TELECOM Paris

IP PARIS

LABORATOIRE INFORMATIQUE D'AVIGNON

Conditional Independence for Pretext Task Selection in Self-Supervised Speech Representation Learning

Salah Zaiem, Titouan Parcollet, Slim Essid
salah.zaiem@telecom-paris.fr

INTERSPEECH 2021

Outline

Speech Self-Supervised Learning

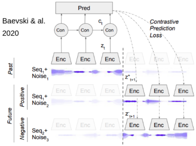


Figure 1: ***CPC-based data augmentation***. *Each speech sequence is encoded twice, one for past one for future, with potentially different augmentations for each. The CPC loss tries to contrastively predict future embeddings z_{t+i} based on past ones, ignoring the noise of the augmentation. Positive and negative sequences may have different augmentations.*

Speech Self-Supervised Learning

Baevski & al. 2020

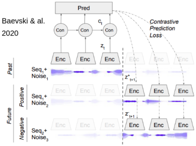
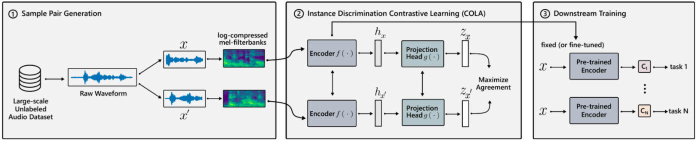


Figure 1: ***CPC-based data augmentation***. *Each speech sequence is encoded twice, one for past one for future, with potentially different augmentations for each. The CPC loss tries to contrastively predict future embeddings z_{t+i} based on past ones, ignoring the noise of the augmentation. Positive and negative sequences may have different augmentations.*

Saeed & al. 2020



Speech Self-Supervised Learning

Baevski & al. 2020

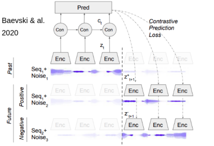
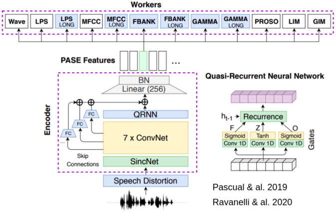


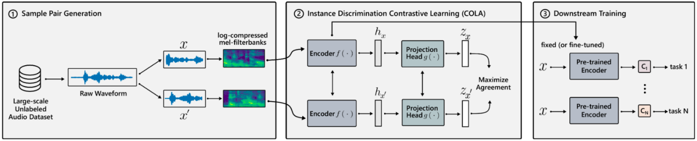
Figure 1: ***CPC-based data augmentation.*** *Each speech sequence is encoded twice, one for past one for future, with potentially different augmentations for each. The CPC loss tries to contrastively predict future embeddings z_{t+i} based on past ones, ignoring the noise of the augmentation. Positive and negative sequences may have different augmentations.*



Pascual & al. 2019

Ravanelli & al. 2020

Saeed & al. 2020



Speech Self-Supervised Learning

Baevski & al. 2020

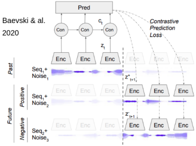
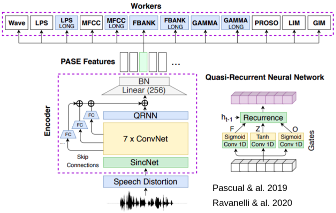


Figure 1: ***CPC-based data augmentation****. Each speech sequence is encoded twice, one for past one for future, with potentially different augmentations for each. The CPC loss tries to contrastively predict future embeddings z_{t+i} based on past ones, ignoring the noise of the augmentation. Positive and negative sequences may have different augmentations.*

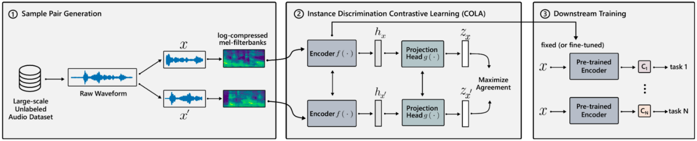
Generated Pseudo-labels



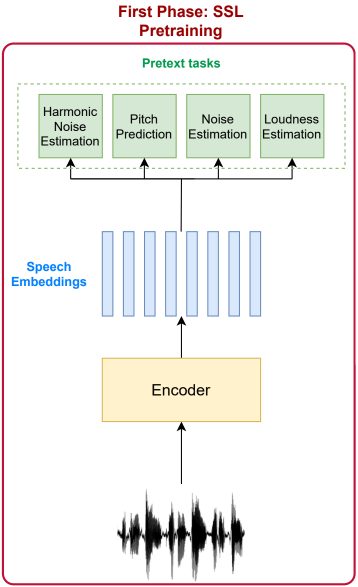
Pascual & al. 2019

Ravanelli & al. 2020

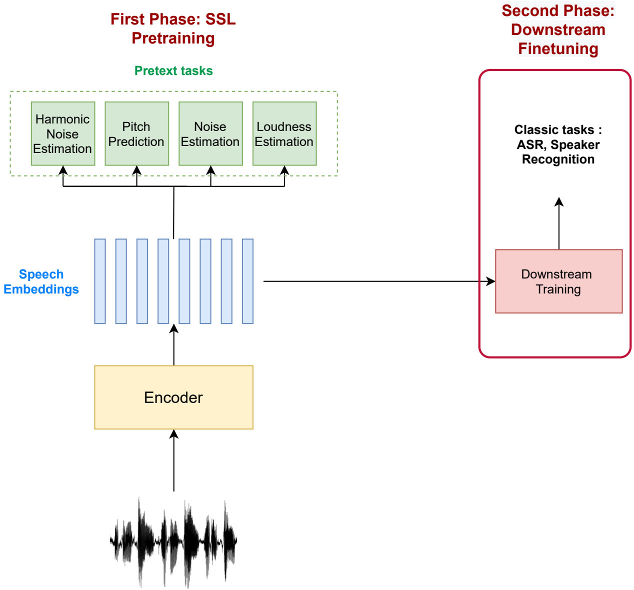
Saeed & al. 2020



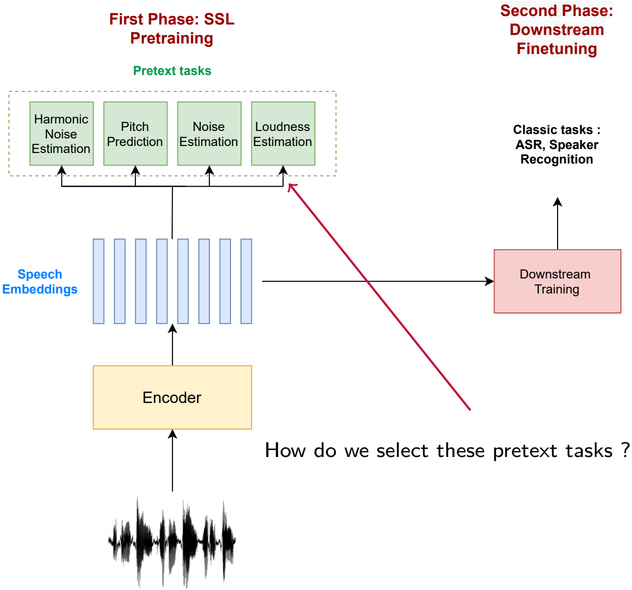
Introduction



Introduction



Introduction



How do we select these pretext tasks ?

Objective

Can we find a function scoring the usefulness of a given pretext task towards solving a downstream one ?

Outline

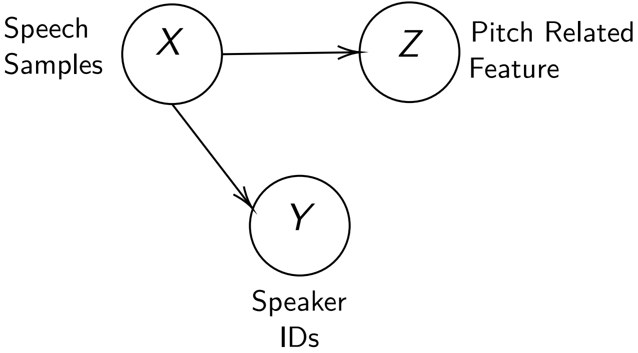
Conditional Independence based estimator

Main Idea

Speech samples $\perp$ Pretext task labels $|$ Downstream labels
$\longrightarrow$ Good pretext task.

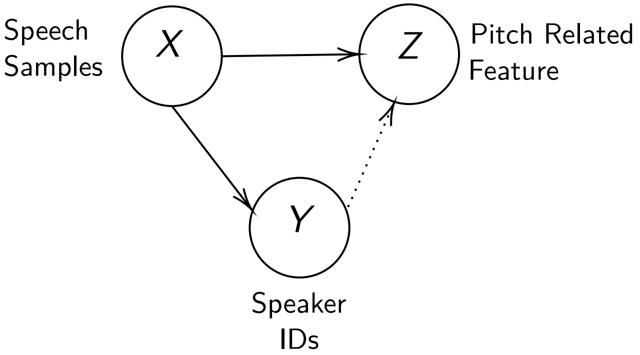
Conditional Independence based estimator

Speech samples $\perp$ Pretext task labels | Downstream labels



Conditional Independence based estimator

Speech samples $\perp$ Pretext task labels | Downstream labels



Issues with Conditional Independence

- Non trivial to compute.
- Even harder with speech data.

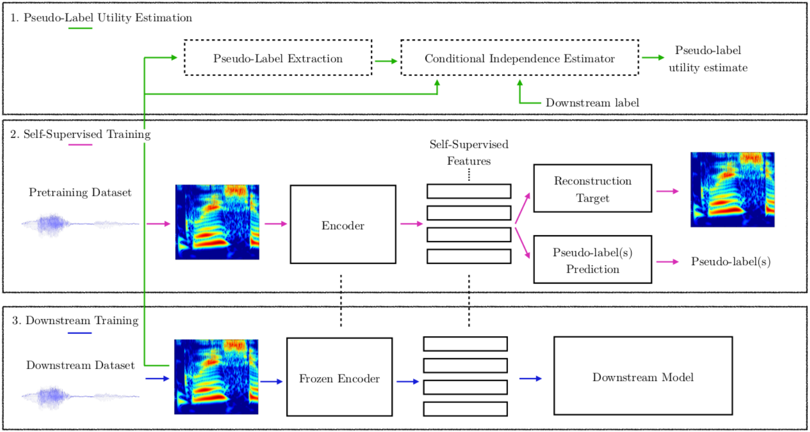
Issues with Conditional Independence

- Non trivial to compute.
- Even harder with speech data.

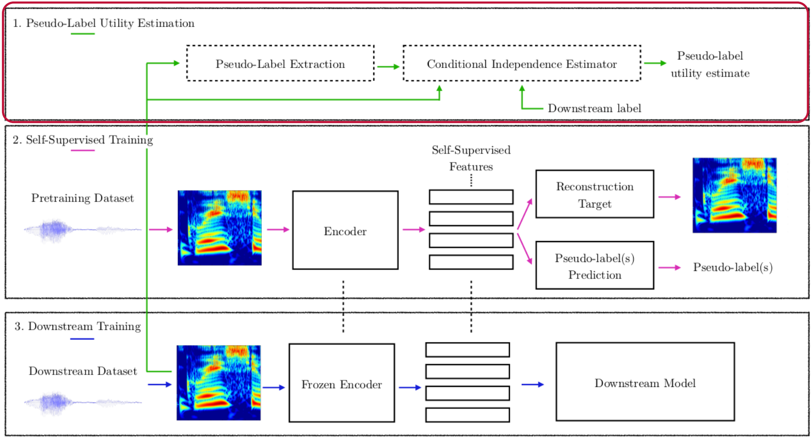
Contribution

Simple method to compute a CI estimate

Three steps validation



First step



1. Pseudo-Label Utility Estimation
Pseudo-Label Extraction
Conditional Independence Estimator
Pseudo-label utility estimate
Downstream label
2. Self-Supervised Training
Pretraining Dataset
Encoder
Self-Supervised Features
Reconstruction Target
Pseudo-label(s) Prediction
Pseudo-label(s)
3. Downstream Training
Downstream Dataset
Frozen Encoder
Downstream Model

Hilbert Schmidt Independence Criterion

- Kernel-based independence testing between speech samples $X = (x_i)_{i \in [0,N]}$ and pseudo labels $Z = (z_i)_{i \in [0,N]}$
- Only need two kernel (similarity) matrices K and L
- Where $K_{ij} = k(x_i, x_j)$ and $L_{ij} = l(z_i, z_j)$

Hilbert Schmidt Independence Criterion

- Kernel-based independence testing between speech samples $X = (x_i)_{i \in [0,N]}$ and pseudo labels $Z = (z_i)_{i \in [0,N]}$
- Only need two kernel (similarity) matrices K and L
- Where $K_{ij} = k(x_i, x_j)$ and $L_{ij} = l(z_i, z_j)$

$$HSIC(X, Z) = \frac{1}{n^2} Trace(KHLH)$$

- $H = I_N - \frac{1}{N} 1_N 1_N^T$

Hilbert Schmidt Independence Criterion

- Kernel-based independence testing between speech samples $X = (x_i)_{i \in [0,N]}$ and pseudo labels $Z = (z_i)_{i \in [0,N]}$
- Only need two kernel (similarity) matrices K and L
- Where $K_{ij} = k(x_i, x_j)$ and $L_{ij} = l(z_i, z_j)$

$$HSIC(X, Z) = \frac{1}{n^2} Trace(KHLH)$$

- $H = I_N - \frac{1}{N} 1_N 1_N^T$
- Hilbert-Schmidt Norm of the Cross Covariance Operator.
- The lower, the more independent.

Hilbert Schmidt Independence Criterion

- Kernel-based independence testing between speech samples $X = (x_i)_{i \in [0,N]}$ and pseudo labels $Z = (z_i)_{i \in [0,N]}$
- Only need two kernel (similarity) matrices K and L
- Where $K_{ij} = k(x_i, x_j)$ and $L_{ij} = l(z_i, z_j)$

$$HSIC(X, Z) = \frac{1}{n^2} Trace(KHLH)$$

- $H = I_N - \frac{1}{N} 1_N 1_N^T$
- Hilbert-Schmidt Norm of the Cross Covariance Operator.
- The lower, the more independent.
- Intuition : points similar in K are similar in L $\longrightarrow$ high HSIC

From Independence to Conditional Independence

- Divide the data points according to the downstream classes.
- Compute the HSIC on every subset.
- Aggregate them in a weighted mean.

$$HSIC(X, Z|Y) = \frac{1}{M} \sum_{c \in \mathscr{C}} HSIC_c(X, Z) \times n_c.$$

Intuition Break

- In our cases, $X \not\perp Z$
- Speaker Recognition as the Downstream Task, Y are speaker ID
- Suppose that Z is a function of Y

Intuition Break

- In our cases, $X \not\perp Z$
- Speaker Recognition as the Downstream Task, Y are speaker ID
- Suppose that Z is a function of Y
- Moving K, Constant $L \longrightarrow$ low $HSIC$

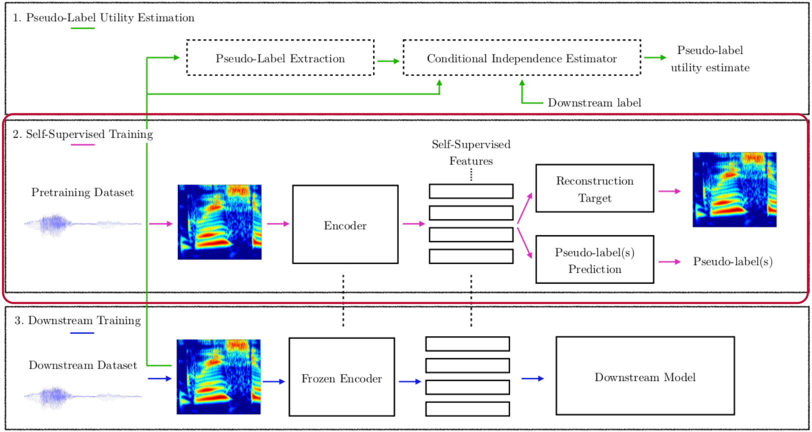
From Independence to Conditional Independence

- Divide the data points according to the downstream classes.
- Compute the HSIC on every subset.
- Aggregate them in a weighted mean.

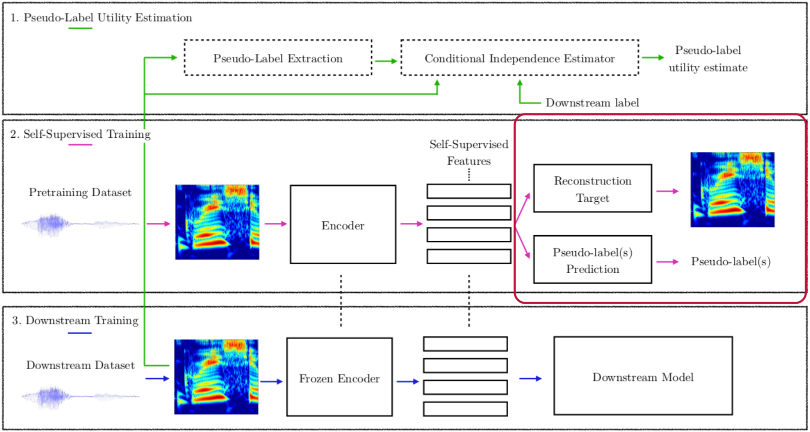
$$HSIC(X,Z|Y) = \frac{1}{M} \sum_{c \in \mathscr{C}} HSIC_c(X,Z) \times n_c.$$

Outline

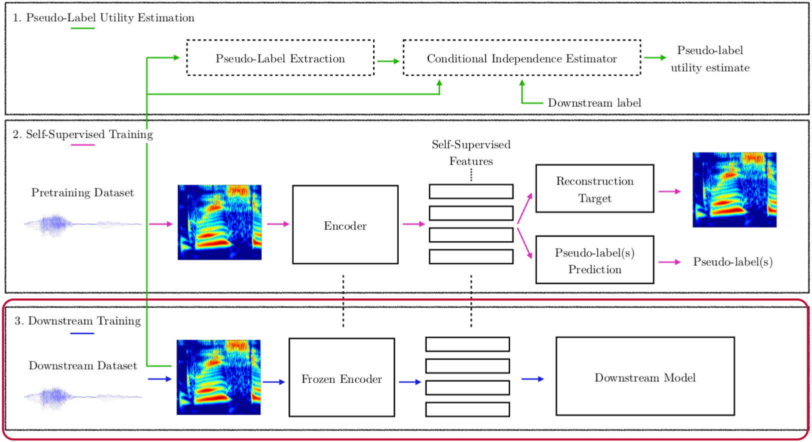
Next steps : Pretraining



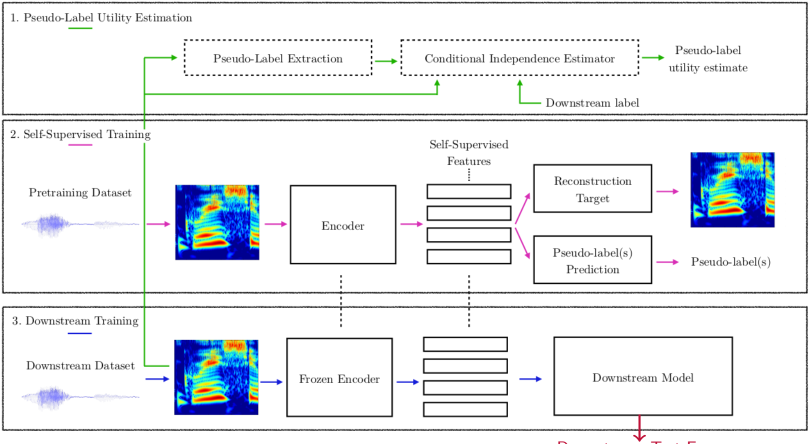
Next steps : Pretraining



Next steps : Finetuning



Next steps : Finetuning



Downstream Test Error

Datasets

Datasets Roles and Descriptions

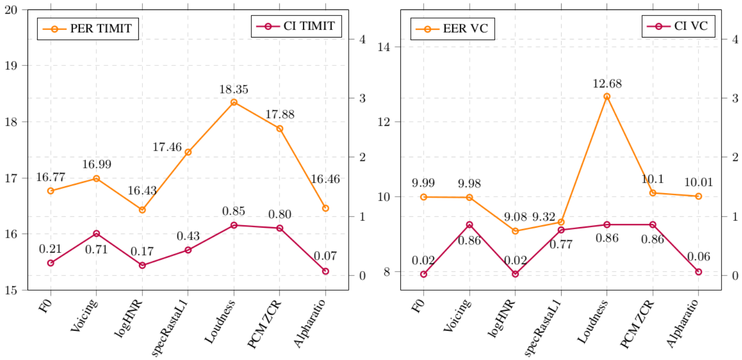
Task	**Dataset**	**$\sim$Dur.(train)**	**Speakers**
Pretraining	CommonVoiceEn6.1	1686 hours	$\sim$66173
ASR	TIMIT	5 hours	462
Speak Recog	VoxCeleb1	148642 utt	1251

Pretext tasks: pseudo-labels prediction

Candidate pseudo-labels and descriptions

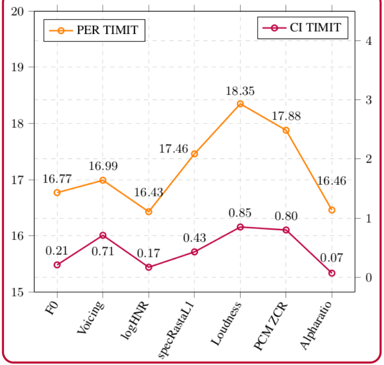
Pseudo-label	**Description**
Loudness	Intensity & approx. loudness
F0	Fundamental Frequency
Voicing	Voicing Decision
Alpha Ratio	Ratio of spectrum intensity % 1000 Hz
Zero Crossing Rate	Zero crossing number per frame
RastaSpec L1Norm	L1 Norm of Rasta Spectrum
log HNR	log of Harmonicity to Noise Ratio

Results

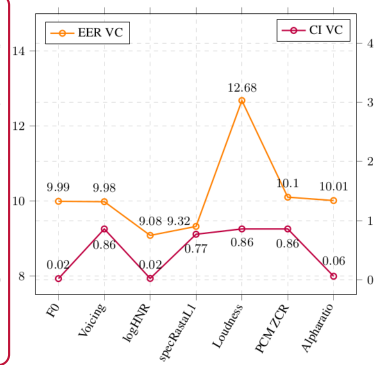


PER TIMIT
CI TIMIT
16.77
16.99
16.43
17.46
18.35
17.88
16.46
0.21
0.71
0.17
0.43
0.85
0.80
0.07
EER VC
CI VC
9.99
9.98
9.08
9.32
12.68
10.1
10.01
0.02
0.86
0.02
0.77
0.86
0.86
0.06
F0
Voicing
logHNR
specRastaL1
Loudness
PCM ZCR
Alpharatio

Results



PER TIMIT
CI TIMIT
20
19
18
17
16
15
4
3
2
1
0
16.77
16.99
16.43
17.46
18.35
17.88
16.46
0.21
0.71
0.17
0.43
0.85
0.80
0.07
F0
Voicing
logHNR
specRastaL1
Loudness
PCM ZCR
Alpharatio

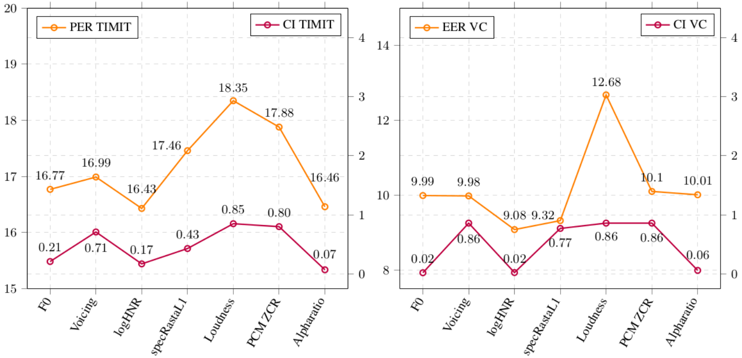


EER VC
CI VC
14
12
10
8
4
3
2
1
0
9.99
9.98
9.08
9.32
12.68
10.1
10.01
0.02
0.86
0.02
0.77
0.86
0.86
0.06
F0
Voicing
logHNR
specRastaL1
Loudness
PCM ZCR
Alpharatio

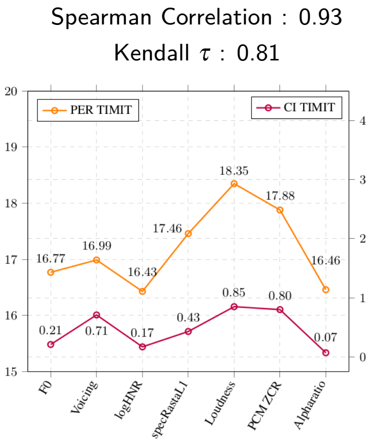
Results

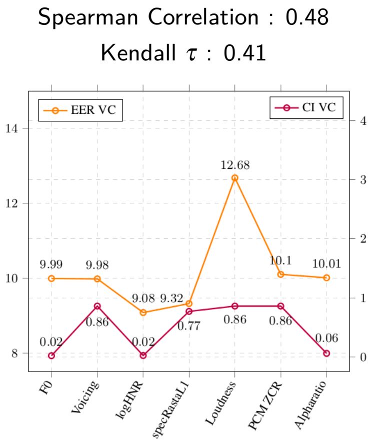
Spearman Correlation : correlation between ranks

Kendall τ : proportion of pairs respecting the monotonic relationship



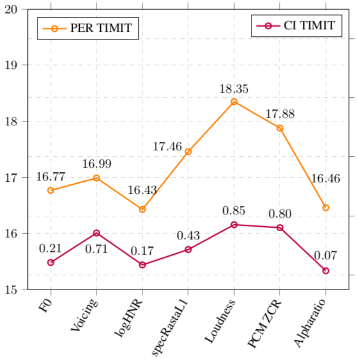
Results


Spearman Correlation : 0.93
Kendall τ : 0.81
PER TIMIT
CI TIMIT
16.77
16.99
16.43
17.46
18.35
17.88
16.46
0.21
0.71
0.17
0.43
0.85
0.80
0.07
F0
Voicing
logHNR
specRastaL1
Loudness
PCM ZCR
Alpharatio

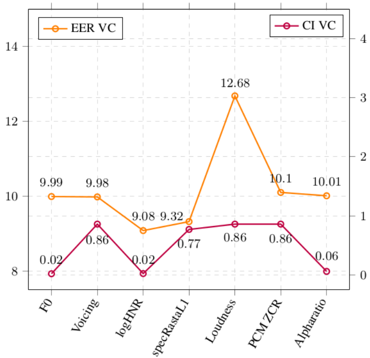

Spearman Correlation : 0.48
Kendall τ : 0.41
EER VC
CI VC
9.99
9.98
9.08
9.32
12.68
10.1
10.01
0.02
0.86
0.02
0.77
0.86
0.86
0.06
F0
Voicing
logHNR
specRastaL1
Loudness
PCM ZCR
Alpharatio

Results

Spearman Correlation : 0.93
Kendall τ : 0.81



Spearman Correlation : 0.48
Kendall τ : 0.41



Multi pretext task selection

What if we learned pretext-tasks simultaneously ?

Multi pretext task selection

What if we learned pretext-tasks simultaneously ?
Simple attempt : Regrouping the best and the worst pretext tasks for every Downstream task.

Multi pretext task selection

What if we learned pretext-tasks simultaneously ?
Simple attempt : Regrouping the best and the worst pretext tasks for every Downstream task.

Experiment	Pseudo Labels	EER/PER
Best VC	F0 /log HNR / AlphaR	6.40
Worst VC	Loud/ZCR/RastaL1/ Voicing	7.33
Best TIM	F0/RastaL1/AlphaR/log HNR	15.35
Worst TIM	Voicing/ Loud/ ZCR	16.77

Conclusion

Can we find a function scoring the usefulness of a given pretext task towards solving a downstream one ?

Conclusion

Can we find a function scoring the usefulness of a given pretext task towards solving a downstream one ?

- Use Conditional Independence to predict the utility of a pretext-task towards solving a given downstream task.
- Efficient way to for SSL pretext-tasks exploration.
- Further works on multi-task pretext task selection.

Thank You

Thank you all for your attention !!

Please feel free to ask any question