Topological deep learning for speech recognition

Joint with Zeyang Ding, Pingyao Feng, Qingrui Qu, Siheng Yi, Zhiwang Yu, and Haiyu Zhang

Yifei Zhu

Southern University of Science and Technology

2026.2.3

Overview

In 1966, Mark Kac asked the famous question:

Can you hear the shape of a drum?

Overview

In 1966, Mark Kac asked the famous question:

Can you hear the shape of a drum?

To hear the shape of a drum is to deduce information about the shape of the drumhead from the sound it makes, using mathematical theory.

Overview

In 1966, Mark Kac asked the famous question:

Can you hear the shape of a drum?

To hear the shape of a drum is to deduce information about the shape of the drumhead from the sound it makes, using mathematical theory.

In this talk, we mirror the question across senses and address instead:

Overview

In 1966, Mark Kac asked the famous question:

Can you hear the shape of a drum?

To hear the shape of a drum is to deduce information about the shape of the drumhead from the sound it makes, using mathematical theory.

In this talk, we mirror the question across senses and address instead:

Can you see the sound of a human speech?

Overview

In 1966, Mark Kac asked the famous question:

Can you hear the shape of a drum?

To hear the shape of a drum is to deduce information about the shape of the drumhead from the sound it makes, using mathematical theory.

In this talk, we mirror the question across senses and address instead:

Can you see the sound of a human speech?

Overview: context & summary

Topological speech (and audio) signal processing

Overview: context & summary

Topological speech (and audio) signal processing

time series data

Overview: context & summary

Topological speech (and audio) signal processing

time series data

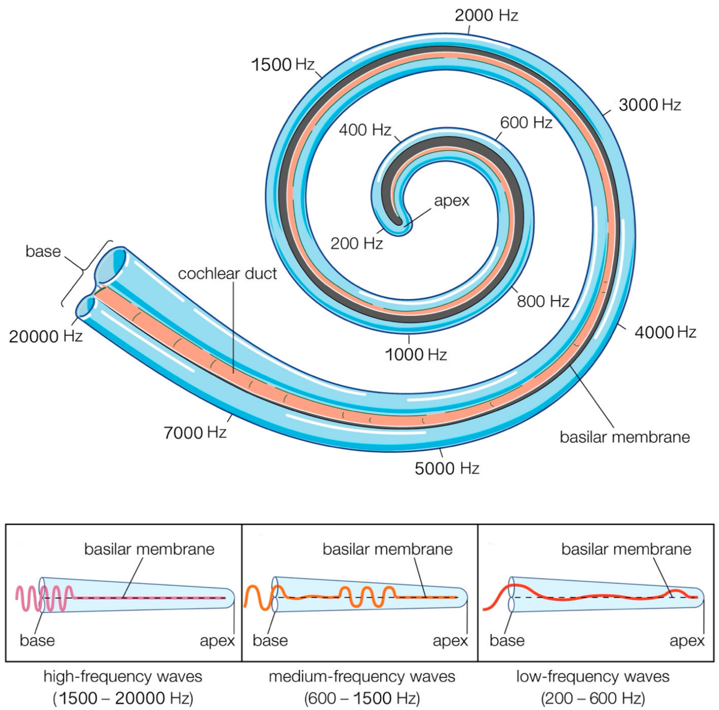
one of the essential components of AI

Overview: context & summary

Topological speech (and audio) signal processing, beyond direct biomimetic engineering

Overview: context & summary

Topological speech (and audio) signal processing, beyond direct biomimetic engineering

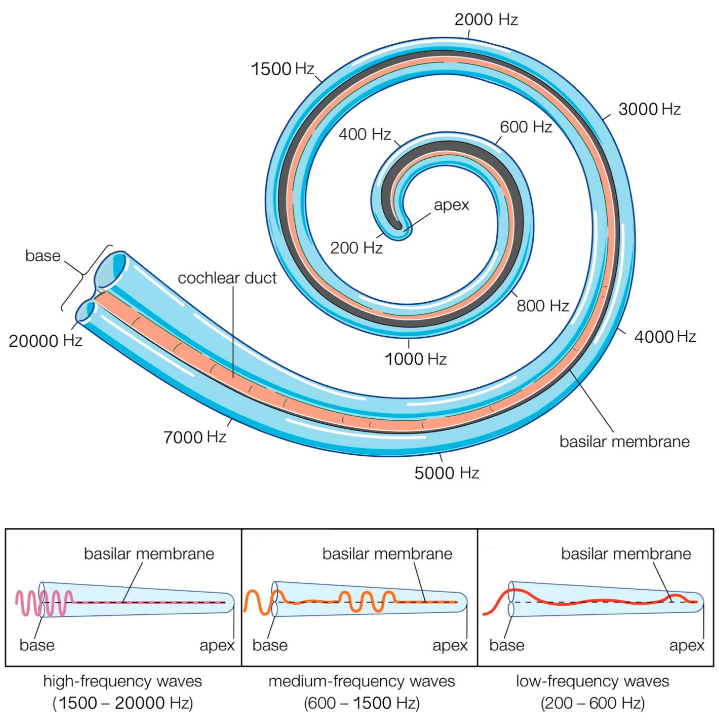


Distribution of frequencies along the basilar membrane of the cochlea, which functions as a natural Fourier analysis device

Overview: context & summary

Topological speech (and audio) signal processing, beyond direct biomimetic engineering: topological features vs. STFT/MFCC

short-time Fourier transform

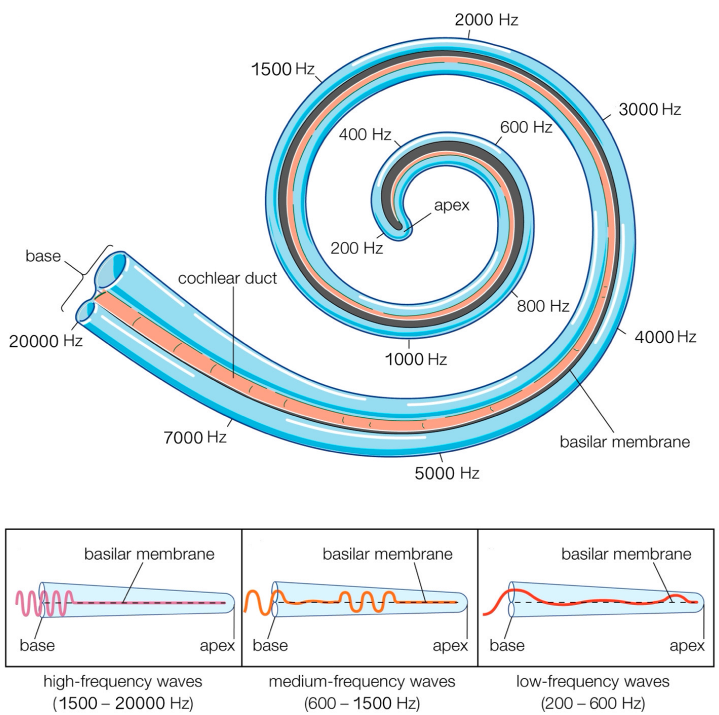


Distribution of frequencies along the basilar membrane of the cochlea, which functions as a natural Fourier analysis device

Overview: context & summary

Topological speech (and audio) signal processing, beyond direct biomimetic engineering: topological features vs. STFT/MFCC *mel-frequency cepstral coefficients*

short-time Fourier transform



Distribution of frequencies along the basilar membrane of the cochlea, which functions as a natural Fourier analysis device

Overview: context & summary

Topological speech (and audio) signal processing, beyond direct biomimetic engineering: topological features vs. STFT/MFCC

Combination of TDA to ML *machine learning*

topological data analysis

Overview: context & summary

Topological speech (and audio) signal processing, beyond direct biomimetic engineering: topological features vs. STFT/MFCC

Combination of TDA to ML:

1. TopCap
2. TopNN
3. TopKer

Overview: context & summary

Topological speech (and audio) signal processing, beyond direct biomimetic engineering: topological features vs. STFT/MFCC

Combination of TDA to ML:

1. TopCap *capability of capturing topological structures of data*
2. TopNN
3. TopKer

Overview: context & summary

Topological speech (and audio) signal processing, beyond direct biomimetic engineering: topological features vs. STFT/MFCC

Combination of TDA to ML:

1. TopCap *capability of capturing topological structures of data*
2. TopNN *topology-enhanced neural network*
3. TopKer

Overview: context & summary

Topological speech (and audio) signal processing, beyond direct biomimetic engineering: topological features vs. STFT/MFCC

Combination of TDA to ML:

1. TopCap *capability of capturing topological structures of data*
2. TopNN *topology-enhanced neural network*
3. TopKer *topology-informed convolution kernel*

Overview: context & summary

Topological speech (and audio) signal processing, beyond direct biomimetic engineering: topological features vs. STFT/MFCC

Combination of TDA to ML:

1. TopCap: stands in comparison, datasets + models
2. TopNN
3. TopKer

Overview: context & summary

Topological speech (and audio) signal processing, beyond direct biomimetic engineering: topological features vs. STFT/MFCC

Combination of TDA to ML:

1. TopCap: stands in comparison, datasets + models
2. TopNN
3. TopKer

Overview: context & summary

Topological speech (and audio) signal processing, beyond direct biomimetic engineering: topological features vs. STFT/MFCC

Combination of TDA to ML:

1. TopCap: stands in comparison, datasets + models
2. TopNN
3. TopKer

Overview: context & summary

Topological speech (and audio) signal processing, beyond direct biomimetic engineering: topological features vs. STFT/MFCC

Combination of TDA to ML:

1. TopCap: stands in comparison, datasets + models
2. TopNN: outperforms, accuracy + convergence of loss function + steadiness + robustness against noise
3. TopKer

Overview: context & summary

Topological speech (and audio) signal processing, beyond direct biomimetic engineering: topological features vs. STFT/MFCC

Combination of TDA to ML:

1. TopCap: stands in comparison, datasets + models
2. TopNN: outperforms, accuracy + convergence of loss function + steadiness + robustness against noise
3. TopKer

Overview: context & summary

Topological speech (and audio) signal processing, beyond direct biomimetic engineering: topological features vs. STFT/MFCC

Combination of TDA to ML:

1. TopCap: stands in comparison, datasets + models
2. TopNN: outperforms, accuracy + convergence of loss function + steadiness + robustness against noise
3. TopKer: superior performance + cross-domain adaptability

phoneme recognition

Overview: context & summary

Topological speech (and audio) signal processing, beyond direct biomimetic engineering: topological features vs. STFT/MFCC

Combination of TDA to ML:

1. TopCap: stands in comparison, datasets + models
2. TopNN: outperforms, accuracy + convergence of loss function + steadiness + robustness against noise
3. TopKer: superior performance + cross-domain adaptability

phoneme recognition *other audio and visual recognition tasks*

Periodic phenomena: a motivating example

Let $T^2 = (\mathbb{R}/\mathbb{Z})^2$ be the 2D torus. Consider the dynamical system given by

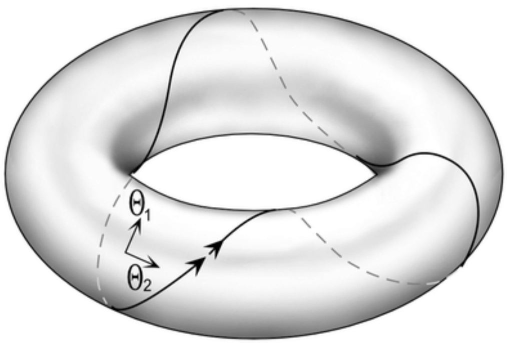
$$\begin{aligned}\Phi_{\sigma} \colon T^2 \times \mathbb{R} &\longrightarrow T^2 \\ \big((a,b),t\big) &\longmapsto (a+t, b+\sigma t)\end{aligned}$$

Periodic phenomena: a motivating example

Let $T^2 = (\mathbb{R}/\mathbb{Z})^2$ be the 2D torus. Consider the dynamical system given by

$$\begin{aligned} \Phi_\sigma \colon T^2 \times \mathbb{R} &\to T^2 \\ \big((a,b),t\big) &\mapsto (a+t, b+\sigma t) \end{aligned}$$

If σ is rational, then every orbit is periodic.

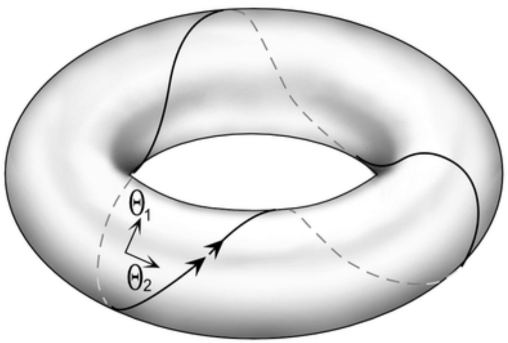


Periodic phenomena: a motivating example

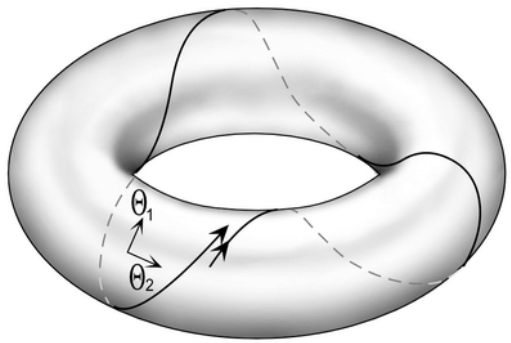
Let $T^2 = (\mathbb{R}/\mathbb{Z})^2$ be the 2D torus. Consider the dynamical system given by

$$\begin{aligned} \Phi_\sigma \colon T^2 \times \mathbb{R} &\to T^2 \\ \big((a,b),t\big) &\mapsto (a+t, b+\sigma t) \end{aligned}$$

If σ is rational, then every orbit is periodic. Otherwise every orbit is dense in T^2.



rational σ



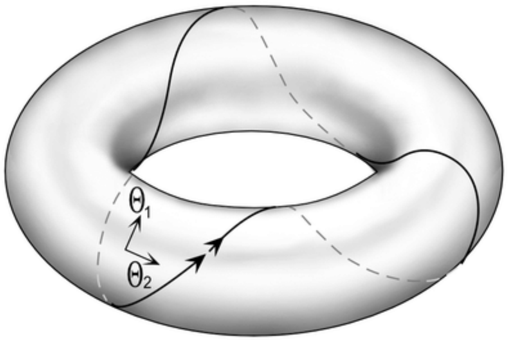
irrational σ

Periodic phenomena: a motivating example

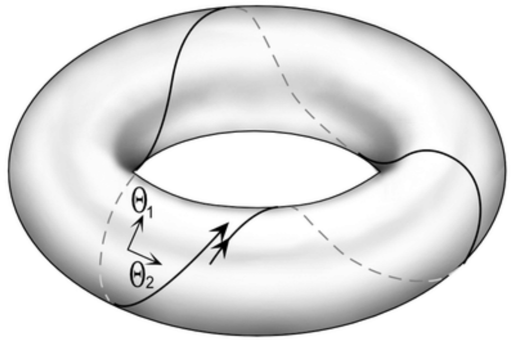
Let $T^2 = (\mathbb{R}/\mathbb{Z})^2$ be the 2D torus. Consider the dynamical system given by

$$\Phi_\sigma \colon T^2 \times \mathbb{R} \to T^2$$
$$\big((a,b),t\big) \mapsto (a+t, b+\sigma t)$$

If σ is rational, then every orbit is periodic. Otherwise every orbit is dense in T^2.



rational σ



irrational σ

From time series to topological shapes

Most periodic time series can be realized by a topological circle S^1 embedded in a Euclidean space of higher dimension.

Topological time series analysis

Let us make the assumption that sampled signals are distributed over a manifold (!)

Topological time series analysis

Let us make the assumption that sampled signals are distributed over a manifold (!) To topologically analyze time series, we then proceed as follows:

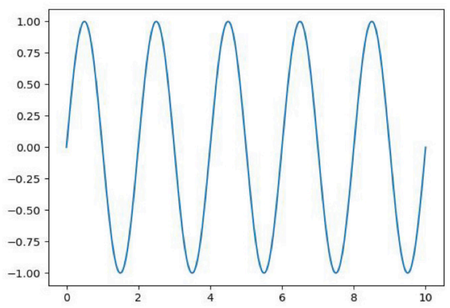
Step 1 Embed the data into a Euclidean space of suitable dimension

Topological time series analysis

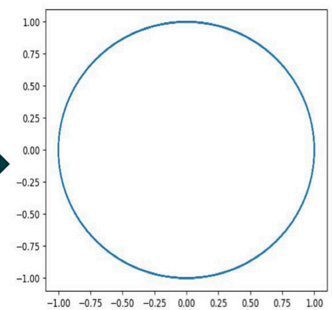
Let us make the assumption that sampled signals are distributed over a manifold (!) To topologically analyze time series, we then proceed as follows:

<u>Step 1</u> Embed the data into a Euclidean space of suitable dimension;

<u>Step 2</u> Compute the algebraic invariants for statistical inference.



realization



computation

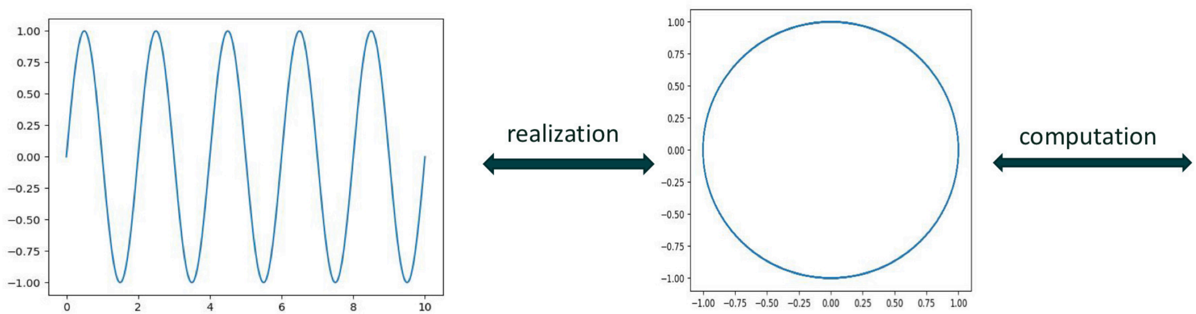
$$H_k(S^1) = \begin{cases} \mathbb{Z} & k = 0 \\ \mathbb{Z} & k = 1 \\ 0 & k > 1 \end{cases}$$

Topological time series analysis

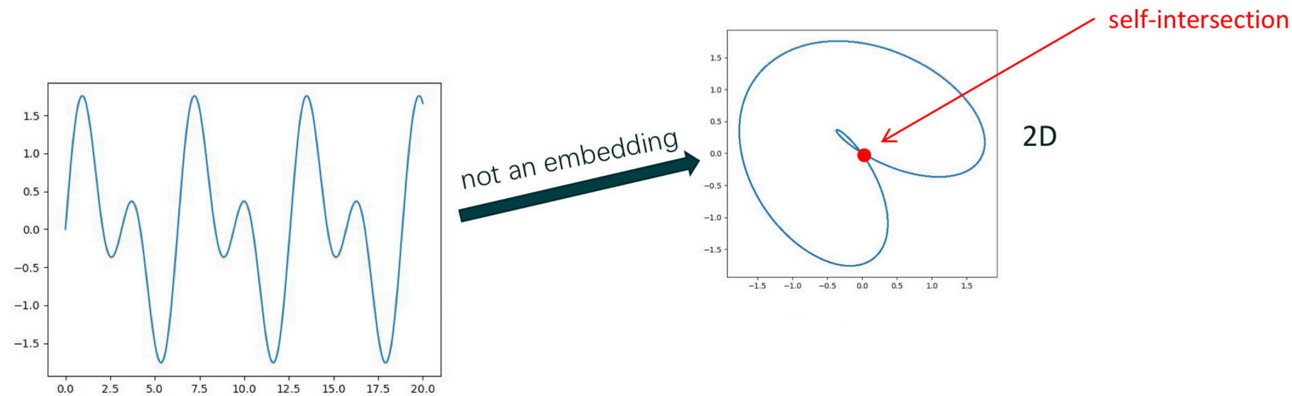
Let us make the assumption that sampled signals are distributed over a manifold (!) To topologically analyze time series, we then proceed as follows:

Step 1 Embed the data into a Euclidean space of suitable dimension;

Step 2 Compute the algebraic invariants for statistical inference.



$$H_k(S^1) = \begin{cases} \mathbb{Z} & k = 0 \\ \mathbb{Z} & k = 1 \\ 0 & k > 1 \end{cases}$$

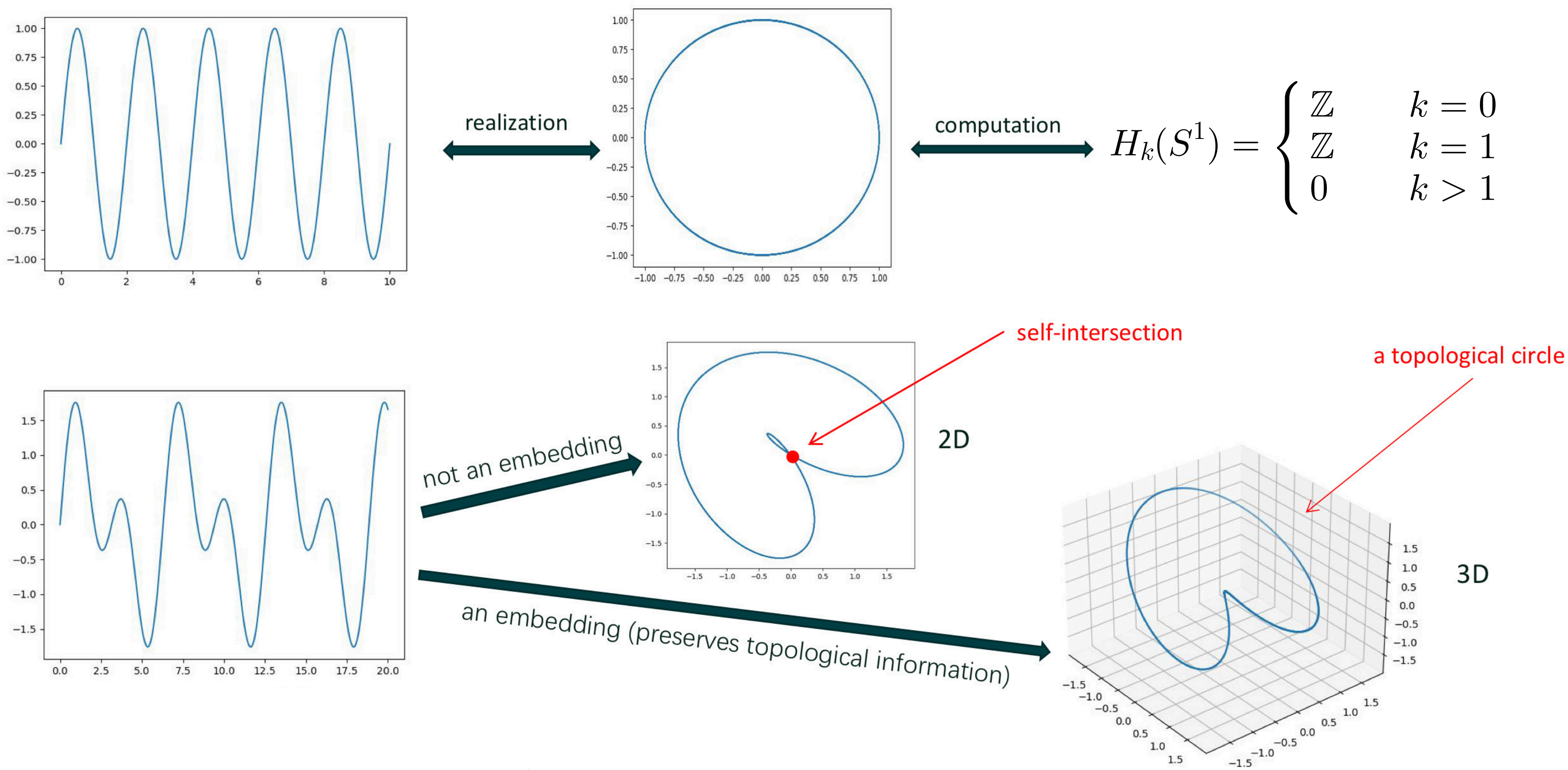


Topological time series analysis

Let us make the assumption that sampled signals are distributed over a manifold (!) To topologically analyze time series, we then proceed as follows:

Step 1 Embed the data into a Euclidean space of suitable dimension;

Step 2 Compute the algebraic invariants for statistical inference.



Topological time series analysis

Let us make the assumption that sampled signals are distributed over a manifold (!) To topologically analyze time series, we then proceed as follows:

Step 1 Embed the data into a Euclidean space of suitable dimension;

Step 2 Compute the algebraic invariants for statistical inference.

Jose A. Perea, *Topological time series analysis*, **Notices of the American Mathematical Society**, 2019.

Jose A. Perea and John Harer, *Sliding windows and persistence: An application of topological methods to signal analysis*, **Foundations of Computational Mathematics**, 2015.

Topological time series analysis

Let us make the assumption that sampled signals are distributed over a manifold (!) To topologically analyze time series, we then proceed as follows:

Step 1 Embed the data into a Euclidean space of suitable dimension;

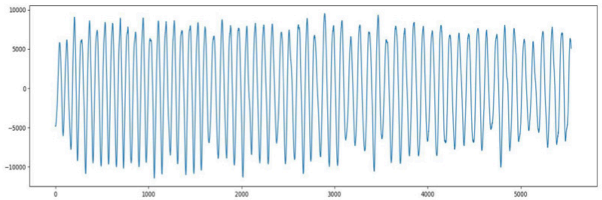
Step 2 Compute the algebraic invariants for statistical inference.

Jose A. Perea, *Topological time series analysis*, **Notices of the American Mathematical Society**, 2019.

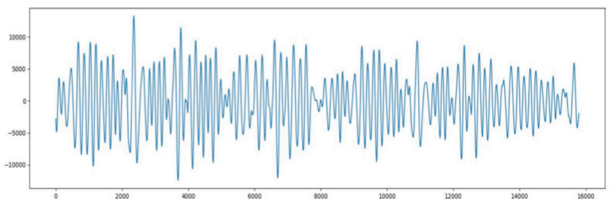
Jose A. Perea and John Harer, *Sliding windows and persistence: An application of topological methods to signal analysis*, **Foundations of Computational Mathematics**, 2015.

An application: detection of wheeze in medical science (pulmonology)

wheeze



normal

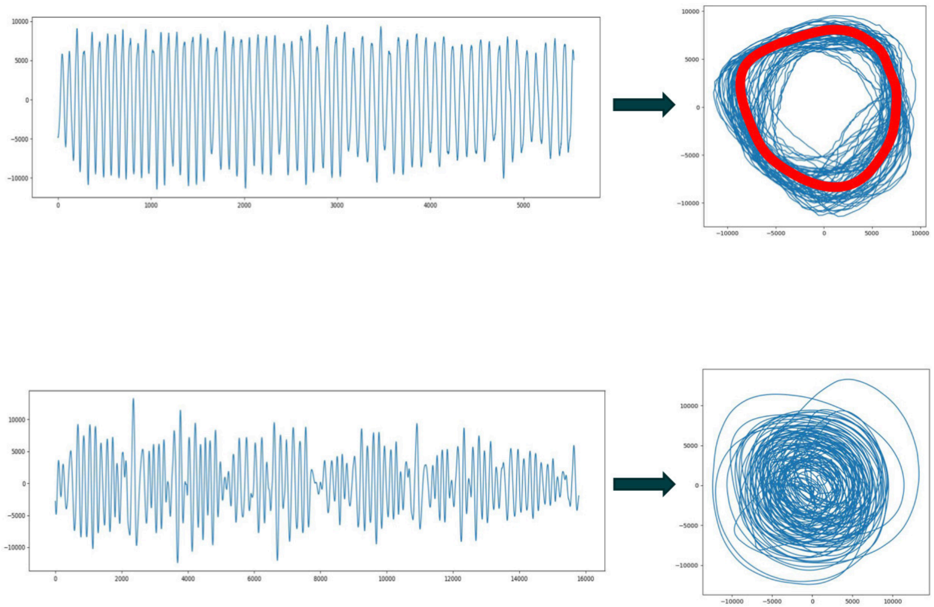


Original sound signals

An application: detection of wheeze in medical science (pulmonology)

wheeze

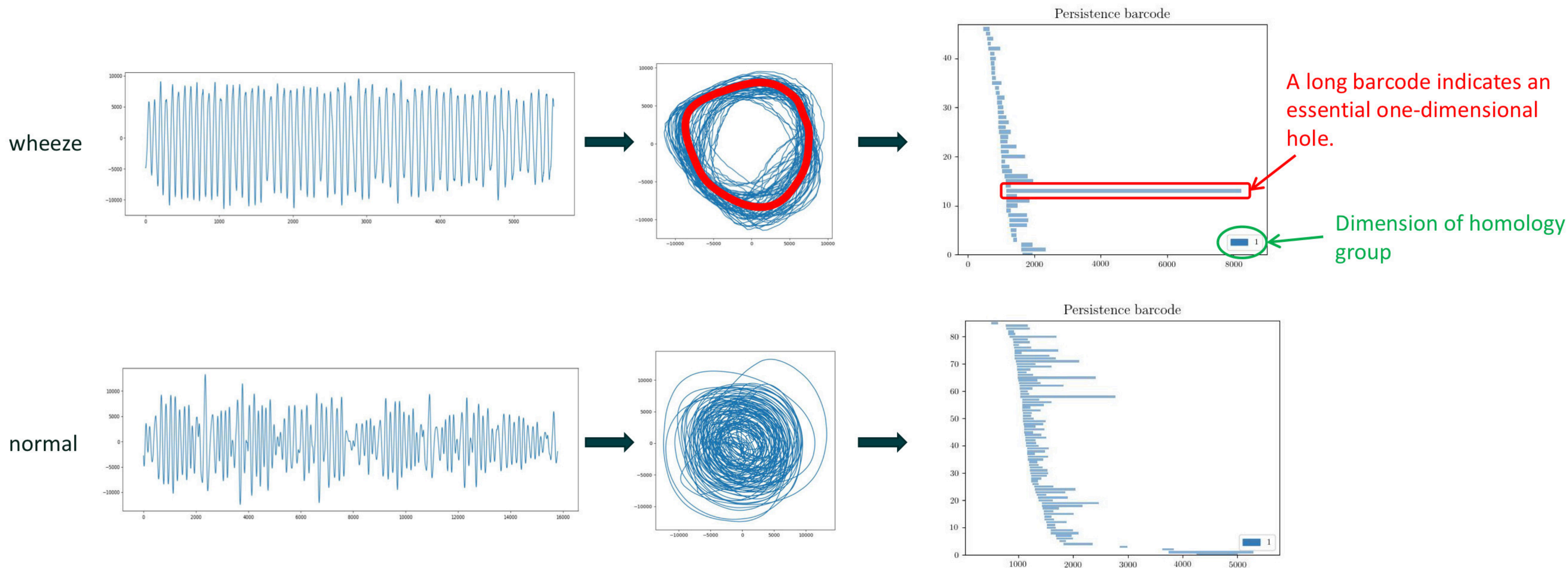
normal



Original sound signals

Realized topological shapes embedded in 2D Euclidean space

An application: detection of wheeze in medical science (pulmonology)



Original sound signals

Realized topological shapes embedded in 2D Euclidean space

Persistence barcodes as representations of the algebraic invariant (1D homology group)

Emrani et al., *Persistent homology of delay embeddings and its application to wheeze detection*, **IEEE Signal Processing Letters**, 2014.

A pipeline for topological time series analysis

Time series data: $x_1, x_2, x_3, x_4, \ldots$

A pipeline for topological time series analysis

Time series data: $x_1, x_2, x_3, x_4, \ldots$

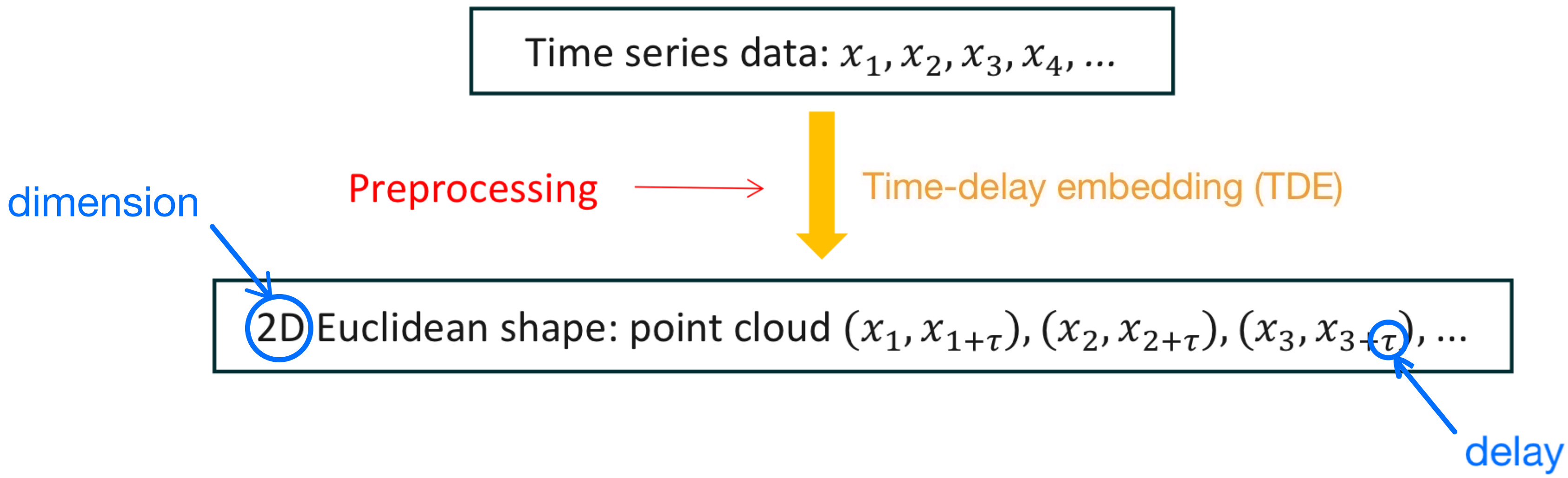
Preprocessing $\longrightarrow$ Time-delay embedding (TDE)

dimension

2D Euclidean shape: point cloud $(x_1, x_{1+\tau}), (x_2, x_{2+\tau}), (x_3, x_{3+\tau}), \ldots$

delay

A pipeline for topological time series analysis



Euclidean embedding of time series data dates back to Takens's work on fluid turbulence.

Theorem (Takens 1981). Let M be a compact manifold of dimension n. Given pairs (ϕ, y) with $\phi : M \to M$ a smooth diffeomorphism and $y : M \to \mathbb{R}$ a smooth function, it is a generic property that the map $\Phi_{(\phi, y)} : M \to \mathbb{R}^{2n+1}$ defined by

$$\Phi_{(\varphi,y)}(x) = \Big(y(x), y\big(\varphi(x)\big), \ldots, y\big(\varphi^{2n}(x)\big)\Big)$$

is an embedding.

A pipeline for topological time series analysis

Time series data: $x_1, x_2, x_3, x_4, \ldots$

Preprocessing → Time-delay embedding (TDE)

dimension

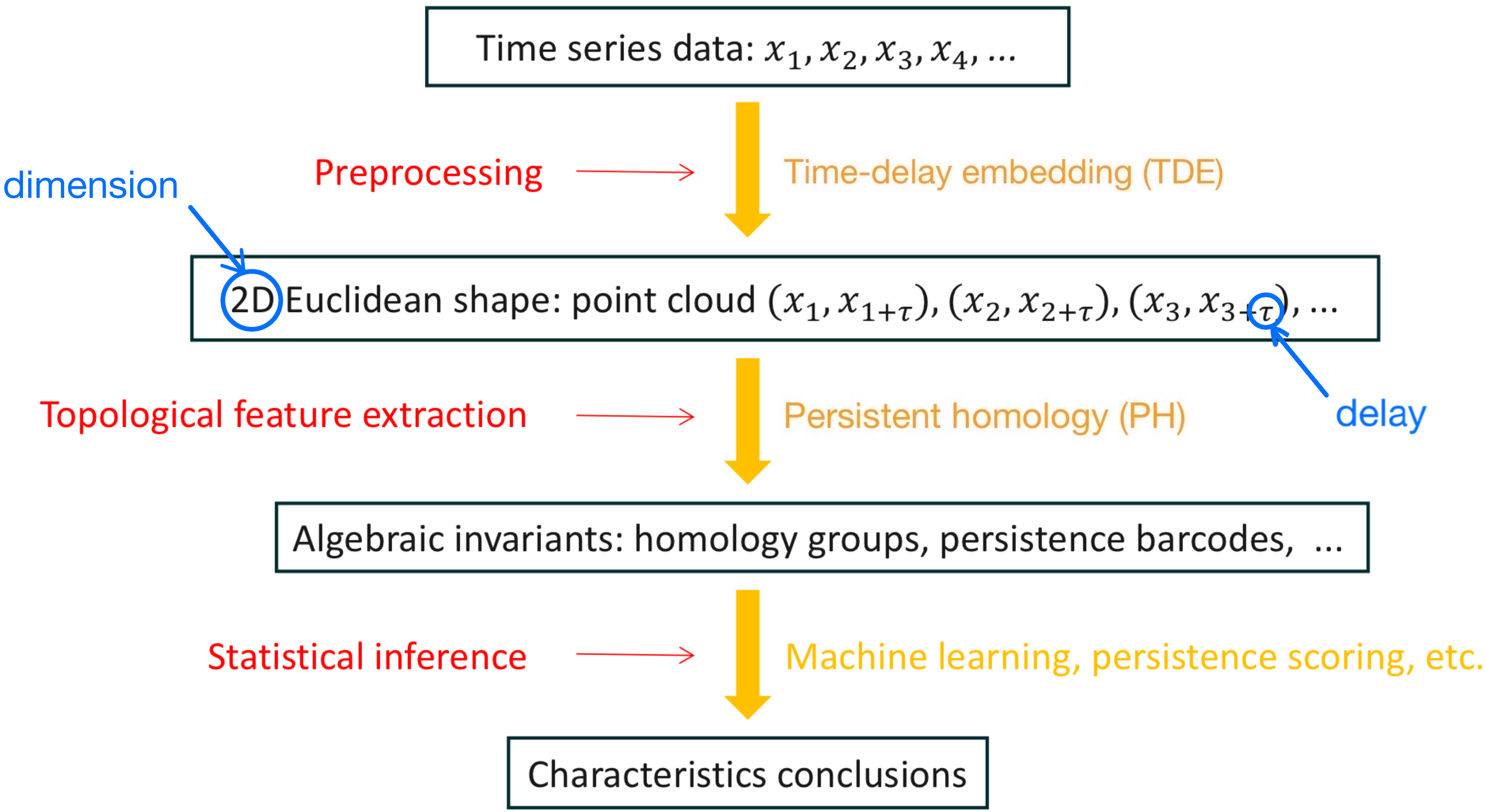
2D Euclidean shape: point cloud $(x_1, x_{1+\tau}), (x_2, x_{2+\tau}), (x_3, x_{3+\tau}), \ldots$

delay

Topological feature extraction → Persistent homology (PH)

Algebraic invariants: homology groups, persistence barcodes, ...

A pipeline for topological time series analysis



Classification of speech signals

In consultation with Meng Yu of Tencent AI Lab, we applied topological methods to classify voiced/voiceless and vowel/consonant speech data

Classification of speech signals

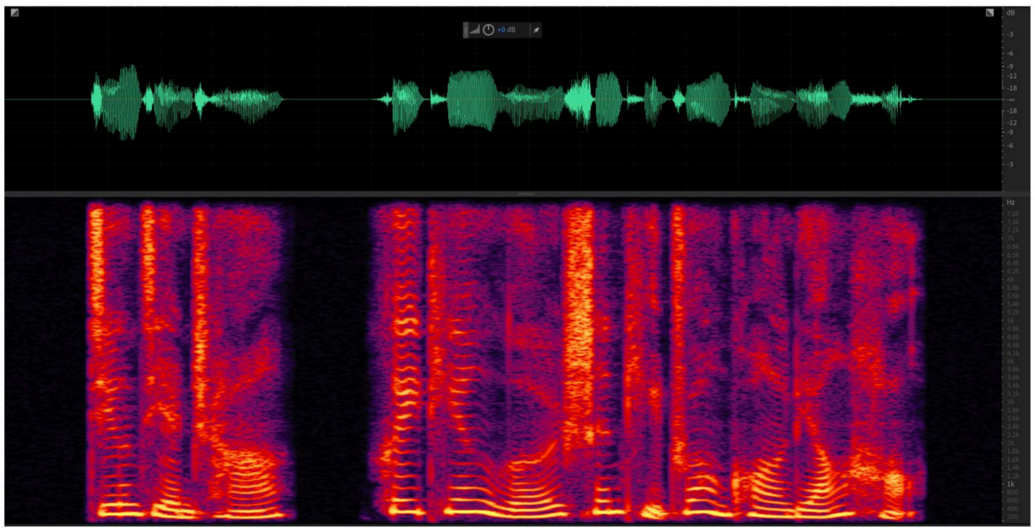
In consultation with Meng Yu of Tencent AI Lab, we applied topological methods to classify voiced/voiceless and vowel/consonant speech data, with motivations from industrial applications, including medical diagnosis, neurophysiology, speaker identification and other AI innovations.

Classification of speech signals

In consultation with Meng Yu of Tencent AI Lab, we applied topological methods to classify voiced/voiceless and vowel/consonant speech data, with motivations from industrial applications, including medical diagnosis, neurophysiology, speaker identification and other AI innovations.

Spectrograms

There are speech signal processing softwares for professional use.

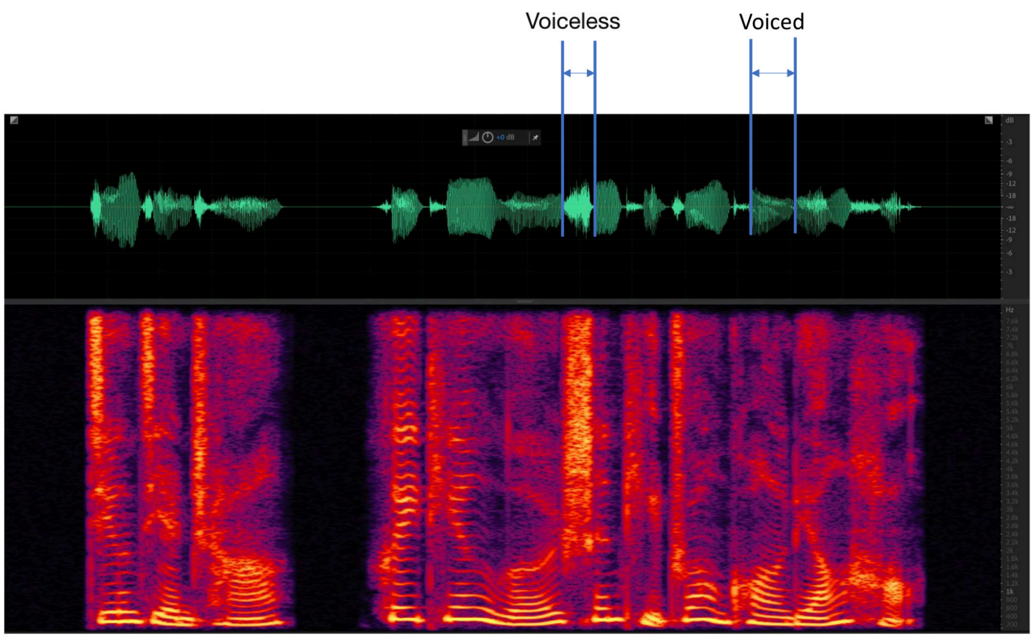


Classification of speech signals

In consultation with Meng Yu of Tencent AI Lab, we applied topological methods to classify voiced/voiceless and vowel/consonant speech data, with motivations from industrial applications, including medical diagnosis, neurophysiology, speaker identification and other AI innovations.

Spectrograms

There are speech signal processing softwares for professional use.



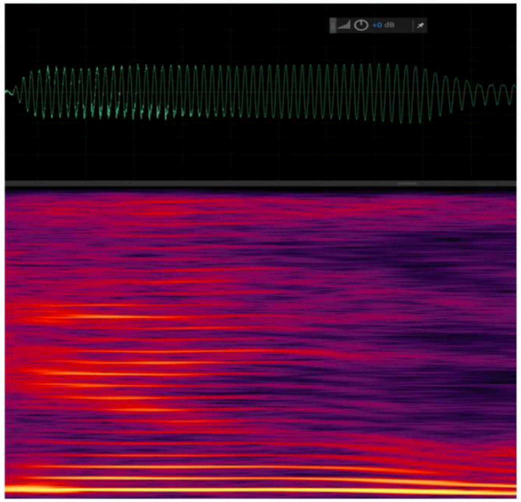
Classification of speech signals

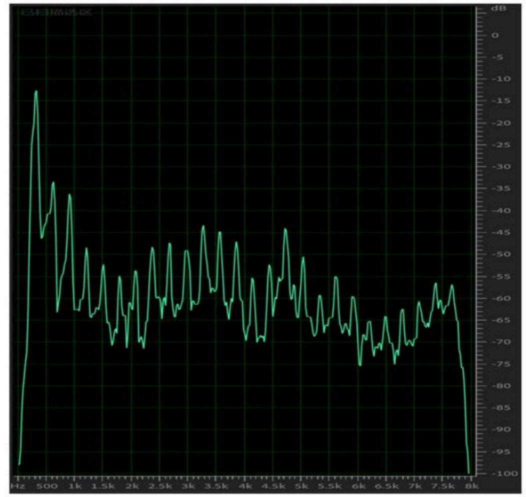
Voiced

[ŋ], [m], [n], [j], [l], [v], [ʒ], *etc.*

Sinusoid in time domain

Harmonics in frequency domain





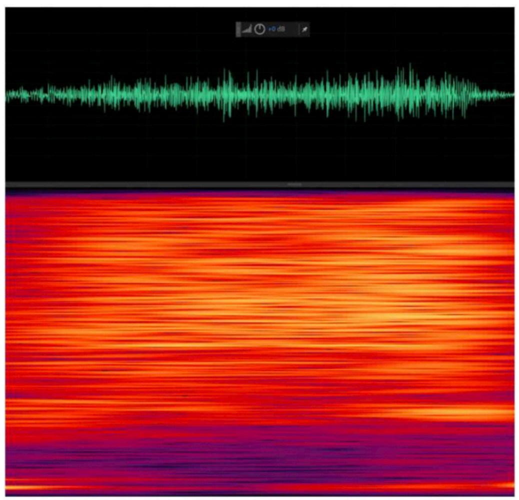
Time and Time-Frequency domain

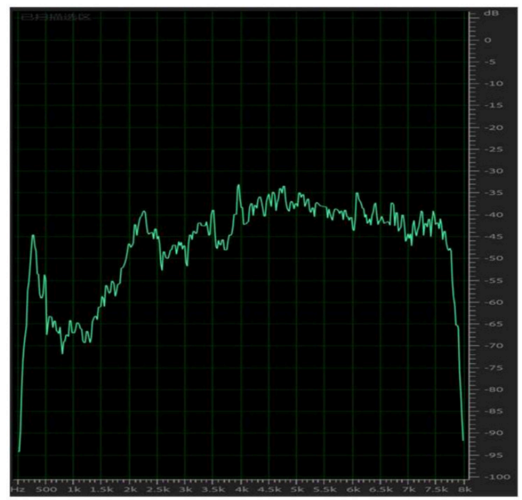
Frequency response

Voiceless

[f], [k], [θ], [t], [s], [tʃ], *etc.*

Like a white noise





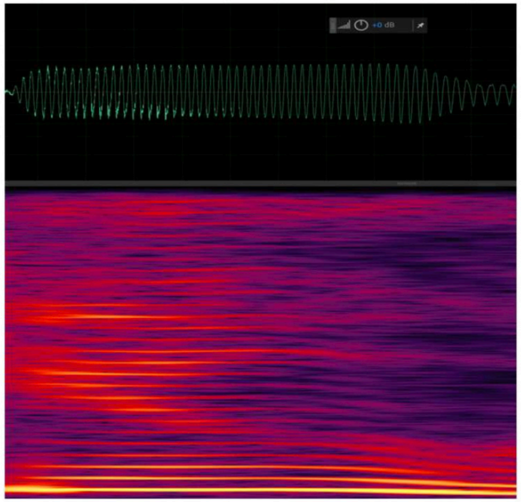
Classification of speech signals

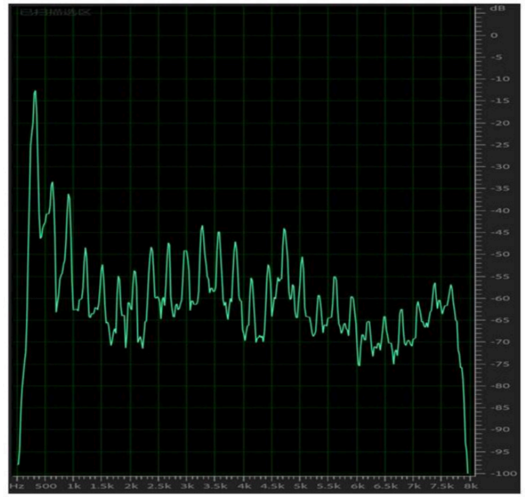
Voiced

exhibit periodic waveforms resulting from *glottal vibrations*

Sinusoid in time domain

Harmonics in frequency domain





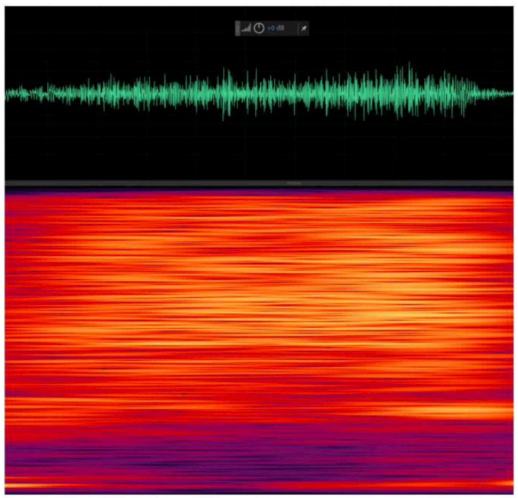
Time and Time-Frequency domain

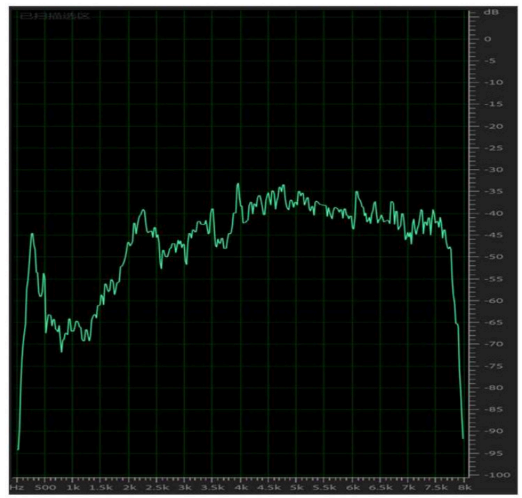
Frequency response

Voiceless

predominantly characterized by aperiodic, turbulence-induced noise

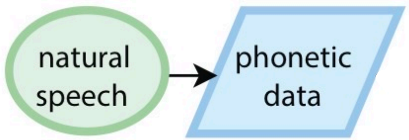
Like a white noise





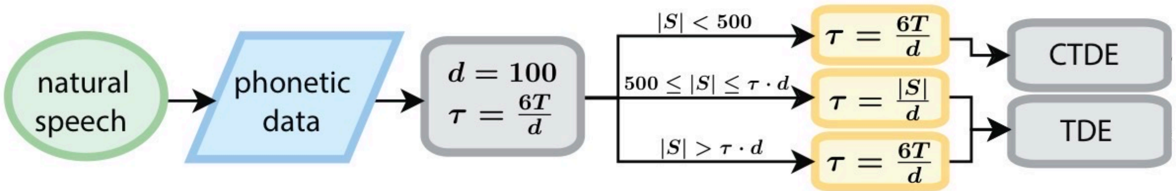
Primary experiments combining topological features with ML models

Here is a flowchart for our method of TopCap:



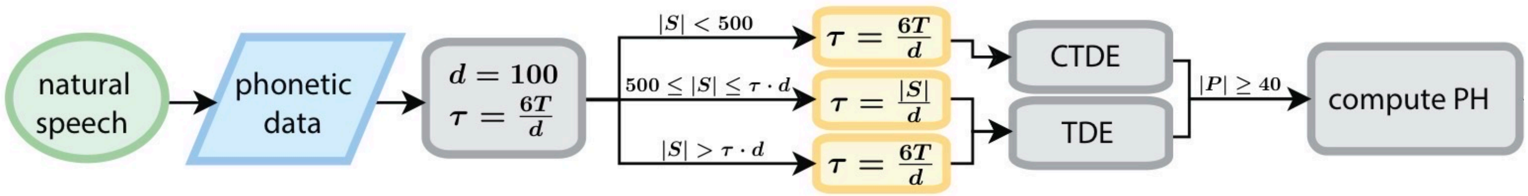
Primary experiments combining topological features with ML models

Here is a flowchart for our method of TopCap:



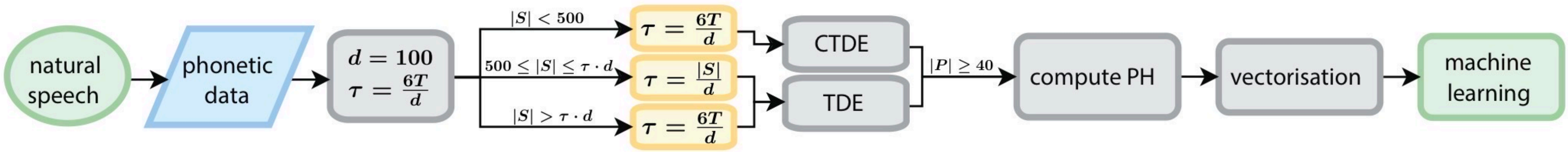
Primary experiments combining topological features with ML models

Here is a flowchart for our method of TopCap:



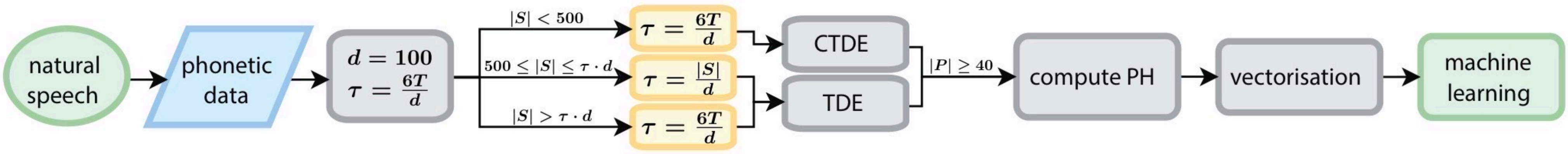
Primary experiments combining topological features with ML models

Here is a flowchart for our method of TopCap:

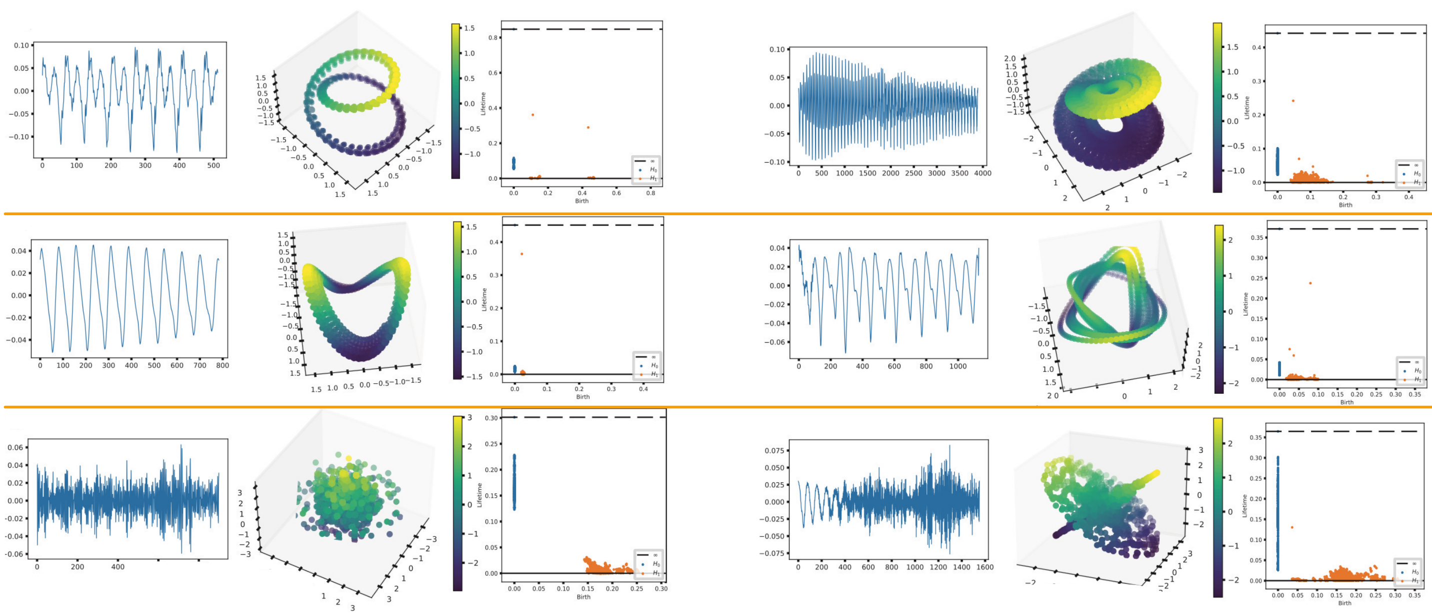


Primary experiments combining topological features with ML models

Here is a flowchart for our method of TopCap:



Topological profiles for vowels and consonants

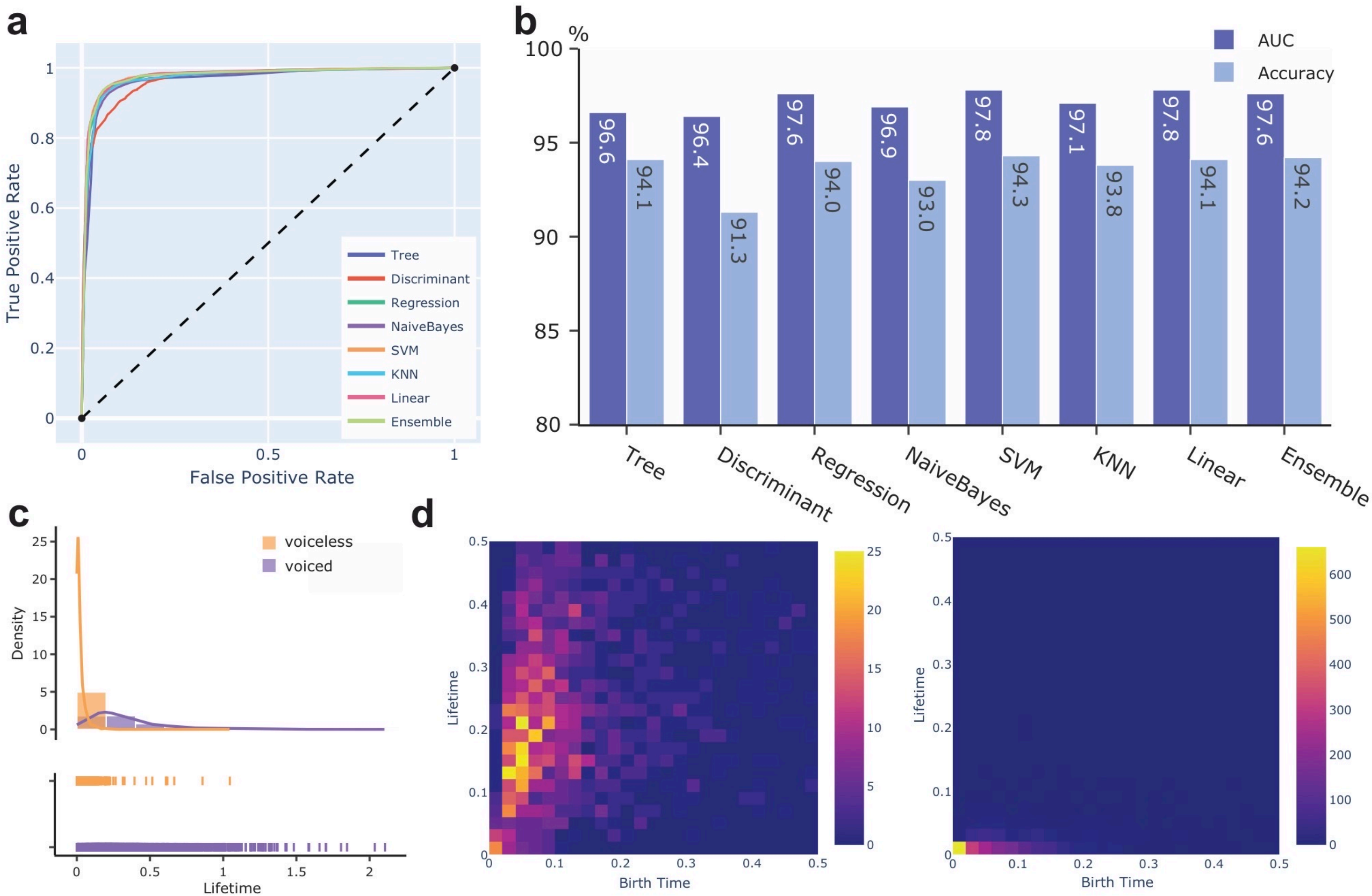


vowels

voiced consonants

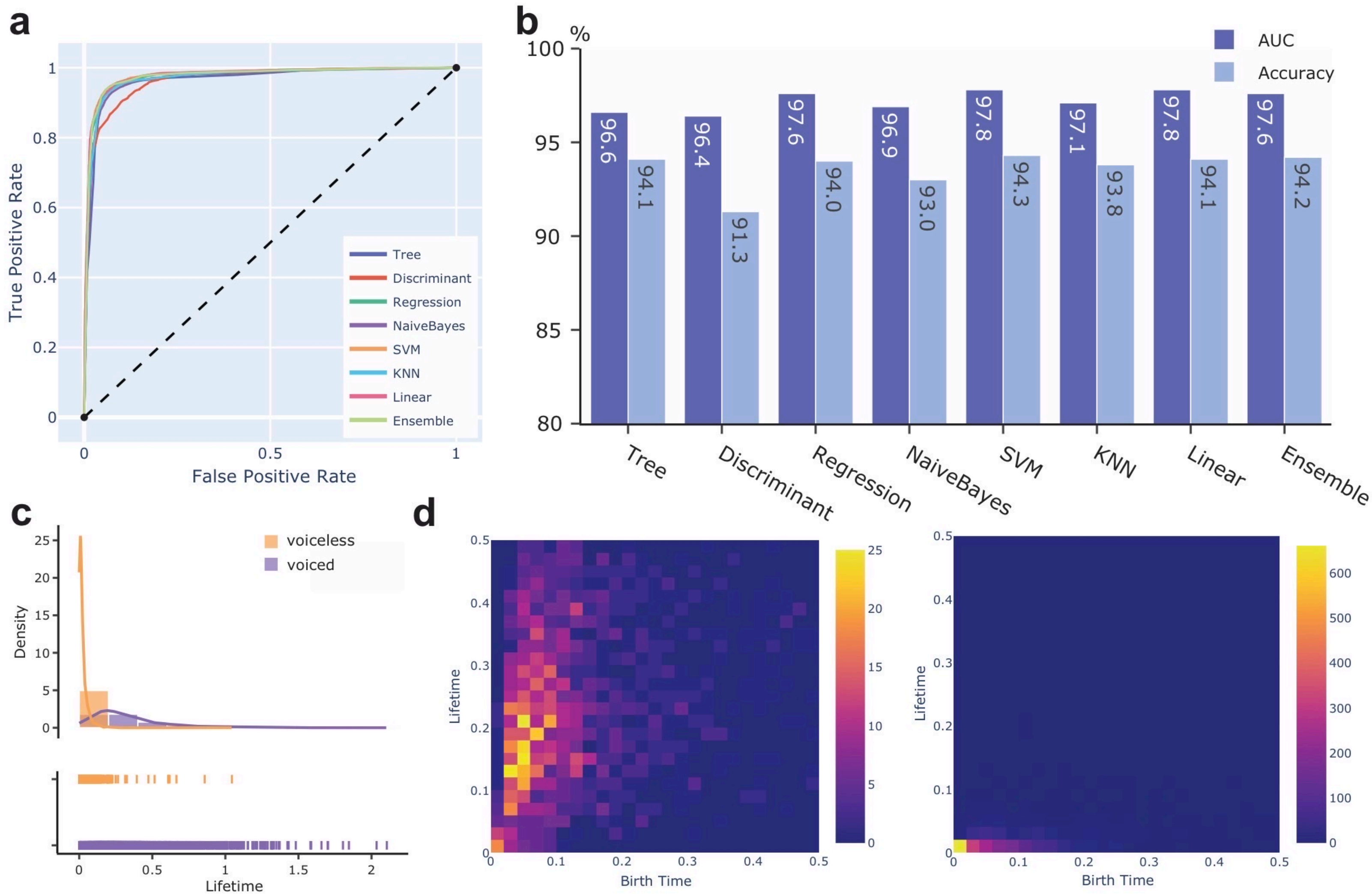
voiceless consonants

Primary experiments combining topological features with ML models



Machine learning results with topological features

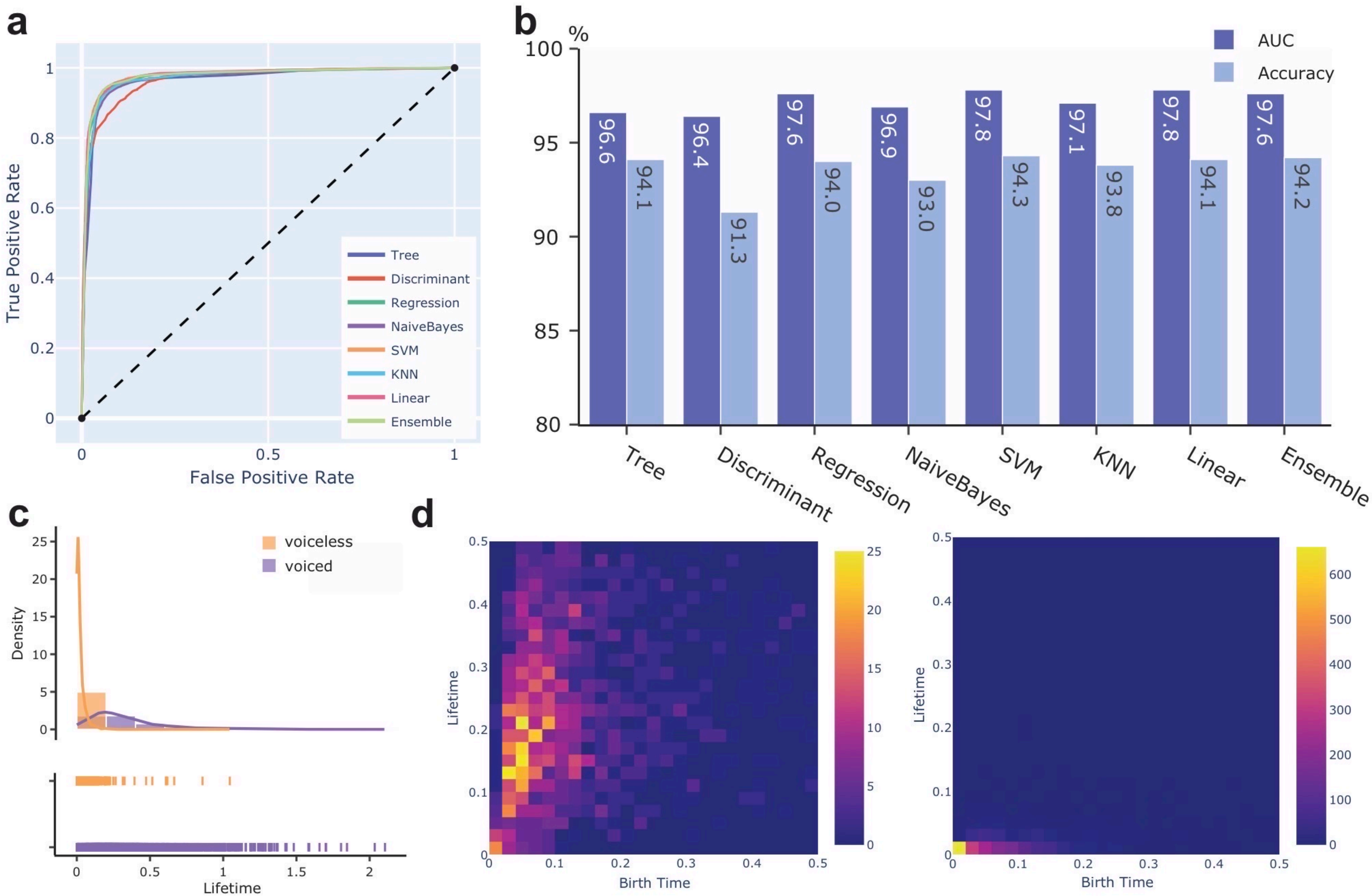
Primary experiments combining topological features with ML models



Machine learning results with topological features

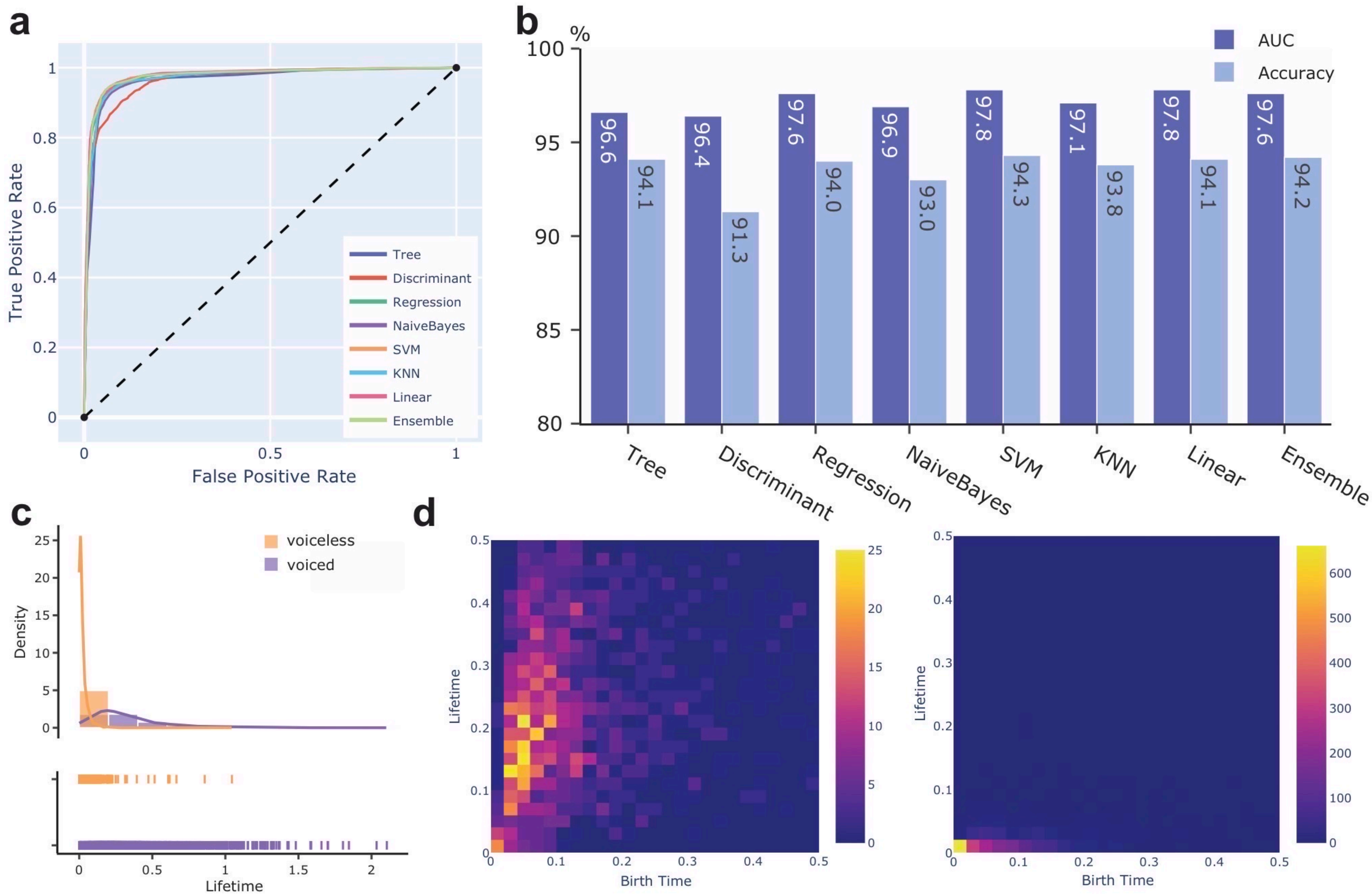
***a**, Receiver operating characteristic curves of traditional machine learning algorithms.*

Primary experiments combining topological features with ML models



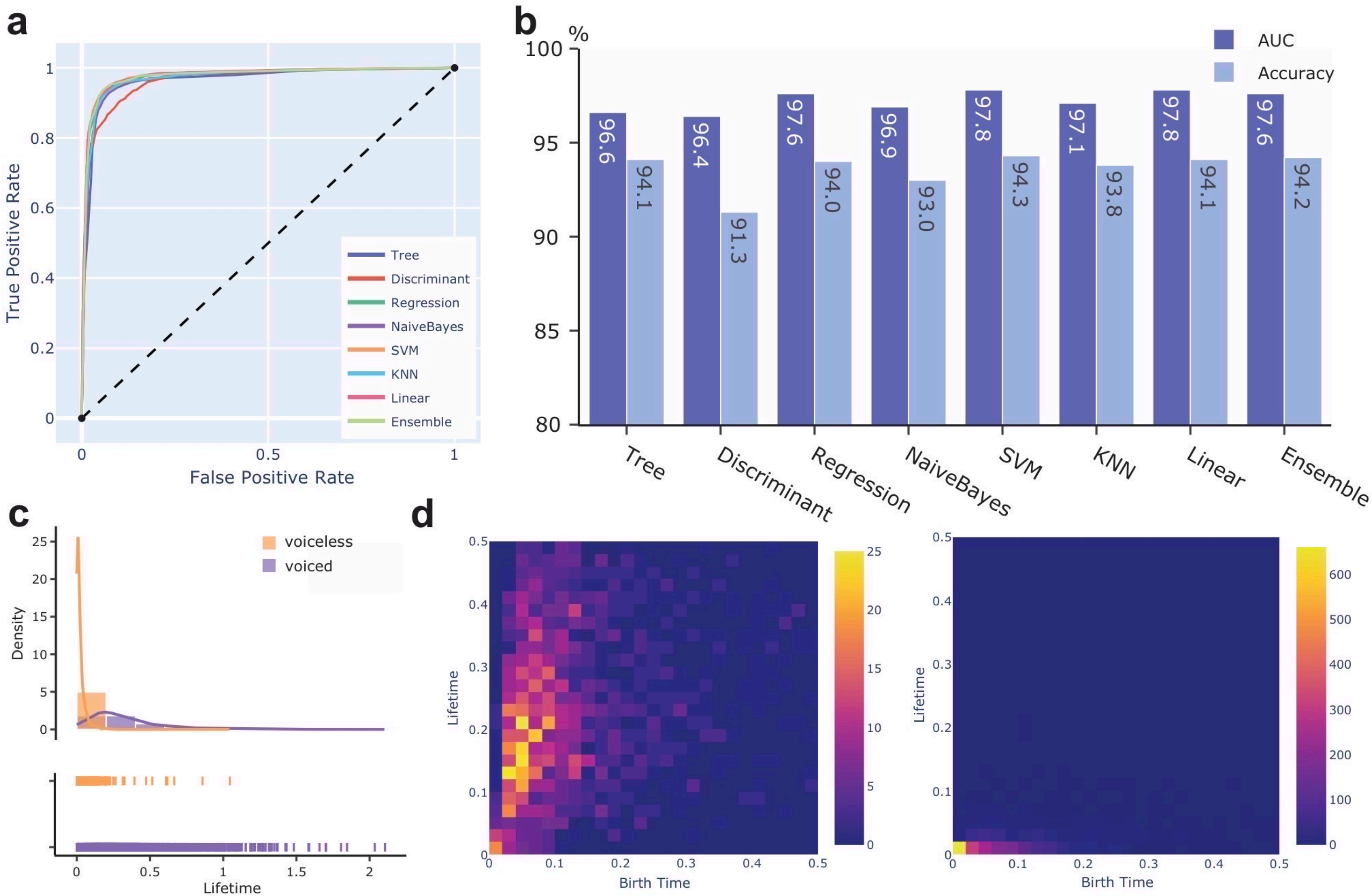
Machine learning results with topological features
***b**, Accuracy and area under the curve of each of these algorithms.*

Primary experiments combining topological features with ML models



Machine learning results with topological features
***c**, Histograms of records represented by their PH-lifetime for voiced and voiceless consonants, together with kernel density estimation and rug plot. The distributions of maximal persistence can distinguish voiced and voiceless consonants.*

Primary experiments combining topological features with ML models



Machine learning results with topological features
***d**, Diagrams of records represented as (birth time, lifetime) for voiced consonants (left) and voiceless consonants (right), where voiced consonants exhibit higher birth time and lifetime. The color represents the density of points in each unit grid box.*

Model comparison on benchmark datasets

	ALLSSTAR corpora					Random samples		
Small dataset	HT1	HT2	DHR	LPP	NWS	LJ	TIMIT	Libri
Number of phones	3200	3000	3600	3800	1800	2000	2000	2000
TopCap	94.3	92.7	92.3	91.9	88.8	94.6	83.9	85.1
MFCC–GRU	93.3	92.2	93.2	91.4	89.8	86.0	70.5	79.0
MFCC–Transformer	96.0	93.9	94.2	92.4	94.4	92.0	96.3	87.5
STFT–CNN-8	87.1	84.0	78.2	79.1	79.9	82.7	76.3	77.5
STFT–CNN-16	96.7	95.1	94.4	92.1	94.0	95.6	89.4	88.7

Large dataset	ALLSSTAR	LJSpeech	TIMIT	LibriSpeech
Number of phones	21000	257000	42000	500000
TopCap	92.5	92.9	92.8	88.7
MFCC–GRU	93.9	96.2	97.4	91.0
MFCC–Transformer	93.7	96.9	97.6	92.1
STFT–CNN-8	81.2	85.4	77.5	80.3
STFT–CNN-16	94.6	96.3	91.4	90.6

Accuracy rates % of TopCap on 8 small datasets and 4 large datasets stand in comparison with state-of-the-art methods.

Model comparison on benchmark datasets

	ALLSSTAR corpora					Random samples		
Small dataset	HT1	HT2	DHR	LPP	NWS	LJ	TIMIT	Libri
Number of phones	3200	3000	3600	3800	1800	2000	2000	2000
TopCap	94.3	92.7	92.3	91.9	88.8	94.6	83.9	85.1
MFCC–GRU	93.3	92.2	93.2	91.4	89.8	86.0	70.5	79.0
MFCC–Transformer	96.0	93.9	94.2	92.4	94.4	92.0	96.3	87.5
STFT–CNN-8	87.1	84.0	78.2	79.1	79.9	82.7	76.3	77.5
STFT–CNN-16	96.7	95.1	94.4	92.1	94.0	95.6	89.4	88.7

Large dataset	ALLSSTAR	LJSpeech	TIMIT	LibriSpeech
Number of phones	21000	257000	42000	500000
TopCap	92.5	92.9	92.8	88.7
MFCC–GRU	93.9	96.2	97.4	91.0
MFCC–Transformer	93.7	96.9	97.6	92.1
STFT–CNN-8	81.2	85.4	77.5	80.3
STFT–CNN-16	94.6	96.3	91.4	90.6

Accuracy rates % of TopCap on 8 small datasets and 4 large datasets stand in comparison with state-of-the-art methods. While MFCC–Transformer and STFT–CNN-16 generally outperform TopCap, it is important to note that TopCap exceeds the performance of MFCC–GRU (gated recurrent unit, which also uses advanced architecture) and STFT–CNN-8 (convolutional neural network, a smaller model than STFT–CNN-16) on small datasets.

Model comparison on benchmark datasets

	ALLSSTAR corpora					Random samples		
Small dataset	HT1	HT2	DHR	LPP	NWS	LJ	TIMIT	Libri
Number of phones	3200	3000	3600	3800	1800	2000	2000	2000
TopCap	94.3	92.7	92.3	91.9	88.8	94.6	83.9	85.1
MFCC–GRU	93.3	92.2	93.2	91.4	89.8	86.0	70.5	79.0
MFCC–Transformer	96.0	93.9	94.2	92.4	94.4	92.0	96.3	87.5
STFT–CNN-8	87.1	84.0	78.2	79.1	79.9	82.7	76.3	77.5
STFT–CNN-16	96.7	95.1	94.4	92.1	94.0	95.6	89.4	88.7

Large dataset	ALLSSTAR	LJSpeech	TIMIT	LibriSpeech
Number of phones	21000	257000	42000	500000
TopCap	92.5	92.9	92.8	88.7
MFCC–GRU	93.9	96.2	97.4	91.0
MFCC–Transformer	93.7	96.9	97.6	92.1
STFT–CNN-8	81.2	85.4	77.5	80.3
STFT–CNN-16	94.6	96.3	91.4	90.6

Accuracy rates % of TopCap on 8 small datasets and 4 large datasets stand in comparison with state-of-the-art methods. While MFCC–Transformer and STFT–CNN-16 generally outperform TopCap, it is important to note that TopCap exceeds the performance of MFCC–GRU (gated recurrent unit, which also uses advanced architecture) and STFT–CNN-8 (convolutional neural network, a smaller model than STFT–CNN-16) on small datasets. For larger datasets, TopCap generally does not match the performance of deep neural networks, primarily due to its use of simpler topological features and basic machine learning models.

Model comparison on benchmark datasets

	ALLSSTAR corpora					Random samples		
Small dataset	HT1	HT2	DHR	LPP	NWS	LJ	TIMIT	Libri
Number of phones	3200	3000	3600	3800	1800	2000	2000	2000
TopCap	94.3	92.7	92.3	91.9	88.8	94.6	83.9	85.1
MFCC–GRU	93.3	92.2	93.2	91.4	89.8	86.0	70.5	79.0
MFCC–Transformer	96.0	93.9	94.2	92.4	94.4	92.0	96.3	87.5
STFT–CNN-8	87.1	84.0	78.2	79.1	79.9	82.7	76.3	77.5
STFT–CNN-16	96.7	95.1	94.4	92.1	94.0	95.6	89.4	88.7

Large dataset	ALLSSTAR	LJSpeech	TIMIT	LibriSpeech
Number of phones	21000	257000	42000	500000
TopCap	92.5	92.9	92.8	88.7
MFCC–GRU	93.9	96.2	97.4	91.0
MFCC–Transformer	93.7	96.9	97.6	92.1
STFT–CNN-8	81.2	85.4	77.5	80.3
STFT–CNN-16	94.6	96.3	91.4	90.6

Accuracy rates % of TopCap on 8 small datasets and 4 large datasets stand in comparison with state-of-the-art methods. While MFCC–Transformer and STFT–CNN-16 generally outperform TopCap, it is important to note that TopCap exceeds the performance of MFCC–GRU (gated recurrent unit, which also uses advanced architecture) and STFT–CNN-8 (convolutional neural network, a smaller model than STFT–CNN-16) on small datasets. For larger datasets, TopCap generally does not match the performance of deep neural networks, primarily due to its use of simpler topological features and basic machine learning models. This limitation motivates the integration TopNN of topological features into neural networks.

Model comparison on benchmark datasets

	ALLSSTAR corpora					Random samples		
Small dataset	HT1	HT2	DHR	LPP	NWS	LJ	TIMIT	Libri
Number of phones	3200	3000	3600	3800	1800	2000	2000	2000
TopCap	94.3	92.7	92.3	91.9	88.8	94.6	83.9	85.1
MFCC–GRU	93.3	92.2	93.2	91.4	89.8	86.0	70.5	79.0
MFCC–Transformer	96.0	93.9	94.2	92.4	94.4	92.0	96.3	87.5
STFT–CNN-8	87.1	84.0	78.2	79.1	79.9	82.7	76.3	77.5
STFT–CNN-16	96.7	95.1	94.4	92.1	94.0	95.6	89.4	88.7

Large dataset	ALLSSTAR	LJSpeech	TIMIT	LibriSpeech
Number of phones	21000	257000	42000	500000
TopCap	92.5	92.9	92.8	88.7
MFCC–GRU	93.9	96.2	97.4	91.0
MFCC–Transformer	93.7	96.9	97.6	92.1
STFT–CNN-8	81.2	85.4	77.5	80.3
STFT–CNN-16	94.6	96.3	91.4	90.6

Accuracy rates % of TopCap on 8 small datasets and 4 large datasets stand in comparison with state-of-the-art methods. While MFCC–Transformer and STFT–CNN-16 generally outperform TopCap, it is important to note that TopCap exceeds the performance of MFCC–GRU (gated recurrent unit, which also uses advanced architecture) and STFT–CNN-8 (convolutional neural network, a smaller model than STFT–CNN-16) on small datasets. For larger datasets, TopCap generally does not match the performance of deep neural networks, primarily due to its use of simpler topological features and basic machine learning models. This limitation motivates the integration TopNN of topological features into neural networks. Overall, while TopCap may not achieve the highest performance across all benchmarks, it produces decent results.

Model comparison on benchmark datasets

	ALLSSTAR corpora					Random samples		
Small dataset	HT1	HT2	DHR	LPP	NWS	LJ	TIMIT	Libri
Number of phones	3200	3000	3600	3800	1800	2000	2000	2000
TopCap	94.3	92.7	92.3	91.9	88.8	94.6	83.9	85.1
MFCC–GRU	93.3	92.2	93.2	91.4	89.8	86.0	70.5	79.0
MFCC–Transformer	96.0	93.9	94.2	92.4	94.4	92.0	96.3	87.5
STFT–CNN-8	87.1	84.0	78.2	79.1	79.9	82.7	76.3	77.5
STFT–CNN-16	96.7	95.1	94.4	92.1	94.0	95.6	89.4	88.7

Large dataset	ALLSSTAR	LJSpeech	TIMIT	LibriSpeech
Number of phones	21000	257000	42000	500000
TopCap	92.5	92.9	92.8	88.7
MFCC–GRU	93.9	96.2	97.4	91.0
MFCC–Transformer	93.7	96.9	97.6	92.1
STFT–CNN-8	81.2	85.4	77.5	80.3
STFT–CNN-16	94.6	96.3	91.4	90.6

Advantages of TopCap:

- ***Structural efficiency.*** *Neural network models require further feature extraction from input MFCC sequences or STFT spectrograms for classification tasks, necessitating a training process which lengthens with the growing dataset.*

Model comparison on benchmark datasets

	ALLSSTAR corpora					Random samples		
Small dataset	HT1	HT2	DHR	LPP	NWS	LJ	TIMIT	Libri
Number of phones	3200	3000	3600	3800	1800	2000	2000	2000
TopCap	94.3	92.7	92.3	91.9	88.8	94.6	83.9	85.1
MFCC–GRU	93.3	92.2	93.2	91.4	89.8	86.0	70.5	79.0
MFCC–Transformer	96.0	93.9	94.2	92.4	94.4	92.0	96.3	87.5
STFT–CNN-8	87.1	84.0	78.2	79.1	79.9	82.7	76.3	77.5
STFT–CNN-16	96.7	95.1	94.4	92.1	94.0	95.6	89.4	88.7

Large dataset	ALLSSTAR	LJSpeech	TIMIT	LibriSpeech
Number of phones	21000	257000	42000	500000
TopCap	92.5	92.9	92.8	88.7
MFCC–GRU	93.9	96.2	97.4	91.0
MFCC–Transformer	93.7	96.9	97.6	92.1
STFT–CNN-8	81.2	85.4	77.5	80.3
STFT–CNN-16	94.6	96.3	91.4	90.6

Advantages of TopCap:

- ***Structural efficiency.*** *Neural network models require further feature extraction from input MFCC sequences or STFT spectrograms for classification tasks, necessitating a training process which lengthens with the growing dataset. In contrast, TopCap mainly utilizes topology-based methods (TDE and PH) which are more straightforward for feature extraction.*

Model comparison on benchmark datasets

	ALLSSTAR corpora					Random samples		
Small dataset	HT1	HT2	DHR	LPP	NWS	LJ	TIMIT	Libri
Number of phones	3200	3000	3600	3800	1800	2000	2000	2000
TopCap	94.3	92.7	92.3	91.9	88.8	94.6	83.9	85.1
MFCC–GRU	93.3	92.2	93.2	91.4	89.8	86.0	70.5	79.0
MFCC–Transformer	96.0	93.9	94.2	92.4	94.4	92.0	96.3	87.5
STFT–CNN-8	87.1	84.0	78.2	79.1	79.9	82.7	76.3	77.5
STFT–CNN-16	96.7	95.1	94.4	92.1	94.0	95.6	89.4	88.7

Large dataset	ALLSSTAR	LJSpeech	TIMIT	LibriSpeech
Number of phones	21000	257000	42000	500000
TopCap	92.5	92.9	92.8	88.7
MFCC–GRU	93.9	96.2	97.4	91.0
MFCC–Transformer	93.7	96.9	97.6	92.1
STFT–CNN-8	81.2	85.4	77.5	80.3
STFT–CNN-16	94.6	96.3	91.4	90.6

Advantages of TopCap:

- ***Structural efficiency.*** *Neural network models require further feature extraction from input MFCC sequences or STFT spectrograms for classification tasks, necessitating a training process which lengthens with the growing dataset. In contrast, TopCap mainly utilizes topology-based methods (TDE and PH) which are more straightforward for feature extraction. Meanwhile, the topological fingerprints (e.g., maximal persistence) are strong enough to characterize phonemes effectively for our classification tasks.*

Model comparison on benchmark datasets

	ALLSSTAR corpora					Random samples		
Small dataset	HT1	HT2	DHR	LPP	NWS	LJ	TIMIT	Libri
Number of phones	3200	3000	3600	3800	1800	2000	2000	2000
TopCap	94.3	92.7	92.3	91.9	88.8	94.6	83.9	85.1
MFCC–GRU	93.3	92.2	93.2	91.4	89.8	86.0	70.5	79.0
MFCC–Transformer	96.0	93.9	94.2	92.4	94.4	92.0	96.3	87.5
STFT–CNN-8	87.1	84.0	78.2	79.1	79.9	82.7	76.3	77.5
STFT–CNN-16	96.7	95.1	94.4	92.1	94.0	95.6	89.4	88.7

Large dataset	ALLSSTAR	LJSpeech	TIMIT	LibriSpeech
Number of phones	21000	257000	42000	500000
TopCap	92.5	92.9	92.8	88.7
MFCC–GRU	93.9	96.2	97.4	91.0
MFCC–Transformer	93.7	96.9	97.6	92.1
STFT–CNN-8	81.2	85.4	77.5	80.3
STFT–CNN-16	94.6	96.3	91.4	90.6

Advantages of TopCap:

- ***Structural efficiency.*** *Neural network models require further feature extraction from input MFCC sequences or STFT spectrograms for classification tasks, necessitating a training process which lengthens with the growing dataset. In contrast, TopCap mainly utilizes topology-based methods (TDE and PH) which are more straightforward for feature extraction. Meanwhile, the topological fingerprints (e.g., maximal persistence) are strong enough to characterize phonemes effectively for our classification tasks. Therefore, TopCap gains higher efficiency, especially when handling larger datasets.*

Model comparison on benchmark datasets

	ALLSSTAR corpora					Random samples		
Small dataset	HT1	HT2	DHR	LPP	NWS	LJ	TIMIT	Libri
Number of phones	3200	3000	3600	3800	1800	2000	2000	2000
TopCap	94.3	92.7	92.3	91.9	88.8	94.6	83.9	85.1
MFCC–GRU	93.3	92.2	93.2	91.4	89.8	86.0	70.5	79.0
MFCC–Transformer	96.0	93.9	94.2	92.4	94.4	92.0	96.3	87.5
STFT–CNN-8	87.1	84.0	78.2	79.1	79.9	82.7	76.3	77.5
STFT–CNN-16	96.7	95.1	94.4	92.1	94.0	95.6	89.4	88.7

Large dataset	ALLSSTAR	LJSpeech	TIMIT	LibriSpeech
Number of phones	21000	257000	42000	500000
TopCap	92.5	92.9	92.8	88.7
MFCC–GRU	93.9	96.2	97.4	91.0
MFCC–Transformer	93.7	96.9	97.6	92.1
STFT–CNN-8	81.2	85.4	77.5	80.3
STFT–CNN-16	94.6	96.3	91.4	90.6

Advantages of TopCap:

- ***Structural efficiency.*** *Neural network models require further feature extraction from input MFCC sequences or STFT spectrograms for classification tasks, necessitating a training process which lengthens with the growing dataset. In contrast, TopCap mainly utilizes topology-based methods (TDE and PH) which are more straightforward for feature extraction. Meanwhile, the topological fingerprints (e.g., maximal persistence) are strong enough to characterize phonemes effectively for our classification tasks. Therefore, TopCap gains higher efficiency, especially when handling larger datasets. On a related note, deep learning methods, as a data-driven approach, require large amounts of data for training and generalization.*

Model comparison on benchmark datasets

	ALLSSTAR corpora					Random samples		
Small dataset	HT1	HT2	DHR	LPP	NWS	LJ	TIMIT	Libri
Number of phones	3200	3000	3600	3800	1800	2000	2000	2000
TopCap	94.3	92.7	92.3	91.9	88.8	94.6	83.9	85.1
MFCC–GRU	93.3	92.2	93.2	91.4	89.8	86.0	70.5	79.0
MFCC–Transformer	96.0	93.9	94.2	92.4	94.4	92.0	96.3	87.5
STFT–CNN-8	87.1	84.0	78.2	79.1	79.9	82.7	76.3	77.5
STFT–CNN-16	96.7	95.1	94.4	92.1	94.0	95.6	89.4	88.7

Large dataset	ALLSSTAR	LJSpeech	TIMIT	LibriSpeech
Number of phones	21000	257000	42000	500000
TopCap	92.5	92.9	92.8	88.7
MFCC–GRU	93.9	96.2	97.4	91.0
MFCC–Transformer	93.7	96.9	97.6	92.1
STFT–CNN-8	81.2	85.4	77.5	80.3
STFT–CNN-16	94.6	96.3	91.4	90.6

Advantages of TopCap:

- ***Structural efficiency.*** *Neural network models require further feature extraction from input MFCC sequences or STFT spectrograms for classification tasks, necessitating a training process which lengthens with the growing dataset. In contrast, TopCap mainly utilizes topology-based methods (TDE and PH) which are more straightforward for feature extraction. Meanwhile, the topological fingerprints (e.g., maximal persistence) are strong enough to characterize phonemes effectively for our classification tasks. Therefore, TopCap gains higher efficiency, especially when handling larger datasets. On a related note, deep learning methods, as a data-driven approach, require large amounts of data for training and generalization. In contrast, comparing the upper and lower halves of the above table, we see that TopCap achieves equally good performance on relatively small datasets.*

Model comparison on benchmark datasets

	ALLSSTAR corpora					Random samples		
Small dataset	HT1	HT2	DHR	LPP	NWS	LJ	TIMIT	Libri
Number of phones	3200	3000	3600	3800	1800	2000	2000	2000
TopCap	94.3	92.7	92.3	91.9	88.8	94.6	83.9	85.1
MFCC–GRU	93.3	92.2	93.2	91.4	89.8	86.0	70.5	79.0
MFCC–Transformer	96.0	93.9	94.2	92.4	94.4	92.0	96.3	87.5
STFT–CNN-8	87.1	84.0	78.2	79.1	79.9	82.7	76.3	77.5
STFT–CNN-16	96.7	95.1	94.4	92.1	94.0	95.6	89.4	88.7

Large dataset	ALLSSTAR	LJSpeech	TIMIT	LibriSpeech
Number of phones	21000	257000	42000	500000
TopCap	92.5	92.9	92.8	88.7
MFCC–GRU	93.9	96.2	97.4	91.0
MFCC–Transformer	93.7	96.9	97.6	92.1
STFT–CNN-8	81.2	85.4	77.5	80.3
STFT–CNN-16	94.6	96.3	91.4	90.6

Advantages of TopCap:

- ***Interpretability.*** *Neural networks are often referred to as "black boxes" due to their low explainability and interpretability, which make it challenging to understand the mechanisms of feature extraction and effectively improve a model for classification. However, TopCap offers a white-box method for visualizing features of time series data, which gives insight to the intrinsic properties and nuanced differences within the data, enabling us to better understand and improve the model.*

Model comparison on benchmark datasets

	ALLSSTAR corpora					Random samples		
Small dataset	HT1	HT2	DHR	LPP	NWS	LJ	TIMIT	Libri
Number of phones	3200	3000	3600	3800	1800	2000	2000	2000
TopCap	94.3	92.7	92.3	91.9	88.8	94.6	83.9	85.1
MFCC–GRU	93.3	92.2	93.2	91.4	89.8	86.0	70.5	79.0
MFCC–Transformer	96.0	93.9	94.2	92.4	94.4	92.0	96.3	87.5
STFT–CNN-8	87.1	84.0	78.2	79.1	79.9	82.7	76.3	77.5
STFT–CNN-16	96.7	95.1	94.4	92.1	94.0	95.6	89.4	88.7

Large dataset	ALLSSTAR	LJSpeech	TIMIT	LibriSpeech
Number of phones	21000	257000	42000	500000
TopCap	92.5	92.9	92.8	88.7
MFCC–GRU	93.9	96.2	97.4	91.0
MFCC–Transformer	93.7	96.9	97.6	92.1
STFT–CNN-8	81.2	85.4	77.5	80.3
STFT–CNN-16	94.6	96.3	91.4	90.6

Advantages of TopCap:

- ***Computational speed.*** *Neural networks involve time-consuming training processes, even with GPU acceleration. For instance, on the TIMIT dataset, a full training cycle of 15 epochs can take approximately 30 minutes with GPU parallelization.*

Model comparison on benchmark datasets

	ALLSSTAR corpora					Random samples		
Small dataset	HT1	HT2	DHR	LPP	NWS	LJ	TIMIT	Libri
Number of phones	3200	3000	3600	3800	1800	2000	2000	2000
TopCap	94.3	92.7	92.3	91.9	88.8	94.6	83.9	85.1
MFCC–GRU	93.3	92.2	93.2	91.4	89.8	86.0	70.5	79.0
MFCC–Transformer	96.0	93.9	94.2	92.4	94.4	92.0	96.3	87.5
STFT–CNN-8	87.1	84.0	78.2	79.1	79.9	82.7	76.3	77.5
STFT–CNN-16	96.7	95.1	94.4	92.1	94.0	95.6	89.4	88.7

Large dataset	ALLSSTAR	LJSpeech	TIMIT	LibriSpeech
Number of phones	21000	257000	42000	500000
TopCap	92.5	92.9	92.8	88.7
MFCC–GRU	93.9	96.2	97.4	91.0
MFCC–Transformer	93.7	96.9	97.6	92.1
STFT–CNN-8	81.2	85.4	77.5	80.3
STFT–CNN-16	94.6	96.3	91.4	90.6

Advantages of TopCap:

- ***Computational speed.*** *Neural networks involve time-consuming training processes, even with GPU acceleration. For instance, on the TIMIT dataset, a full training cycle of 15 epochs can take approximately 30 minutes with GPU parallelization. In contrast, TopCap bypasses the need for iterative training and achieves significantly faster computation. TopCap performs lightweight machine learning with negligible runtime overhead, completing both feature extraction and classification in just 2 minutes when utilizing 16-thread CPU parallelization.*

Model comparison on benchmark datasets

	ALLSSTAR corpora					Random samples		
Small dataset	HT1	HT2	DHR	LPP	NWS	LJ	TIMIT	Libri
Number of phones	3200	3000	3600	3800	1800	2000	2000	2000
TopCap	94.3	92.7	92.3	91.9	88.8	94.6	83.9	85.1
MFCC–GRU	93.3	92.2	93.2	91.4	89.8	86.0	70.5	79.0
MFCC–Transformer	96.0	93.9	94.2	92.4	94.4	92.0	96.3	87.5
STFT–CNN-8	87.1	84.0	78.2	79.1	79.9	82.7	76.3	77.5
STFT–CNN-16	96.7	95.1	94.4	92.1	94.0	95.6	89.4	88.7

Large dataset	ALLSSTAR	LJSpeech	TIMIT	LibriSpeech
Number of phones	21000	257000	42000	500000
TopCap	92.5	92.9	92.8	88.7
MFCC–GRU	93.9	96.2	97.4	91.0
MFCC–Transformer	93.7	96.9	97.6	92.1
STFT–CNN-8	81.2	85.4	77.5	80.3
STFT–CNN-16	94.6	96.3	91.4	90.6

Advantages of TopCap:

- ***Computational speed.*** *Neural networks involve time-consuming training processes, even with GPU acceleration. For instance, on the TIMIT dataset, a full training cycle of 15 epochs can take approximately 30 minutes with GPU parallelization. In contrast, TopCap bypasses the need for iterative training and achieves significantly faster computation. TopCap performs lightweight machine learning with negligible runtime overhead, completing both feature extraction and classification in just 2 minutes when utilizing 16-thread CPU parallelization. TopCap's efficiency advantage comes from avoiding gradient-based optimization and using computationally cheaper topologically derived features, along with a highly parallelizable pipeline. These make it significantly faster and more scalable especially for large datasets or real-time applications.*

From topological data analysis to topological deep learning

From topological data analysis to topological deep learning

Using persistent homology, Carlsson, Ishkhanov, de Silva, and Zomorodian qualitatively analyzed approximately 4.5×10^6 high-contrast local patches of natural images obtained by van Hateren and van der Schaaf and previously studied by Lee, Mumford, and Petersen.

From topological data analysis to topological deep learning

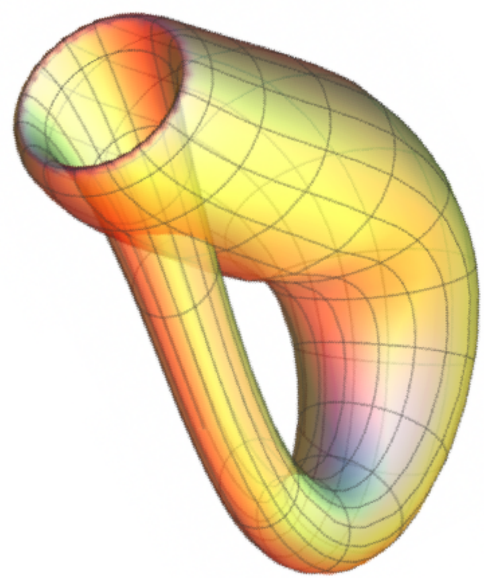
Using persistent homology, Carlsson, Ishkhanov, de Silva, and Zomorodian qualitatively analyzed approximately 4.5×10^6 high-contrast local patches of natural images obtained by van Hateren and van der Schaaf and previously studied by Lee, Mumford, and Petersen.

From topological data analysis to topological deep learning

Using persistent homology, Carlsson, Ishkhanov, de Silva, and Zomorodian qualitatively analyzed approximately 4.5×10^6 high-contrast local patches of natural images obtained by van Hateren and van der Schaaf and previously studied by Lee, Mumford, and Petersen. In their 2008 article, they discovered that as vectors of pixels, the image data were unevenly distributed over a Klein bottle within the 7-dimensional Euclidean sphere.

*Gunnar Carlsson et al., On the local behavior of spaces of natural images, **International Journal of Computer Vision**, 2008.*

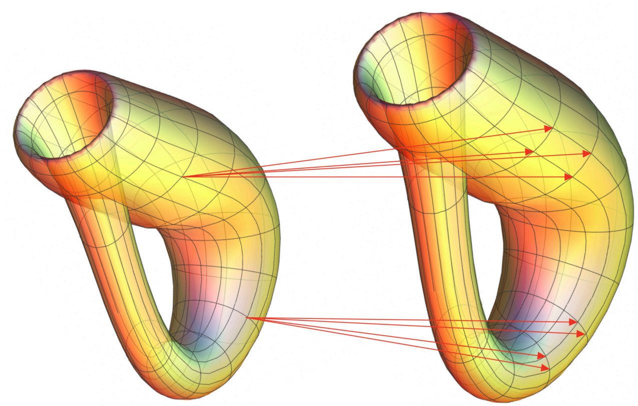
*Gunnar Carlsson, Topology and data, **Bulletin of the American Mathematical Society**, 2009.*



From topological data analysis to topological deep learning

Using persistent homology, Carlsson, Ishkhanov, de Silva, and Zomorodian qualitatively analyzed approximately 4.5×10^6 high-contrast local patches of natural images obtained by van Hateren and van der Schaaf and previously studied by Lee, Mumford, and Petersen. In their 2008 article, they discovered that as vectors of pixels, the image data were unevenly distributed over a Klein bottle within the 7-dimensional Euclidean sphere.

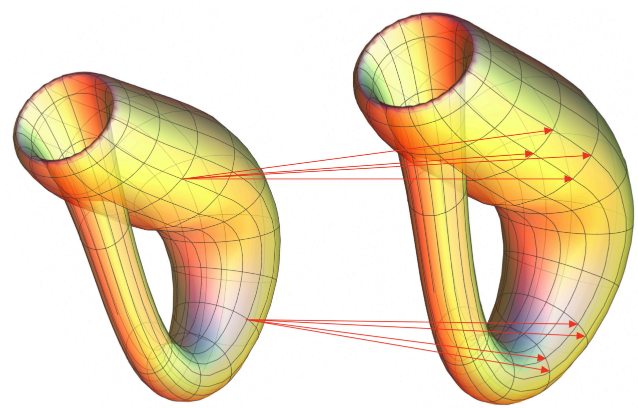
A decade later, Love, Filippenko, Maroulas, and Carlsson have made the Klein bottle as a topological input for designing convolutional layers in neural networks that learn image data.



From topological data analysis to topological deep learning

Using persistent homology, Carlsson, Ishkhanov, de Silva, and Zomorodian qualitatively analyzed approximately 4.5×10^6 high-contrast local patches of natural images obtained by van Hateren and van der Schaaf and previously studied by Lee, Mumford, and Petersen. In their 2008 article, they discovered that as vectors of pixels, the image data were unevenly distributed over a Klein bottle within the 7-dimensional Euclidean sphere.

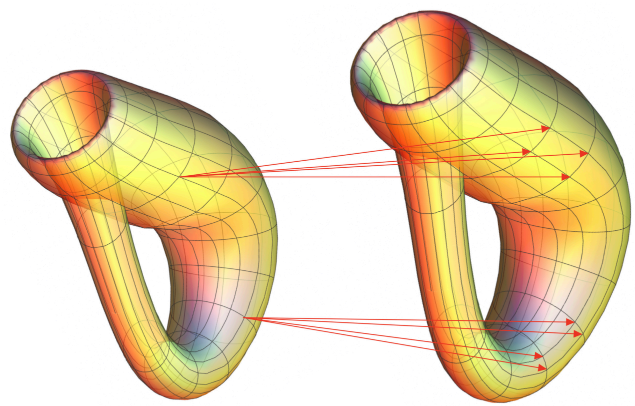
A decade later, Love, Filippenko, Maroulas, and Carlsson have made the Klein bottle as a topological input for designing convolutional layers in neural networks that learn image data. Moreover, they have incorporated the tangent bundle of a Klein bottle into TCNNs for learning video data.



From topological data analysis to topological deep learning

Using persistent homology, Carlsson, Ishkhanov, de Silva, and Zomorodian qualitatively analyzed approximately 4.5×10^6 high-contrast local patches of natural images obtained by van Hateren and van der Schaaf and previously studied by Lee, Mumford, and Petersen. In their 2008 article, they discovered that as vectors of pixels, the image data were unevenly distributed over a Klein bottle within the 7-dimensional Euclidean sphere.

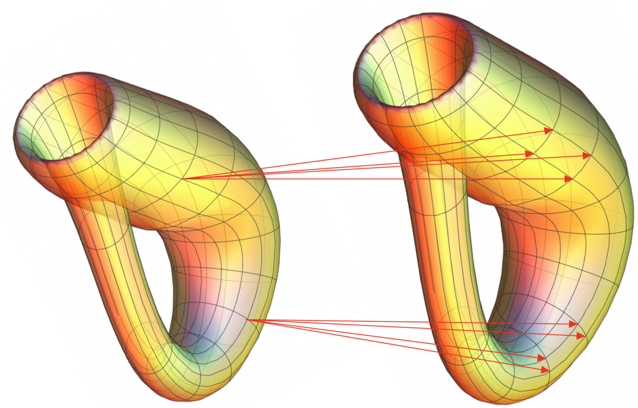
A decade later, Love, Filippenko, Maroulas, and Carlsson have made the Klein bottle as a topological input for designing convolutional layers in neural networks that learn image data. Moreover, they have incorporated the tangent bundle of a Klein bottle into TCNNs for learning video data. Both learnings achieved higher accuracies with smaller training sets.



From topological data analysis to topological deep learning

Using persistent homology, Carlsson, Ishkhanov, de Silva, and Zomorodian qualitatively analyzed approximately 4.5×10^6 high-contrast local patches of natural images obtained by van Hateren and van der Schaaf and previously studied by Lee, Mumford, and Petersen. In their 2008 article, they discovered that as vectors of pixels, the image data were unevenly distributed over a Klein bottle within the 7-dimensional Euclidean sphere.

A decade later, Love, Filippenko, Maroulas, and Carlsson have made the Klein bottle as a topological input for designing convolutional layers in neural networks that learn image data. Moreover, they have incorporated the tangent bundle of a Klein bottle into TCNNs for learning video data. Both learnings achieved higher accuracies with smaller training sets.



*Ephy R. Love et al., Topological convolutional layers for deep learning, **Journal of Machine Learning Research**, 2023.*

*Gunnar Carlsson and Rickard Brüel Gabrielsson, Topological approaches to deep learning, **Topological Data Analysis: The Abel Symposium**, 2018.*

From topological data analysis to topological deep learning

Motivated by the works of Carlsson and his collaborators', we have been investigating analogous questions for speech signals

From topological data analysis to topological deep learning

Motivated by the works of Carlsson and his collaborators', we have been investigating analogous questions for speech signals, with the additional tool of time-delay embedding for turning time series data to point clouds in Euclidean spaces

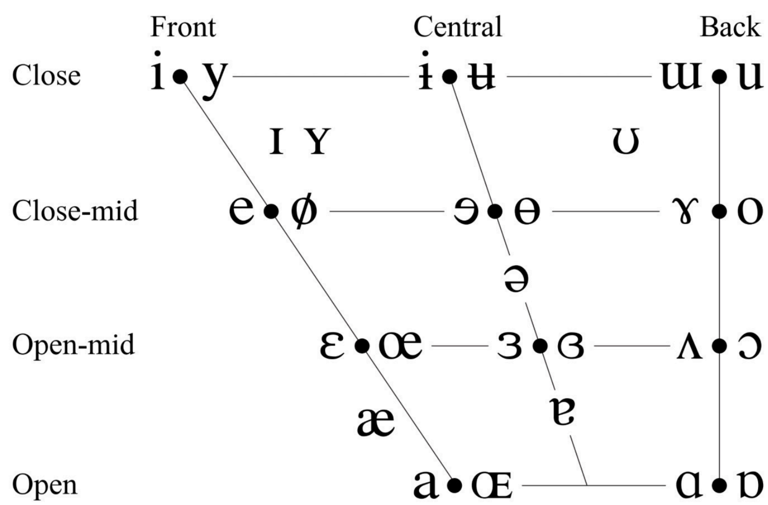
From topological data analysis to topological deep learning

Motivated by the works of Carlsson and his collaborators', we have been investigating analogous questions for speech signals, with the additional tool of time-delay embedding for turning time series data to point clouds in Euclidean spaces, as well as spectrograms as their imagery representations.

From topological data analysis to topological deep learning

Motivated by the works of Carlsson and his collaborators', we have been investigating analogous questions for speech signals, with the additional tool of time-delay embedding for turning time series data to point clouds in Euclidean spaces, as well as spectrograms as their imagery representations.

- For phonetic data, linguists created a charted "distribution space" of vowels:

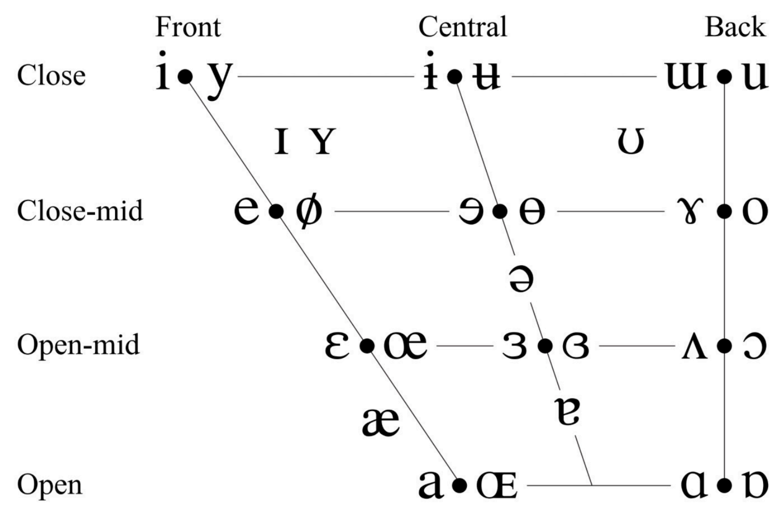


From topological data analysis to topological deep learning

Motivated by the works of Carlsson and his collaborators', we have been investigating analogous questions for speech signals, with the additional tool of time-delay embedding for turning time series data to point clouds in Euclidean spaces, as well as spectrograms as their imagery representations.

- For phonetic data, linguists created a charted "distribution space" of vowels:

The vertical axis of the chart denotes vowel height. Vowels pronounced with the tongue lowered are at the bottom and raised are at the top.

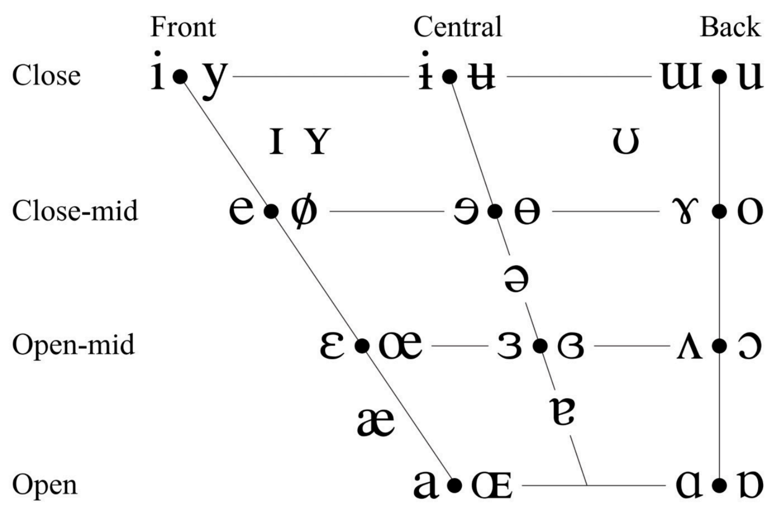


From topological data analysis to topological deep learning

Motivated by the works of Carlsson and his collaborators', we have been investigating analogous questions for speech signals, with the additional tool of time-delay embedding for turning time series data to point clouds in Euclidean spaces, as well as spectrograms as their imagery representations.

- For phonetic data, linguists created a charted "distribution space" of vowels:

The vertical axis of the chart denotes vowel height. Vowels pronounced with the tongue lowered are at the bottom and raised are at the top. The horizontal axis of the chart denotes vowel backness. Vowels with the tongue moved towards the front of the mouth are in the left of the chart, while those with the tongue moved to the back are I placed in right.

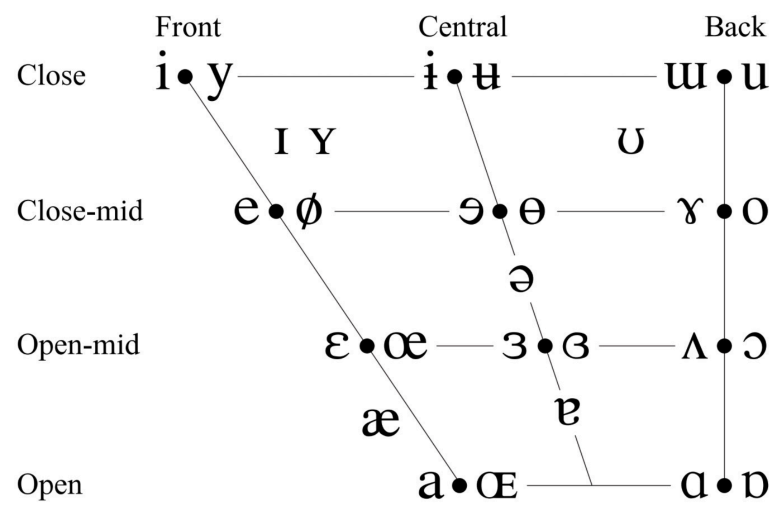


From topological data analysis to topological deep learning

Motivated by the works of Carlsson and his collaborators', we have been investigating analogous questions for speech signals, with the additional tool of time-delay embedding for turning time series data to point clouds in Euclidean spaces, as well as spectrograms as their imagery representations.

- For phonetic data, linguists created a charted "distribution space" of vowels:

The vertical axis of the chart denotes vowel height. Vowels pronounced with the tongue lowered are at the bottom and raised are at the top. The horizontal axis of the chart denotes vowel backness. Vowels with the tongue moved towards the front of the mouth are in the left of the chart, while those with the tongue moved to the back are I placed in right. The last parameter is whether the lips are rounded. At each given spot, vowels on the right and left are rounded and unrounded, respectively.



From topological data analysis to topological deep learning

Motivated by the works of Carlsson and his collaborators', we have been investigating analogous questions for speech signals, with the additional tool of time-delay embedding for turning time series data to point clouds in Euclidean spaces, as well as spectrograms as their imagery representations.

- For phonetic data, linguists created a charted "distribution space" of vowels.
- A main goal remains to use topological methods to reveal a distribution space for speech (and audio) data

From topological data analysis to topological deep learning

Motivated by the works of Carlsson and his collaborators', we have been investigating analogous questions for speech signals, with the additional tool of time-delay embedding for turning time series data to point clouds in Euclidean spaces, as well as spectrograms as their imagery representations.

- For phonetic data, linguists created a charted "distribution space" of vowels.

- A main goal remains to use topological methods to reveal a distribution space for speech (and audio) data, even a digraph on it modeling the complex network of speech-signal sequences

From topological data analysis to topological deep learning

Motivated by the works of Carlsson and his collaborators', we have been investigating analogous questions for speech signals, with the additional tool of time-delay embedding for turning time series data to point clouds in Euclidean spaces, as well as spectrograms as their imagery representations.

- For phonetic data, linguists created a charted "distribution space" of vowels.

- A main goal remains to use topological methods to reveal a distribution space for speech (and audio) data, even a digraph on it modeling the complex network of speech-signal sequences, and apply these topological inputs for smarter learning.

From topological data analysis to topological deep learning

Motivated by the works of Carlsson and his collaborators', we have been investigating analogous questions for speech signals, with the additional tool of time-delay embedding for turning time series data to point clouds in Euclidean spaces, as well as spectrograms as their imagery representations.

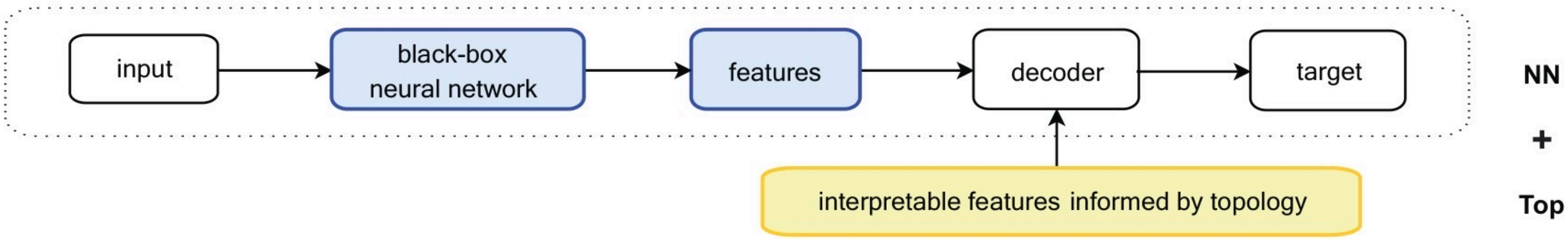
- For phonetic data, linguists created a charted "distribution space" of vowels.
- A main goal remains to use topological methods to reveal a distribution space for speech (and audio) data, even a digraph on it modeling the complex network of speech-signal sequences, and apply these topological inputs for smarter learning.
- In a related direction, based on TopCap, we developed topology-enhanced neural networks.

From topological data analysis to topological deep learning

Motivated by the works of Carlsson and his collaborators', we have been investigating analogous questions for speech signals, with the additional tool of time-delay embedding for turning time series data to point clouds in Euclidean spaces, as well as spectrograms as their imagery representations.

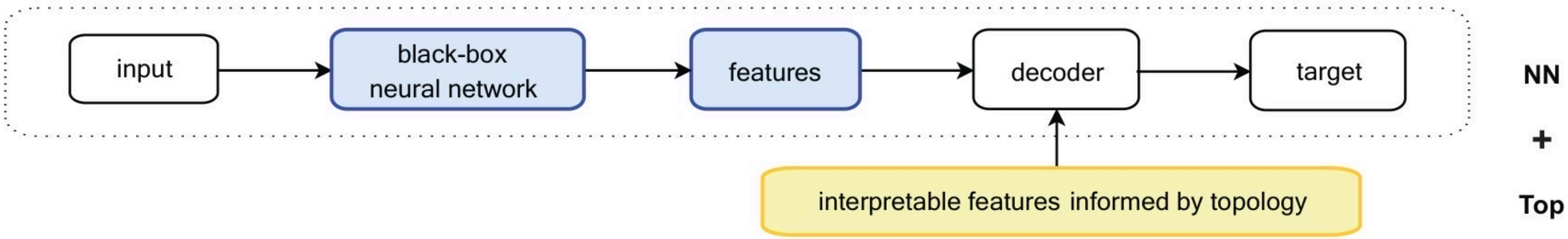
- For phonetic data, linguists created a charted "distribution space" of vowels.
- A main goal remains to use topological methods to reveal a distribution space for speech (and audio) data, even a digraph on it modeling the complex network of speech-signal sequences, and apply these topological inputs for smarter learning.
- In a related direction, based on TopCap, we developed topology-enhanced neural networks.
- Moreover, we exploited the reduced symmetry of spectrograms and designed topological convolutional layers for deep learning speech data.

Topology-enhanced neural networks

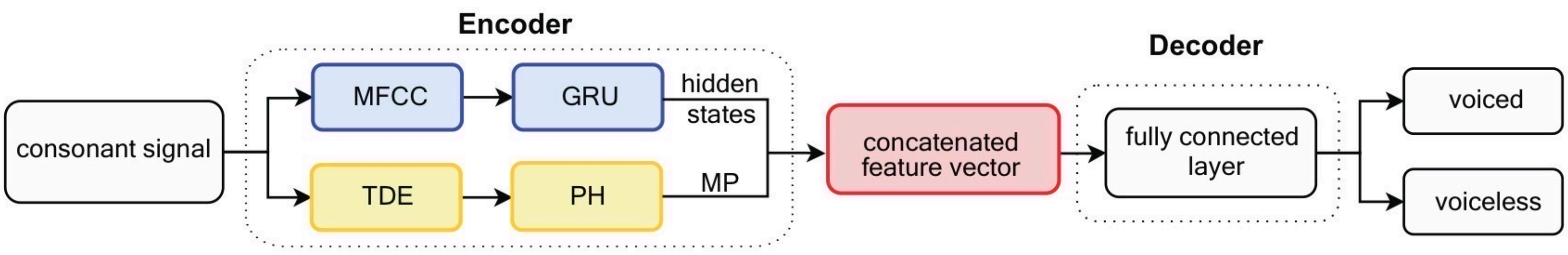


A generic flow chart for enhancing neural networks with topological features

Topology-enhanced neural networks

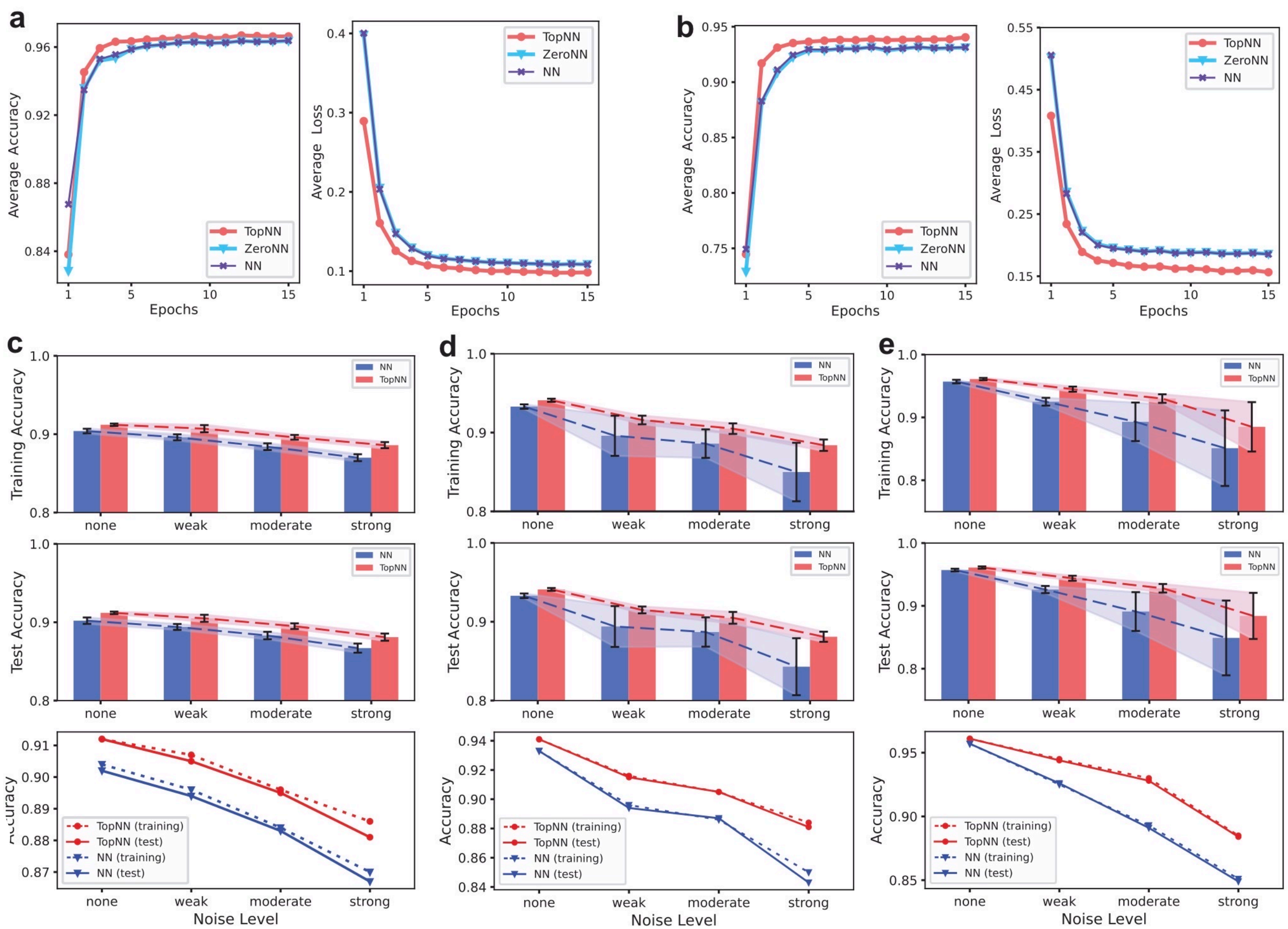


A generic flow chart for enhancing neural networks with topological features



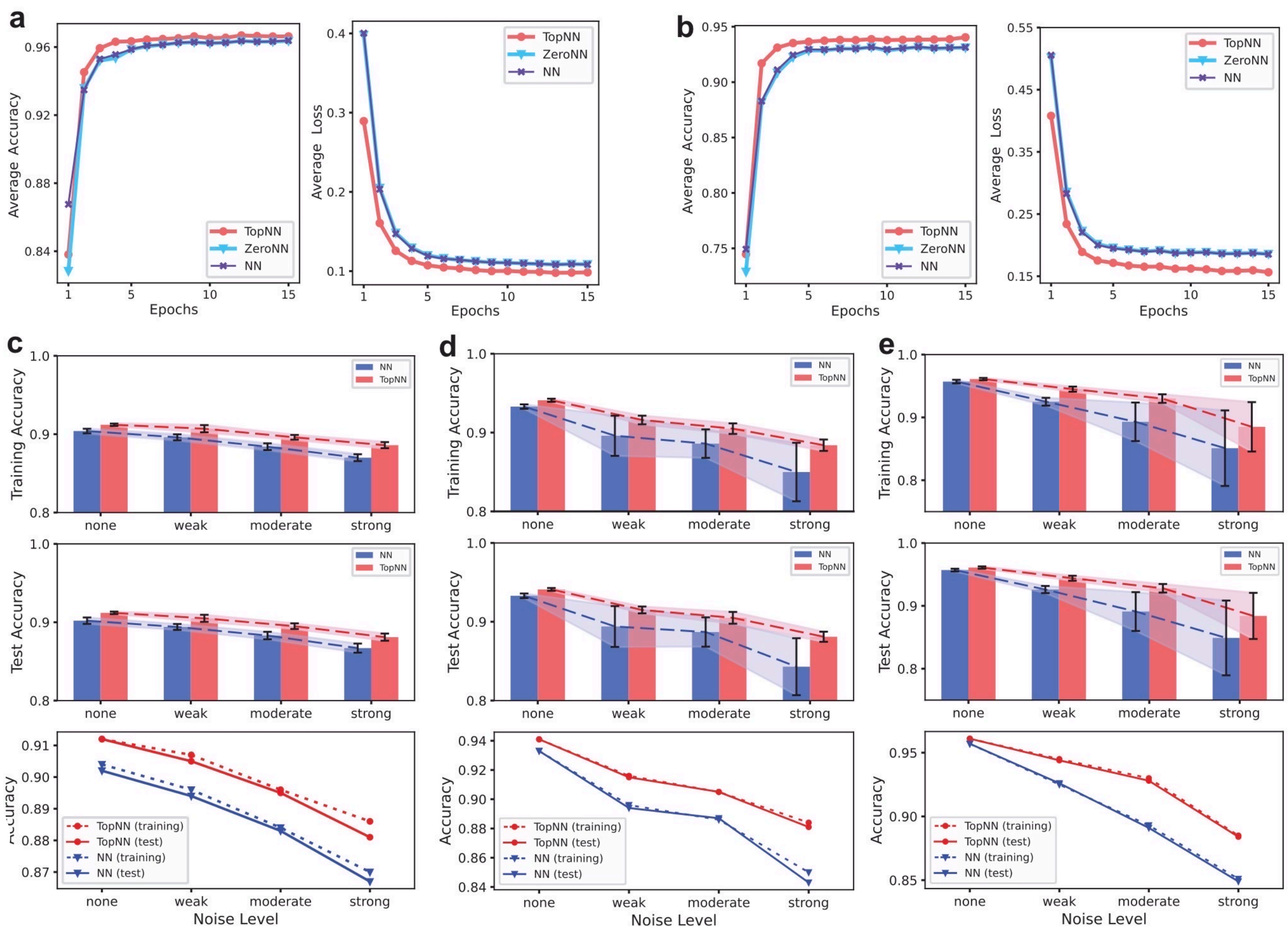
Architecture of a specific TopNN, concatenating GRU and TopCap features

Topology-enhanced neural networks



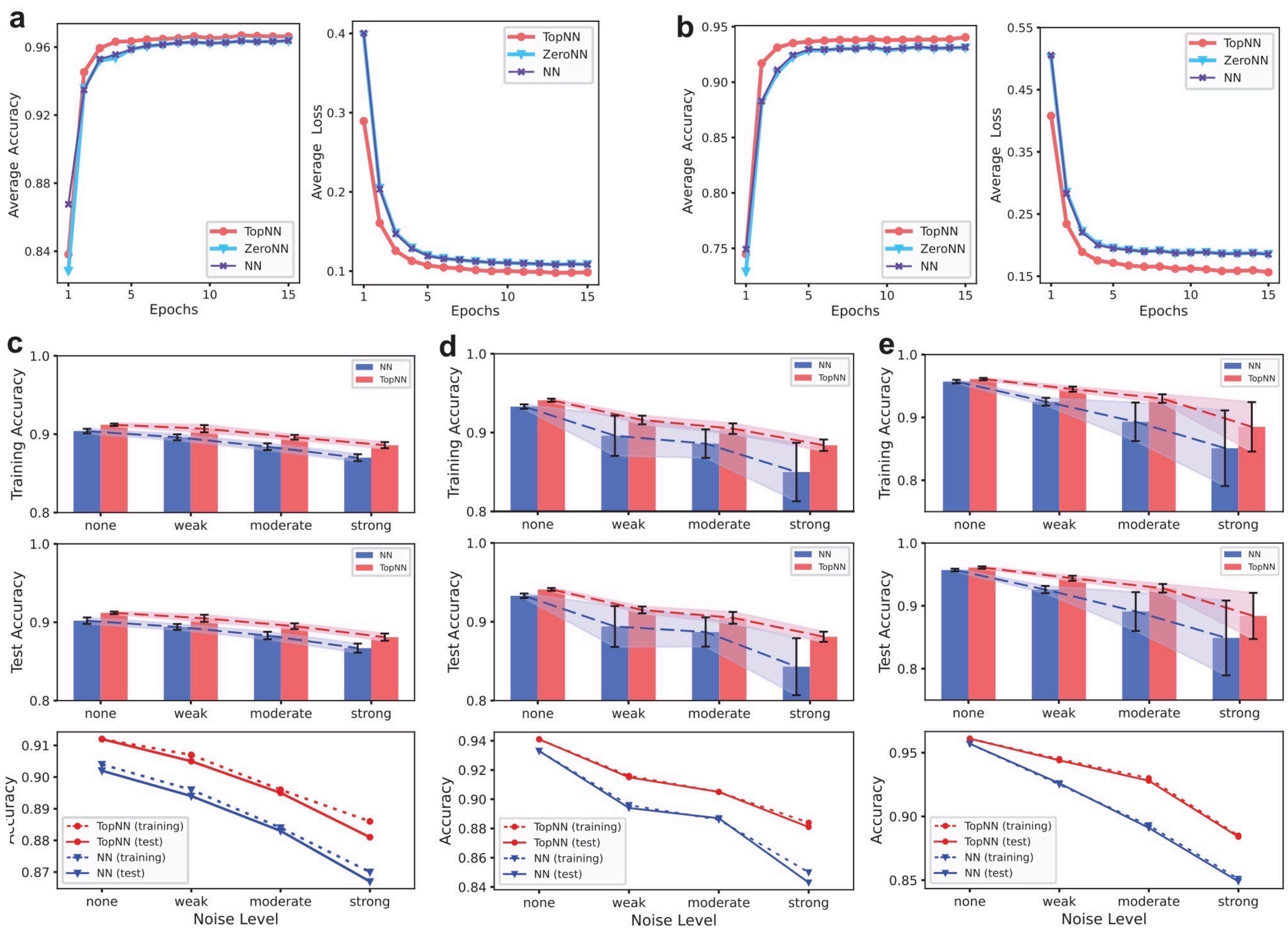
Visual analytics of experiments with TopNN

Topology-enhanced neural networks



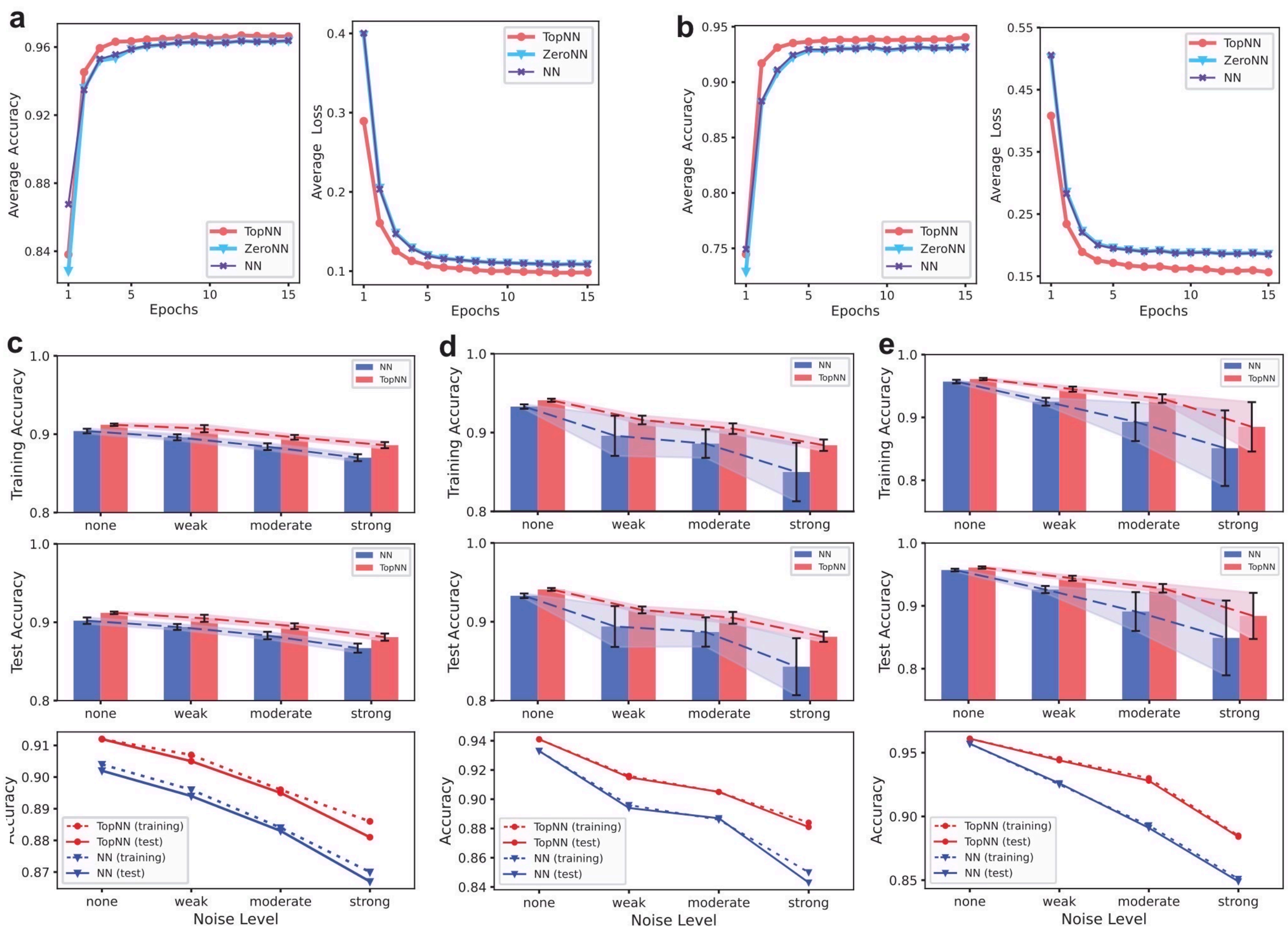
Visual analytics of experiments with TopNN
a, *Training curves of TopNN, ZeroNN (NN features concatenated with null topological feature, as a sanity check), and NN on 36000 speech data from the TIMIT dataset.*

Topology-enhanced neural networks



Visual analytics of experiments with TopNN
a, *Training curves of TopNN, ZeroNN (NN features concatenated with null topological feature, as a sanity check), and NN on 36000 speech data from the TIMIT dataset. They demonstrate that TopNN has higher accuracy and faster convergence in loss function than ZeroNN and NN.*

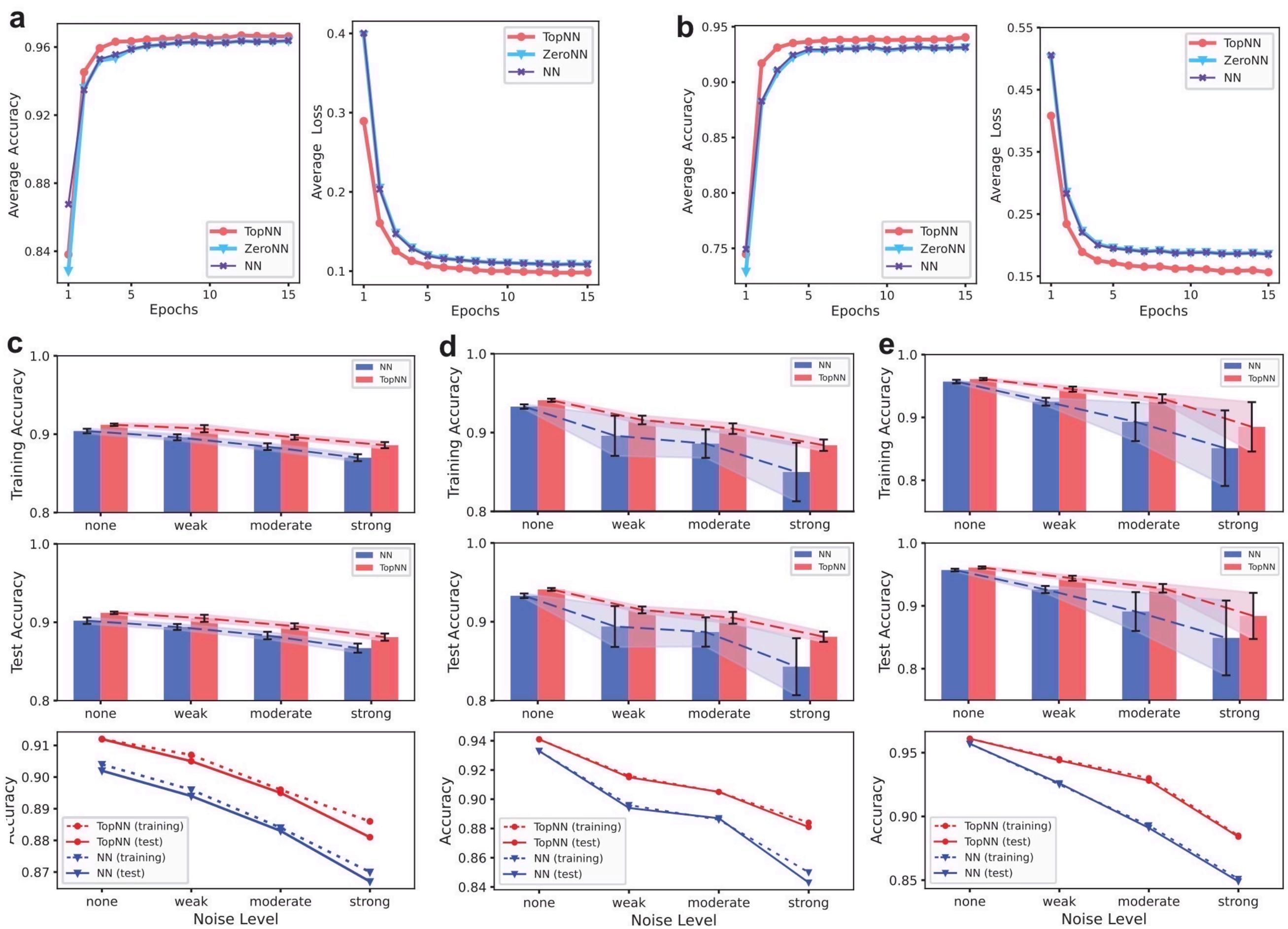
Topology-enhanced neural networks



Visual analytics of experiments with TopNN

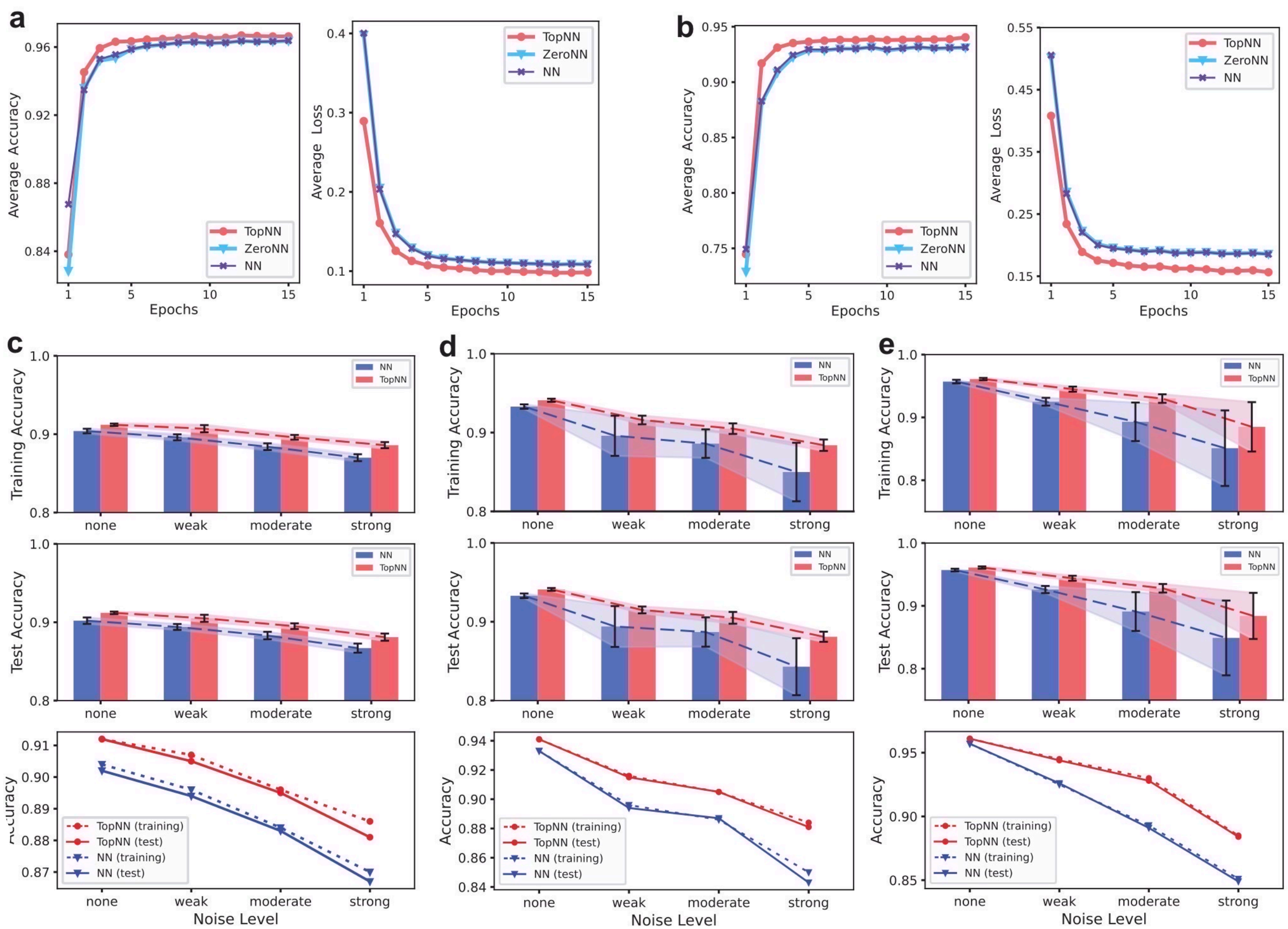
***b**, Training curves of TopNN, ZeroNN, and NN with the same set up as in **a** and including noise (signal-to-noise ratio = 5dB).*

Topology-enhanced neural networks



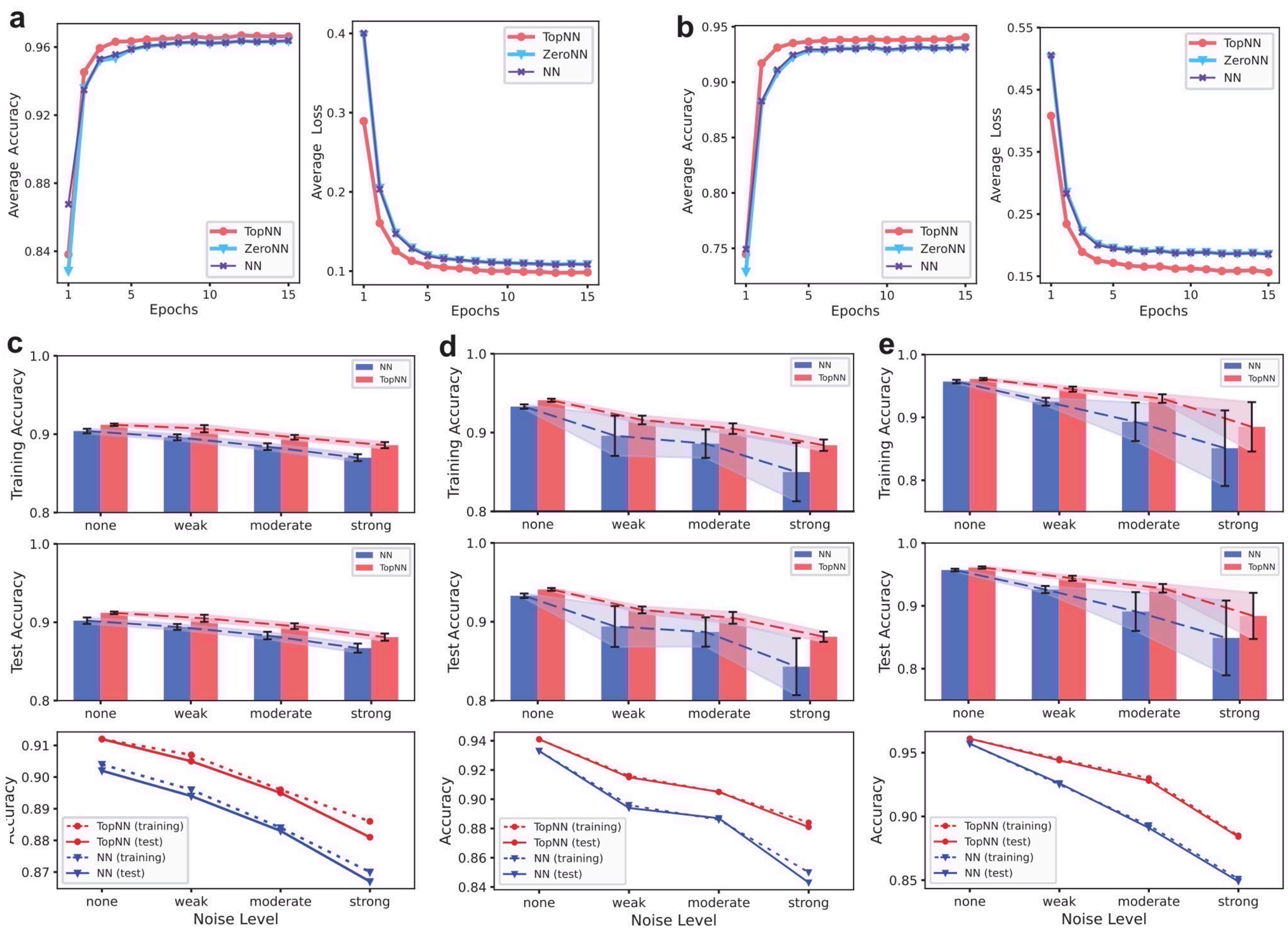
Visual analytics of experiments with TopNN
b, *Training curves of TopNN, ZeroNN, and NN with the same set up as in **a** and including noise (signal-to-noise ratio = 5dB). With noise added, TopNN's improvement in accuracy and loss decrease are more prominent compared with the results in **a**.*

Topology-enhanced neural networks



Visual analytics of experiments with TopNN
***c**, **d**, and **e**, Comprehensive performance comparison and noise robustness analysis of TopNN and NN based on training and test accuracy rates with the large datasets ALLSSTAR, LJSpeech, and TIMIT, respectively.*

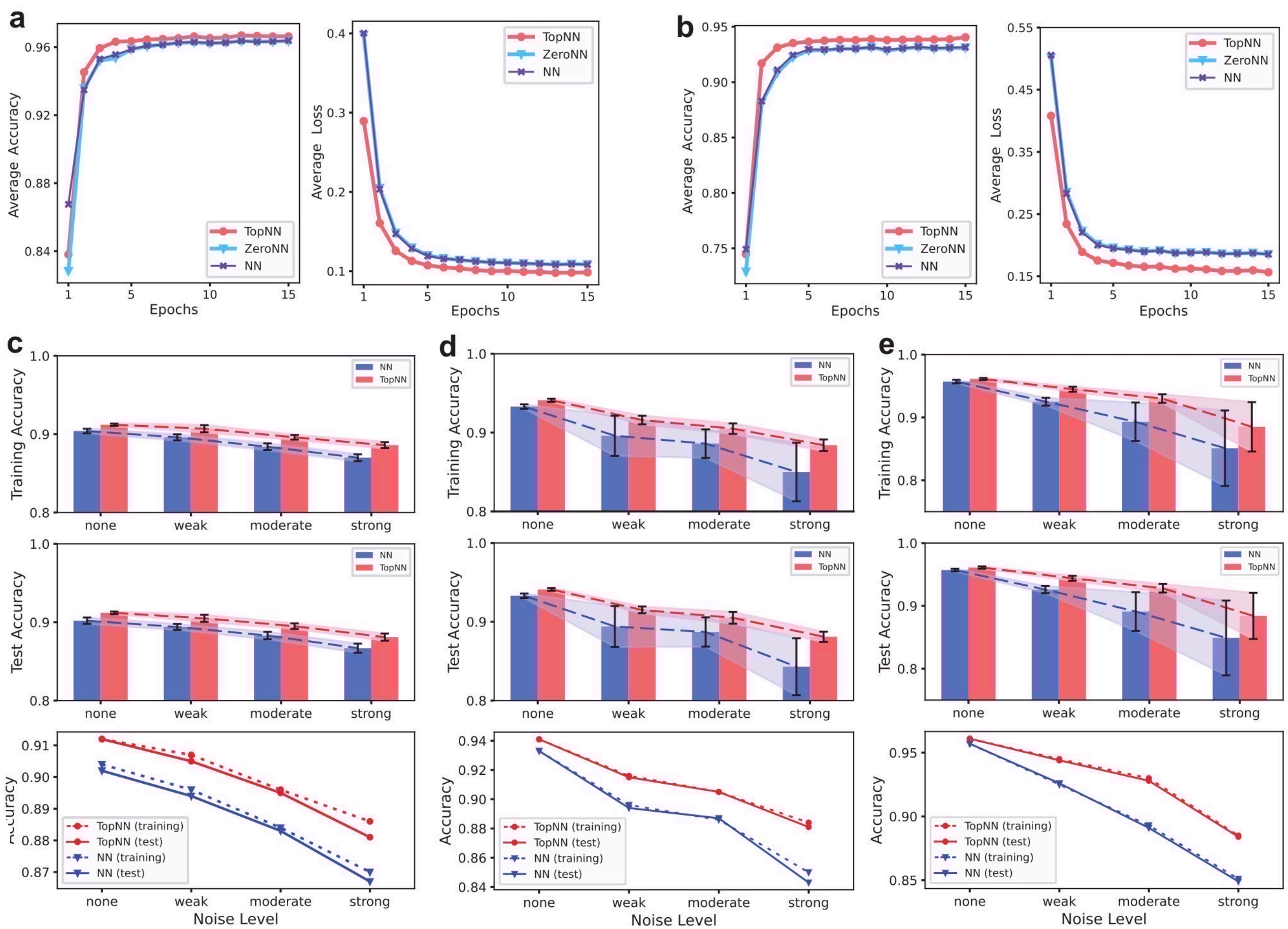
Topology-enhanced neural networks



Visual analytics of experiments with TopNN

c, ***d***, *and* ***e***, *Comprehensive performance comparison and noise robustness analysis of TopNN and NN based on training and test accuracy rates with the large datasets ALLSSTAR, LJSpeech, and TIMIT, respectively. Noise levels include none, weak (SNR = 10dB), moderate (SNR = 5dB), and strong (SNR = 0dB).*

Topology-enhanced neural networks

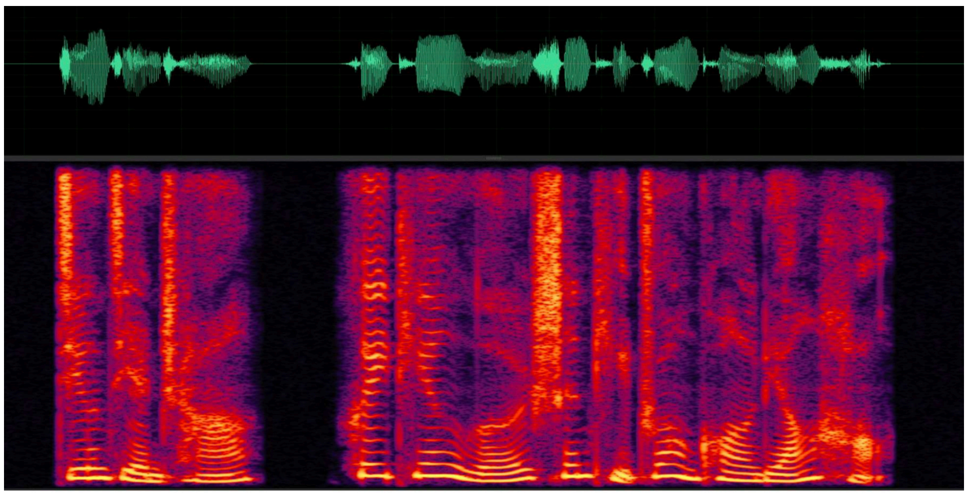


Visual analytics of experiments with TopNN

***c**, **d**, and **e**, Comprehensive performance comparison and noise robustness analysis of TopNN and NN based on training and test accuracy rates with the large datasets ALLSSTAR, LJSpeech, and TIMIT, respectively. Noise levels include none, weak (SNR = 10dB), moderate (SNR = 5dB), and strong (SNR = 0dB). TopNN consistently achieves higher accuracy, steadier performance, and more robustness against noise.*

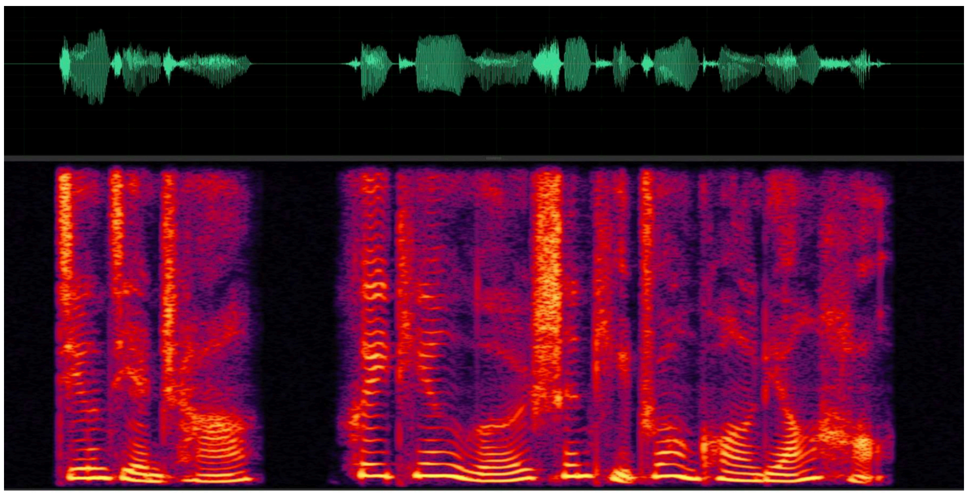
Topology-informed convolution kernels for speech recognition

We defined a notion of *contrast* for 3 x 3 convolution kernels that process spectrograms, and introduced rigid constraints (unit norm and zero-sum of column vectors) to define a space *V* of kernels.



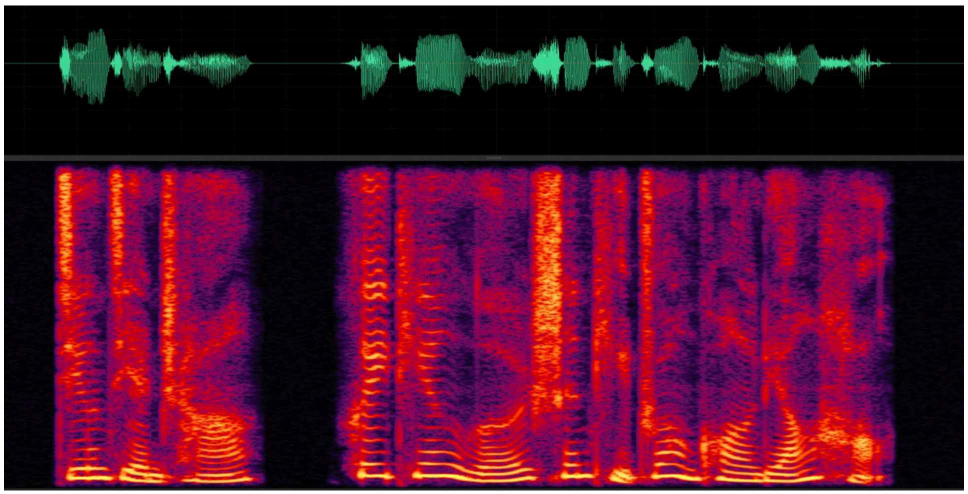
Topology-informed convolution kernels for speech recognition

We defined a notion of *contrast* for 3 x 3 convolution kernels that process spectrograms, and introduced rigid constraints (unit norm and zero-sum of column vectors) to define a space V of kernels. We showed that V is homeomorphic to S^5 and that the natural SO(3)-action on V induces a quotient space B that is homeomorphic to a disk D^2.

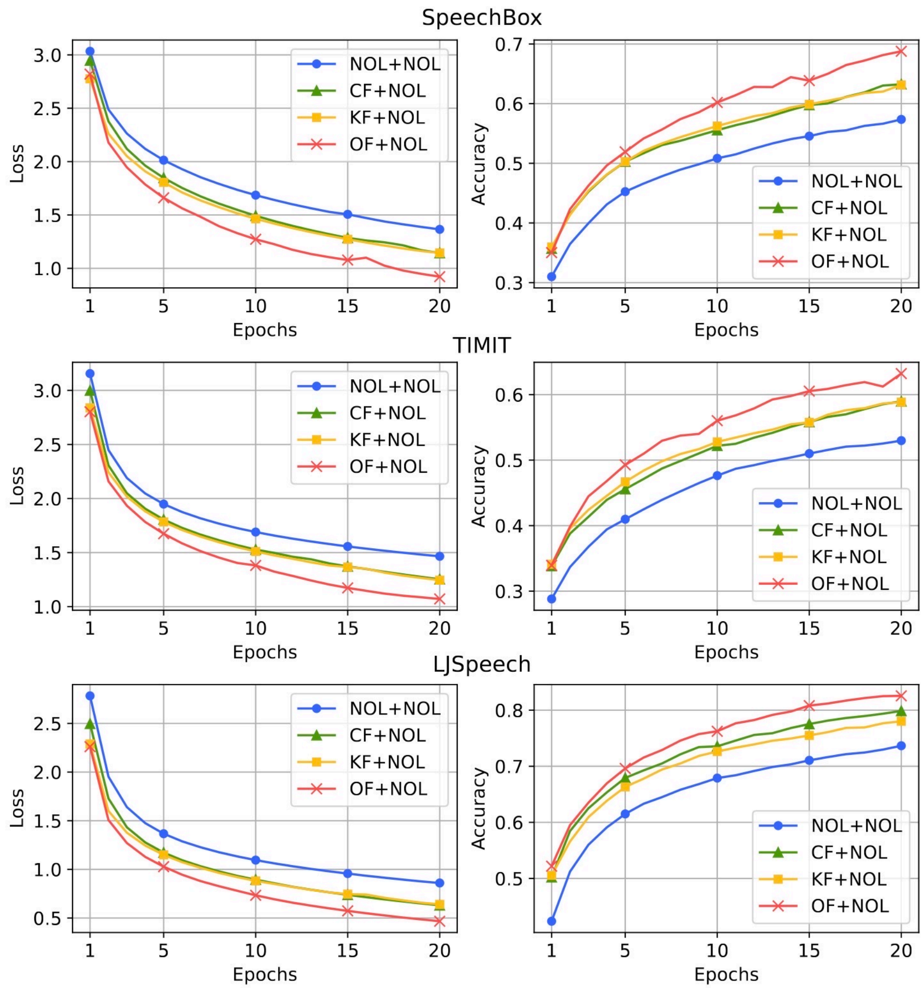


Topology-informed convolution kernels for speech recognition

We defined a notion of *contrast* for 3 x 3 convolution kernels that process spectrograms, and introduced rigid constraints (unit norm and zero-sum of column vectors) to define a space V of kernels. We showed that V is homeomorphic to S^5 and that the natural SO(3)-action on V induces a quotient space B that is homeomorphic to a disk D^2. We then defined untrained Orthogonal Filter (OF) layer with convolution kernels informed by this topology.

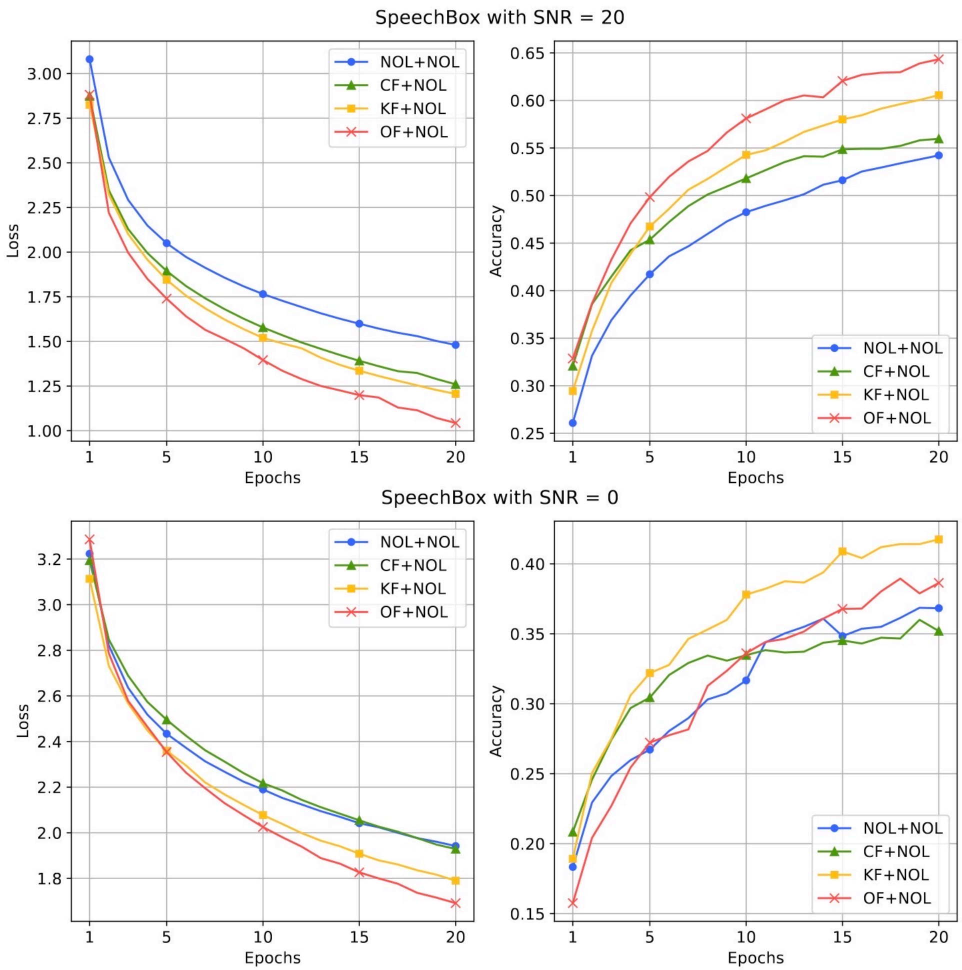


Topology-informed convolution kernels for speech recognition



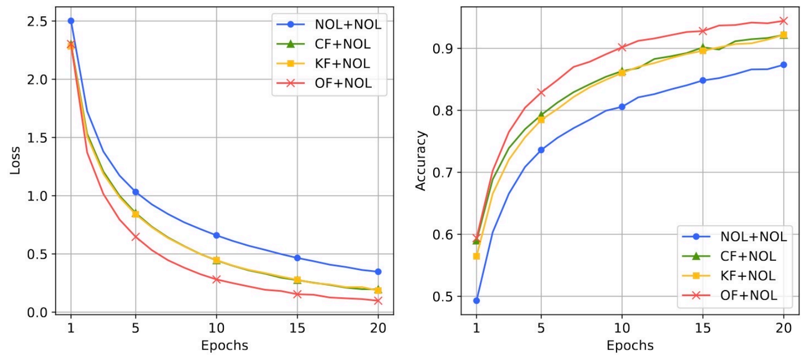
Comparisons of normal (NOL), Love et al.'s circle filter (CF) and Klein-bottle filter (KF), and our orthogonal filter (OF) convolutional layers for phoneme classification tasks via loss and accuracy on datasets SpeechBox, TIMIT, and LJSpeech

Topology-informed convolution kernels for speech recognition

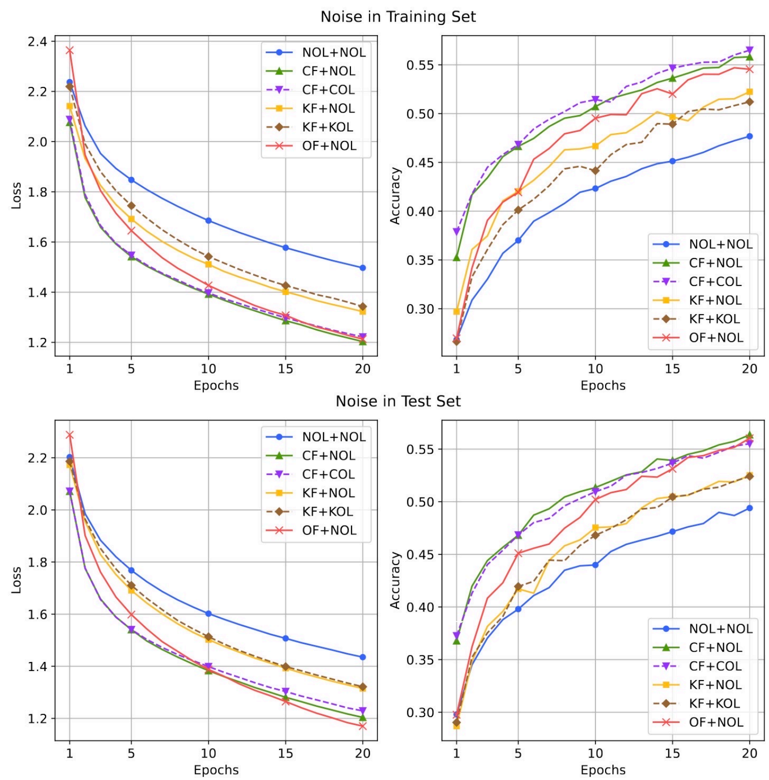


Comparisons with noise added
Our proposed OF layer enables superior performance in phoneme recognition, particularly in low-noise scenarios.

Topology-informed convolution kernels for speech recognition



Comparison for word classification on SpeechCommands



Comparisons for image classification on CIFAR10

Thank you.