AI in Speech Recognition and Voice Control

Iván López-Espejo

Open Day for Artificial Intelligence (ODAI'24)
Université Antonine

iloes@ugr.es

Thursday 18th April, 2024

Université Antonine

UNIVERSIDAD DE GRANADA

Overview

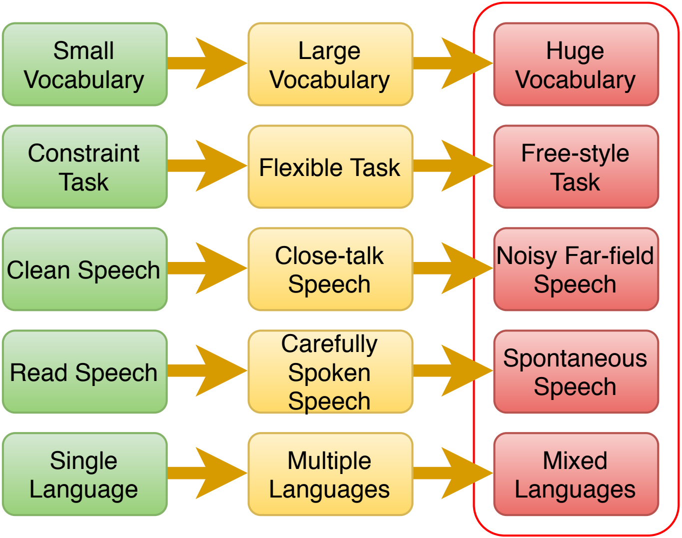
Introduction

- Upswing of **automatic speech recognition** (**ASR**) over the last decade: *deep learning has revolutionized ASR!*
  1. Availability of a huge amount of speech data
  2. Powerful computational resources (**GPUs**)

[1] IWSDS, `http://www.iwsds.org/`

- Tons of applications:
  1. Search-by-voice, voice assistants, gaming, dictation, in-vehicle systems...
  2. Low-resource **keyword spotting** (**KWS**) for hearing assistive devices ([2] I. López-Espejo *et al.*, "Improved External Speaker-Robust Keyword Spotting for Hearing Assistive Devices". IEEE/ACM Transactions on Audio, Speech, and Language Processing, 2020)

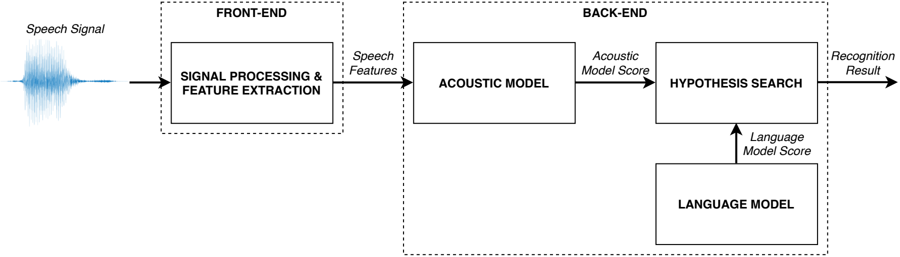
Introduction



[3] D. Yu and L. Deng, "Automatic Speech Recognition: A Deep Learning Approach". Springer, 2015

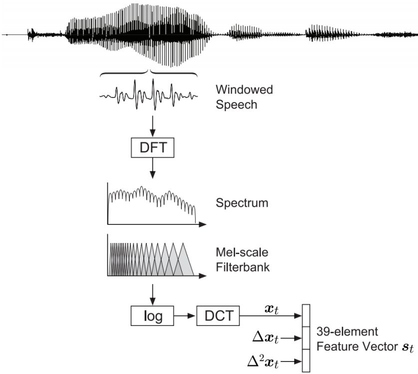
ASR Overview

- Architecture overview of an ASR system:



- Basic ASR components:
  1. *Signal processing & feature extraction*: log-Mel spectra, Mel-frequency cepstral coefficients (**MFCCs**)...
  2. *Acoustic model (**AM**)*: it integrates knowledge about acoustics and phonetics
  3. *Language model (**LM**)*: it estimates the probability of a hypothesized word sequence (LM score) by learning the correlation among words from text corpora
  4. *Hypothesis search*: it outputs the word sequence with the highest score as the recognition result

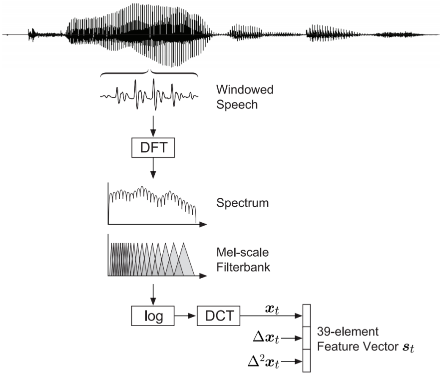
ASR Overview: Front-end



[4] H. Liao, "Uncertainty Decoding for Noise Robust Speech Recognition". Ph.D. thesis (University of Cambridge), 2007

- Desirable properties of speech features: **discriminative**, **compact** and **robust** to acoustic distortions (e.g., ambient/background noise)
- Depending on acoustic modeling...
  1. Gaussian mixture models (**GMMs**): use of coefficient derivatives
  2. Deep neural networks (**DNNs**): use of temporal context

ASR Overview: Front-end

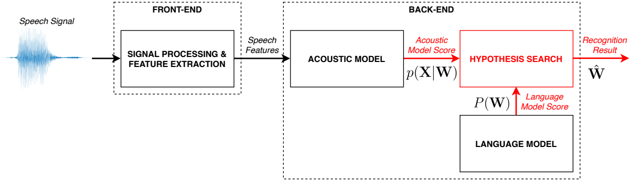


[4] H. Liao, "Uncertainty Decoding for Noise Robust Speech Recognition". Ph.D. thesis (University of Cambridge), 2007

- MFCCs better fit GMM-based acoustic models (diagonal covariance matrices, less complexity)
- Log-Mel spectra better fit DNN-based acoustic models (exploitation of spectro-temporal correlations)

ASR Overview: Back-end

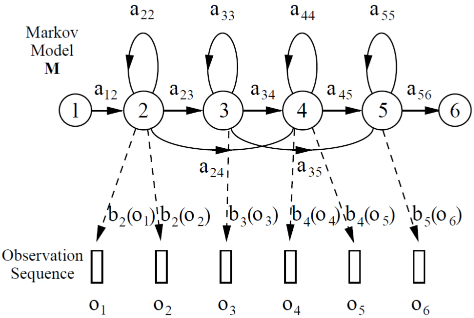
- The goal in ASR is to find the most likely sequence of words $\mathbf{W} = (w_1, w_2, ..., w_m)$ from a set of feature vectors $\mathbf{X} = (\mathbf{x}_1, ..., \mathbf{x}_T)$
- Maximum a posteriori (MAP) estimation problem: $\hat{\mathbf{W}} = \arg\max_{\mathbf{W}} P(\mathbf{W}|\mathbf{X}) = \arg\max_{\mathbf{W}} p(\mathbf{X}|\mathbf{W})P(\mathbf{W})$
- The **Viterbi algorithm** allows us the decoding of $\mathbf{W}$ from the observations $\mathbf{X}$



- To find out $p(\mathbf{X}|\mathbf{W})$, we require both the lexicon (i.e., the mapping between the written words that can be recognized and the word phonetic transcriptions) and the acoustic model

ASR Overview: Back-end

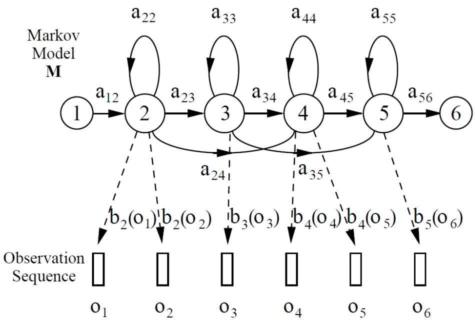
- The **acoustic model** is responsible for providing $p(\mathbf{X}|\mathbf{W})$
- Every word $w_i \in \mathbf{W}$, $i = 1, ..., m$, is normally decomposed into simpler acoustic units (i.e., monophones or triphones) from the lexicon
- Each of these acoustic units is modeled by a **hidden Markov model** (**HMM**) with continuous density functions (*variable speed of speech*)
- *Remember*: HMM parameters are obtained by maximum likelihood estimation using the **Baum-Welch** (EM) algorithm



[5] S. Young *et al.*, "The HTK Book (for HTK Version 3.4)". Cambridge University Engineering Department, 2006

ASR Overview: Back-end

- Each distribution $b_j(\mathbf{o} = \mathbf{x}_t)$ expresses the probability that the feature vector $\mathbf{x}_t$ is observed at state s_j
- Modeling of the output observation distributions of the HMM states
  1. Using GMMs: $b_j(\mathbf{x}_t|s_j) = \sum_{k=1}^{\mathcal{K}} P(k|s_j)\mathcal{N}\left(\mathbf{x}_t \middle| \boldsymbol{\mu}_{s_j}^{(k)}, \boldsymbol{\Sigma}_{s_j}^{(k)}\right)$
  2. *It is much better to use* **DNNs** *to produce the state emission likelihoods!*



[5] S. Young *et al.*, "The HTK Book (for HTK Version 3.4)". Cambridge University Engineering Department, 2006

ASR Overview: Back-end

- $p(\mathbf{X}|\mathbf{W})$ can be calculated by summing over all the possible state sequences $\mathbf{q} = (q_1, ..., q_T)$ that can produce $\mathbf{W}$:
  $p(\mathbf{X}|\mathbf{W}) = \sum_{\mathbf{q}} \prod_{t=1}^{T} p(\mathbf{x}_t|q_t) P(q_t|q_{t-1})$

- $P(\mathbf{W})$ depends on the linguistic task. Under the N-gram approach (usually, $N = 2$ or $N = 3$): $P(\mathbf{W}) = \prod_{i=1}^{m} P(w_i|w_{i-1}, ..., w_{i-N+1})$

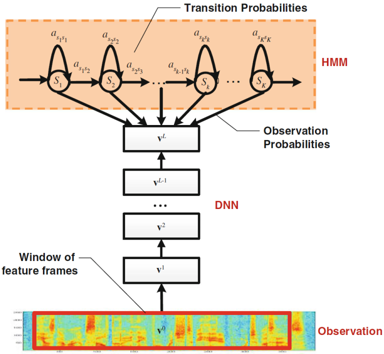
- From N-gram to connectionist approaches: recurrent neural networks (**RNNs**) are widely used to fit a probabilistic model to compute $P(\mathbf{W})$

- Macromodel λ integrates the acoustic and language models

- The optimal state sequence $\hat{\mathbf{q}}$ from which $\mathbf{W}$ is recovered is estimated by the Viterbi algorithm:
  $\hat{\mathbf{q}} = \arg\max_{\mathbf{q}} p(\mathbf{q}, \mathbf{X}|\lambda)$

DNN-HMM Hybrid ASR

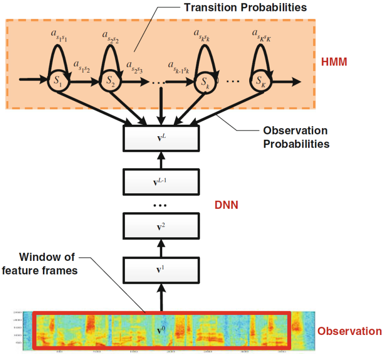
- Context-dependent (CD) DNN-HMM systems significantly outperform classical GMM-HMM systems on many large-vocabulary continuous speech recognition (**LVCSR**) tasks:
  - The output units of the DNN are senones (i.e., tied triphone states) instead of monophone states



[3] D. Yu and L. Deng, "Automatic Speech Recognition: A Deep Learning Approach". Springer, 2015

DNN-HMM Hybrid ASR

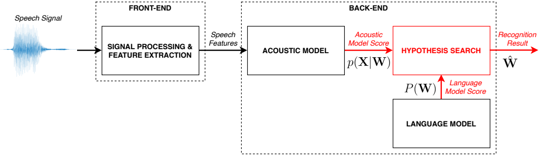
- *Important!* We do **not** use one DNN per state: a single DNN is trained to estimate the conditional state posterior probability $p(q_t = s_j | \mathbf{x}_t)$ for all states $\{s_j;\ j = 1, ..., S\}$



[3] D. Yu and L. Deng, "Automatic Speech Recognition: A Deep Learning Approach". Springer, 2015

DNN-HMM Hybrid ASR

- **Decoding in DNN-HMM ASR systems:**
  - As for GMM-HMM ASR, $\hat{\mathbf{W}} = \arg\max_{\mathbf{W}} p(\mathbf{X}|\mathbf{W})P(\mathbf{W})$, where the AM score is $p(\mathbf{X}|\mathbf{W}) = \sum_{\mathbf{q}} \prod_{t=1}^{T} p(\mathbf{x}_t|q_t)P(q_t|q_{t-1})$
  - In DNN-HMM ASR,
    $$p(\mathbf{x}_t|q_t = s_j) = \frac{p(q_t = s_j|\mathbf{x}_t)P(\mathbf{x}_t)}{P(s_j)} \Rightarrow \bar{p}(\mathbf{x}_t|q_t = s_j) = \frac{p(q_t = s_j|\mathbf{x}_t)}{P(s_j)}$$
  - The prior probability of each senone, $P(s_j) = T_{s_j}/T$, is estimated from the training set
  - $p(q_t = s_j|\mathbf{x}_t)$ is given by the DNN!



DNN-HMM Hybrid ASR

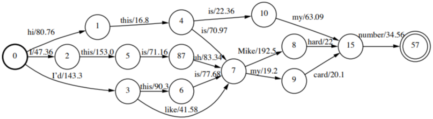
- **Training DNN-HMM ASR systems (I):**
  - Embedded Viterbi algorithm ($\mathbb{S}$ is the training set):
    1. *hmm0* $\leftarrow$ TrainCD-GMM-HMM($\mathbb{S}$);
    2. *stateAlignment* $\leftarrow$ ForcedAlignmentWithGMMHMM($\mathbb{S}$,*hmm0*);
    3. *stateToSenoneIDMap* $\leftarrow$ GenerateStateToSenoneIDMap(*hmm0*);
    4. *featureSenoneIDPairs* $\leftarrow$ GenerateDNNTrainingSet(*stateToSenoneIDMap*,*stateAlignment*);
    5. *ptdnn* $\leftarrow$ PretrainDNN($\mathbb{S}$);
    6. *hmm* $\leftarrow$ ConvertGMMHMMToDNNHMM(*hmm0*,*stateToSenoneIDMap*);
    7. *prior* $\leftarrow$ EstimatePriorProbability(*featureSenoneIDPairs*);
    8. *dnn* $\leftarrow$ Backpropagate(*ptdnn*,*featureSenoneIDPairs*);
    9. Return *dnnhmm* $= \{$*dnn*,*hmm*,*prior*$\}$
  - The embedded Viterbi algorithm minimizes the average cross entropy for each speech utterance with T frames:

$$\mathcal{L}_{\mathrm{CE}}(\theta) = -\sum_{t=1}^{T} \log p(q_t|\mathbf{x}_t;\theta)$$

DNN-HMM Hybrid ASR

- **Training DNN-HMM ASR systems (II):**
  - The cross-entropy criterion treats each frame independently
  - Nevertheless, ASR is a sequence classification problem!
  - Sequence-discriminative training techniques:
    1. Maximum mutual information (**MMI**)
    2. Boosted maximum mutual information (**BMMI**)
    3. Minimum phone error (**MPE**)
    4. State minimum Bayes risk (**sMBR**)
  - **Example: MMI**
    - MMI aims at maximizing the mutual information between the distributions of the observation and word sequences (*highly correlated to minimizing the expected sentence error*)

$$\mathcal{J}_{\mathrm{MMI}}(\theta;\mathbb{S})=\sum_{u=1}^{\mathcal{U}}\log P\left(\mathbf{W}^u|\mathbf{X}^u;\theta\right)=\sum_{u=1}^{\mathcal{U}}\log\frac{p\left(\mathbf{X}^u|\mathbf{s}^u;\theta\right)^{\kappa}P(\mathbf{W}^u)}{\sum_{\mathbf{W}}p\left(\mathbf{X}^u|\mathbf{s}^w;\theta\right)^{\kappa}P(\mathbf{W})}$$



[6] M. Mohri, `https://cs.nyu.edu/~mohri/asr12/lecture_12.pdf`

DNN-HMM Hybrid ASR

- **Training DNN-HMM ASR systems (II):**
  - The cross-entropy criterion treats each frame independently
  - Nevertheless, ASR is a sequence classification problem!
  - Sequence-discriminative training techniques:
    1. Maximum mutual information (**MMI**)
    2. Boosted maximum mutual information (**BMMI**)
    3. Minimum phone error (**MPE**)
    4. State minimum Bayes risk (**sMBR**)

Training criterion	WER (%)
GMM-BMMI	18.6
DNN-CE	14.2
DNN-MMI	12.9
DNN-BMMI	12.9
DNN-MPE	12.9
DNN-sMBR	12.6

Word error rate (%) on the Switchboard dataset, [7] K. Veselỳ *et al.*, "Sequence-discriminative training of deep neural networks". In Proc. of Interspeech 2013

DNN-HMM Hybrid ASR: Key Issues

- Directly modeling context-dependent phone states (i.e., senones) is key (*overfitting alleviation*):

Model	Monophones	Senones
CD-GMM-HMM	—	23.6
CD-DNN-HMM (7x2k)	34.9	17.1

Word error rate (%) on the Switchboard dataset, [8] F. Seide *et al.*, "Conversational Speech Transcription Using Context-Dependent Deep Neural Networks". In Proc. of Interspeech 2011

- Deeper is better!

$L \times N$	WER (%)	$1 \times N$	WER (%)
1x2k	24.2	—	—
3x2k	18.4	—	—
5x2k	17.2	1x3,772	22.5
7x2k	17.1	1x4,634	22.6
—	—	1x16k	22.1

Word error rate (%) on the Switchboard dataset, [8]

- Use of temporal context:

Model	1 frame	11 frames
CD-DNN-HMM (7x2k)	23.2	17.1

Word error rate (%) on the Switchboard dataset, [8]

Robust ASR

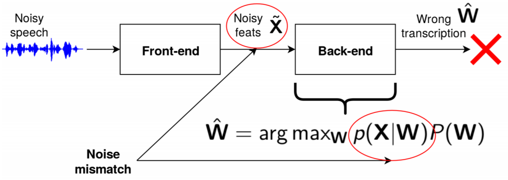
- Gap in performance between humans and machines due to mismatch between the training and testing conditions of ASR systems:
  1. **Speaker variabilities:** intra- (mood, illness...) and inter-speaker (vocal tract length, tone...) variability
  2. **Environment variabilities:** background noise, reverberation...
- Speaker variability compensation:
  1. Vocal tract length normalization (**VTLN**)
  2. Feature-space maximum likelihood linear regression (**fMLLR**)

Model	No compensation	VTLN	fMLLR
CD-GMM-HMM	23.6	21.5	20.4
CD-MLP-HMM (1x2,048)	24.2	22.5	21.5
CD-DNN-HMM (7x2,048)	17.1	16.8	16.4

Word error rate (%) on the Switchboard dataset, [9] F. Seide *et al.*, "Feature engineering in context-dependent deep neural networks for conversational speech transcription". In Proc. of ASRU 2011

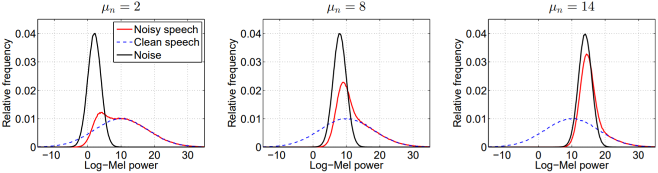
Robust ASR

- Environment variability compensation:
  - *Example*: acoustic models are trained with clean speech data and we try to recognize noisy speech data $\rightarrow$ mismatch will cause a wrong transcription



  - The statistical distribution of the speech energy is affected in the presence of background noise (when $\mathbf{h} = \mathbf{0}$):

$$y(m) = h(m) * x(m) + n(m) \rightarrow \mathbf{y} = \mathbf{x} + \mathbf{h} + \log\left(\mathbf{1} + \exp\{\mathbf{n} - \mathbf{x} - \mathbf{h}\}\right)$$

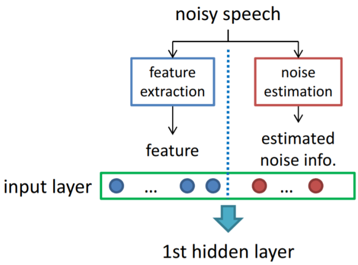


Robust ASR

- Environment variability compensation:
  - **THERE ARE PLENTY OF APPROACHES!**
  - **Feature-space approaches:** noise-robust features (RASTA-PLP, TANDEM...), normalization of feature statistical moments (CMN, HEQ...) and speech/feature enhancement (Wiener filtering, DNN-based speech enhancement, beamforming...)
  - **Model-based approaches:** model adaptation (CMLLR...) and adaptive training (fNAT, SAT...)
  - **Compensation with explicit distortion modeling:** model adaptation or feature compensation (VTS...)
  - **Missing data approaches:** ignoring unreliable elements during recognition (marginalization, SFD...) and data imputation (TGI...)
  - **...**

Robust ASR

- Environment variability compensation:
  - **Multi-condition training:** training the acoustic model with distorted speech data from different acoustic conditions (*very effective if we can cover rather all the test acoustic conditions!*)
  - **Noise-aware training (NAT):**



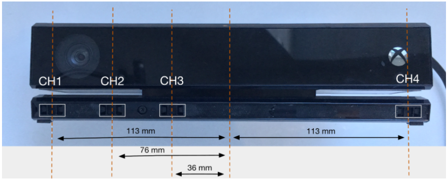
[10] A. Abe *et al.*, "Robust Speech Recognition using DNN-HMM Acoustic Model Combining Noise-aware Training with Spectral Subtraction". In Proc. of Interspeech 2015

DNN-HMM System (7x2,048)	WER (%)
Multi-condition (MC) training	13.4
MC+Feature enhancement	13.8
MC+NAT	13.1
MC+Dropout	12.9
MC+NAT+Dropout	12.4

Word error rate (%) on the Aurora-4 dataset, [11] M. Seltzer *et al.*, "An Investigation of Deep Neural Networks for Noise Robust Speech Recognition". In Proc. of ICASSP 2013

End-to-end ASR

- **End-to-end ASR:** a deep learning model is trained to directly map an input speech feature sequence to a sequence of characters/tokens
- End-to-end ASR systems are "simpler"/cleaner: there is no need for specific acoustic and language models with pronunciation lexicons
- **CHiME-6 Challenge:** distant microphone conversational speech recognition in everyday home environments (`https://chimechallenge.github.io/chime6/overview.html`)

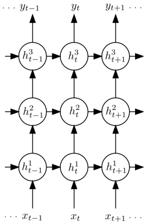


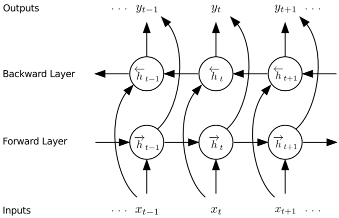
- In CHiME-6, DNN-HMM hybrid ASR systems still outperformed end-to-end ASR approaches (in 2020!)

End-to-end ASR: Basics

Recurrent neural networks (RNNs)

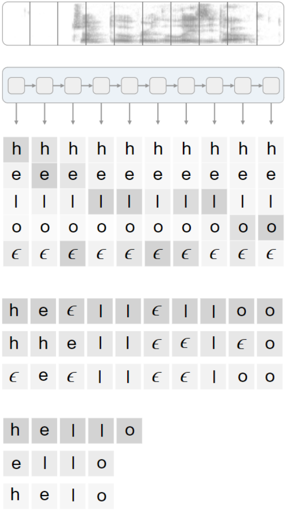
- Standard RNNs (general idea):
  $\mathbf{h}_t = \sigma\left(\mathbf{W}_{ih}\mathbf{x}_t + \mathbf{W}_{hh}\mathbf{h}_{t-1} + \mathbf{b}_h\right)$
  $\mathbf{y}_t = \mathbf{W}_{ho}\mathbf{h}_t + \mathbf{b}_o$
- Bidirectional RNNs:
  $\overrightarrow{\mathbf{h}}_t = \sigma\left(\mathbf{W}_{i\overrightarrow{h}}\mathbf{x}_t + \mathbf{W}_{\overrightarrow{h}\overrightarrow{h}}\overrightarrow{\mathbf{h}}_{t-1} + \mathbf{b}_{\overrightarrow{h}}\right)$
  $\overleftarrow{\mathbf{h}}_t = \sigma\left(\mathbf{W}_{i\overleftarrow{h}}\mathbf{x}_t + \mathbf{W}_{\overleftarrow{h}\overleftarrow{h}}\overleftarrow{\mathbf{h}}_{t+1} + \mathbf{b}_{\overleftarrow{h}}\right)$
  $\mathbf{y}_t = \mathbf{W}_{\overrightarrow{h}o}\overrightarrow{\mathbf{h}}_t + \mathbf{W}_{\overleftarrow{h}o}\overleftarrow{\mathbf{h}}_t + \mathbf{b}_o$
- Long short-term memory (**LSTM**), bidirectional LSTM (**BiLSTM**), gated recurrent units (**GRUs**)





[12] A. Graves and N. Jaitly, "Towards End-to-End Speech Recognition with Recurrent Neural Networks". In Proc. of ICML 2014

End-to-end ASR: CTC



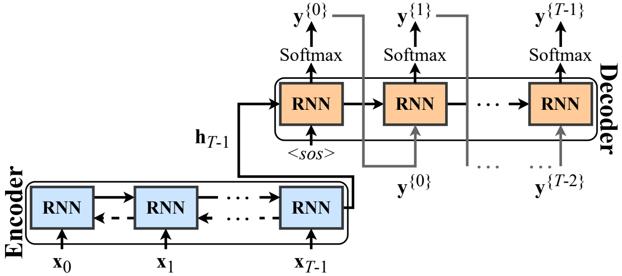
[13] A. Hannun, `https://distill.pub/2017/ctc/`

- Let $\mathbf{C} = (c_1, \ldots, c_m)$ be the sequence of characters/tokens corresponding to $\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_T)$
- We ignore an accurate alignment between $\mathbf{C}$ and $\mathbf{X}$, and $m < T$
- Connectionist temporal classification (**CTC**) is an *alignment-free* algorithm
- CTC introduces the so-called blank token (ϵ)
- **CTC objective:** maximizing $P(\mathbf{C}|\mathbf{X}) = \sum_{A \in \mathcal{A}_{X,C}} \prod_{t=1}^{T} P_t(\mathbf{c}|\mathbf{X})$ (e.g., $\mathbf{c} = \{\mathrm{h,e,l,o},\epsilon\}$)
- Decoding as usual, $\hat{\mathbf{C}} = \arg\max_{\mathbf{C}} P(\mathbf{C}|\mathbf{X})$

End-to-end ASR: Encoder-decoder Framework

Encoder-decoder framework

- The encoder is normally a BiLSTM, while the decoder, an LSTM:
  $\mathbf{h}_t = \text{Encoder}\left(\mathbf{x}_t, \mathbf{h}_{t-1}\right)$
  $\mathbf{s}_i = \text{Decoder}\left(\mathbf{s}_{i-1}, \mathbf{y}_{i-1}\right)$

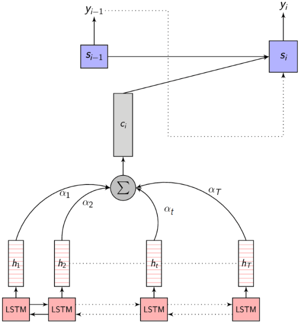


[14] I. López-Espejo *et al.*, "Deep Spoken Keyword Spotting: An Overview". IEEE Access, 2021

- **Potential issue:** the encoder needs to condense all the required information (regardless the length of the input sequence) into a fixed-dimensional vector

End-to-end ASR: Attention

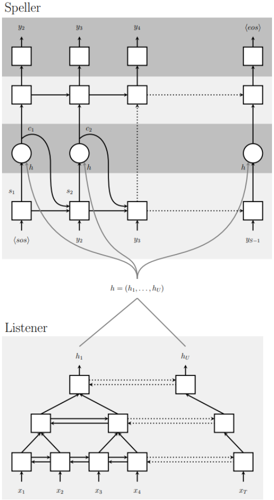
- We can **attend** to a context-relevant subset of $\{\mathbf{h}_1, ..., \mathbf{h}_T\}$ instead of $\mathbf{h}_T$ to "help" the decoder:
  $\mathbf{s}_i = \text{Decoder}\left(\mathbf{s}_{i-1}, \mathbf{y}_{i-1}, \mathbf{C}_i\right)$



[15] S. Nadig, `https://medium.com/intel-student-ambassadors/attention-in-end-to-end-automatic-speech-recognition-9f9e42718d21`

$\mathbf{C}_i = \sum_{t=1}^{T} \alpha_{it} \mathbf{h}_t$ $\qquad$ $\alpha_{it} = \text{softmax}\left(\text{AttentionFunction}\left(\mathbf{s}_{i-1}, \mathbf{h}_t\right)\right)$

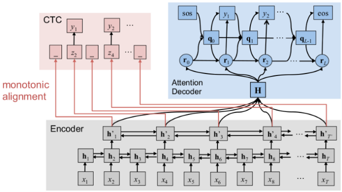
End-to-end ASR: Attention Encoder-decoder



[16] W. Chan *et al.*, "Listen, Attend and Spell: A Neural Network for Large Vocabulary Conversational Speech Recognition". In Proc. of ICASSP 2016

- *Problem with CTC*: conditional label independence at decoding $\rightarrow$ CTC needs an external language model to work well
- **Attention-based encoder-decoder ASR:** the alignment between **C** and **X** is learned using an attention mechanism $P(\mathbf{C}|\mathbf{X}) = \prod_{i=1}^{m} P(c_i|\mathbf{X}, c_1, \ldots, c_{i-1})$
- Attention-based encoder-decoder ASR is less robust to noise than CTC-based ASR $\rightarrow$ **CTC-attention ASR** has proven to be effective to improve recognition performance:

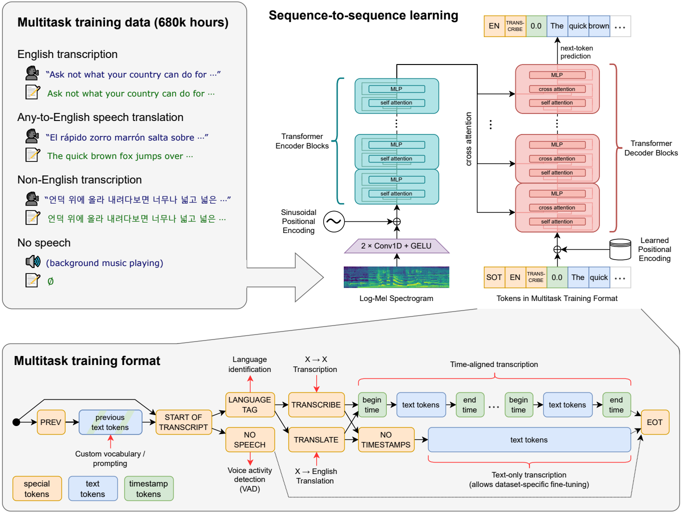
Multitask learning: $\mathcal{L}_{\mathrm{MTL}} = \lambda \mathcal{L}_{\mathrm{CTC}} + (1-\lambda)\mathcal{L}_{\mathrm{Attention}}$



CTC guides attention alignment to be monotonic

[15] S. Nadig, https://medium.com/intel-student-ambassadors/attention-in-end-to-end-automatic-speech-recognition-9f9e42718d21

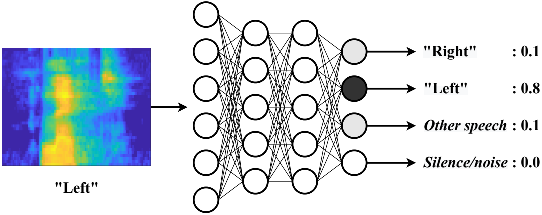
Whisper

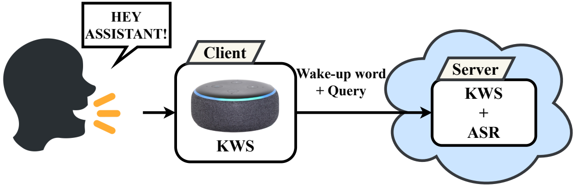


[17] A. Radford *et al.*, "Robust Speech Recognition via Large-Scale Weak Supervision". In Proc. of ICML 2023

Voice Control

Keyword Spotting Technology



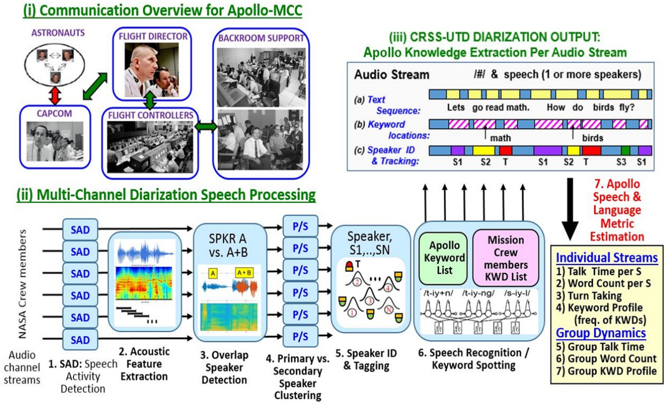


- Voice control is typically implemented through spoken **keyword spotting** (**KWS**)
- Spoken KWS can be defined as the task of identifying keywords in audio streams comprising speech

[14] I. López-Espejo *et al.*, "Deep Spoken Keyword Spotting: An Overview". IEEE Access, 2021

Applications and Ongoing Work

Topic Identification in NASA's Apollo Missions Audio



[18] A. Joglekar *et al.*, "Fearless Steps APOLLO: Challenges in keyword spotting and topic detection for naturalistic audio streams". The Journal of the Acoustical Society of America, 2023

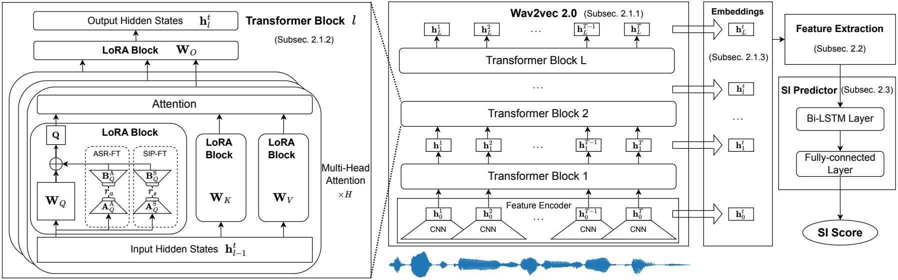
AGILE-KWS: A Giant Leap for Keyword Spotting

European Commission, Marie Curie Global Fellowships (HORIZON-MSCA-2021-PF-01)

UT DALLAS The University of Texas at Dallas

Applications and Ongoing Work

No-Reference Speech Intelligibility Prediction

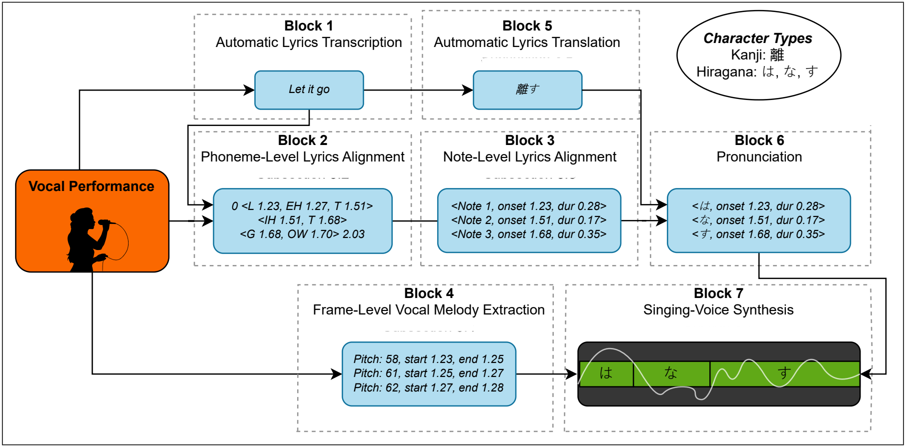


[19] H. Wang *et al.*, "No-Reference Speech Intelligibility Prediction Leveraging a Noisy-Speech ASR Pre-Trained Model". Submitted to Interspeech 2024

- **Self-supervised speech representation learning**
  1. Wav2vec 2.0 (🤗 https://huggingface.co/docs/transformers/model_doc/wav2vec2)
  2. HuBERT (🤗 https://huggingface.co/docs/transformers/model_doc/hubert)
  3. WavLM (🤗 https://huggingface.co/docs/transformers/model_doc/wavlm)

Applications and Ongoing Work

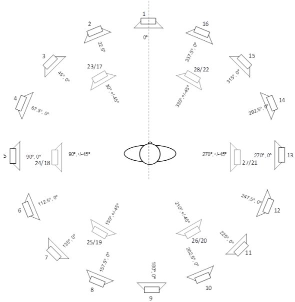
Singing-Voice to Singing-Voice Translation

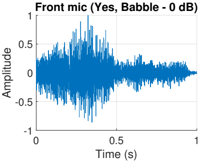


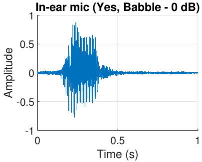
[20] S. Antonisen and I. López-Espejo, "PolySinger: Singing-Voice to Singing-Voice Translation from English to Japanese". Submitted to ISMIR 2024

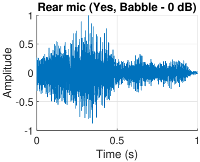
Applications and Ongoing Work

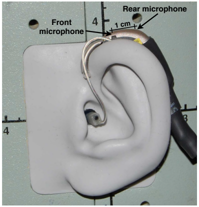
Noise-Robust Hearing Aid Voice Control











🤗 **Wanna collaborate?** Reach out to `iloes@ugr.es`

AI in Speech Recognition and Voice Control

Iván López-Espejo

Open Day for Artificial Intelligence (ODAI'24)
Université Antonine

iloes@ugr.es

Thursday 18^{th} April, 2024