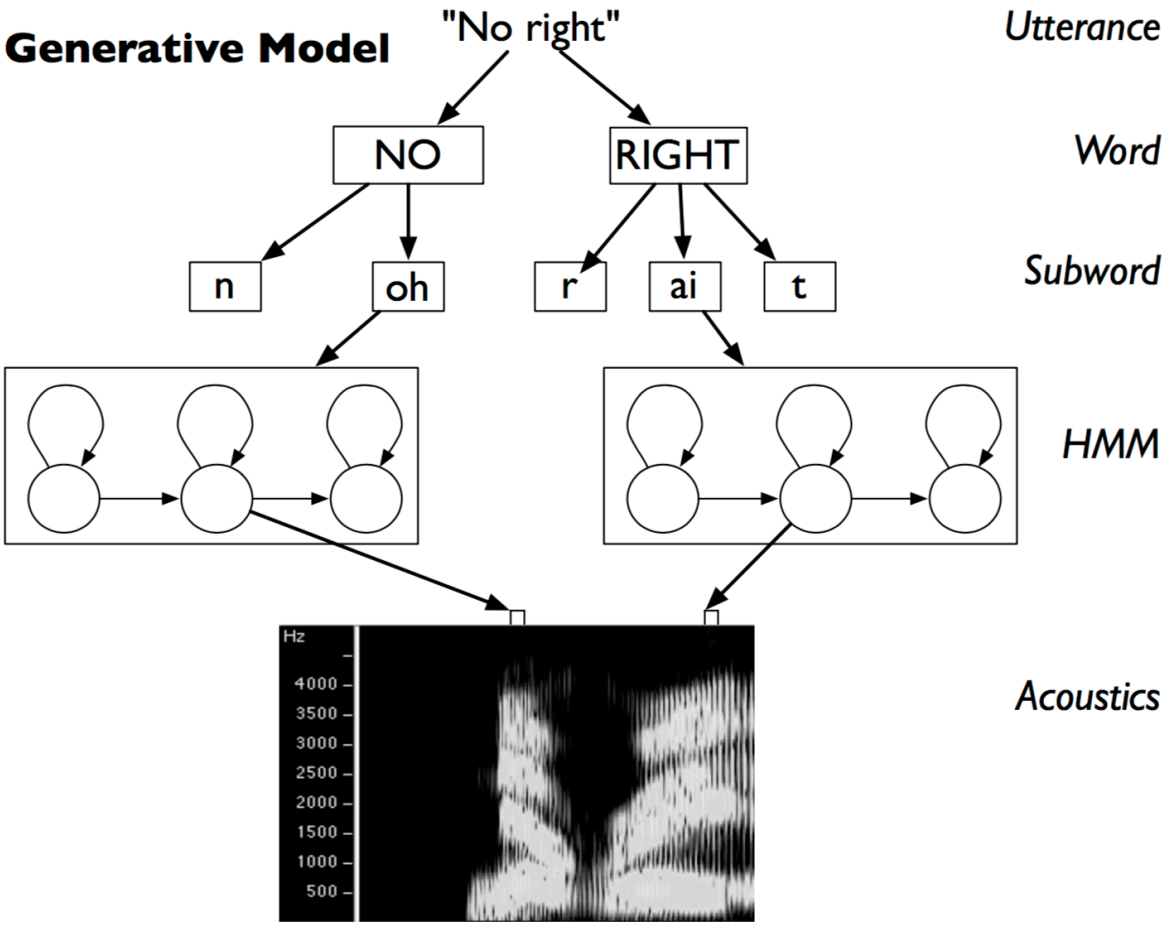


Figure: From speech to utterances[3]

[3] Image from Steve Renals's lecture on ASR

ASR overview

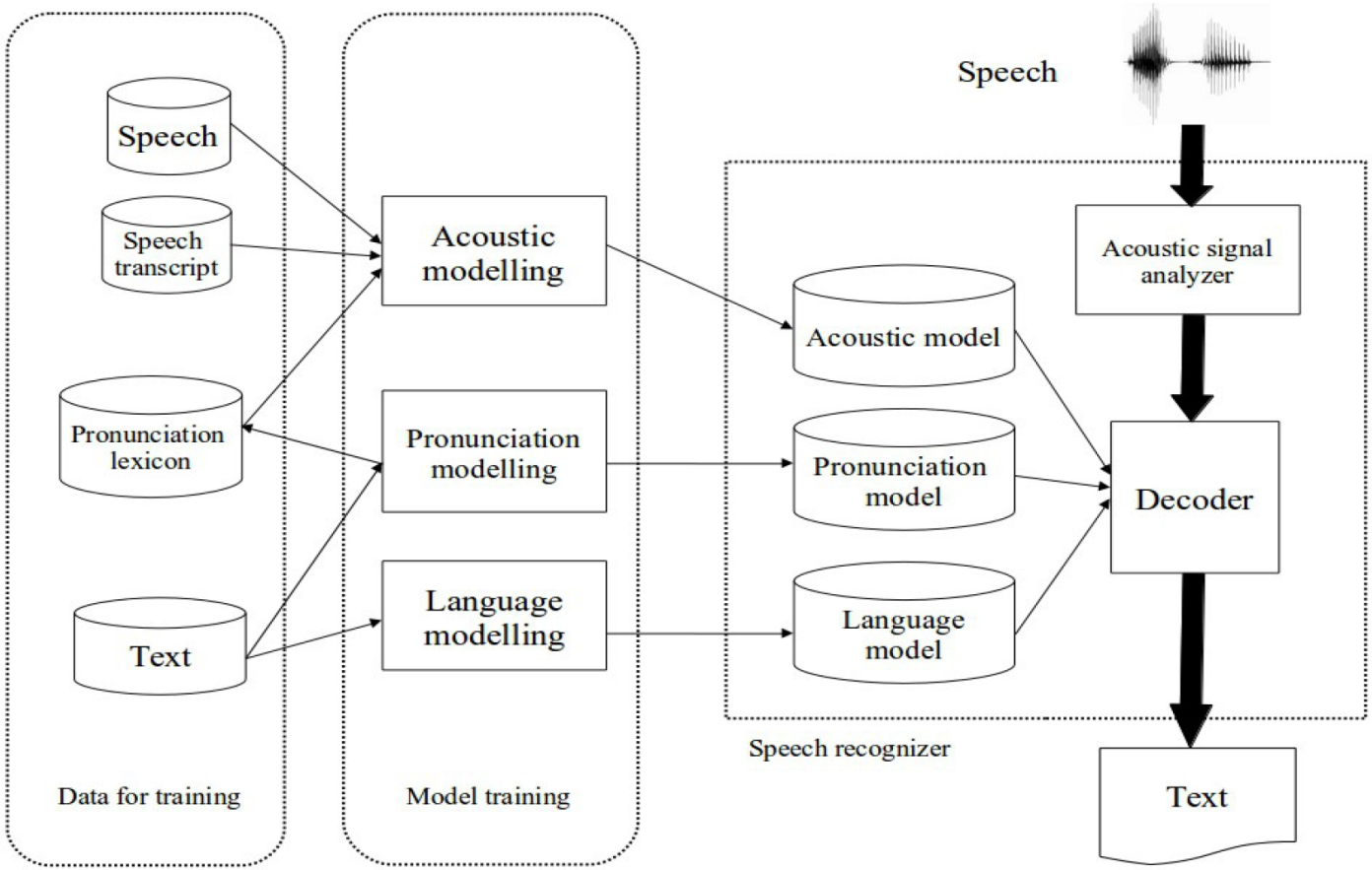


Figure: ASR Overview

Acoustic modeling: HMM/GMM

- Complex sequential patterns of speech decomposed into piecewise stationary segments
- Sequential structure of the data described by a sequence of states
  - HMM (Hidden Markov Models) transitions
- Local characteristics of the data described by a distribution associated to each state
  - GMM (Gaussian Mixture Models) observations (outputs)

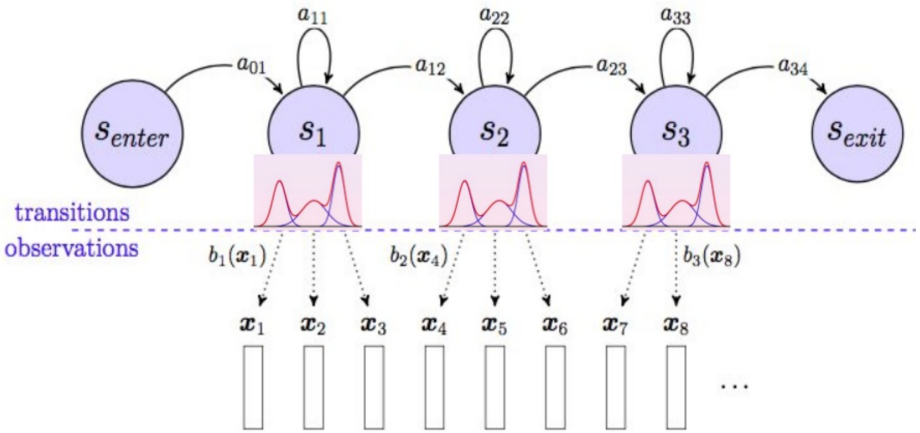


Figure: HMM/GMM approach

HMMs

- Well known algorithms for
  - *training* the model parameters (Baum-Welch algo.)
  - *decoding* the most probable hidden state sequence (Viterbi algo.)
  - *evaluate* the likelihood of an observation being generated by a HMM (Forward algo.)
- Phonemes are generally modeled in context (1 phoneme = N HMMs)
  - triphones or quintphones (model co-articulation)
  - state or parameter tying to reduce model complexity

Language models: from N-grams to RNNs

For a sequence of T words $W = w_1, w_2, ..., w_T$

$$P(W) = \prod_{k=1}^{T} P(w_k | w_1, w_2, ..., w_{k-1}) \tag{2}$$

$$P(W) = \prod_{k=1}^{T} P(w_k | h) \tag{3}$$

n-gram LM: $h = w_{k-n+1}, w_{k-n+2}, ..., w_{k-1}$

recurrent neural network LM: $h = rnn\_state(E(w_1), E(w_2), ..., E(w_{k-1}))$

NNs in the 90s and 00s

- Introduced in the 80s and 90s to speech recognition, but extremely slow and poor in performance compared to the state-of-the-art HMM/GMM
- Several papers published by ICSI, CMU, IDIAP several decades ago!
- Pros: no assumption about a specific data distribution
- Cons: slow and do not scale to large tasks

NNs for acoustic modeling (1990-2010)

- In most approaches, NNs model the posterior probability $p(s|x)$ of an HMM state s given an acoustic observation x
- Existing HMM speech recognizers can be used
- This model is known as hybrid NN-HMM and was introduced by Renals et al. (1994)

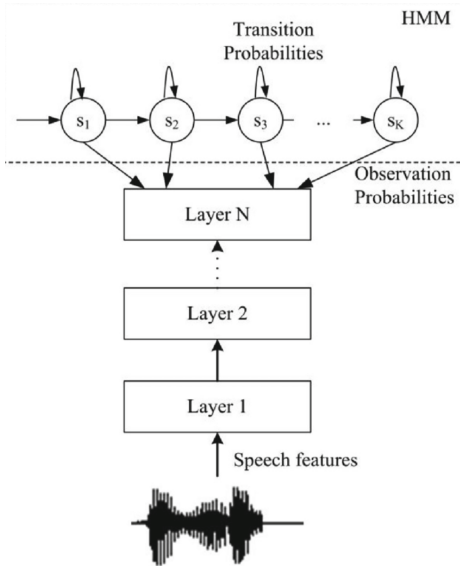


Figure: Hybrid NN-HMM

NNs for language modeling (1990-2010)

- Rescoring a lattice of output hypotheses using NN LM instead of N-gram
- Introduced by Bengio et al. (2003)
- Extended to large vocabulary speech recognition (Schwenk, 2007)
- Reducing computational complexity
  - using shortlist at output layer (Schwenk, 2007)
  - hierarchical decomposition of output probabilities (Morin and Bengio, 2005; Mnih and Hinton, 2008; Le et al., 2011)
- Recurrent neural networks were used in LM training (Mikolov et al., 2010)

Deep learning breakthrough

Like in vision, due to

- More data
  - ex: (2015) Librispeech (en) 1,000h (Panayotov et al., 2015)
  - ex: (2016) Baidu Deep Speech 2 (en) 12,000h (Amodei et al., 2016)
  - ex: (2017) Google Home (en) 18,000h (from a Google presentation)
  - ex: (2018) Google wav2words (en) >100.000h (informal discussion)
  - ex: (2021) Meta XLS-R >436,000h (Babu et al., 2021) (self-supervised)
  - ex: (2022) OpenAI Whisper model trained on 680,000 hours of multilingual speech[4] (Alec Radford, 2022)
- Computation (ex: GPU)
- Better optimization algorithms and training objectives
- ASR Toolkits (ex: Kaldi (Povey et al., 2011) and DL frameworks (Tensorflow, Pytorch)

[4] >75 years of speech !

End-to-end ASR (get rid of HMMs)

Approaches for end-to-end ASR

- Connectionist Temporal Classification (CTC)
  - Solves the problem of unaligned input and output sequences by marginalizing the conditional likelihood of the output sequence given the input over all possible alignments
- Attention Modeling
  - Simultaneously optimize alignment and grapheme (or word) decoding using attention weights (linear combination of hidden states) to influence the generated output
- Transducer-based
  - Allow to decouple the acoustic model from the language model; elegant to leverage larger amounts of raw text (for LM)

CTC loss function

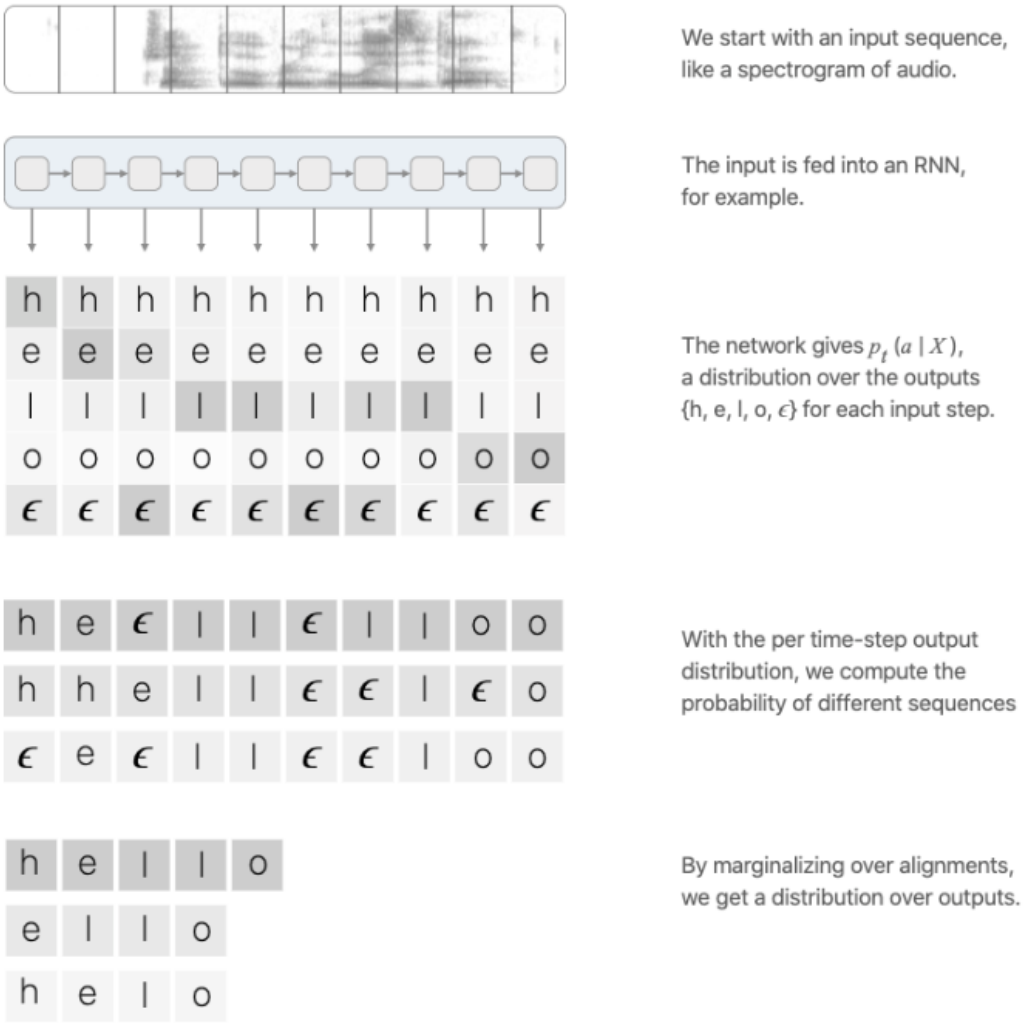


Figure: CTC Overview [5]

CTC overview

- Connectionist Temporal Classification
- Model will learn to align the transcript itself during training (Graves et al., 2006)
- Defined over a label sequence z (of length M)
- *blank or* _ symbol allows M-length target sequence to be mapped to a T-length sequence x[6]
- z can be represented by a set of all possible CTC paths (sequence of labels, at frame level) that are mapped to z
  - ex: M=2 ($z = hi$) and T=3 (3 frames): possible sequences are 'hhi', 'hii', '_hi', 'h_i', 'hi_'
- Probability $p(z/x)$ evaluated as sum of probabilities over all possible CTC paths (using Forward-Backward)
- Generate frame posteriors at decoding time

[6] gives the model the ability to say that a certain audio frame did not produce a character

CTC loss function

$$p(Y \mid X) = \sum_{A \in \mathcal{A}_{X,Y}} \prod_{t=1}^{T} p_t(a_t \mid X)$$

The CTC conditional **probability** | **marginalizes** over the set of valid alignments | computing the **probability** for a single alignment step-by-step.

Figure: CTC Loss [7]

- CTC loss can be very expensive to compute
- The problem is there can be a massive number of alignments
- We can compute the loss much faster with a dynamic programming algorithm

[7] from https://www.assemblyai.com/blog/end-to-end-speech-recognition-pytorch

CTC inference

- Greedy decoding

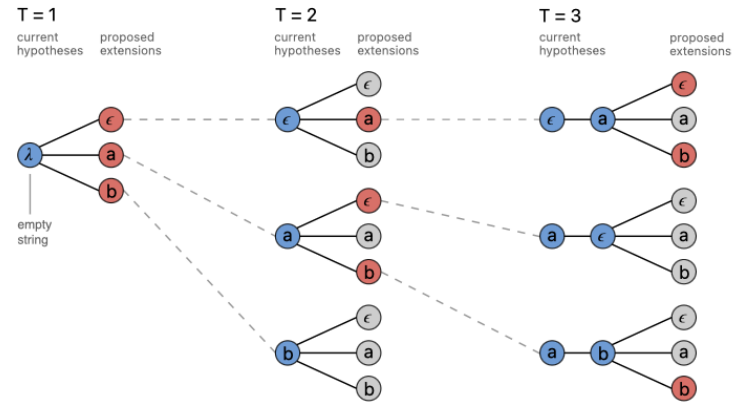
$$Y^* = \underset{Y}{\operatorname{argmax}}\ p(Y \mid X)$$

One heuristic is to take the most likely output at each time-step. This gives us the alignment with the highest probability:

$$A^* = \underset{A}{\operatorname{argmax}} \prod_{t=1}^{T} p_t(a_t \mid X)$$

We can then collapse repeats and remove ϵ tokens to get Y.

- Beam-Search decoding



A standard beam search algorithm with an alphabet of $\{\epsilon, a, b\}$ and a beam size of three.

Figure: from `https://www.assemblyai.com/blog/`

Attention modeling

- Architecture similar to neural machine translation
- Speech encoder based on CNNs or pyramidal LSTMs **?**

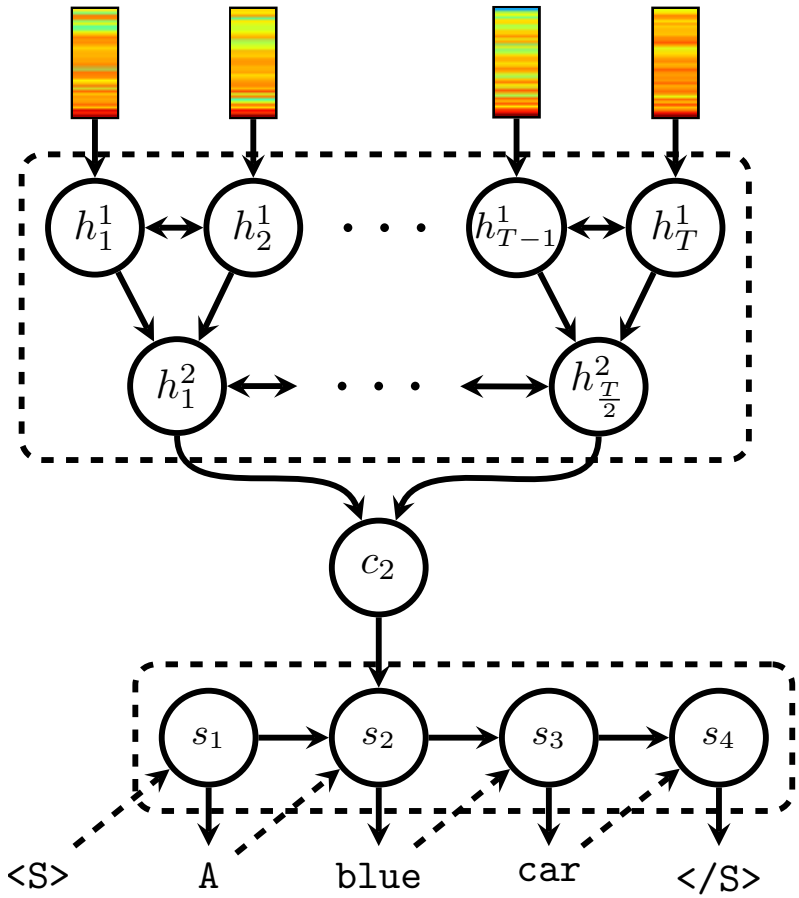


Image from Alexandre Berard's thesis

Attention modeling

Initially proposed for (neural) machine translation (Bahdanau et al., 2014) and introduced for ASR by Chorowski et al. (2015)

- A context (attention) model is a function of the encoder codes and of the previous decoded tokens
- A speech encoder is defined (CNNs, pyramidal LSTMs)
- While CTC generates frame-level posteriors, attention models generate L predictions until the end-of-sequence symbol (no posterior for a given frame)
- Well-known issue with attention and CTC models is the thin lattices we end up with

Attention modeling (different view)

- Also called LAS: **L**isten (encode) ; **A**ttend (attention) and **S**pell (decode)

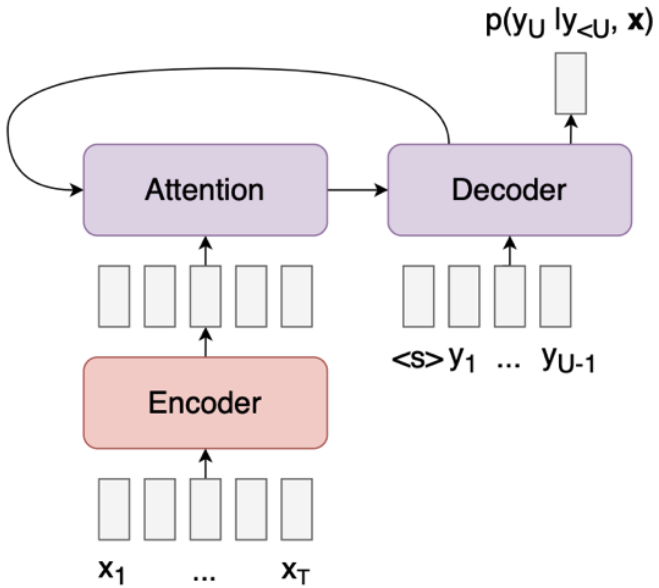


Image from
https://lorenlugosch.github.io/posts/2020/11/transducer/

Attention modeling (alignment)

- Allows non-monotonic alignements
- As opposed to CTC (monotonic)

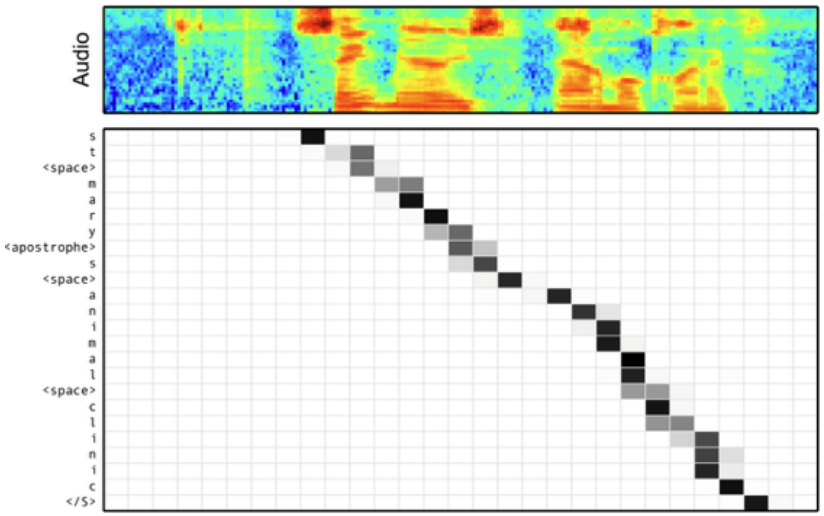


Image from
https://lorenlugosch.github.io/posts/2020/11/transducer/

CTC versus Attention

- CTC provides monotonic alignements while attention allows non-monotonic alignements
  - Attention more suitable for speech translation
- Attention will induce Auto-Regressive decoding (one token at a time) as attention depends on decoder's state
  - slow inference
- CTC does not have this constraint and is much simpler
  - faster inference
- Can we do better ?
  - Yes, transducer models

Transducer models

- Problems with CTC
  - The output sequence length M has to be smaller than the input sequence length T (prevents models that do a lot of input pooling)
  - The outputs are assumed to be independent of each other. CTC models often produce wrong outputs like "I eight food"

Transducer models

- Solve both problems
- Predictor is a language model
- Joiner is a simple feed-forward network

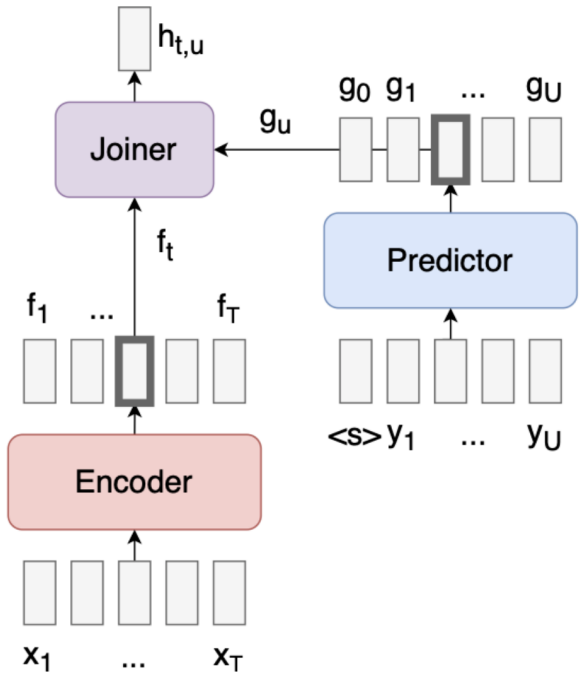
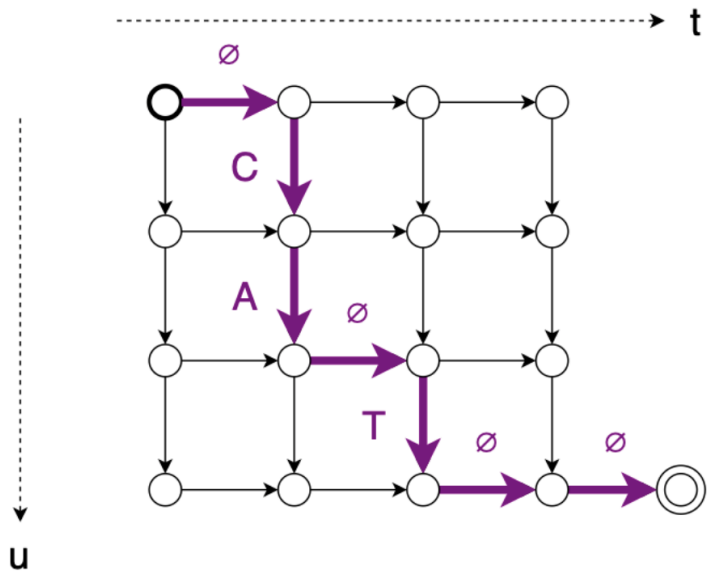


Image from `https://lorenlugosch.github.io/posts/2020/11/transducer/`

Transducer models

- Interesting features
  - If encoder is causal (not using something like a bidirectional RNN), then search can run in online/streaming fashion
  - The predictor only has access to y (not x) unlike the decoder in an attention model, so we can easily pre-train the predictor on text-only data
  - Naturally defines alignment between x and y

Here's one alignment: $\mathbf{z} = \varnothing, C, A, \varnothing, T, \varnothing, \varnothing$



Self supervised representation learning

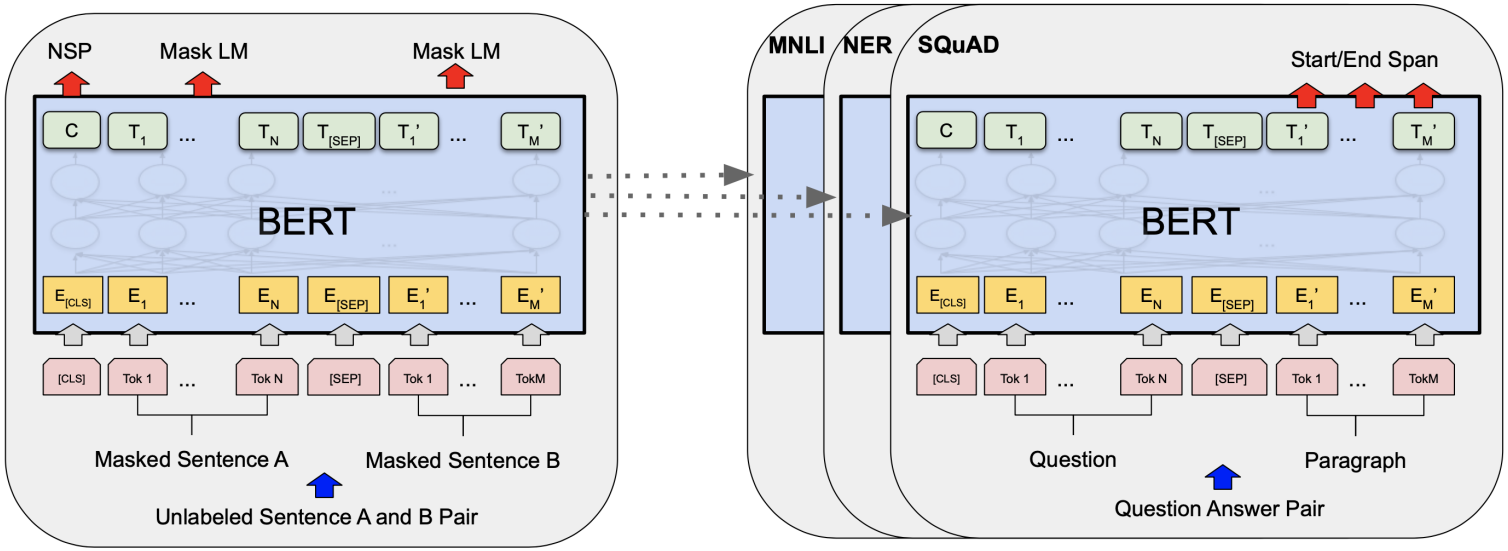
- Using huge unlabeled data for training ; targets are computed from the signal itself
  - *"learn representations using objective functions similar to those used for supervised learning, but train networks to perform pretext tasks where both the inputs and labels are derived from an unlabeled dataset"* (from Chen et al. (2020))
- Introduced for vision: see for instance (Chen et al., 2020)
  - learn representations by contrasting positive pairs against negative pairs
- Introduced also in NLP: see for instance (Devlin et al., 2018)
  - learn representations by predicting tokens that were masked in an input sequence

Previous works

- Stacked restricted Boltzman machines (RBM) (Hinton and Salakhutdinov, 2006)
  - hidden layer extracts relevant features from the observations that serve as input to next RBM that is stacked on top of it forming a deterministic feed-forward neural network
- Denoising autoencoders (AE) (Vincent et al., 2008)
  - networks which are tasked with reconstructing outputs from their (noisy) input versions
- Variational autoencoders (VAE) (Kingma and Welling, 2013)
  - VAE is like a traditional AE in which the encoder produces distributions over latent representations (rather than deterministic encodings) while the decoder is trained on samples from this distribution
  - both encoder and decoder are trained jointly
  - VQ-VAE (van den Oord et al., 2017) replaces continuous latent vectors with deterministically quantized versions

Pre-trained language models

- Leverage large amount of freely available unlabeled text to facilitate transfer learning in NLP
- Yield state-of-the-art results on a wide range of NLP tasks + save time and computational resources
- Example of BERT (Devlin et al., 2018) based on the Transformer model (Vaswani et al., 2017)



Self supervised representation learning from speech

- Autoregressive predictive coding (APC) (Chung et al., 2019; Chung and Glass, 2020)
  - Considers the sequential structure of speech and predicts information about a future frame
- Contrastive Predictive Coding (CPC) (Baevski et al., 2019; Schneider et al., 2019a; Kahn et al., 2019)
  - Easier learning objective which consists in distinguishing a true future audio frame from negatives
- Other approaches for feature representation learning using multiple self supervised tasks (Pascual et al., 2019; Ravanelli et al., 2020) or bidirectional encoders (Song et al., 2019; Liu et al., 2020; Wang et al., 2020)

Autoregressive predictive coding (APC)

- Predicting the spectrum of a future frame (rather than a wave sample) (Chung et al., 2019)
- Somewhat inspired by language models (LMs) for text, which are typically a probability distribution over sequences of T tokens $(t_1, t_2, ..., t_T)$

$$P(sequence) = \prod_{k=1}^{T} P(t_k | t_1, t_2, ..., t_{k-1}) \tag{4}$$

$$P(sequence) = \prod_{k=1}^{T} P(t_k | h) \tag{5}$$

- Recurrent neural network LM:
  $h = rnn\_state(E(t_1), E(t_2), ..., E(t_{k-1}))$
- For speech, each token t_k corresponds to a frame rather than a word or character token

Autoregressive predictive coding (APC)

- No final set of target tokens (softmax layer replaced by a regression layer)
- Learnable parameters in APC are the RNN parameters θ_{rnn} and the regression layer parameters θ_r
- Encourage APC to infer more global structures rather than the local information in the signal
  - ask the model to predict a frame n steps ahead of the current one
- Model is optimized by minimizing the L1 loss between sequence $(x_1, x_2, ..., x_T)$ and the predicted sequence $(y_1, y_2, ..., y_T)$:

$$\sum_{i=1}^{T-n} |t_i - y_i|, t_i = x_{i+n} \tag{6}$$

Autoregressive predictive coding (APC)

- Chung et al. (2019) models APC with a multi-layer unidirectional LSTM with residual connections
- After training, RNN hidden states are taken as the learned representations
- A follow-up work (Chung and Glass, 2020) adds an auxiliary objective that serves as regularization to improve generalization

Feature	ASR (WER ↓)	ST (BLEU ↑)
log Mel	18.3	12.9
APC w/ L_f	15.2	13.8
APC w/ L_m	14.2	14.5

Table 2: Automatic speech recognition (ASR) and speech translation (ST) results using different types of features as input to a seq2seq with attention model. Word error rates (WER, ↓) and BLEU scores (↑) are reported for the two tasks, respectively.

Differences between speech and text SSL

- Input speech representations (MFCCs for instance) are already in a vector form (no embedding layer)
- More uncertainty
  - text (discrete): finite number of possible outcomes (target tokens)
  - speech and video (continuous): infinite number of frames that can plausibly follow a given audio (or video) clip

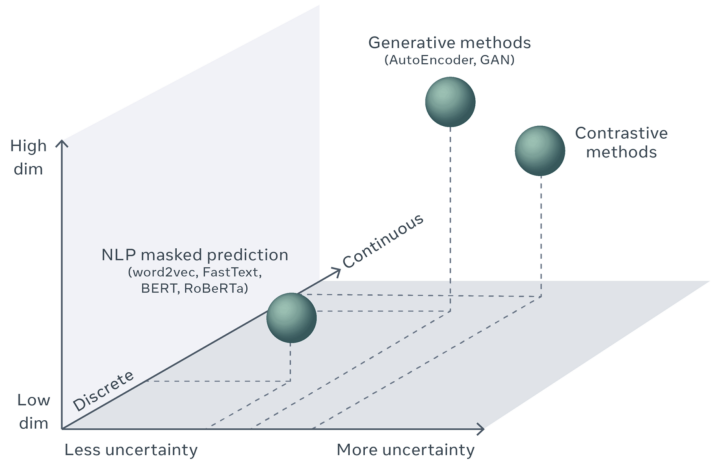
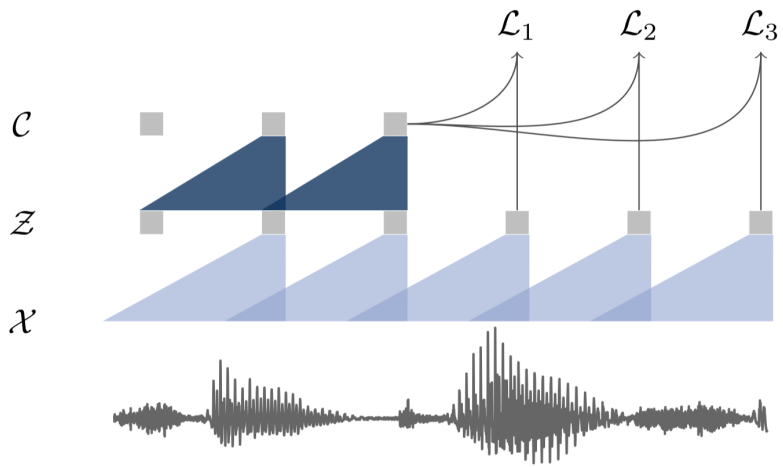


Figure: Figure from `https://ai.facebook.com/blog/self-supervised-learning-the-dark-matter-of-intelligence`

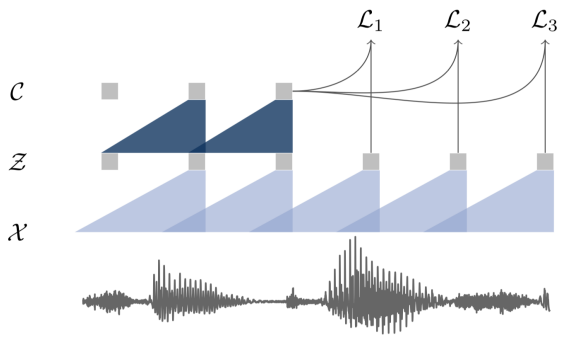
Contrastive Predictive Coding (CPC)

- Idea proposed by van den Oord et al. (2018)
- Maybe an easier learning objective (classification instead of regression)
- Use of a contrastive loss that distinguishes a true future audio sample from negatives
- Example of *wav2vec* (Schneider et al., 2019b) that relies on a fully convolutional architecture
- Applied the learned representations to improve a supervised ASR system



Contrastive Predictive Coding (CPC)

- Encoder network $Z = f(X)$; 5 (causal) convolution layers ; local feature representations z_i encode 30 ms of audio every 10ms
- Context network $C = g(Z)$; 9 (causal) convolution layers ; mix multiple z_i (receptive field of dimension v corresponding to 210ms) into a single contextualized representation c_i
- Model trained to distinguish a sample z_{i+k} that is k steps in the future from distractor samples $\tilde{z}$ drawn from a proposal distribution p_n by minimizing a contrastive loss for each step $k = 1, ..., K$
- Negatives examples sampled by uniformly choosing distractors from each audio sequence: is $p_n(z) = 1/T$ where T is the sequence length



Representation learning with multiple self-supervised tasks

- Problem-agnostic speech encoder (PASE) (Pascual et al., 2019)
- PASE+: robust speech recognition in noisy and reverberant environments (Ravanelli et al., 2020)

Representation learning with multiple self-supervised tasks

- Problem-agnostic speech encoder (PASE) (Pascual et al., 2019)
- Jointly tackle multiple self-supervised tasks using an ensemble of neural networks that cooperate to discover good speech representations
- Approach requires consensus across tasks, more likely to learn general, robust, and transferable features
- Authors find that such representations outperform more traditional hand-crafted features in different speech classification tasks such as speaker identification, emotion classification, and ASR

Problem-agnostic speech encoder (PASE)

- Encoder: SincNet (Ravanelli and Bengio, 2018) + Convblocks (receptive field 150ms)
- Workers: one for each task (see next slide)

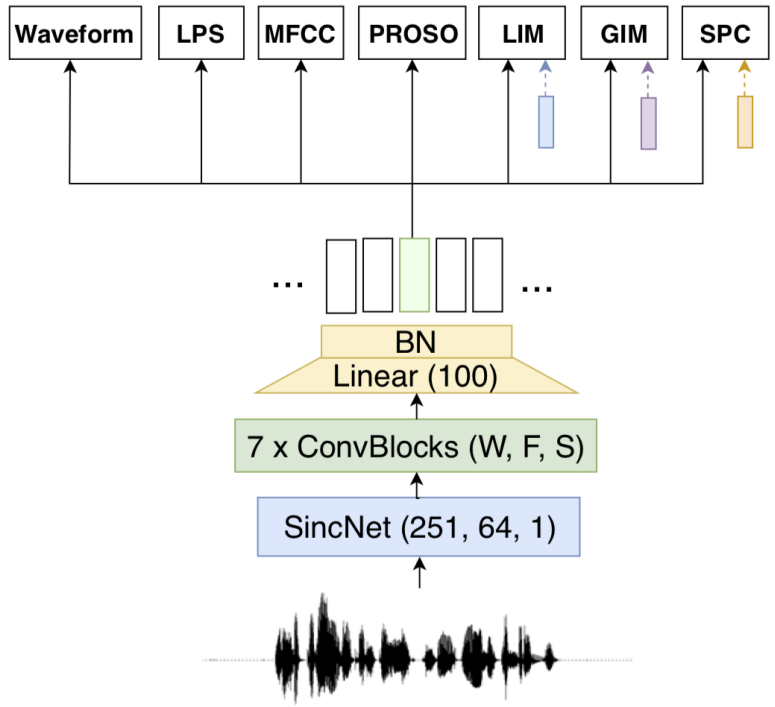


Figure 1: *The PASE architecture, with the considered workers.*

Figure: Figure from (Pascual et al., 2019)

Problem-agnostic speech encoder (PASE)

- Regression workers that solve 7 self-supervised tasks
- Trained to minimize the mean squared error (MSE) between the target features and the network predictions
  - **Waveform** learns to reconstruct waveforms
  - **LPS** reconstruct log power spectrum
  - **MFCC** reconstruct mel-frequency cepstral coefficients
  - **Prosody** predicts 4 basic prosodic features per frame
  - **LIM** (local info max) contrastive task where positive sample is drawn from the same utterance and a negative sample is drawn from another random utterance (that likely belongs to a different speaker)
  - **GIM** (global info max) similar to LIM using global representations (averaged over 1s) instead of local ones
  - **SPC** sequence predicting coding: similar to contrastive predictive coding (CPC) introduced earlier

Problem-agnostic speech encoder (PASE)

- Experiments on speaker identification, emotion recognition and ASR

Table 2: *Accuracy comparison on the considered classification tasks using MLPs and RNNs as classifiers.*

Model	Classification accuracy [%]					
	Speaker-ID *(VCTK)*		Emotion *(INTERFACE)*		ASR *(TIMIT)*	
	MLP	RNN	MLP	RNN	MLP	RNN
MFCC	96.9	72.3	90.8	91.1	81.1	84.8
FBANK	98.4	75.1	94.1	92.8	80.9	85.1
PASE-Supervised	97.0	80.5	93.8	92.8	82.1	84.7
PASE-Frozen	97.3	82.5	91.5	92.8	81.4	84.7
PASE-FineTuned	**99.3**	**97.2**	**97.7**	**97.0**	**82.9**	**85.3**

Table 3: *Word error rate (WER) obtained on the DIRHA corpus.*

	WER [%]
MFCC	35.8
FBANK	34.0
PASE-Supervised	33.5
PASE-Frozen	32.5
PASE-FineTuned	**29.8**

Figure: Table from (Pascual et al., 2019)

Many follow-up approaches

- Speech-XLNet: Unsupervised Acoustic Model Pretraining For Self-Attention Networks (Song et al., 2019)
- Mockingjay: Unsupervised speech representation learning with deep bidirectional transformer encoders (Liu et al., 2020)
- Unsupervised pre-training of bidirectional speech encoders via masked reconstruction (Wang et al., 2020)
- **Wav2vec 2.0: A Framework for Self-Supervised Learning of Speech Representations (Baevski et al., 2020)**
- **HuBERT: Self-Supervised Speech Representation Learning by Masked Prediction of Hidden Units (?)**

Speech-XLNet

- Speech-XLNet (Song et al., 2019)
- Learn speech representations with self-attention networks
- BERT-like autoencoding (AE) scheme to train a bi-directional speech representation model (not only left-to-right)
- Mask and reconstruct speech frames rather than word tokens (regression instead of classification task)
- Encourage network to learn global structures by shuffling speech frame orders (can be also seen as dynamic data augmentation)
- Training using a Mean Absolute Error (MAE) loss over several permutations of the input frames
- (Unfortunately) not compared with previous APC and CPC approaches

Speech-XLNet

- Experiments on Hybrid and end-to-end ASR
- Results of Hybrid ASR on TIMIT are reported below

Table 2: *PER comparison with previous pretrain methods. We approximate the number of parameters based on the description in the previous studies.*

Pretrain Method	Pretrain Data	Pretrain Params	Dev/Test PER(%)
VQ-Wav2vec ([8])	libri (960h)	34M	15.34 / 17.78
RBM-DBN ([21])	timit (8h)	$\approx$ 34.2M	15.90 / 16.80
Ours (Randomly Init)	-	19.9M	13.20 / 15.10
Wav2vec ([7])	libri+wsj (1041h)	34M	12.90 / 14.70
Ours (Pretrained)	libri+wsj+ted (1248h)	19.9M	11.70 / 12.80
VQ-Wav2vec+BERT ([8])	libri (960h)	$\approx$ 71.8M	9.64 / 11.64

Figure: Table from (Song et al., 2019)

Unsupervised speech representation learning with deep bidirectional transformer encoders

- Predict the current frame through jointly conditioning on both past and future contexts (Mockingjay (Liu et al., 2020))
- Masked acoustic modeling task (rand. mask 15% of input frames)[8]
- Use multi-layer transformer encoders and multi-head self-attention
- Add a prediction head (2 layers of feed-forward network with layer-norm) using last encoder layer as input

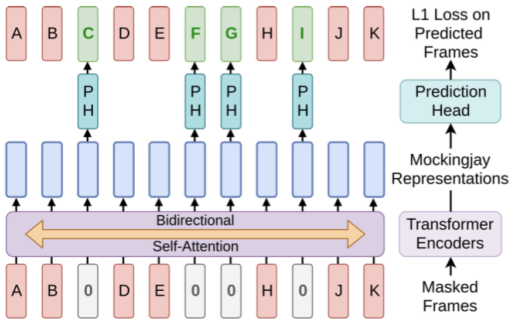


Figure: Table from (Liu et al., 2020)

[8] Use of additional consecutive masking where they mask consecutive frames C_{num} to zero. The model is required to infer on global rather than local structure

Unsupervised speech representation learning with deep bidirectional transformer encoders

- Experiments on phoneme classification
- With different amount of annotated data for training

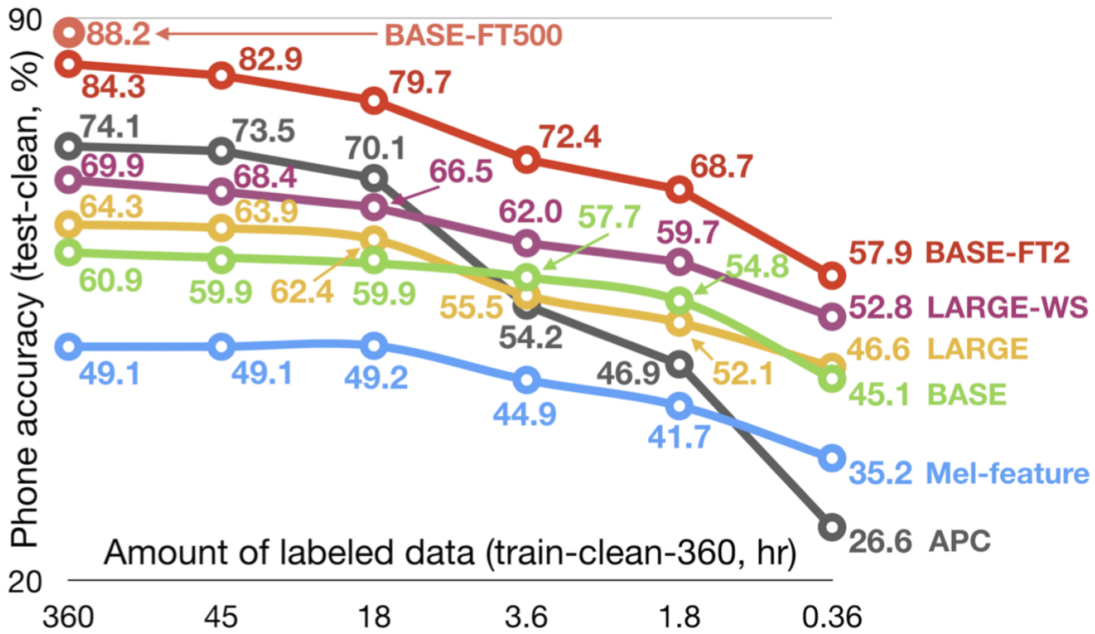


Figure: Figure from (Liu et al., 2020)

Unsupervised Pre-training of Bidirectional Speech Encoders via Masked Reconstruction

- Pre-training speech representations via a masked reconstruction loss (Wang et al., 2020)
- Masking in both frequency and time to encourage model to exploit spatio-temporal info
- Elegant extension of data augmentation technique SpecAugment (Park et al., 2019)

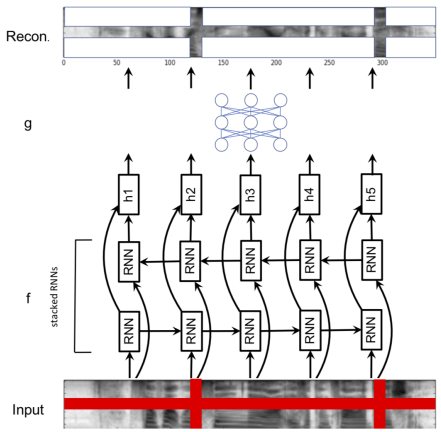


Figure: Figure from (Wang et al., 2020)

Unsupervised Pre-training of Bidirectional Speech Encoders via Masked Reconstruction

Table 3: Dev set %CERs of character-based systems pre-trained on LibriSpeech, and fine-tuned with different amounts of supervised data.

	Baseline	Pre-train *Libri.* w/o LIN	Pre-train *Libri.* w/ LIN
si84	15.23	14.02	**13.29**
+ SpecAug	12.98	12.26	**11.70**
si284	7.01	6.90	**6.48**
+ SpecAug	6.29	6.19	**5.61**

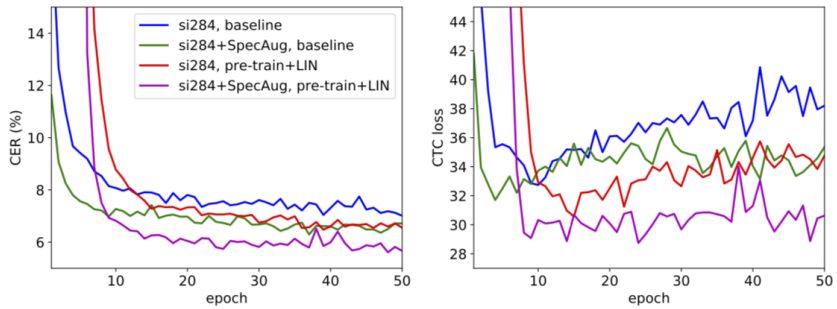


Fig. 2: Dev set learning curves (%CER and CTC loss) of different systems pre-trained on *LibriSpeech*. The first 5 epochs of fine-tuning update only the LIN and softmax layers.

Figure: From (Wang et al., 2020)

Wav2vec 2.0: A Framework for Self-Supervised Learning of Speech Representations

- Encode speech with CNN layers and then mask spans of the resulting latent speech representations (cf masked LM)
- Learn *discrete* speech units as latent representations[9]
- Latent representations fed to a Transformer network to build contextualized representations
- Model trained with a contrastive task (true latent to be distinguished from distractors)

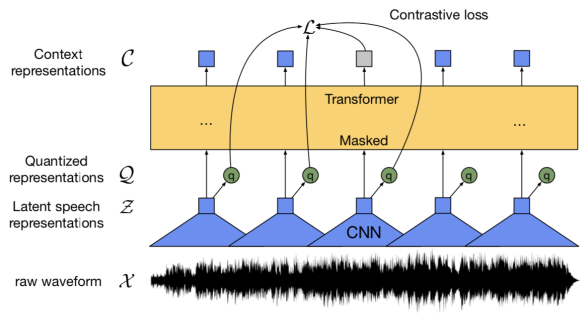
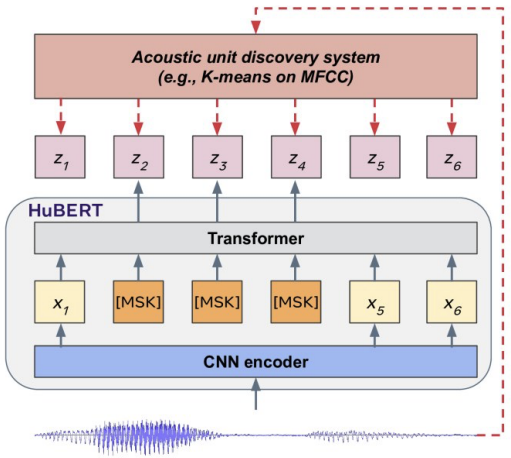


Figure: Figure from (Baevski et al., 2020)

[9] Authors found this more effective than non-quantized targets

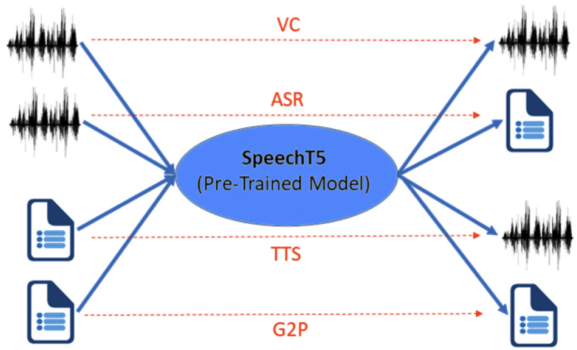
HuBERT: Self-Supervised Speech Representation Learning by Masked Prediction of Hidden Units

- Similar Conv+Transf encoder but
- Uses cross-entropy loss (same as BERT) instead of contrastive loss
- Discrete targets are built through a separate clustering process
- Learnt *discrete* speech units are refined at each iteration (3 iterations for large models)
- X-LARGE version of HuBERT as 1 billion parameters
- Model recently outperformed SOTA techniques for speech recognition, generation, and compression



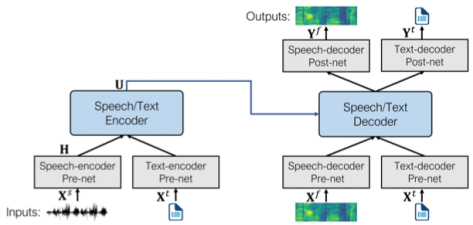
SpeechT5 (Ao et al., 2021)

- A multimodal extension of transformer encoder-decoder models such as T5
- Encode or decode both speech and text with a single model
- Maps both acoustic and text information in a shared vector space
- Used to initialize ASR (speech-to-text), TTS (text-to-speech), Voice Conversion (VC – speech-to-speech), etc.



SpeechT5 (Ao et al., 2021)

- A single transformer encoder/decoder backbone
- Several modality-specific pre-post nets
  - standard for text
  - encoder pre-net for speech similar to the CNN blocks of wav2vec2.0
  - decoder pre-net for speech is different (fully connected net + ReLU) as the model will output slices of filterbank features (no speech directly)
  - a speaker embedding is concatenated to the output of the speech-decoder pre-net to support voice conversion and multi-speaker TTS



(a) The model architecture of SpeechT5

SpeechT5 (Ao et al., 2021)

- A composite loss with multiple pre-training objectives
  - a masked language modeling (MLM) loss on discrete latent speech representations (*à la HuBERT*)
  - a speech reconstruction L1 loss (in the continuous filterbank space)
  - a cross-entropy loss specific to the prediction of the stop token
  - a text denoising objective (*à la BART*)
  - a cross-modal objective to better align speech and text representations (unclear in the paper)

The final pre-training loss with unlabeled speech and text data can be formulated as

$$\mathcal{L} = \mathcal{L}^s_{mlm} + \mathcal{L}^s_1 + \mathcal{L}^s_{bce} + \mathcal{L}^t_{mle} + \gamma \mathcal{L}_d. \quad (6)$$

where γ is set to 0.1 during pre-training.

SpeechT5 Ao et al. (2021)

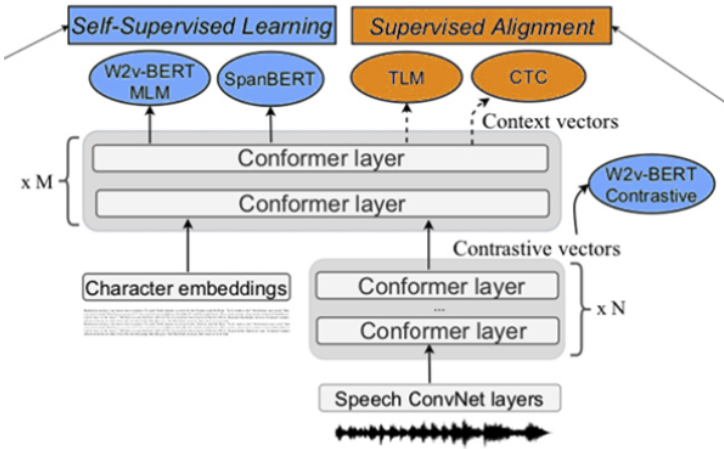
- Experiments on several downstream speech tasks (ASR, VC, TTS, speaker id.) show slightly better results than speech-only pre-training

Model	LM	dev-clean	dev-other	test-clean	test-other
wav2vec 2.0 BASE (Baevski et al., 2020)	-	6.1	13.5	6.1	13.3
HuBERT BASE (Hsu et al., 2021) †	-	5.5	13.1	5.8	13.3
Baseline (w/o CTC)	-	5.8	12.3	6.2	12.3
Baseline	-	4.9	11.7	5.0	11.9
SpeechT5 (w/o CTC)	-	5.4	10.7	5.8	10.7
SpeechT5	-	**4.3**	**10.3**	**4.4**	**10.4**
DiscreteBERT (Baevski et al., 2019)	4-gram	4.0	10.9	4.5	12.1
wav2vec 2.0 BASE (Baevski et al., 2020)	4-gram	2.7	7.9	3.4	8.0
HuBERT BASE (Hsu et al., 2021)	4-gram	2.7	7.8	3.4	8.1
wav2vec 2.0 BASE (Baevski et al., 2020)	Transf.	2.2	6.3	2.6	6.3
Baseline	Transf.	2.3	6.3	2.5	6.3
SpeechT5	Transf.	**2.1**	**5.5**	**2.4**	**5.8**

Table 1: Results of ASR (speech to text) on the LibriSpeech dev and test sets when training on the 100 hours subset of LibriSpeech. † indicates that results are not reported in the corresponding paper and evaluated by ourselves.

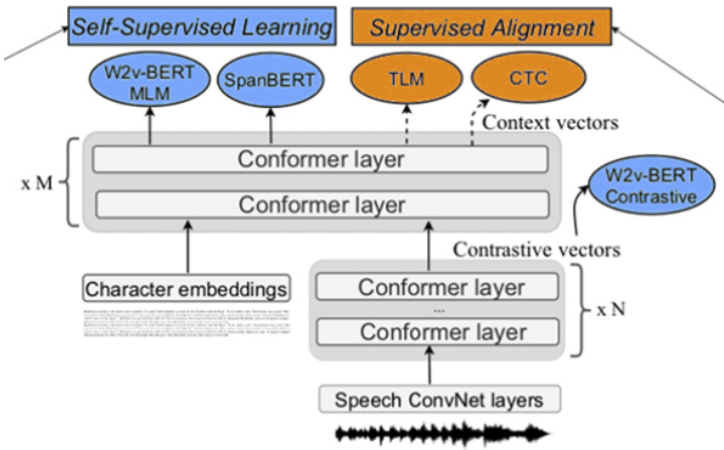
mSLAM Bapna et al. (2022)

- Extension of SLAM Bapna et al. (2021) architecture where speech and text unified in a common encoder model
  - Use of convolution-augmented transformer (conformer) blocks, introduced earlier for ASR Gulati et al. (2020)
  - Input is speech, text, or concatenated speech-text
  - Speech-text pre-training is a mix of self-supervised learning objectives (rather similar to SpeechT5) and supervised cross-modal learning objectives (which leverage aligned speech-text pairs)



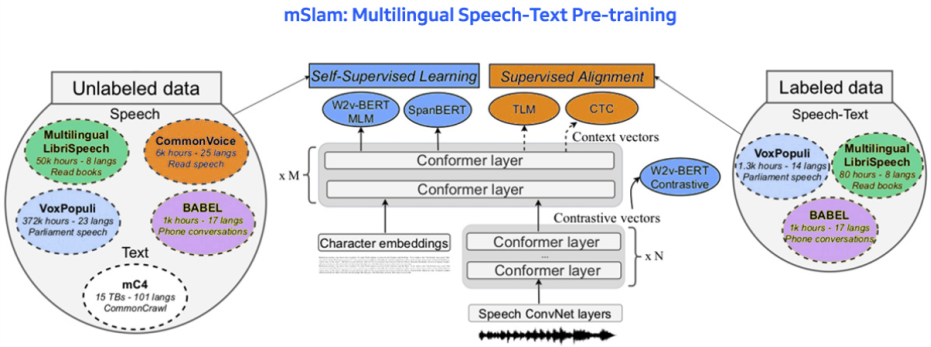
mSLAM Bapna et al. (2022)

- SSL learning objectives for speech (*à la* HuBERT) and text (BERT)
- Speech-text objectives
  - translation language modeling (TLM): predicts masked text or speech spans from a concatenated speech-text input (to encourage use of cross-modal context)
  - Connectionist Temporal Classification (CTC) loss is applied on the speech part of concatenated speech-text using character transcript as a target (ASR loss to learn better speech-text



mSLAM Bapna et al. (2022)

- Massively multilingual (51 lang. speech; 101 lang. text), 2B param
- Downstream task experiments
  - ASR, speech translation, spoken lang. id., spoken intent classification and text classification
  - Speech-text pre-training better than speech-only pre-training for multilingual ASR and translation



mSLAM Bapna et al. (2022)

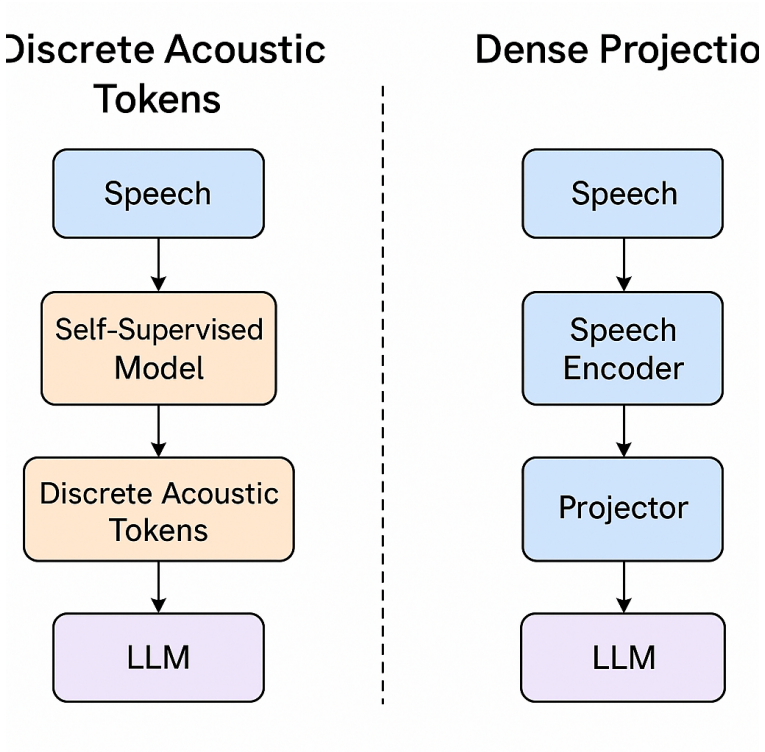
- Zero shot cross-modal properties
- Zero shot text translation from a fine-tuned speech translation model
- The model has never seen src-txt/tgt-txt parallel data but it has seen src-speech/tgt-text + monolingual src-txt
- ... but a system fine-tuned only on text cannot translate speech

Table 6: Zero-shot Performance - CoVoST 2 translation results with X→Y indicating X as the fine tuning modality and Y as the testing modality: S=Speech, T=Text. CAE is our CTC zero-shot character auto-encoding probe.

	Hours	BLEU ↑			CER ↓	
Lang	Paired	S→S	S→T	T→S	S→T	CAE
ar	0	13.3	0.0	0.0		82.6
fa	0	6.2	0.0	0.0		80.0
ja	0	1.6	0.0	0.0		100.0
zh	0	8.7	0.0	0.0		100.0
cy	0	6.1	0.1	0.0		24.3
mn	0	0.5	0.1	0.0		78.4
id	0	3.9	5.1	0.0		10.4
lv	0	19.4	8.2	0.0		18.4
et	0	17.2	8.3	0.0		16.5
sv	0	33.1	15.2	0.0		13.9
ca	0	33.4	16.7	0.0		10.0
ru	0	41.7	21.9	0.0		85.9
sl	6	24.9	7.8	0.0		10.6
pt	10	34.2	17.2	0.0		9.0
nl	41	32.6	16.8	0.0		11.3
ta	63	0.3	0.0	0.0		91.2
tr	69	11.7	1.7	0.0		12.6
it	79	35.0	19.7	0.0		11.2
es	140	39.1	21.2	0.0		7.9
fr	179	36.7	20.0	0.0		9.4
de	197	32.7	16.8	0.0		8.3

Integrate speech into text-only LLMs

- **Two main approaches:**
  1. Discrete token integration (e.g., HuBERT, w2v-BERT)
  2. Dense speech projection into LLM embedding space



Integrate speech into text-only LLMs

1. Discrete Acoustic Tokens

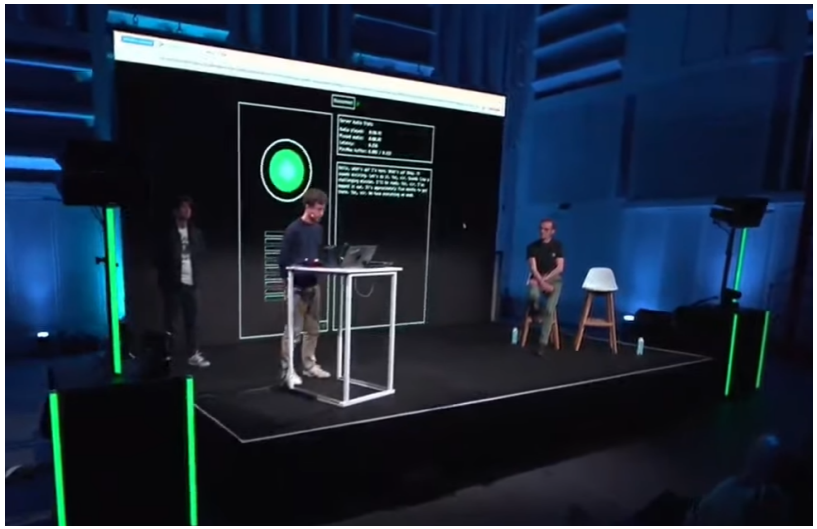
- Use self-supervised models (e.g. HuBERT) to produce discrete units
- Map speech to "phoneme-like" tokens
- Feed tokens as if they were words into the LLM
- Pros: LLM-compatible, simple integration
- Cons: May lose acoustic details, long discrete sequences

2. Dense Projection (Speech Encoder + Projector)

- Encode speech into dense representations
- Use a learned projector to align with LLM's embedding space
- Enables richer, acoustic-aware integration
- Pros: Retains acoustic detail
- Cons: Requires architecture adaptation or fine-tuning

Moshi Speech-2-Speech Model [Defossez et al., 2024]

- Moshi is a Multi-stream Low-latency Speech-to-Speech Dialogue Model Défossez et al. (2024)
  - **Multi-stream**: no explicit turns
  - **Low-latency**: they train causal models, which work on windows of 160ms of speech
  - **Speech-to-Speech**: Model receives as input speech, and produces both text and speech
  - **Dialogue Model**: Trained mostly on conversational data



Moshi: Architecture

- At its core is **Helium**, a 7B parameter text language model (LLM), trained on 2T tokens. The model undergoes specialized training for multistream and full-duplex capabilities.

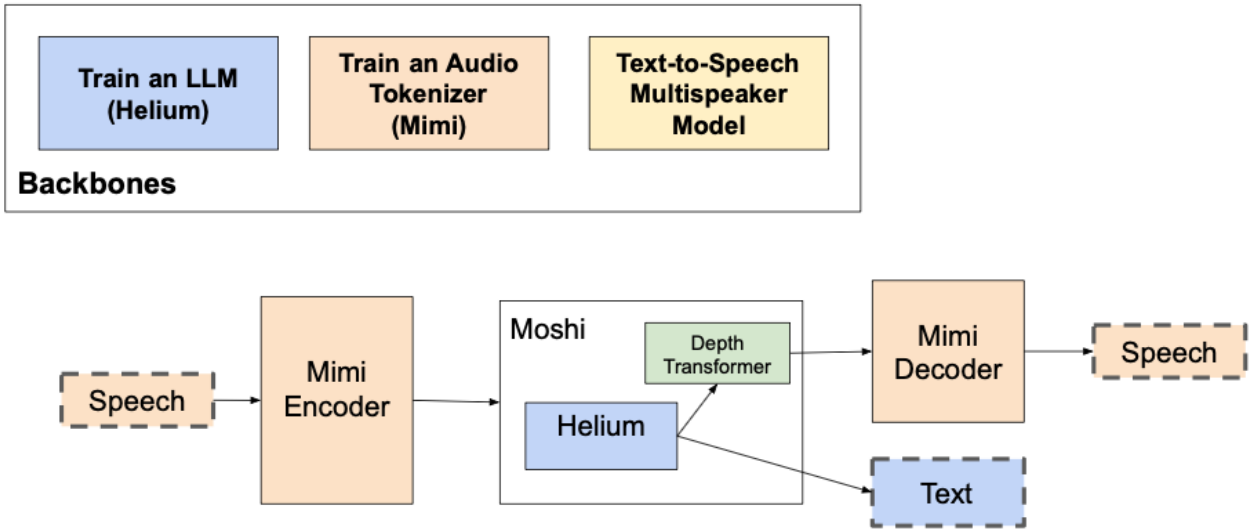


Figure: Figure from Marcely Zanon-Boito

Moshi: Mimi – The Audio Encoder/Decoder

Moshi's unique audio encoder/decoder, **Mimi**, models both speech semantics and acoustics. Key features include:

- Causal convolution followed by transformer block for signal encoding.
- Latent space discretization through vector quantization (VQ).
- Semantic speech units distilled from WavLM speech model.
- Output: 8 discrete substreams—7 for acoustics, 1 for semantics.

Moshi: Full-Duplex Communication and Inner Monologue

Moshi achieves full-duplex communication by modeling both user and system audio streams into one unified sequence. An optional **inner monologue** component improves speech generation by:

- Predicting time-aligned text tokens as a prefix to audio tokens.
- Enhancing factual accuracy and linguistic quality in generated speech.

Moshi: Training

Moshi is pre-trained on:

- 7 million hours of English speech, transcribed using Whisper.
- Fisher conversational dataset augmented for multistream post-training.
- 20k hours of synthetic speech interactions.

Moshi: Performance and Limitations

Key performance insights:

- Moshi outperforms models like SpeechGPT in spoken QA but shows knowledge degradation compared to Helium.
- Inner monologue enhances real-time speech tasks with 5.7% WER on Librispeech (clean).
- Quantization reduces linguistic performance but minimally impacts audio quality.
- Future challenges include managing safety and preventing toxic audio generation.

Conclusion

- Training such models requires access to powerful computing platforms
- Simpler and more efficient pre-training approaches needed
- Standardization in the evaluation process also needed (need for a multimodal and multilingual GLUE)
- Only scratched the surface of zero-shot capabilities of these models (transfer from text to speech tasks)
- More research needed on the decoder side (especially to generate expressive speech with adequate prosody)

ASR Remaining Challenges in 2024

Hello, give ma list of 10 remaining challenges for Automatic SPeech Recognition (ASR) that remained to be solved in 2024. Be concise and sharp

Here are 10 key challenges for Automatic Speech Recognition (ASR) in 2024:

1. **Handling Noisy Environments**: Robust performance in diverse, real-world noisy conditions.
2. **Accurate Multilingual Support**: Seamless recognition across multiple languages and dialects without training separate models.
3. **Low-Resource Languages**: Improved ASR for languages with limited training data.
4. **Speaker Variability**: Handling variations in accents, gender, age, and emotional tone.
5. **Domain Adaptation**: Fine-tuning models for specific industries or applications without full retraining.
6. **Real-Time Processing**: Reducing latency for real-time transcription, especially in low-power devices.
7. **Code-Switching**: Effective recognition of speech that switches between languages within a single utterance.
8. **Privacy and Security**: Ensuring data privacy during live ASR processing, especially for cloud-based models.
9. **Low Computational Footprint**: Deploying high-accuracy ASR on edge devices with limited computational resources.
10. **Bias and Fairness**: Mitigating biases in ASR outputs, particularly for underrepresented groups or accents.

These remain crucial focus areas for advancing ASR technology.

Figure: ASR challenges in 2024 according to openai o-1

Language coverage

- Google addresses (only) 100 languages (ASR)
- Language technology issues: 300 languages (95 % population)
- Language coverage / revitalisation / documentation issues: $>$ 6000 languages !

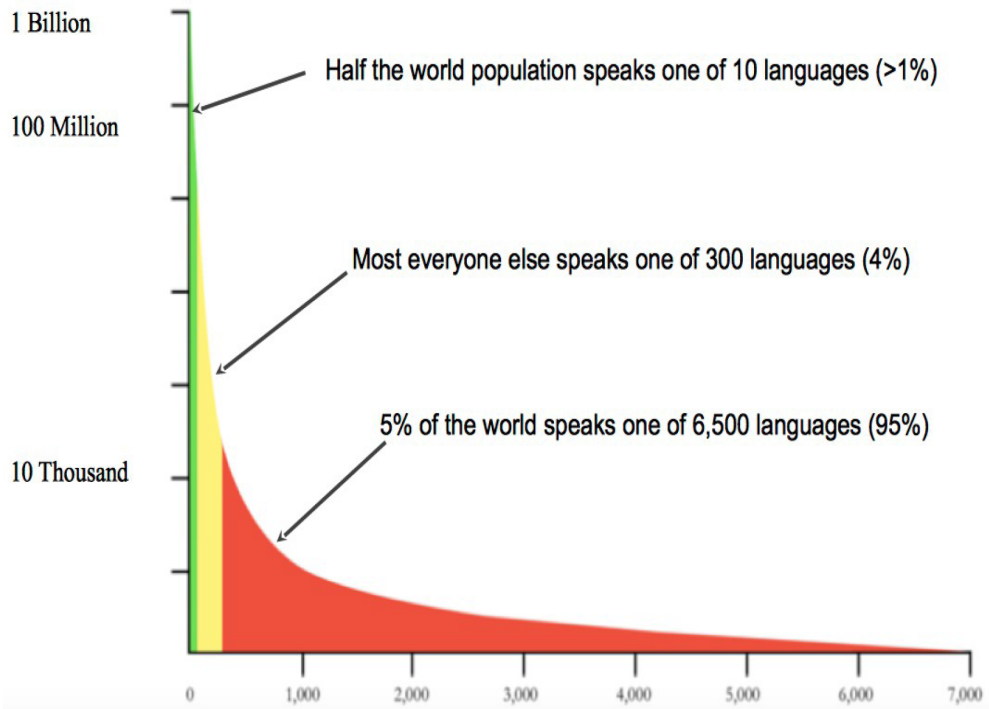
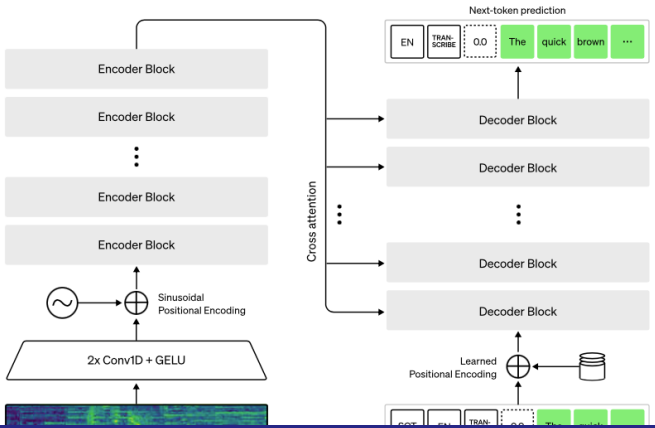


Figure: from Laura Welcher - Big Data for Small Languages The Rosetta Project

Whisper

A massively multilingual ASR system based on weakly-supervised learning Radford et al. (2022)

- trained on 680,000 hours of multilingual and multitask supervised data collected from the web
- use of such a large and diverse dataset leads to improved robustness to accents, background noise and technical language
- enables transcription in multiple languages, as well as translation from those languages into English
- *whisper* architecture is a simple end-to-end approach, implemented as an encoder-decoder Transformer



Low resource ASR

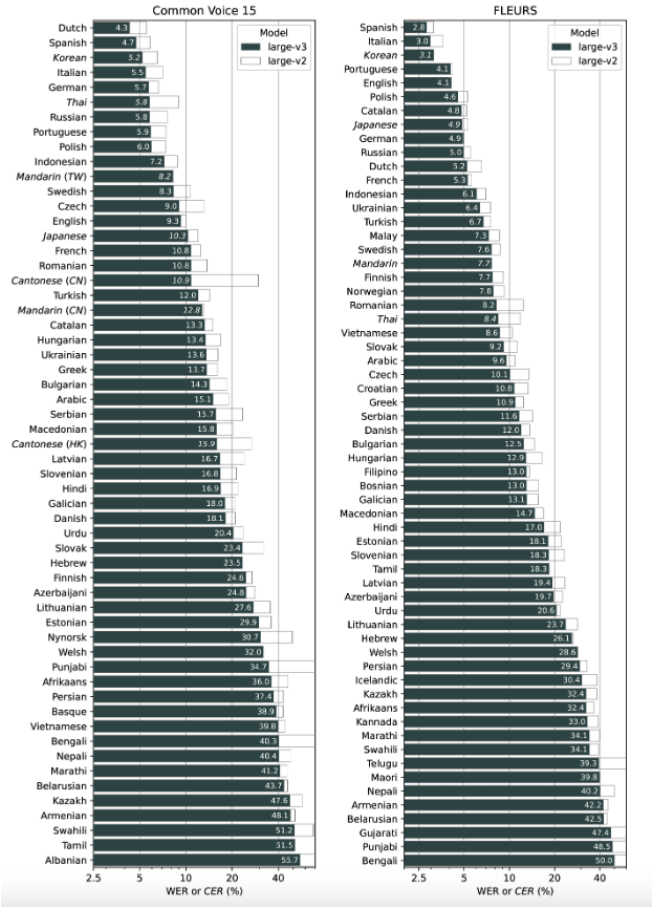


Figure: Performance of Whisper models on multiple languages

On par with human transcription ?

Read Speech			
Test set	DS1	DS2	Human
WSJ eval'92	4.94	3.60	5.03
WSJ eval'93	6.94	4.98	8.08
LibriSpeech test-clean	7.89	5.33	5.83
LibriSpeech test-other	21.74	13.25	12.69

Figure: Comparison of WER for two speech systems and human level performance on **read** speech (from (Amodei et al., 2016)

Accented Speech			
Test set	DS1	DS2	Human
VoxForge American-Canadian	15.01	7.55	4.85
VoxForge Commonwealth	28.46	13.56	8.15
VoxForge European	31.20	17.55	12.76
VoxForge Indian	45.35	22.44	22.15

Figure: Comparison of WER for two speech systems and human level performance on **accented** speech (from (Amodei et al., 2016)

On par with human transcription ?

Noisy Speech			
Test set	DS1	DS2	Human
CHiME eval clean	6.30	3.34	3.46
CHiME eval real	67.94	21.79	11.84
CHiME eval sim	80.27	45.05	31.33

Figure: Comparison of WER for two speech systems and human level performance on **noisy** speech (from (Amodei et al., 2016)

Zero resource ASR

In an unknown language, from unannotated raw speech, discover:[10]

- Invariant subword units (phone units ?)
- Words/terms (lexicon/semantic units ?)

Technological challenge

- Can we build useful speech technologies without any textual resources ?
- Unsupervised ASR / autonomous systems

Scientific challenge

- Can we build algorithms that learn languages like infants do ?
- Can we build algorithms that extract meaningful units from unknown languages ?

[10] The zero resource challenge: `http://zerospeech.com` (Dunbar et al., 2017)

Resources

- Follow me on Bluesky: @lbesacier.bsky.social
- Recent thread on Moshi (on Twitter): Recent Thread on Moshi
- Blog on Multimodal Speech-Text Models: Read the Blog Post
- Research on Multimodal NLP at Naver Labs Europe: Read Naver Labs Research Page on Multimodal NLP
- **Collab Notebook with a few simulations (ASR, speech representations, ...)** [Notebook]

Questions?

Thank you

References I

Alec Radford, Jong Wook Kim, T. X. G. B. C. M. I. S. (2022). Robust speech recognition via large-scale weak supervision.

Amodei, D., Anubhai, R., Battenberg, E., Case, C., Casper, J., Catanzaro, B., Chen, J., Chrzanowski, M., Coates, A., Diamos, G., Elsen, E., Engel, J., Fan, L., Fougner, C., Hannun, A. Y., Jun, B., Han, T., LeGresley, P., Li, X., Lin, L., Narang, S., Ng, A. Y., Ozair, S., Prenger, R., Qian, S., Raiman, J., Satheesh, S., Seetapun, D., Sengupta, S., Wang, C., Wang, Y., Wang, Z., Xiao, B., Xie, Y., Yogatama, D., Zhan, J., and Zhu, Z. (2016). Deep speech 2 : End-to-end speech recognition in english and mandarin. In Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016, pages 173–182.

Ao, J., Wang, R., Zhou, L., Liu, S., Ren, S., Wu, Y., Ko, T., Li, Q., Zhang, Y., Wei, Z., et al. (2021). Speecht5: Unified-modal encoder-decoder pre-training for spoken language processing. arXiv preprint arXiv:2110.07205.

Babu, A., Wang, C., Tjandra, A., Lakhotia, K., Xu, Q., Goyal, N., Singh, K., von Platen, P., Saraf, Y., Pino, J., Baevski, A., Conneau, A., and Auli, M. (2021). Xls-r: Self-supervised cross-lingual speech representation learning at scale. arXiv, abs/2111.09296.

Baevski, A., Auli, M., and Mohamed, A. (2019). Effectiveness of self-supervised pre-training for speech recognition.

References II

Baevski, A., Zhou, Y., Mohamed, A., and Auli, M. (2020). wav2vec 2.0: A framework for self-supervised learning of speech representations. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H., editors, Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual.

Bahdanau, D., Cho, K., and Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. CoRR, abs/1409.0473.

Bapna, A., Cherry, C., Zhang, Y., Jia, Y., Johnson, M., Cheng, Y., Khanuja, S., Riesa, J., and Conneau, A. (2022). mslam: Massively multilingual joint pre-training for speech and text. CoRR, abs/2202.01374.

Bapna, A., Chung, Y., Wu, N., Gulati, A., Jia, Y., Clark, J. H., Johnson, M., Riesa, J., Conneau, A., and Zhang, Y. (2021). SLAM: A unified encoder for speech and language modeling via speech-text joint pre-training. CoRR, abs/2110.10329.

Bengio, Y., Ducharme, R., Vincent, P., and Janvin, C. (2003). A neural probabilistic language model. J. Mach. Learn. Res., 3:1137–1155.

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. (2020). A simple framework for contrastive learning of visual representations.

Chorowski, J., Bahdanau, D., Serdyuk, D., Cho, K., and Bengio, Y. (2015). Attention-based models for speech recognition. CoRR, abs/1506.07503.

References III

Chung, Y., Hsu, W., Tang, H., and Glass, J. R. (2019). An unsupervised autoregressive model for speech representation learning. CoRR, abs/1904.03240.

Chung, Y.-A. and Glass, J. (2020). Improved speech representations with multi-target autoregressive predictive coding.

Devlin, J., Chang, M., Lee, K., and Toutanova, K. (2018). BERT: pre-training of deep bidirectional transformers for language understanding. CoRR, abs/1810.04805.

Dunbar, E., Cao, X., Benjumea, J., Karadayi, J., Bernard, M., Besacier, L., Anguera, X., and Dupoux, E. (2017). The zero resource speech challenge 2017. CoRR, abs/1712.04313.

Défossez, A., Mazaré, L., Orsini, M., Royer, A., Pérez, P., Jégou, H., Grave, E., and Zeghidour, N. (2024). Moshi: a speech-text foundation model for real-time dialogue.

Graves, A., Fernández, S., Gomez, F. J., and Schmidhuber, J. (2006). Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In ICML, volume 148 of ACM International Conference Proceeding Series, pages 369–376. ACM.

Gulati, A., Qin, J., Chiu, C., Parmar, N., Zhang, Y., Yu, J., Han, W., Wang, S., Zhang, Z., Wu, Y., and Pang, R. (2020). Conformer: Convolution-augmented transformer for speech recognition. In Meng, H., Xu, B., and Zheng, T. F., editors, Interspeech 2020, 21st Annual Conference of the International Speech Communication Association, Virtual Event, Shanghai, China, 25-29 October 2020, pages 5036–5040. ISCA.

References IV

Hinton, G. and Salakhutdinov, R. (2006). Reducing the dimensionality of data with neural networks. Science, 313(5786):504 – 507.

Kahn, J., Rivière, M., Zheng, W., Kharitonov, E., Xu, Q., Mazaré, P.-E., Karadayi, J., Liptchinsky, V., Collobert, R., Fuegen, C., Likhomanenko, T., Synnaeve, G., Joulin, A., Mohamed, A., and Dupoux, E. (2019). Libri-light: A benchmark for asr with limited or no supervision.

Kingma, D. P. and Welling, M. (2013). Auto-encoding variational bayes.

Le, H. S., Oparin, I., Allauzen, A., Gauvain, J., and Yvon, F. (2011). Structured output layer neural network language model. In Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP 2011, May 22-27, 2011, Prague Congress Center, Prague, Czech Republic, pages 5524–5527.

Liu, A. T., Yang, S.-w., Chi, P.-H., Hsu, P.-c., and Lee, H.-y. (2020). Mockingjay: Unsupervised speech representation learning with deep bidirectional transformer encoders. ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP).

Mikolov, T., Karafiát, M., Burget, L., Černocký, J., and Khudanpur, S. (2010). Recurrent neural network based language model. In Interspeech.

Mnih, A. and Hinton, G. (2008). A scalable hierarchical distributed language model. In In NIPS.

Morin, F. and Bengio, Y. (2005). Hierarchical probabilistic neural network language model. In AISTATS'05, pages 246–252.

References V

Panayotov, V., Chen, G., Povey, D., and Khudanpur, S. (2015). Librispeech: An ASR corpus based on public domain audio books. In ICASSP, pages 5206–5210. IEEE.

Park, D. S., Chan, W., Zhang, Y., Chiu, C.-C., Zoph, B., Cubuk, E. D., and Le, Q. V. (2019). Specaugment: A simple data augmentation method for automatic speech recognition. Interspeech 2019.

Pascual, S., Ravanelli, M., Serrà, J., Bonafonte, A., and Bengio, Y. (2019). Learning problem-agnostic speech representations from multiple self-supervised tasks. CoRR, abs/1904.03416.

Povey, D., Ghoshal, A., Boulianne, G., Burget, L., Glembek, O., Goel, N., Hannemann, M., Motlicek, P., Qian, Y., Schwarz, P., Silovsky, J., Stemmer, G., and Vesely, K. (2011). The kaldi speech recognition toolkit. In IEEE 2011 Workshop on Automatic Speech Recognition and Understanding. IEEE Signal Processing Society. IEEE Catalog No.: CFP11SRW-USB.

Radford, A., Kim, J. W., Xu, T., Brockman, G., McLeavey, C., and Sutskever, I. (2022). Robust speech recognition via large-scale weak supervision.

Ravanelli, M. and Bengio, Y. (2018). Speaker recognition from raw waveform with sincnet.

Ravanelli, M., Zhong, J., Pascual, S., Swietojanski, P., Monteiro, J., Trmal, J., and Bengio, Y. (2020). Multi-task self-supervised learning for robust speech recognition.

References VI

Renals, S., Morgan, N., Bourlard, H., Cohen, M., and Franco, H. (1994). Connectionist probability estimators in HMM speech recognition. IEEE Trans. Speech and Audio Processing, 2(1):161–174.

Schneider, S., Baevski, A., Collobert, R., and Auli, M. (2019a). wav2vec: Unsupervised Pre-Training for Speech Recognition. In Proc. Interspeech 2019, pages 3465–3469.

Schneider, S., Baevski, A., Collobert, R., and Auli, M. (2019b). wav2vec: Unsupervised pre-training for speech recognition. CoRR, abs/1904.05862.

Schwenk, H. (2007). Continuous space language models. Computer Speech & Language, 21(3):492–518.

Song, X., Wang, G., Wu, Z., Huang, Y., Su, D., Yu, D., and Meng, H. (2019). Speech-xlnet: Unsupervised acoustic model pretraining for self-attention networks.

van den Oord, A., Li, Y., and Vinyals, O. (2018). Representation learning with contrastive predictive coding. CoRR, abs/1807.03748.

van den Oord, A., Vinyals, O., and kavukcuoglu, k. (2017). Neural discrete representation learning. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R., editors, Advances in Neural Information Processing Systems 30, pages 6306–6315. Curran Associates, Inc.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. (2017). Attention is all you need. CoRR, abs/1706.03762.

References VII

Vincent, P., Larochelle, H., Bengio, Y., and Manzagol, P.-A. (2008). Extracting and composing robust features with denoising autoencoders.

Wang, W., Tang, Q., and Livescu, K. (2020). Unsupervised pre-training of bidirectional speech encoders via masked reconstruction.