Keyword spotting

Plan

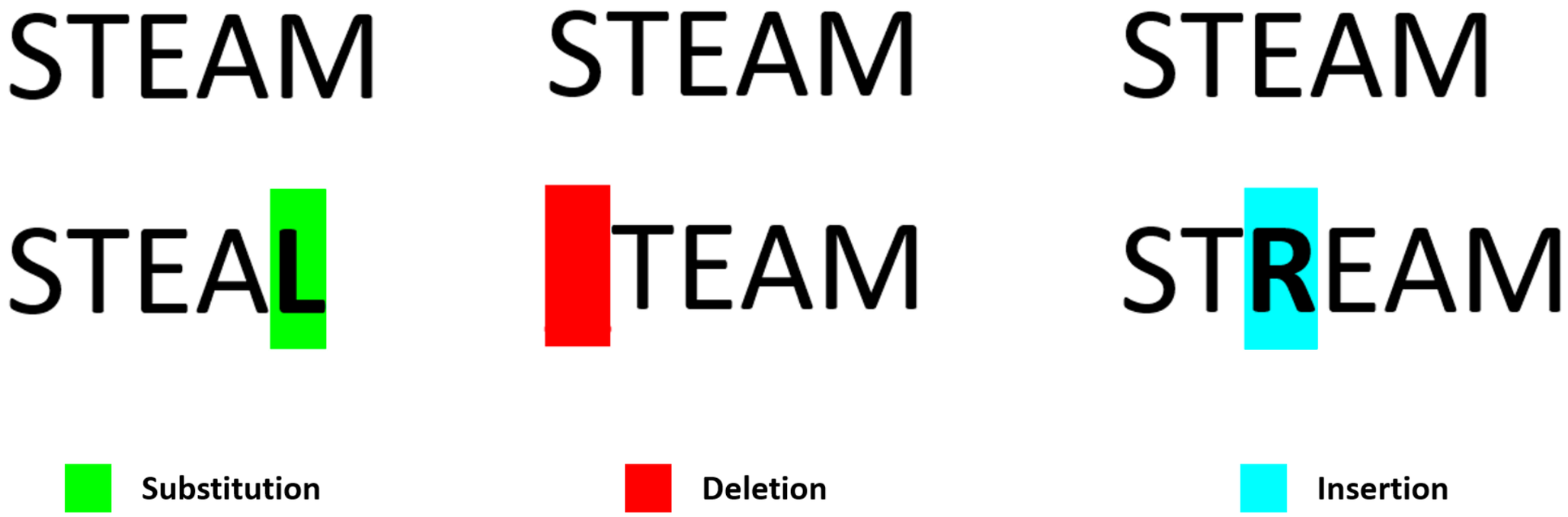
- ASR Metrics: CER, WER, SER
- Keyword spotting
- Spotter metrics: FA/h, FRR
- Neural KWS
- Modern KWS papers

ASR Metrics

We need to measure ASR quality.

Your suggestions?

Character Error Rate (CER)



CER

$$CER = \frac{S + D + I}{N}$$

Character Error Rate (CER) formula

where:

- **S** = Number of **S**ubstitutions
- **D** = Number of **D**eletions
- **I** = Number of **I**nsertions
- **N** = **N**umber of characters in reference text (aka ground truth)

Bonus Tip: The denominator N can alternatively be computed with:
***N** = **S** + **D** + **C** (where **C** = number of **correct** characters)*

WER

$$WER = \frac{S_w + D_w + I_w}{N_w}$$

For example:

- Ground Truth: **‘my name is kenneth’**
- OCR Output: **‘myy nime iz kenneth’**

From the above, the **CER** is **16.67%,** whereas the **WER** is **75%**. The WER value of 75% is clearly understood since 3 out of 4 words in the sentence were wrongly transcribed.

Levenshtein Distance

$$\operatorname{lev}_{a,b}(i,j) = \begin{cases} \max(i,j) & \text{if } \min(i,j) = 0, \\ \min \begin{cases} \operatorname{lev}_{a,b}(i-1,j) + 1 \\ \operatorname{lev}_{a,b}(i,j-1) + 1 \\ \operatorname{lev}_{a,b}(i-1,j-1) + 1_{(a_i \neq b_j)} \end{cases} & \text{otherwise.} \end{cases}$$

Levenshtein Distance

An example that features the comparison of “HONDA” and “HYUNDAI”,

		H	Y	U	N	D	A	I
	0	1	2	3	4	5	6	7
H	1	0	1	2	3	4	5	6
O	2	1	1	2	3	4	5	6
N	3	2	2	2	2	3	4	5
D	4	3	3	3	3	2	3	4
A	5	4	4	4	4	3	2	3

Following are two representations: Levenshtein distance between “HONDA” and “HYUNDAI” is 3.

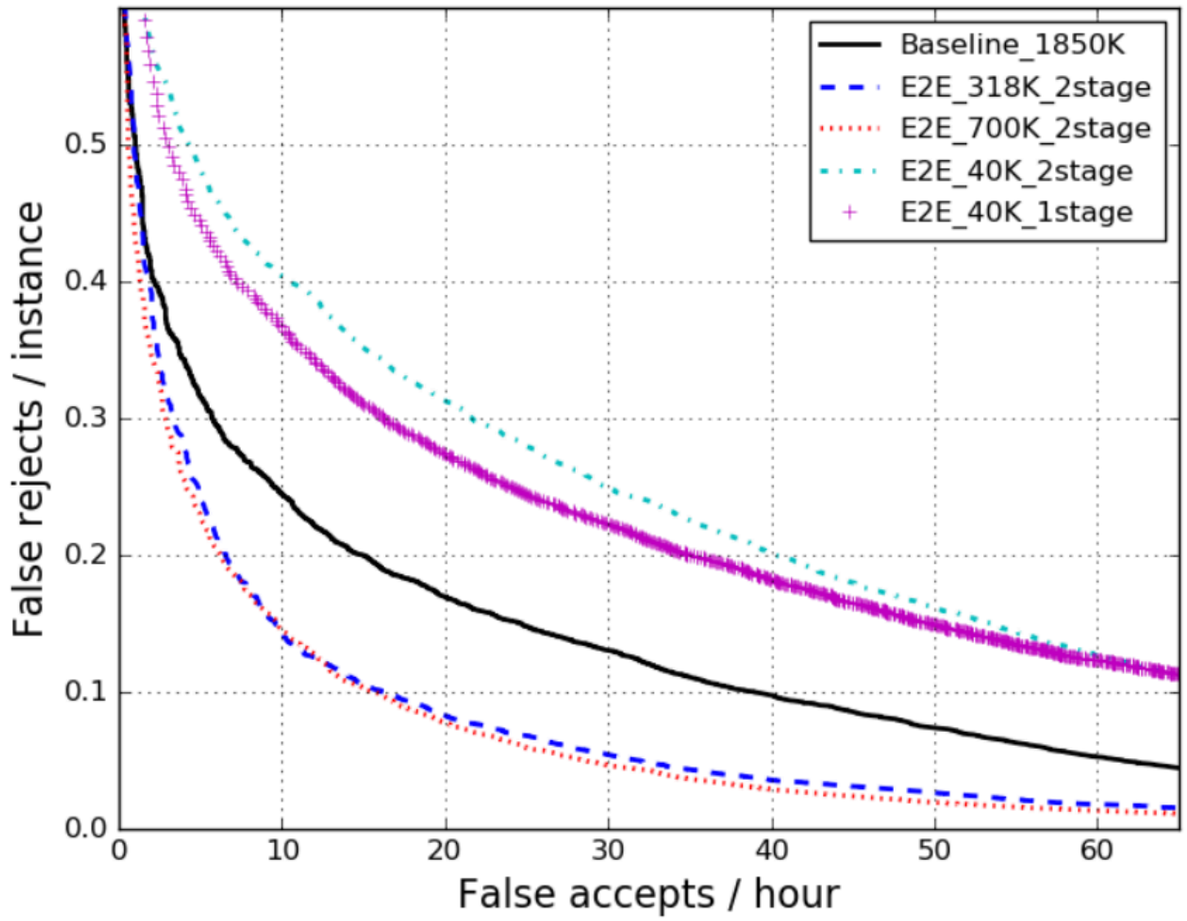
H	O		N	D	A	
H	Y	U	N	D	A	I

H	Y	U	N	D	A	I
H		O	N	D	A	

Keyword spotting

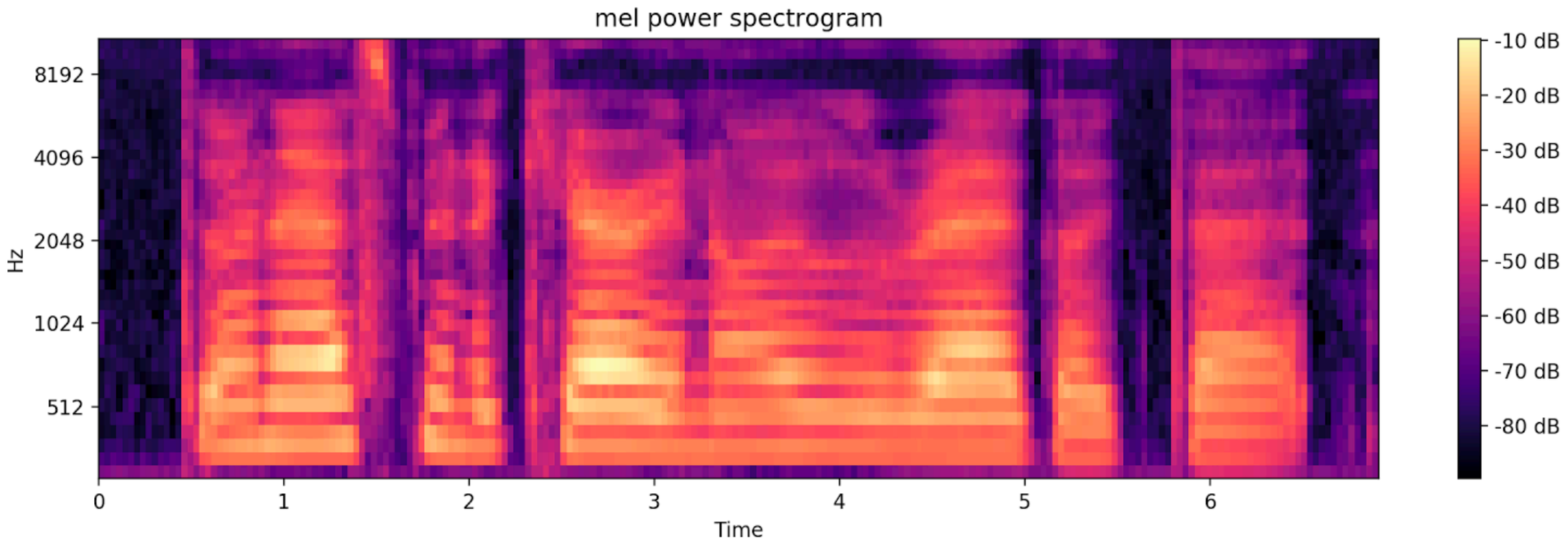
Spotter metrics

- FA/h: False activations hour
- FRR: False rejects rate

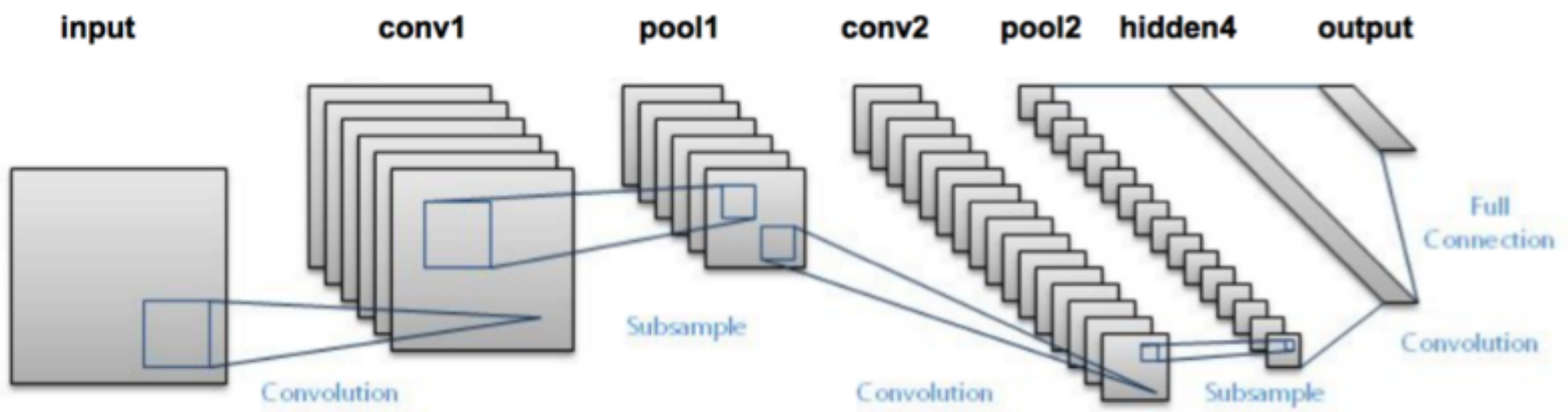


(d) Anonymous query logs

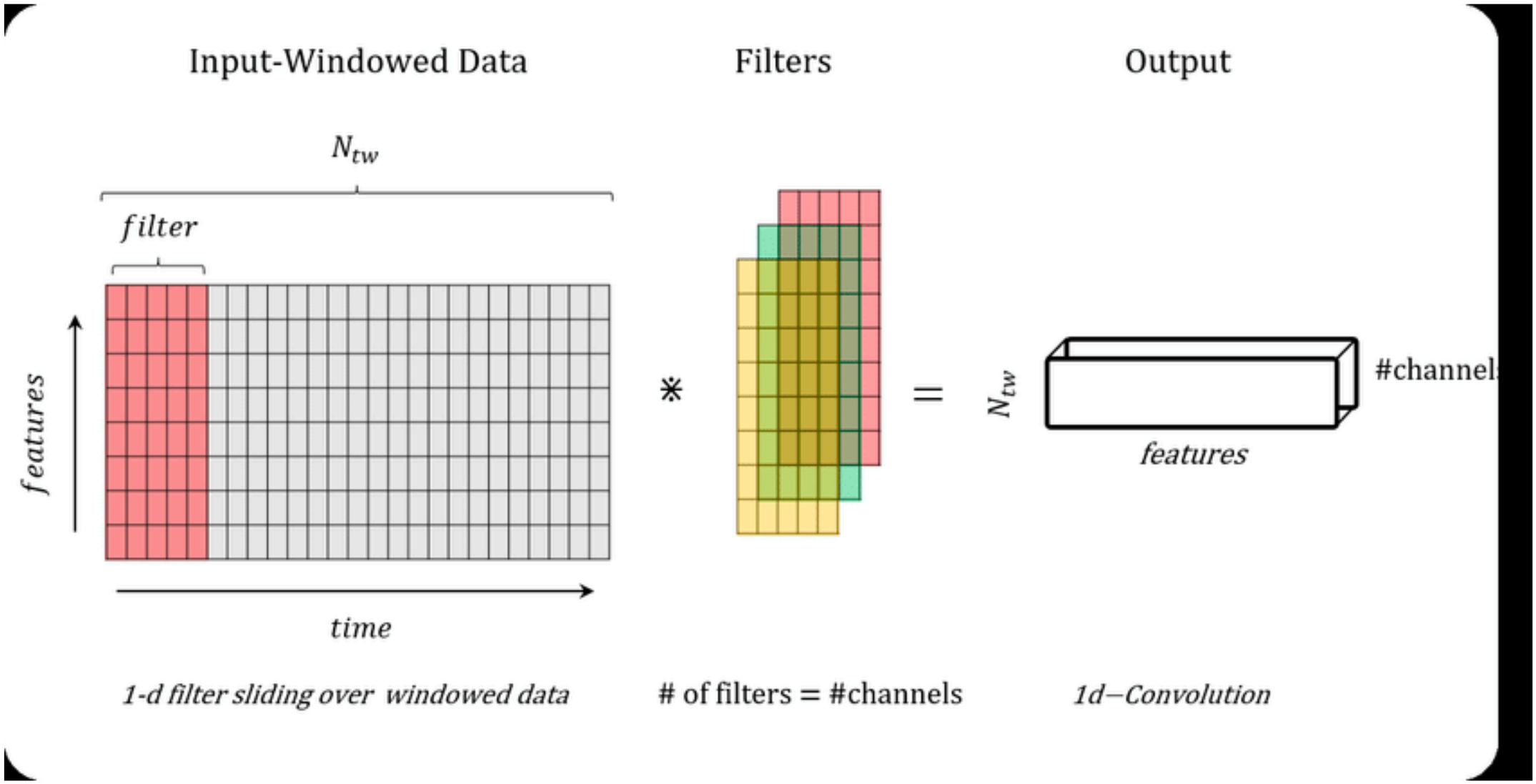
Log Mel Spectrogram



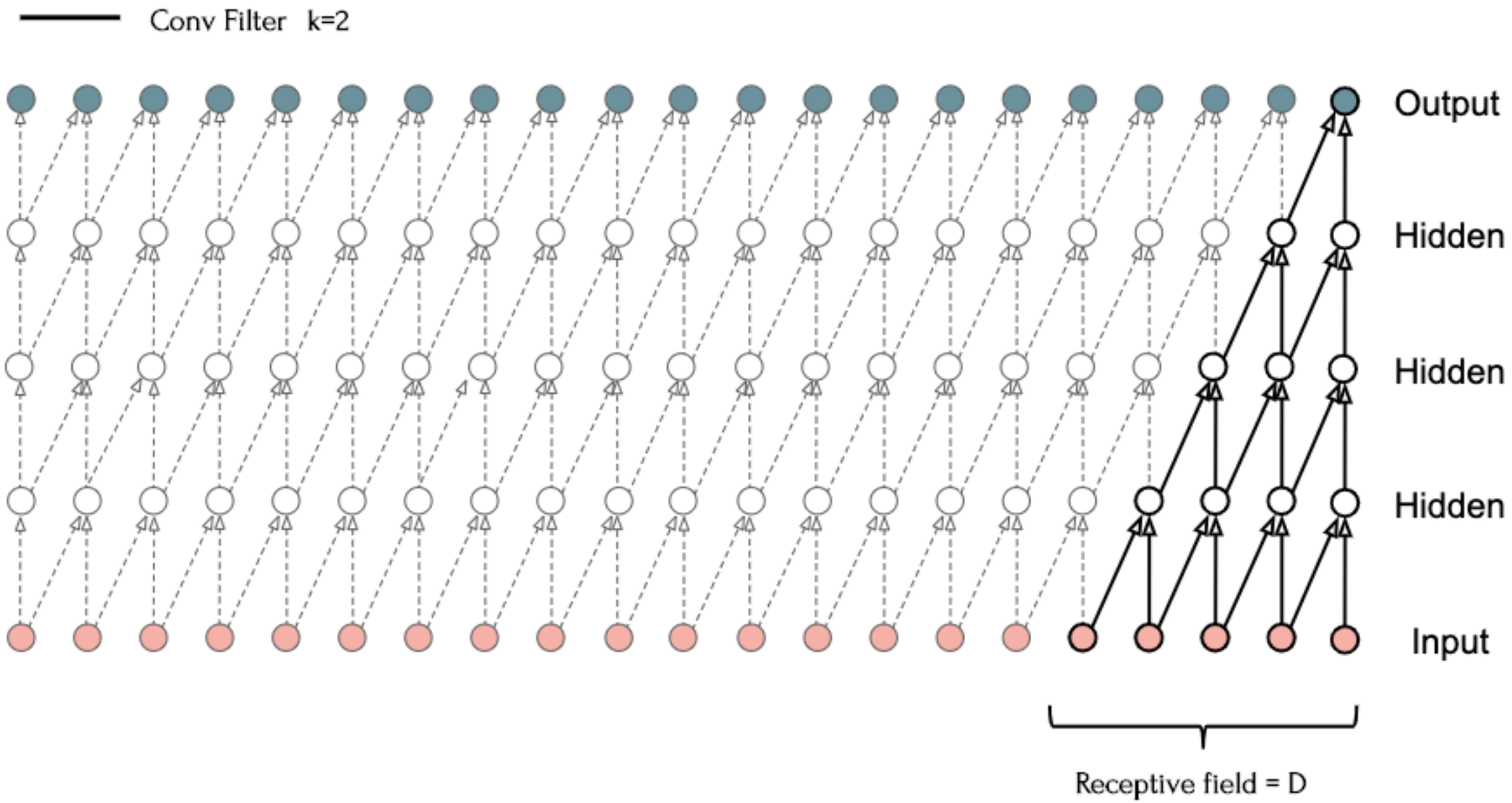
2D Convolutions



Temporal 1D Convolutions



Temporal 1D Convolutions



Deep Learning in KWS

SMALL-FOOTPRINT KEYWORD SPOTTING USING DEEP NEURAL NETWORKS

Guoguo Chen[*1] *Carolina Parada*[2] *Georg Heigold*[2]

[1] Center for Language and Speech Processing, Johns Hopkins University, Baltimore, MD
[2] Google Inc., Mountain View, CA
guoguo@jhu.edu carolinap@google.com heigold@google.com

Convolutional Neural Networks for Small-footprint Keyword Spotting

Tara N. Sainath, Carolina Parada

Google, Inc. New York, NY, U.S.A
{tsainath, carolinap}@google.com

Deep Learning in KWS

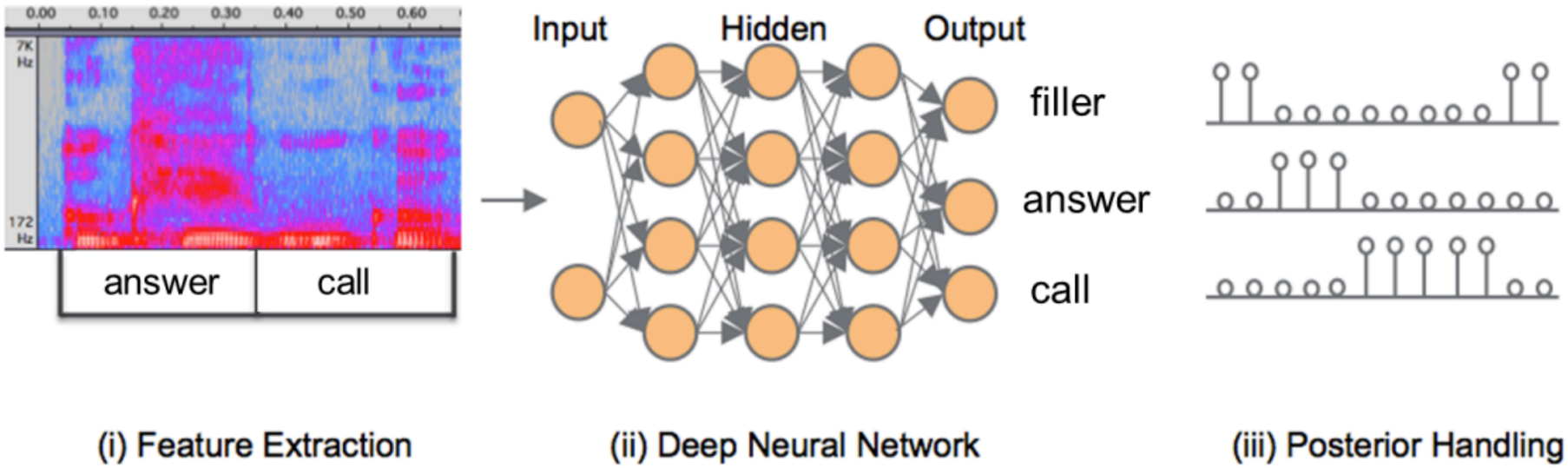


Figure 1: Framework of Deep KWS system, components from left to right: (i) Feature Extraction (ii) Deep Neural Network (iii) Posterior Handling

Posterior Handling

Posterior smoothing. Raw posteriors from the neural network are noisy, so we smooth the posteriors over a fixed time window of size w_{smooth}. Suppose p'_{ij} is the smoothed posterior of p_{ij} (Eq. 1). The smoothing is done with the following formula:

$$p'_{ij} = \frac{1}{j - h_{smooth} + 1} \sum_{k=h_{smooth}}^{j} p_{ik} \tag{2}$$

where $h_{smooth} = \max\{1, j - w_{smooth} + 1\}$ is the index of the first frame within the smoothing window.

Confidence. The confidence score at j^{th} frame is computed within a sliding window of size w_{max}, as follows

$$confidence = \sqrt[n-1]{\prod_{i=1}^{n-1} \max_{h_{max} \le k \le j} p'_{ik}} \tag{3}$$

where p'_{ij} is the smoothed state posterior in Eq. (2), $h_{max} = \max\{1, j - w_{max} + 1\}$ is the index of the first frame within the sliding window. We use $w_{smooth} = 30$ frames, and $w_{max} = 100$, as this gave best performance on the dev set; however the performance was not very sensitive to the window sizes. Eq. (3) does not enforce the order of the label sequence, however the stacked frames fed as input to the neural network help encode contextual information.

CRNN

Convolutional Recurrent Neural Networks for Small-Footprint Keyword Spotting

Sercan Ö. Arık[1,*], *Markus Kliegl*[1,*], *Rewon Child*[1], *Joel Hestness*[1], *Andrew Gibiansky*[1], *Chris Fougner*[1], *Ryan Prenger*[1], *Adam Coates*[1]

[1]Baidu Silicon Valley Artificial Intelligence Lab, 1195 Bordeaux Dr. Sunnyvale, CA 94089, USA
[*]Equal contribution
sercanarik@baidu.com, klieglmarkus@baidu.com

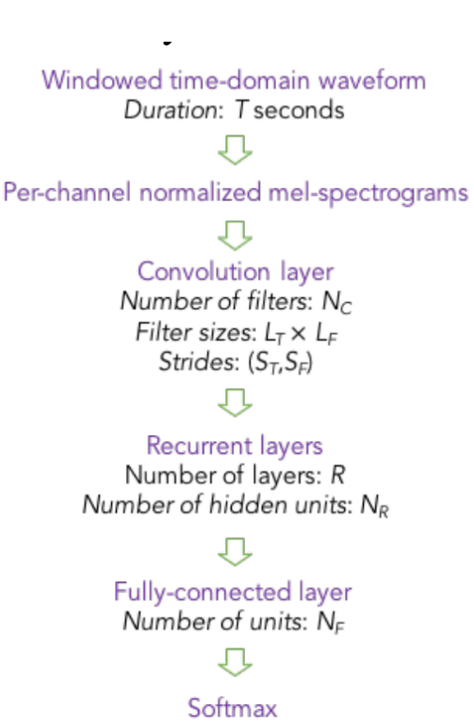


Figure 1: *End-to-end CRNN architecture for KWS.*

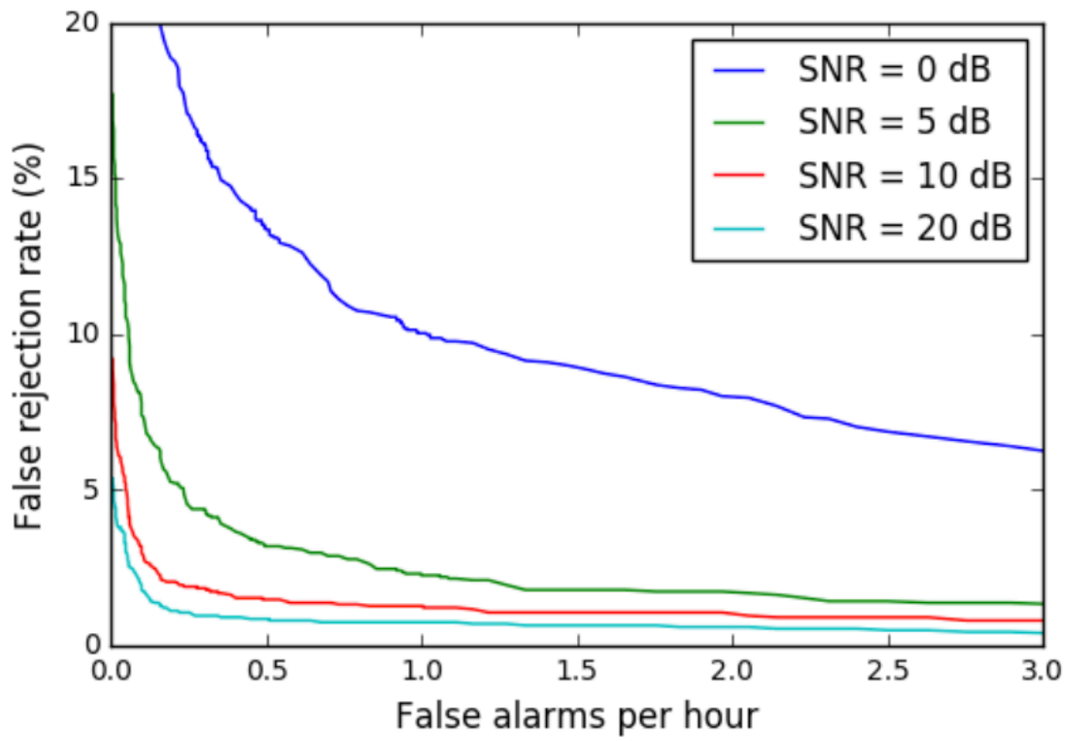


Figure 3: *FRR vs. FA per hour for the test set with various SNR values.*

https://arxiv.org/abs/1703.05390

MatchboxNet

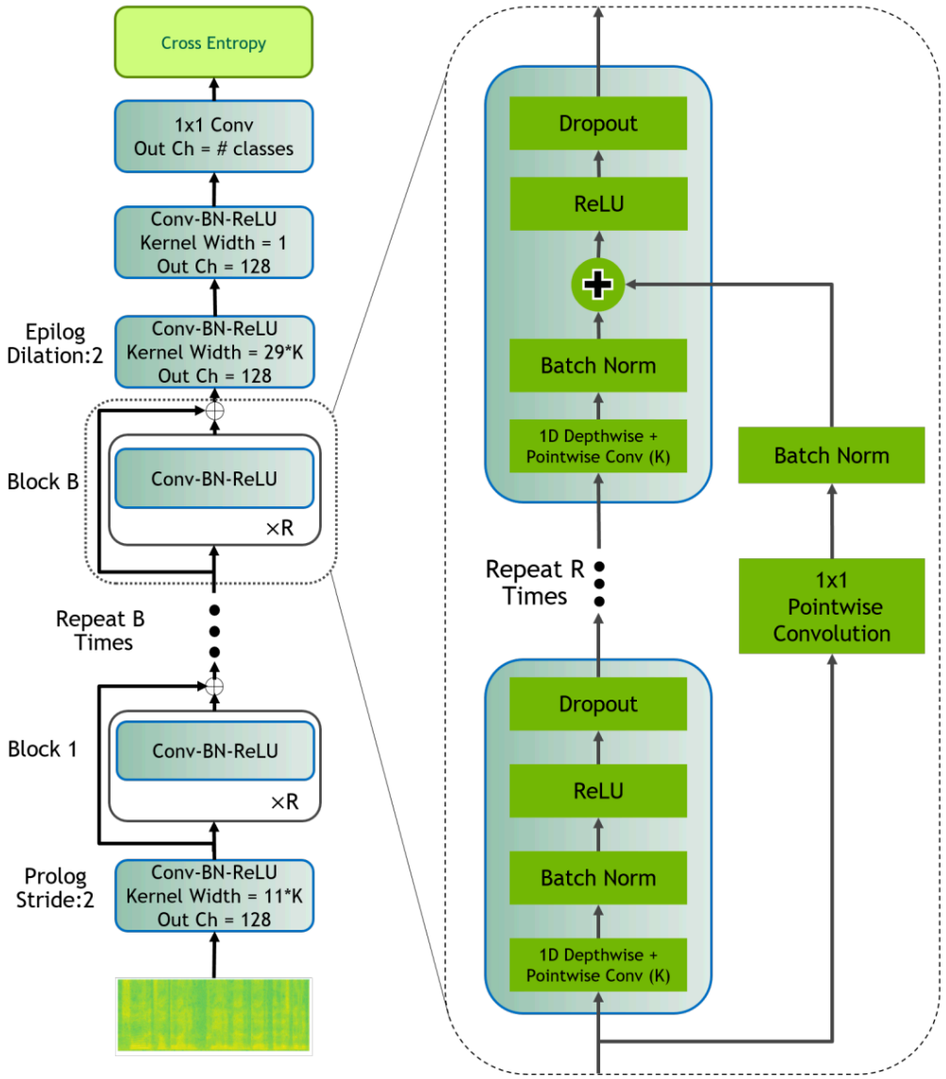


Figure 1: *MatchboxNet BxRxC model: B - number of blocks, R - number of sub-blocks, C - the number of channels.*

MatchboxNet: 1D Time-Channel Separable Convolutional Neural Network Architecture for Speech Commands Recognition

Somshubra Majumdar, Boris Ginsburg

NVIDIA, Santa Clara, USA

{smajumdar,bginsburg}@nvidia.com

Table 3: *MatchboxNet on Google Speech Commands dataset v2, the accuracy is averaged over 5 trials (95% Confidence Interval).*

Model	# Parameters, K	Accuracy, %	Reference
Attention RNN	202	94.30	[33]
Harmonic Tensor 2D-CNN	-	96.39	[30]
"Embedding + Head" Model	385	97.7	[31]
MatchboxNet-3x1x64	77	96.91 ± 0.101	
MatchboxNet-3x2x64	93	97.21 ± 0.072	
MatchboxNet-6x2x64	140	97.37 ± 0.110	

https://arxiv.org/abs/2004.08531