Topology-enhanced machine learning for consonant recognition

Pingyao Feng, Qingrui Qu, Siheng Yi, Zhiwang Yu, Haiyu Zhang, Yifei Zhu

Abstract—In artificial-intelligence-aided signal processing, existing deep learning models often exhibit a black-box structure. The integration of topological methods serves a dual purpose of making models more interpretable as well as extracting structural information from time-dependent data for smarter learning. Here, we provide a transparent and broadly applicable methodology, TopCap, to capture topological features inherent in time series for machine learning. Rooted in high-dimensional ambient spaces, TopCap is capable of capturing features rarely detected in datasets with low intrinsic dimensionality. Compared to prior approaches, we obtain descriptors which probe finer information such as the vibration of a time series. This information is then vectorised and fed to multiple machine learning algorithms. Notably, in classifying voiced and voiceless consonants, TopCap achieves an accuracy exceeding 96%, consistently standing comparison with and sometimes significantly outperforming state-of-the-art deep learning neural networks in both accuracy and efficiency.

1 Introduction

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 infer information about the shape of the drumhead from the sound it makes, using mathematical theory. In this article, we venture to flip and mirror the question across senses and address instead: “Can we see the sound of a human speech?”

As a major task of natural language processing (NLP), speech recognition is one of the essential components of artificial intelligence (AI). In turn, AI advancements have led to a widespread adoption of voice recognition technologies, encompassing applications such as speech-to-text conversion and music generation. The rise of topological data analysis (TDA) [1] has integrated topological methods into many areas including AI [2, 3], which makes neural networks more interpretable and efficient, with a focus on structural information. In the field of voice recognition [4, 5], more specifically consonant recognition [6, 7, 8, 9, 10], prevalent methodologies frequently revolve around the analysis of energy and spectral information. While topological approaches are still rare in this area, we combine TDA and machine learning to obtain a classification for speech data, based on geometric patterns hidden within phonetic segments. The method we propose, TopCap (referring to the capability of capturing topological structures of data), is not only applicable to audio data but also to general-purpose time series data that require extraction of structural information for machine learning algorithms. Initially, we endow phonetic time series with point-cloud structure in a high-dimensional Euclidean space via time-delay embedding (TDE, see Fig. 1a) with appropriate choices of parameters. Subsequently, 1-dimensional persistence diagrams are computed using persistent homology (see Sec. S.2.2 for an explanation of the terminologies). We then conduct evaluations with nine machine learning algorithms, in comparison with multiple deep learning models without topological inputs, to demonstrate the significant capabilities of TopCap in the desired classification.

Conceptually, TDA is an approach that examines data structure through the lens of topology. This discipline was originally formulated to investigate the *shape* of data, particularly point-cloud data in high-dimensional spaces [11]. Characterised by a unique insensitivity to metrics, robustness against noise, invariance under continuous deformation, and coordinate-free computation [1], TDA has been combined with machine learning algorithms to uncover intricate and concealed information within datasets [12, 3, 13, 14, 15, 16]. In these contexts, topological methods have been employed to extract structural information from the dataset, thereby enhancing the efficiency of the original algorithms. Notably, TDA excels in identifying patterns such as clusters, loops, and voids in data, establishing it as a burgeoning tool in the realm of data analysis [17]. Despite being a nascent field of study, with its distinctive emphasis on the shape of data, TDA has led to novel applications in various far-reaching fields, as evidenced in the literature. These include image recognition [18, 19, 20], time series forecasting [21] and classification [22], brain activity monitoring [23, 24], protein structural analysis [25, 26], speech recognition [27], signal processing [28, 29], neural networks [30, 31, 32, 2], among others. It is anticipated that further development of TDA will pave a new direction to enhance numerous aspects of daily life.

The task of extracting features that pertain to structural information is both intriguing and formidable. This process is integral to a multitude of practical applications [33, 34, 35, 36], as scholars strive to identify the most effective representatives and descriptors of shape within a given dataset. Despite the fact that TDA is specifically designed for shape capture, there are several hurdles that persist in this newly developed field of study. These include (1) the nature and sensitivity of descriptors obtained by methods in TDA, (2) the dimensionality of the data and other parameter

10/10/2024

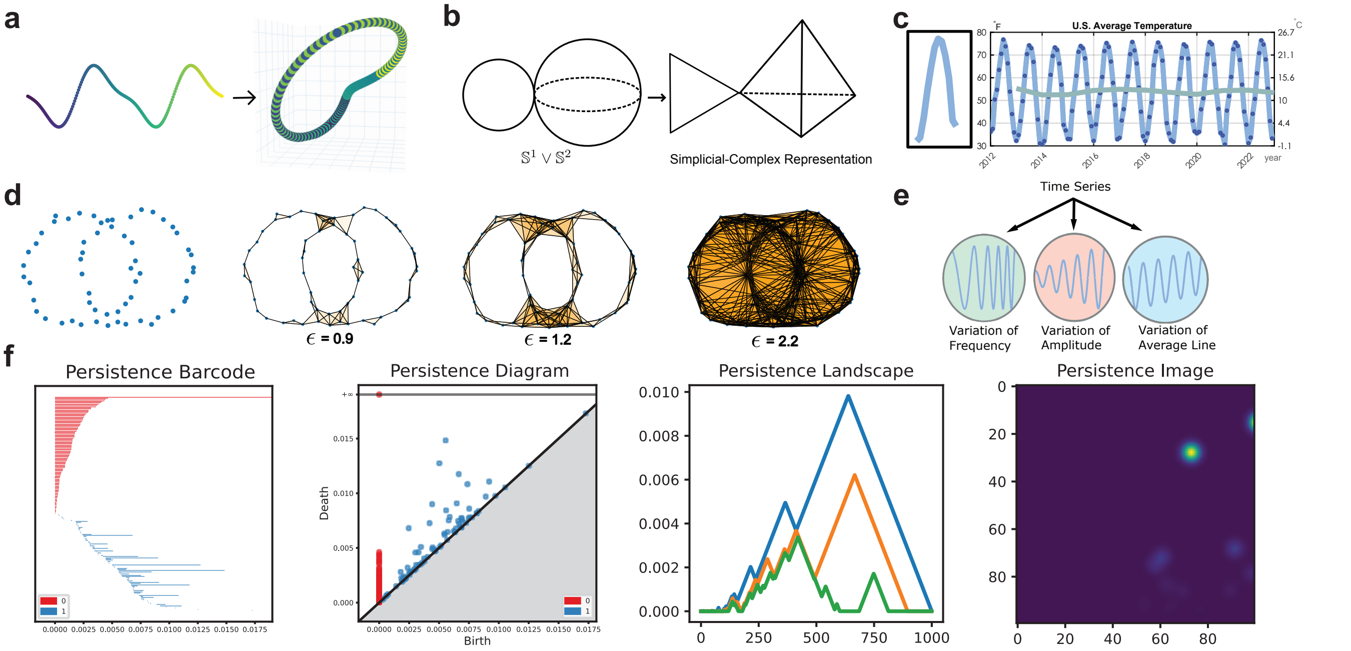

Fig. 1: Illustrations of methodology. **a**, Time-delay embedding (dimension=3, delay=10, skip=1) of $f(t_n) = \sin(2t_n) - 3\sin(t_n)$, with $t_n = \frac{\pi}{50}n$ $(0 \leqslant n \leqslant 200)$. Resulting point clouds lay on a closed curve in 3-dimensional Euclidean space. The colour indicates their original locations in the time series. **b**, A topological space and its triangulation. On the left is a topological space consisting of a 1-dimensional sphere (i.e., a circle) and a 2-dimensional sphere with a single point of contact, denoted as $\mathbb{S}^1 \vee \mathbb{S}^2$. The right depicts a triangulation of this topological space. **c**, Average temperature in the U.S. with monthly values (dark blue dots) and yearly values (green curve). The left panel shows a single-year section of average temperature. **d**, Computing PH. The four plots consecutively show how a diagram or a barcode is computed: Connect each pair of points with a distance less than ϵ by a line segment, fill in each triple of points with mutual distances less than ϵ with a triangular region, etc., and compute the corresponding homology groups. In this way, as "time" ϵ increases, points in the diagram or intervals in the barcode record the "birth" and "death" of each generator of a homology group, i.e., the occurrence and disappearance of a loop (or a higher-dimensional hole), thereby revealing the essential topological features of the point cloud that persist. **e**, Characterising the vibration of a time series in terms of its variability of frequency, amplitude, and average line. **f**, Commonly used representations for PH, with an example of 100 points uniformly distributed over a bounded region in 2D Euclidean space.

choices, (3) the vectorisation of topological features, and (4) computational cost. These challenges will be elaborated in the following paragraphs within this section. Subsequently, we will demonstrate how our proposed methodology, Top-Cap, addresses these challenges through an application to consonant classification.

When applying TDA, the most imminent question is to comprehend the characteristics and nature of descriptors extracted via topological methods. TDA is grounded in the pure-mathematical field of algebraic topology (AT) [37, 38], with persistent homology (PH) being its primary tool [39, 40]. While AT can quantify topological information to a certain extent [38, 1, 17], it is vitally important to understand both the capabilities and limitations of TDA. Generally speaking, TDA methods distinguish objects based on continuous deformation. For example, PH cannot differentiate a disk from a filled rectangle, given that one can continuously deform the rectangle into a disk by pulling out its four edges. In contrast, PH can distinguish between a filled rectangle and an unfilled one due to the presence of a "hole" in the latter, preventing a continuous deformation between the two. In certain circumstances, these methods are considered excessively ambiguous to capture the structural information in data, thereby necessitating a more precise descriptor of shapes. To draw an analogy, TDA can be conceptualised as a scanner with diverse inputs encompassing time series, graphs, pictures, videos, etc. The output of this scanner is a multiset of intervals in the extended real line, referred to as a persistence diagram (PD)[1] or a persistence barcode (PB) [11, 41, 42] (cf. Fig. 1f). In particular, by *maximal persistence* (MP) we mean the maximal length of the intervals. The precision of the topological descriptor depends on two factors: (1) the association of a topological space, i.e., the process of transforming the input data into a topological space (see Fig. 1b for a simplicial-complex representation of spaces; typically, the original datasets are less structured, and one should find a suitable representation of the data), and (2) the vectorisation of PD or PB, i.e., how to perform statistical inference with PD/PB. Despite there are many theoretical results which provide a solid foundation for TDA, few can elucidate the practical implications of PD

[1] In this article, we shall freely use the usual birth-by-death PDs and their birth-by-lifetime variants, whichever better serve our purposes. See Sec. S.2.2 for details.

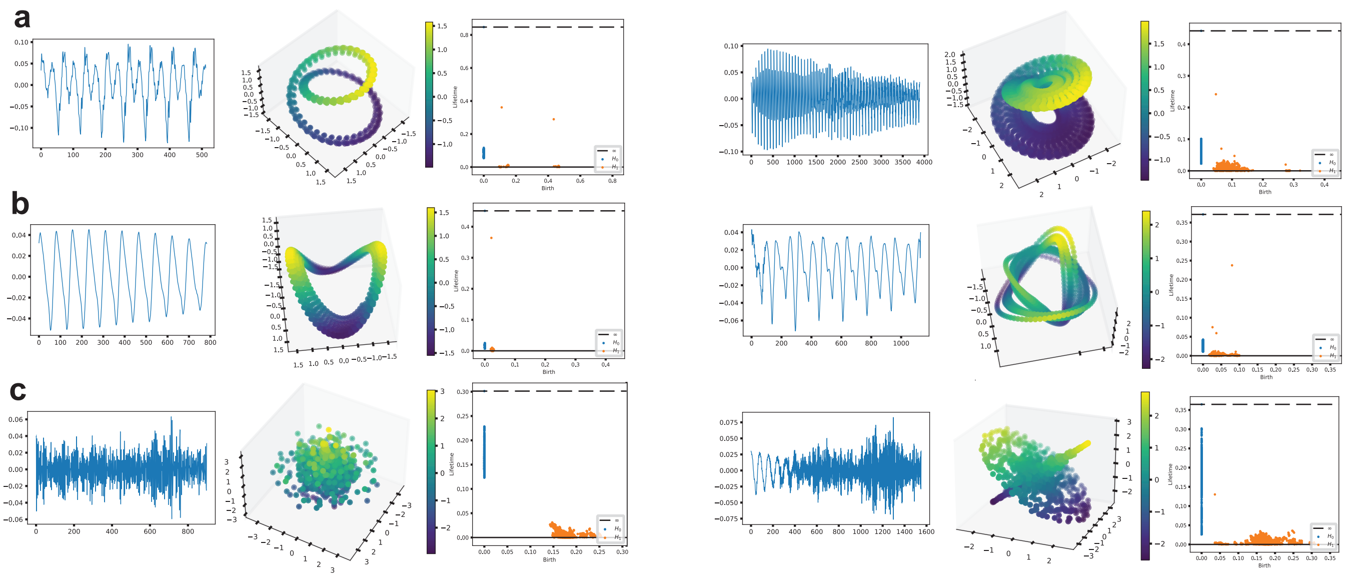

Fig. 2: The varied shapes of vowels, voiced consonants, and voiceless consonants. **a**, the left 3 panels and the right 3 panels depict 2 vowels, respectively. For each, the first picture is the time series of the vowel, the second picture corresponds to the 3-dimensional principal component analysis of the point cloud resulting from performing TDE (dimension=100, delay=1, skip=1) on this time series, and the third picture is the PD of this point cloud. **b**, The analogous features for 2 voiced consonants. **c**, Those for 2 voiceless consonants.

and PB. For example, what does it mean if many points are distributed near the birth–death diagonal line in a PD? Extensive studies have been conducted on short-lived bars in PH, including those related to molecular data [43, 44], hierarchical structures [45], and protein structures [46, 43, 47], among others. The significance of points distributed near the birth–death diagonal line is particularly relevant in real-world applications. In this article, we present a new way to examine those data, with an example of simulated time series to demonstrate that points distributed in such regions indeed encode important information, such as vibration patterns of the time series, and a different distribution in these regions leads to a different pattern of vibration. This serves as a motivation for proposing TopCap and is further discussed in Sec. 2.1. It turns out that topological descriptors can be sharpened by noting patterns in these regions.

In view of the capability of topological methods to discern vibration patterns in time series, we apply them to classify consonant signals into voiced and voiceless categories. As a first demonstration of our findings, to *visualise* vowels, voiced consonants, and voiceless consonants in TDE and PD, see Fig. 2 (cf. Sec. S.1 for details of phonetic categories).

The first challenge, as many researchers may encounter when applying topological methods, is to determine the dimension of point clouds derived from input data [48, 49, 50]. This essentially involves transforming the input into a topological space. In situations where the dimensionality of the data is large, researchers often project the data into a lower-dimensional topological space to facilitate visualisation and reduce computational cost [23, 24, 51]. On the other hand, as in this study and other applications with time series analysis [52, 53, 54, 55, 22, 56, 27], low-dimensional data are embedded into a higher-dimensional space. In both scenarios, deciding on the data dimensionality is both critical and challenging. Often, tuning the dimension is a tremendous task. In Sec. 4 of Discussion below, we delve into the issue of data dimensionality. In our case, as it might seem counterintuitive compared to most algorithms, when the data are embedded into a higher-dimensional space, the computation will be a little faster, the point cloud appears smoother and more regular, and most importantly, more salient topological features can be spotted, which seldom happen in lower-dimensional spaces. When encountering the dimensionality of data, researchers would think of the well-known curse of dimensionality [57]: As a typical algorithm grapple, with the increase of dimension, more data are needed to be involved, often growing exponentially and thereby escalating computational cost. Even worse, the computational cost of the algorithm itself normally rises as the dimension goes higher. However, topological methods do not necessarily prefer data of lower dimension. For computing PH (see Fig. 1d for the process of computing PD/PB from point clouds), a commonly used algorithm [58, 59] sees complexity grow with an increase in the number n of simplices during the process, with a worst-case polynomial time-complexity of $O(n^3)$. As such, the computational cost is directly related to the number of simplices formed during filtration. Our observation shows that computation time may not increase much given an increase of dimension of data, because the latter may have little effect on the size (i.e., number of points) of the point cloud and thus neither on the number of simplices formed during filtration.

Having obtained a suitable topological space from input data, one can derive a PD/PB from the topological space, which constitutes a multiset of intervals. The subsequent challenge lies in the vectorisation of the PD/PB for its integration into a machine-learning algorithm. The vectorisation process is essentially linked to the construction of the topological space, as the combination of different methods for constructing the topological space and vectorisation together determine the descriptor utilised in machine learning. A plethora of vectorisation methods exist, such as persistent entropy [60], persistence curve [61], persistence landscape [62], and persistence image (PI) [63], among others, as documented in various studies [40, 64] (cf. Fig. 1f). The selection of these methods requires careful consideration. Additionally, one can design more customised quantification techniques tailored to specific experimental conditions and physical properties to meet specific requirements [65, 66, 67]. In Sec. 3 of Methods, we employ MP and its corresponding birth time as two features. These have been integrated into nine traditional machine learning algorithms to classify voiced and voiceless consonants, yielding an accuracy that exceeds 96% with each algorithm. This vectorisation method is quite simple, primarily due to our construction of topological spaces from phonetic time series, as detailed in the Method section. This construction enables PH to capture significant topological features within the time series. In Sec. 2.1, we also observe a pattern of vibration which could potentially be vectorised by PI into a matrix. As one of its strengths, PI emphasises regions where the weighting function scores are high, which makes it a computationally flexible method. Future work may involve a more precise recognition of such patterns using PI.

To place our results in a more specific context as well as to acknowledge earlier efforts made by other researchers to which we are indebted, let us now give an overview of closely related work in the field.

Time series analysis [68] is a prevalent tool for various applied sciences. The recent surge in TDA has opened new avenues for the integration of topological methods into time series analysis [21, 69, 70]. Much literature has contributed to the theoretical foundation in this area. For example, theoretical frameworks for processing periodic time series have been proposed by Perea and Harer [71], followed by their and their collaborators' implementation in discovering periodicity in gene expressions [72]. Their article [71] studied the geometric structure of truncated Fourier series of a periodic function and its dependence on parameters in time-delay embedding (TDE), providing a solid background for TopCap. In addition to periodic time series, towards more general and complex scenarios, quasi-periodic time series have also been the subject of scholarly attention. Research in this direction has primarily concentrated on the selection of parameters for geometric space reconstruction [73] and extended to vector-valued time series [74].

In this article, a topological space is constructed from data using TDE, a technique that has been widely employed in the reconstruction of time series (see Fig. 1a and cf. Sec. S.2.1 for more background). Thanks to the topological invariance of TDE, the general construction of simplicial-complex representation (see Fig. 1b) and computation of PH from point clouds (see Fig. 1d) apply to time series data, although this transformation involves subtle technical issues in practice. For instance, Emrani et al. utilised TDE and PH to identify the periodic structure of dynamical systems, with applications to wheeze detection in pulmonology [52]. They selected the embedded dimension d as 2, and their delay parameter τ was determined by an autocorrelation-like (ACL) function, which provided a range for the delay between the first and second critical points of the ACL function. Pereira and de Mello proposed a data clustering approach based on PD [53]. The data were initially reconstructed by TDE, with $d = 2$ and $\tau = 3$, so as to obtain the corresponding PD, which was then subjected to k-means clustering. The delay τ was determined using the first minimum of an auto mutual information, and the embedded dimension d was set to be 2 as using 3 dimensions did not significantly improve the results. Khasawneh and Munch introduced a topological approach for examining the stability of a class of nonlinear stochastic delay equations [54]. They used false nearest neighbours to determine the embedded dimension $d = 3$ and chose the delay to equal the first zeros of the ACL function. Subsequently, the longest persistence lifetime in PD was used as a vectorisation to quantify periodicity. Umeda focused on a classification problem for volatile time series by extracting the structure of attractors, using TDA to represent transition rules of the time series [22]. He assigned $d = 3$, $\tau = 1$ in his study and introduced a novel vectorisation method, which was then applied to a convolutional neural network (CNN) to achieve classification. Gidea and Katz employed TDA to detect early signs prior to financial crashes [56]. They studied multi-dimensional time series with $\tau = 1$ and used persistence landscape as a vectorisation method. For speech recognition, Brown and Knudson examined the structure of point clouds obtained via TDE of human speech signals [27]. The TDE parameters were set as $d = 3$, $\tau = 20$, after which they examined the structure of point clouds and their corresponding PB.

Upon reviewing the relevant literature, we see that currently there lacks a general framework for systematically choosing d and τ, and researchers often have to make choices in an ad hoc fashion for practical needs. While the TDE–PH topological approach to handling time series data is not new, TopCap extracts features from high-dimensional spaces. For example, in our experiments $d = 100$ based on several considerations (see Sec. 2.3.1). It happens in some cases that in a low-dimensional space, regardless of how optimal the choice of τ is, the structure of the time series cannot be adequately captured. In contrast, given a high-dimensional space, feature extraction from data becomes simpler. Of course, operating in a high-dimensional space comes with its own cost. For example, the adjustment of τ then requires careful consideration. Nonetheless, it also offers advantages, which we will elucidate step by step in the subsequent sections.

2 Results

This research drew inspiration from Carlsson and his collaborators' discovery of the Klein-bottle distribution of high-contrast, local patches of natural images [20], as well as their subsequent recent work on topological CNNs for learning image and even video data [2]. By analogy, we aim to understand a distribution space for speech data, even a directed graph structure on it modeling the complex network of speech-signal sequences for practical purposes such as speaker diarisation, and how these topological inputs may enable smarter learning (cf. Sec. S.1). Here are some of our first findings in this direction, set in the context of topological time series analysis.

2.1 Detection of vibration patterns

The impetus behind TopCap lies in an observation of how PD can capture vibration patterns within time series. To begin with, our aim is to determine which sorts of information can be extracted using topological methods. As the name indicates, topological methods quantify features based on topology, which distinguishes spaces that cannot continuously deform to each other. In the context of time series, we conduct a series of experiments to scrutinise the performance of topological methods, their limitations as well as their potential.

Given a periodic time series, its TDE target is situated on a closed curve (i.e., a loop) in a sufficiently high-dimensional Euclidean space (see Fig. 1a). Despite the satisfactory point-cloud representation of a periodic time series, it remains rare in practical measurement and observation to capture a truly periodic series. Often, we find ourselves dealing with time series that are not periodic yet exhibit certain patterns within some time segments. For instance, Fig. 1c portrays the average temperature of the United States from the year 2012 to 2022, as documented in [75]. Although the temperature does not adhere strictly to a periodic pattern, it does display a noticeable cyclical trend on an annual basis. Typically, the temperature tends to rise from January to July and fall from August to December, with each year approximately comprising one cycle of the variation pattern. One strength of topological methods is their ability to capture "cycles". A question then arises naturally: Can these methods also capture the cycle of temperature as well as subtle variations within and among these cycles? To be more precise, we first observe that variations occur in several ways. For instance, the amplitude (or range) of the annual temperature variation may fluctuate slightly, with the maximum and minimum annual temperatures varying from year to year. Additionally, the trend line for the annual average temperature also shows fluctuations, such as the average temperature in 2012 surpassing that of 2013. Despite each year's temperature pattern bearing resemblance to that depicted in the left panel in Fig. 1c (representing a single cycle of temperature within a year), it may be more beneficial for prediction and response strategies to focus on the evolution of this pattern rather than its specific form. In other words, attention should be directed towards how this cycle varies over the years. This leads to several questions. How can we consistently capture these subtle changes in the pattern's evolution, such as variations in the frequency, amplitude, and trend line of cycles? How can we describe the similarities and differences between time series that possess distinct evolutionary trajectories? In applications, these are crucial inquiries that warrant further exploration.

To address these questions, we propose three kinds of "fundamental variations" which are utilised for depicting the evolutionary trace of a time series. Consider a series of a periodic function $f(t_n) = f(t_n + T)$, where T is a period.

(1) *Variation of frequency*. Denote the frequency by $F = T^{-1}$. Note that the series is not necessarily periodic in the mathematical sense. Rather, it exhibits a recurring pattern after the period T. For instance, the average temperature from Fig. 1c is not a periodic series, but we consider its period to be one year since it follows a specific pattern, i.e., the one displayed in the left panel of Fig. 1c. This 1-year pattern always lasts for a year as time progresses. Hence, there is no frequency variation in this example. This type of variations can be represented as $g_1(t_n) = f\big(F(t_n) \cdot t_n\big)$, where $F(t_n)$ is a series representing the changing frequency. This type of variation occurs, for example, when one switches their vocal tone or when one's heartbeats experience a transition from walking mode to running mode.

(2) *Variation of amplitude*. The amplitudes of temperature in the years 2014 and 2015 are 42.73°F and 40.93°F, respectively. So the variation of amplitude from 2014 to 2015 is -1.80°F. This can be represented by $g_2(t_n) = A(t_n) \cdot f(t_n)$, where $A(t_n)$ is a series of the changing amplitude. This type of variation is observed when a particle vibrates with resistance or when there is a change in the volume of a sound.

(3) *Variation of average line*. The average temperatures through the years 2012 and 2013 are 55.28°F and 52.43°F, respectively. The variation of average line from 2012 to 2013 is -2.85°F. Let $g_3(t_n) = f(t_n) + L(t_n)$, where $L(t_n)$ is a series representing the variation of average line. This type of variation is observed when a stock experiences a downturn over several days or when global warming causes a year-by-year increase in temperature.

To summarise, Fig. 1e provides a visual representation of the three fundamental variations. It is important to note that these variations are not utilised to depict the pattern itself but rather to illustrate the variation within the pattern or how the time series oscillates over time. This approach offers a dynamic perspective on the evolution of the time series, capturing changes in patterns that static analyses may overlook.

Using three simulated time series corresponding to the above three fundamental types of variation (see Sec. 3.1 for detailed construction), we demonstrate that PD can distinguish these variations and detect how significant they are. See Fig. 3, where a smaller value of c indicates a more rapid fundamental variation. Here, regardless of which value c takes, each individual diagram features a prominent single point at the top and a cluster of points with relatively short duration, except when $F(t_n) = 1$ (i.e., $c = 4$). In this case, the series represents a cosine function, and thus the diagram consists of a single point. Normally, one tends to overlook the points in a PD that exhibit a short duration as they are sometimes inferred as noise. However, in this example, the distribution of those points holds valuable information regarding the three fundamental variations. As shown in Fig. 3, each fundamental variation has its distinct pattern of distribution in the lower region of a diagram, which leads to refined inferences: If the points spiral along the vertical axis of lifetime, it is probably due to a variation of amplitude; if every two or four points stay close to form a "shuttle", it probably indicates a variation of average line; otherwise the points just seem to randomly spread over, which more likely results from a variation of frequency. It

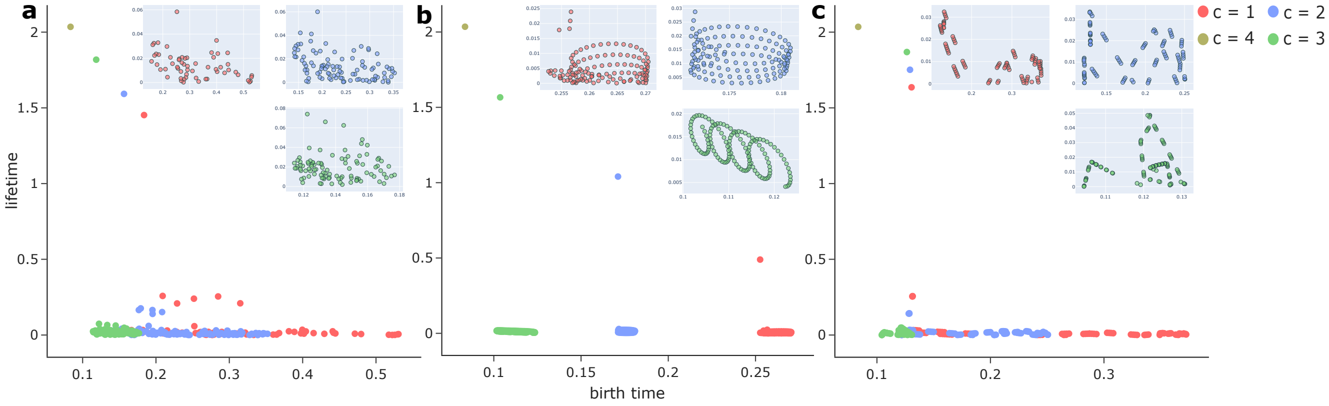

Fig. 3: 1-dimensional PH reveals three fundamental variations. **a**, Detecting variation of frequency. Upper-right panels zoom in to show the barcode distribution in the lower dense region, where the position and colour of each value of c in the main legend corresponds to those of its panel. Note that when $c = 4$, there is a single point, and so the panel for this value is omitted. **b**, Detecting variation of amplitude. **c**, Detecting variation of average line.

is also straightforward to distinguish the values of c for a specific fundamental variation, by their most significant point in the diagram: Longer lifetime for the barcode of the solitary point indicates slower variation. The lower region of a diagram also gives some hints in this respect.

In this simulated example, we demonstrated how PD could be utilised as a uniform means to distinguish three fundamental variations of the cosine series and their respective rates of change. However, it is important to note that in general scenarios, identifying the fundamental variations in a time series using topological methods may encounter significant challenges. Although topological methods are indeed capable of capturing this information, vectorising this information for subsequent utilisation remains a complex task at this stage. Having recognised the potential of topological methods, we resort to an alternative algorithm for handling time series. Specifically, despite the difficulty in vectorising PD to measure each fundamental variation, we have developed a simplified algorithm to measure the vibration of time series as a whole. This approach provides a comprehensive understanding of the overall behaviour of a time series, bypassing the need for complex vectorisation.

2.2 The three fundamental variations gleaned from a persistence diagram

A PD for 1-dimensional PH encodes much more information beyond the birth time and lifetime of the point of MP. The three fundamental variations examined in Sec. 2.1 also manifest themselves in certain regions of the PD, which can in turn be vectorised.

To capture these variations, we perform an experiment with two records of the vowel [ɑ]. Specifically, we demonstrate the fundamental variations by comparing the PDs of (a) the record of [ɑ] relatively unstable with respect to the fundamental variations and (b) the other record of the same vowel that is relatively stable. To better illustrate the results, we crop each record into 4 overlapping intervals, each starting from time 0 and ending at 600, 800, 1000, 1200, respectively. When adding a new segment of 200 units into the original sample each time, the amplitude and frequency of the series altered more drastically in case (a). A more rapid changing rate may lead to more points distributed in the lower region of the diagram. The outcomes are presented in Fig. 4. The plots in Fig. 4c show that the spectral frequency of (a) indeed varies faster than that of (b).

We should also mention that the 1-dimensional PD here serves as a profile for the collective effect of the fundamental variations. Currently, it is unclear how the points in the lower region change in response to a specific variation.

2.3 Traditional machine learning methods with novel topological features

In this subsection, we present our results on consonant recognition using topology-enhanced machine learning methods, notably, the streamlined approach of TopCap. The classification of voiced and voiceless consonants serves as a significant, relevant application of our methodology, showcasing its efficacy and advantages. Meanwhile, as a hands-on example originating directly from industrial innovation, it makes various technical considerations in developing our methods more transparent and highlights potential for further investigation and enhancement.

Voiced and voiceless regions of speech have distinct speech production processes and energy patterns. Segmentation of voiced and voiceless speech is a fundamental and important process for various speech processing applications [76]. In medical diagnosis, researchers can detect common cold and other diseases by studying voiceless and voiced sounds [77, 78]. The detection of voiced and voiceless sounds can also be used to reveal whether musical expertise leads to an altered neurophysiological processing of sub-segmental information available in the speech signal [79]. It is particularly important to study the segmentation of voiced and voiceless sounds in linguistics, and a variety of methods have been developed and applied [80, 81, 82, 83]. Moreover, there are applications geared towards AI innovations, for example, speaker identification via voiceless consonants [84]. Thus, it has become imperative to research the characteristics of voiced and voiceless sounds and distinguish them, which can ensure the accuracy of

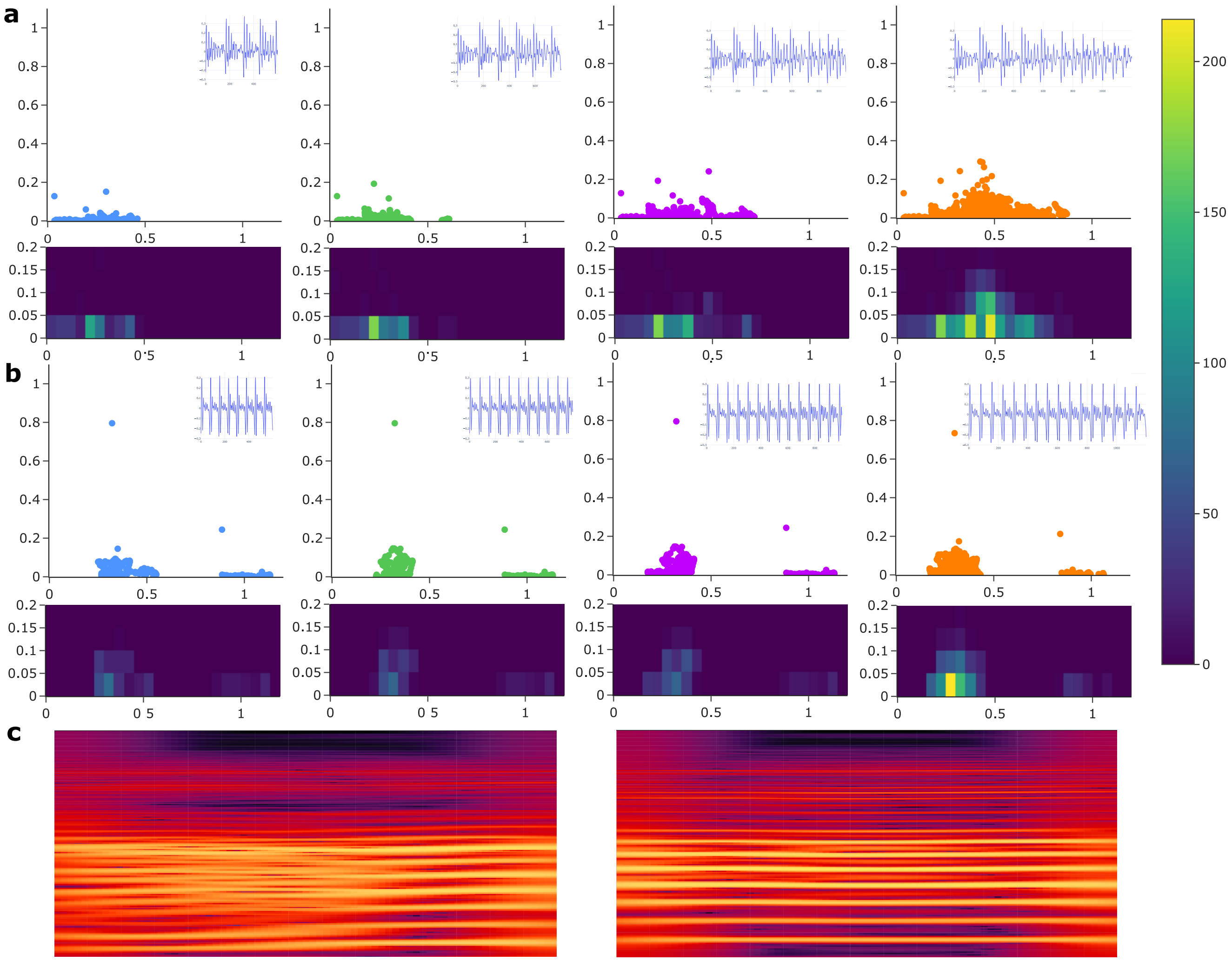

Fig. 4: Variation of 1-dimensional PDs due to the fundamental variations of time series. **a**, PDs of drastic fundamental variations. The small panel on top right of each diagram shows the original time series, with 4 segments extracted from the same record of [ɑ], each starting from time 0 and ending at time 600, 800, 1000, 1200, respectively. It can directly be seen from the time series that the variation of amplitude in (a) is bigger than (b); for frequency, see **c**; normally, we do not discuss the average line of phonetic data as it is assumed to be constant. Below, each diagram shows the clustering density of points in the lower region of the PD. **b**, PDs of mild fundamental variations for 4 time-series segments extracted from the other record of [ɑ], with the same ending and starting times as in (a). The lower density diagrams demonstrate that unstable time series are characterised by a higher density of points in the lower region of PD. Moreover, stable series tend to attain high MP. **c**, Spectral frequency plots of the time series with rapid variations (left) and with mild variations (right).

the segmentation and enable other applications. Placed in a broader context, this analysis for speech recognition at the phonemic level precedes the type of higher-order language processing typically associated with NLP.

Given consonant recognition as a significant problem originating and posed to us from the industry, we performed multiple topology-enhanced machine learning experiments and obtained the following.

2.3.1 Main experiment

Using datasets comprising human speech, we initially employ the Montreal Forced Aligner (MFA) [85] to align natural speech into phonetic segments. Following preprocessing of these phonetic segments, TDE is conducted with dimension parameter $d = 100$ and delay parameter τ set to equal $6T/d$, where T approximates the (minimal) period of the time series. Following additional refinement procedures, PDs are computed for these segments and are then vectorised based on MP and its corresponding birth time. The comprehensive procedural framework is expounded in Secs. 3.2 and 3.3, while the corresponding workflow is shown in Fig. 5e. It is worth noting that in the applications of TDE, the dimension parameter d is usually determined through some algorithms designed to identify the minimal appropriate dimension [50, 86]. Here, the embedded dimension $d = 100$ was chosen to be as large as possible within the constraints of our data. More specifically, in our experiments, using lower dimensions such as $d = 5$, 10, or 20 yielded poor results, as those dimensions were insufficient to capture the complex underlying structure of the time series. In higher dimensions, important features that are not apparent in lower dimensions become much easier to identify. However, the dimension cannot be too large either, otherwise the embedded point cloud obtained following the theoretical framework of Perea and Harer [71] (see Sec. 3.3.2 below for details) may consist of too few points to adequately represent the original data structure. The delay parameter τ is determined by an ACL function with no specific rule, but in many cases $\tau = mT/d$ for some positive integer m. In our pursuit of enhanced extraction of topological features, a relatively high dimension is chosen (see Sec. 4 for more discussion on dimension in TDE). Given this higher dimension, the usual case of $\tau = T/d$ with $m = 1$ may prove excessively diminutive, particularly in light of the time series only taking values in discrete time steps. Consequently, in TopCap we adopt an adjusted parametrisation for $\tau = mT/d$ with a relatively large value $m = 6$.

We input the pair of MP and birth time from 1-dimensional PD for each sound record to multiple traditional classification algorithms: Tree, Discriminant, Logistic Regression, Naive Bayes, Support Vector Machine, k-Nearest Neighbours, Kernel, Ensemble, and Neural Network. We use the application of the MATLAB (R2022b) Classification Learner, with 5-fold cross-validation, and set aside 30% records as test data. This application performs machine learning algorithms in an automatic way. There are a total of 1016 records, with 712 training samples and 304 test samples. Among them, 694 records are voiced consonants and the remaining are voiceless consonants. The models we choose in this application are Optimizable Tree, Optimizable Discriminant, Efficient Logistic Regression, Optimizable Naive Bayes, Optimizable SVM, Optimizable KNN, Kernel, Optimizable Ensemble, and Optimizable Neural Network.

The results are shown in Fig. 5a–d. The receiver operating characteristic curve (ROC), area under the curve (AUC), and accuracy metrics collectively demonstrate the efficacy of these topological features as inputs for a variety of machine learning algorithms. Each of the algorithms incorporating topological inputs attains AUC and accuracy surpassing 96%. The ROC and AUC together depict the high performance of our classification model across all classification thresholds. The 2D histograms depicted in Fig. 5c–d collectively illustrate the distinct distributions of voiced and voiceless consonants. Voiced consonants tend to exhibit a relatively higher birth time and lifetime, which provides an explanation for the high performance of these algorithms. Despite the intricate structure that a PD may present, appropriately extracted topological features enable traditional machine learning algorithms to separate complex data effectively. This highlights the potential of TDA in enhancing the performance of machine learning models.

2.3.2 Model comparison on benchmark datasets

We next demonstrate the advantages of TopCap by comparing it with state-of-the-art methods in speech recognition that are not based on topology, over a diverse range of benchmark datasets.

In the above main experiment, our analysis solely utilised the HT1 corpus sourced from the broader ALLSSTAR dataset of SpeechBox [87] (see Sec. 3.2 for details). We extend this by conducting a series of experiments across a diverse array of datasets using the same methodology, with the aim of enhancing the robustness and credibility of our results. These datasets encompass renowned benchmark repositories such as LJSpeech [88], TIMIT [89], and LibriSpeech [90], in addition to supplementary corpora sourced from ALLSSTAR. Collectively, they contain a substantial amount of phones, numbering in the hundreds of thousands: LJSpeech provides around 200000, TIMIT around 40000, LibriSpeech over 7000000 (1000 hours of speech), and ALLSSTAR around 20000 in total.

In terms of comparative analysis with existing methodologies, we have placed our approach alongside three methods that are not based on topology. We combine standard audio processing methods for feature extraction with state-of-the-art deep learning methods for classification tasks. The former methods include short-time Fourier transform (STFT) and mel-frequency cepstral coefficients (MFCC). The latter methods include CNNs, gated recurrent unit (GRU) networks, and Transformers. As such, we perform experiments on the above datasets using the methods of STFT–CNN, MFCC–GRU, and MFCC–Transformer, in comparison with those with TopCap. In more detail, TopCap comprises TDE–PH and an array of traditional, accessible machine learning methods. The coupling of TDE and PH serves to extract the latent topological features inherent in the time series, while STFT and MFCC each extract features through analytic methods. Our selection of the multiple machine learning and deep learning architectures in each experimental pipeline is informed by the nature of the extracted features. Specifically, the output spectrograms from STFT are imagery representations, making them well-suited for CNNs. In particular, we design and compare two models for this method, denoted by STFT–CNN and STFT–CNN^{+}: The former resizes each grey-scale spectrogram of 124×129 pixels through bilinear interpolation down to 8×8 with 386177 parameters, while the latter to 16×16 with 435329 parameters (a 90% reduction of parameters from the original), both consisting of 5 layers with 3 convolutional and 2 fully connected. In contrast, MFCC features, characterised by their lower dimensionality, are more appropriate for recurrent-neural-network architectures, such as GRUs and Transformers.

Tab. 1 presents the results of our experiments with TopCap and the comparison models on benchmark datasets listed above. In each table, on the leftmost column, the various datasets are displayed. The remaining columns record the data sizes (i.e., numbers of phones) along with the corresponding accuracy rates of TopCap and of the comparison models applied to these datasets. In the upper half of Tab. 1, we focus on small-scale datasets. The 5 subsets of ALLSSTAR each comprise their entire phones, while LJSpeech, TIMIT, and LibriSpeech datasets are sampled randomly, each containing 2000 samples with a half voiced consonants and the other half voiceless. The lower half of Tab. 1 displays the results from large-scale datasets.

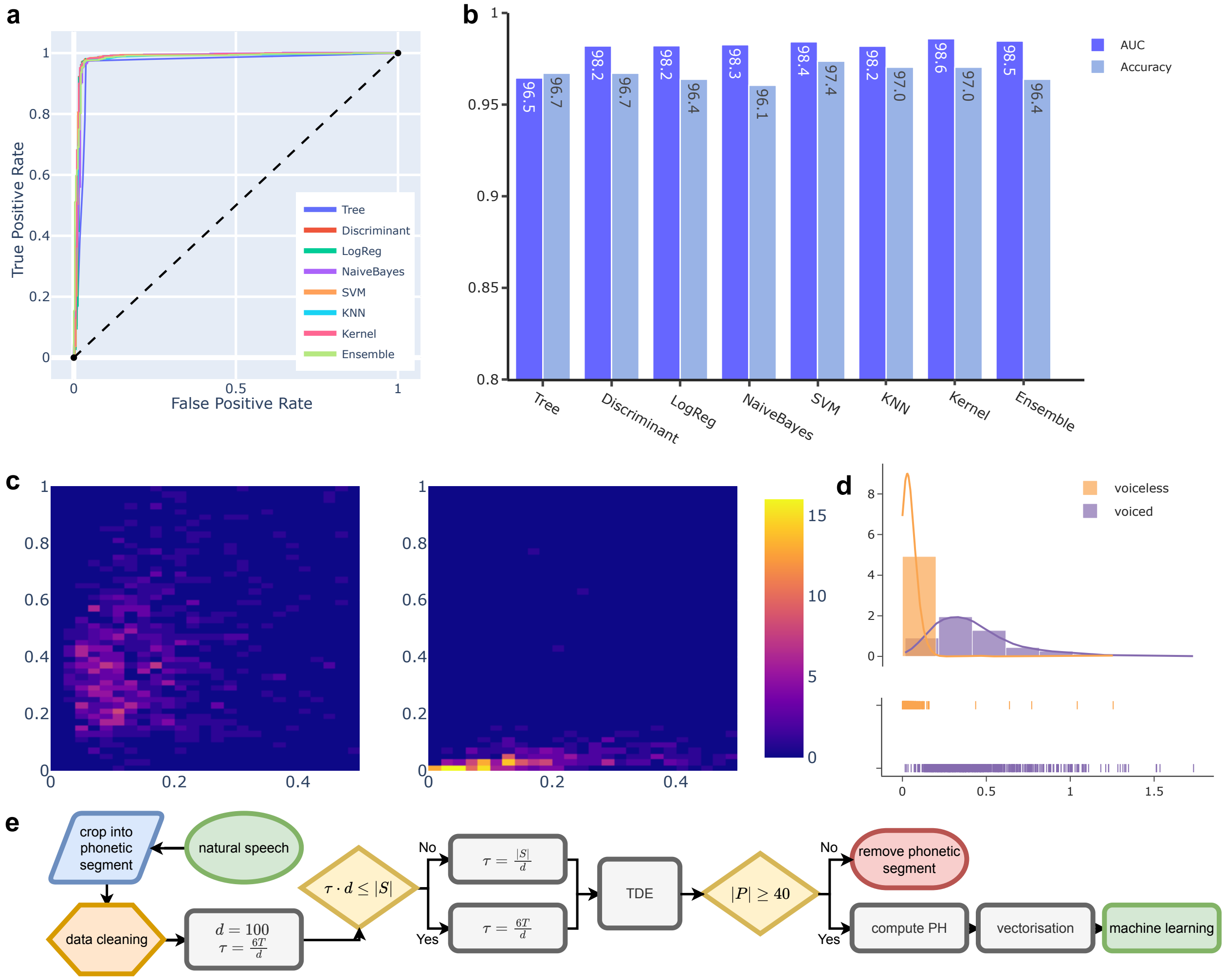

Fig. 5: Machine learning results with topological features. **a**, ROCs of traditional machine learning algorithms. **b**, Accuracy and AUC of each of these algorithms. **c**, Diagrams of records represented as (birth time, lifetime) for voiced consonants (left) and voiceless consonants (right), where voiced consonants exhibit relatively higher birth time and lifetime. The colour represents the density of points in each unit grid box. **d**, Histograms of records represented by their lifetime for voiced and voiceless consonants, together with kernel density estimation and rug plot. The distributions of MP can distinguish voiced and voiceless consonants. **e**, Flow chart of experiment. Here $|S|$ denotes the number of samples in a time series, $|P|$ denotes the number of points in the point cloud, and T denotes the (minimal) period of the time series computed by the ACL function.

	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	96.8	94.3	91.0	93.2	94.4	93.3	87.2	86.1
MFCC–GRU	92.0	91.3	88.9	88.7	92.0	87.7	85.3	80.0
MFCC–Transformer	96.9	95.2	96.3	92.2	97.2	96.3	96.6	92.5
STFT–CNN	84.0	85.0	83.7	84.8	84.2	79.7	78.1	77.6
STFT–CNN$^+$	95.1	96.4	95.8	92.4	92.4	94.8	90.1	91.2

Large dataset	ALLSSTAR	LJSpeech	TIMIT	LibriSpeech
Number of phones	21000	257000	42000	500000
TopCap	94.1	94.4	93.0	90.6
MFCC–GRU	94.0	96.7	96.3	93.8
MFCC–Transformer	95.3	97.8	97.1	95.0
STFT–CNN	84.6	84.5	77.6	80.3
STFT–CNN$^+$	95.0	96.5	91.1	93.6

Tab. 1: Performance of TopCap on 8 small datasets and 4 large datasets in comparison with state-of-the-art methods. The random samples are taken from the large datasets listed in the lower half of the table. In particular, in the second row, LJ and Libri are abbreviations for LJSpeech and LibriSpeech, respectively.

Among them, ALLSSTAR, LJSpeech, and TIMIT each contribute their entire data for analysis, while LibriSpeech does 500000 phones out of 1800000 from its speech data (we obtained 1800000 phonetic segments from a half of the 500-hour speech data). A main consideration for dividing the experiments into small and large datasets lies in the nature of training and generalisation for neural networks, which depend on the size of a dataset and correlate with the networks' performances.

The above results show that, in classification of voiced and voiceless consonants, our topology-enhanced model TopCap achieved an outstanding accuracy on small datasets and sustained a good performance on larger ones, in comparison with state-of-the-art models that are not based on topology. Besides, our topology-enhanced approach shows significant advantages in the following two areas.

- 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 utilises 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 characterise phonemes directly and effectively for our classification tasks (see also Sec. 2.3.3 below). 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 generalisation. In contrast, comparing the upper and lower halves of Tab. 1, we see that TopCap achieves equally good performance on relatively small datasets.
- 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 visualising features of time series data, which gives insight of the intrinsic properties and nuanced differences within the data, enabling us to better understand and improve the model.

To date, our TopCap method has incorporated only basic machine learning algorithms for accessibility. In future work, as explained at the beginning of Sec. 2, we aim to integrate local and global topological features with deep learning architectures and devise models with improved accuracy, robustness, generalisability, efficiency, and interpretability. Moreover, our work complements and potentially enhances applications in NLP, particularly those reliant on speech input.

2.3.3 Feature analysis

Finally, to further demonstrate our model's advantage in feature extraction, we carry out feature analysis by comparing clustering results based on our topological fingerprints and latent-space features from the above comparison model STFT–CNN^{+}.

Specifically, we present a comparative analysis between the features extracted by TDE–PH from our proposed method TopCap, i.e., (birth time, lifetime) of MP, and those derived from latent spaces of the CNNs. This analysis is performed on both TIMIT and LJSpeech datasets from the lower half of Tab. 1. We employ two prominent dimensionality reduction techniques: t-distributed stochastic neighbour embedding (t-SNE) and uniform manifold approximation and projection (UMAP), the latter being grounded in the principles of manifold learning and TDA [91]. These techniques enable us to visualise high-dimensional data in a lower-dimensional space, thereby offering insight into the underlying structure and characteristics within the data. The results are shown in Fig. 6.

In summary, from our topological detection results, the most significant distinction between voiced and voiceless consonants is that the former exhibit higher MP. This can scarcely be detected in lower dimensions regardless of how we tune the delay parameter τ in TDE. Besides Fig. 5, see also Fig. 2 for a sample of the recognition of vowels as well as consonants in terms of their *shapes*. To demonstrate the advantages of our proposed approach TopCap, we performed comparative experiments and feature analysis with state-of-the-art methods on multiple examples of datasets, as presented in Tab. 1 and Fig. 6.

3 Methods

3.1 Constructing vibrating time series

There are three kinds of fundamental variations mentioned in Sec. 2.1. In order to substantiate our argument, let $t_n = 0.01n$ with $0 \leqslant t_n \leqslant 7\pi$ and for each $c \in \{1, 2, 3, 4\}$ define

$$\begin{aligned} f(t_n) &= \cos(t_n) \\ F(t_n) &= \frac{c}{4} + \frac{1 - \frac{c}{4}}{7\pi} \cdot t_n \\ g_1(t_n) &= f\big(F(t_n) \cdot t_n\big) \end{aligned}$$

Note that $F(t_n) = c/4$ when $t_n = 0$ and $F(t_n) = 1$ when $t_n = 7\pi$. In fact, $F(t_n)$ is a sequence of line segments connecting $(0, c/4)$ and $(7\pi, 1)$. Correspondingly, the frequency of $g_1(t_n)$ changes more slowly as c increases. In the extreme case when $c = 4$, we have $F(t_n) = 1$, so

$$g_1(t_n) = f\big(F(t_n) \cdot t_n\big) = f(t_n) = \cos(t_n)$$

which is a periodic function. For each value of c, we applied TDE to the series $g_1(t_n)$ with dimension 3, delay 100, skip 10 and computed the 1-dimensional PD of the embedded point cloud. See Fig. 3a for the results. Replacing $F(t_n)$ by $A(t_n)$ and $L(t_n)$, we obtained the diagrams in Figs. 3b and 3c, respectively.

3.2 Obtaining phonetic data from natural speech

We used speech files sourced from SpeechBox [87], ALLSSTAR Corpus, task HT1 language English L1 file, retrieved on 28th January 2023. SpeechBox is a web-based system providing access to an extensive collection of digital speech corpora developed by the Speech Communication

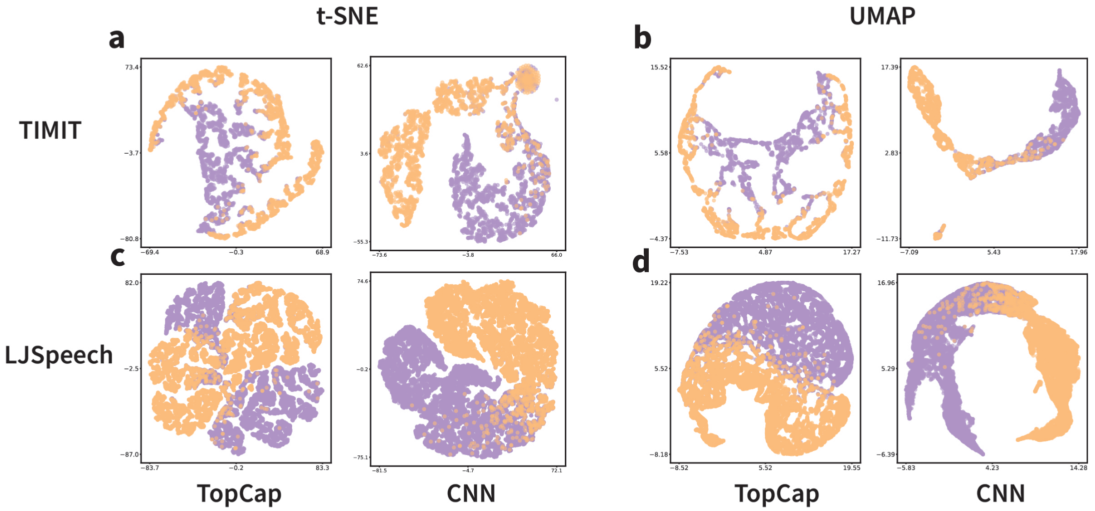

Fig. 6: Feature analysis for TopCap in comparison with STFT–CNN$^+$. Voiced consonants are labeled in purple and voiceless ones in orange, of which 10 percent from each dataset are plotted. **a**, Plots for TopCap (left) and STFT–CNN$^+$ (right) using t-SNE on TIMIT. **b**, Plots with UMAP on TIMIT. **c**, Plots with t-SNE on LJSpeech. **d**, Plots with UMAP on LJSpeech.

Research Group in the Department of Linguistics at Northwestern University. This section contains a total of 25 individual files, comprising 14 files from women and 11 files from men. The age range of these speakers spans from 18 to 26 years, with an average of 19.92. Each file is presented in the WAV format and is accompanied by its corresponding aligned file in Textgrid format, which features three tiers of sentences, words, and phones. Collectively, these 25 speech files amount to a total duration of 41.21 minutes. The speech file contains each individual reading the same sentences consecutively for a duration ranging from 80 to 120 seconds, contingent upon each person's pace. The original .wav file has a sampling frequency of 22050 and comprises only one channel. Since MFA [85] is trained in a sampling frequency of 16000, we opted to adjust the sampling frequency of the .wav files accordingly. We then extracted the "words" tier from Textgrid and aligned words into phones using English_MFA dictionary and acoustic model (MFA version 2.0.6). Thus we obtained corresponding phonetic data from these speech files.

Subsequently, we used voiced and voiceless consonants in those segments as our dataset. Voiced consonants are consonants for which vocal cords vibrate in the throat during articulation, while voiceless consonants are pronounced otherwise (see also Sec. S.1). Specifically, using Praat [92], we extracted voiced consonants [ŋ], [m], [n], [j], [l], [v], and [ʒ]; for voiceless consonants, we selected [f], [k], [ɵ], [t], [s], and [tʃ]. These phones were then read as time series. Our selection was limited to these voiced and voiceless consonants, as we aimed to balance the ratio of voiced and voiceless consonant records in these speech files. Additionally, some consonants, such as [d] and [h], appeared difficult to classify by our methods.

3.3 Deriving topological features from phonetic data

Prior to the extraction of topological features from a time series, we first imbued this 1-dimensional time series with a (Euclidean) topological structure through TDE. It is noteworthy that this technique also applies to multi-dimensional time series. The ambient space throughout this article is always a Euclidean space. By establishing the topological structure there, or more precisely, the distance matrices, we subsequently calculated PH. We elaborate on the following main steps. See Fig. 5e for the flow chart of this section.

3.3.1 Data cleaning

This involved eliminating the initial and final segments of a time series until the first point with an amplitude exceeding 0.03 occurred. This approach was aimed at mitigating the impact of environmental noise at the beginning and end of a phone. Any resulting series with fewer than 500 points will be disregarded, as such series were considered insufficiently long or to contain excessive environmental noise.

3.3.2 Parameter selection for time-delay embedding

We selected suitable parameters for TDE to capture the theoretically optimal MP of a given time series. The dimension of the embedding was fixed to be 100. Our principle for determining an appropriate dimension is that we want to choose the embedded dimension to be large for a time series of limited length. As discussed in Sec. 4 and cf. Sec. S.2.1, a higher dimension results in a more accurate approximation. This approach also aimed to enhance computational efficiency and the occurrence of more prominent MP. Nonetheless, it is imperative to exercise caution when selecting the dimension, as excessively large dimensions may lead to empty point clouds and other uncontrollable factors. For instance, with a time series consisting of approximately 1200 points, setting the dimension to 100, delay to 5, and skip to 1 results in around 700 points in the corresponding point cloud. However, increasing the dimension to 200 under the same parameters would yield only 200 points, which may be too few to adequately represent the original data structure. Thus, the dimension was chosen to be as large as possible while maintaining sufficient data points in the point cloud.

With a proper dimension, we then computed the delay for the embedding. According to Perea and Harer [71], in the case of a periodic function, the optimal delays τ can be expressed as

$$\tau = m \cdot \frac{T}{d}$$

where T denotes the (minimal) period, d represents the dimension of the embedding, and m is a positive integer.

Under these conditions, we could obtain the theoretically optimal MP. The time series under consideration in our case was far from periodic, however, so we used the first peak of the ACL function to represent the period T and set $m = 6$, thus obtaining a relatively proper delay τ. The common choice of τ is to let window size equal the (minimal) period. However, in the case of a discrete time series, one often obtains $\tau = 0$ or $\tau = 1$ in this way, since the dimension of TDE is too large in comparison. Therefore, one strategy is to increase m to get a relatively reasonable τ. The performance of delay obtained in this way is presented in Sec. 4.

Then τ was rounded to the nearest integer (if it equals 0, take 1 instead). It was common that $\tau \cdot d$ exceeded the number of points in the series, resulting in an empty embedding. In this case, we adopted $\tau = |S|/d$, where $|S|$ denotes the number of points (i.e., the point capacity of the time series), and then rounded it downwards. This enabled us to obtain the appropriate delay for each time series, thereby facilitating the attainment of significant MP for the specified dimension.

Lastly, we let skip equal to 5. We chose this skip mainly to reach a satisfactory computation time. The impact of the skip parameter in TDE on MP and computation time is expounded upon in Sec. S.3.1.

Once the parameters were set, the time series were transformed into point clouds. If the number $|P|$ of points in a point cloud was less than 40, we excluded this time series from further analysis, considering that there were too few points to represent the original structure of the time series. The problem of lacking points is also discussed in Sec. 4.

3.3.3 Computing persistent homology

Using Ripser [93, 94], we could compute the PDs of the point clouds in a fast and efficient way. We then extracted MP from each 1-dimensional PD, using persistence birth time and lifetime as two features of a time series. The process of vectorising a PD presents a challenge due to the indeterminate (and potentially large) number of intervals in the barcode, coupled with the ambiguous information they contain. This ambiguity arises from our lack of knowledge about the types of information that can be derived from different parts of the PD. Here we only extracted the MP and corresponding birth time. This decision was informed by our prior selection of an appropriate set of parameters, which ensured that the MP reached its optimal.

4 Discussion

In the realm of applying topological methods to analyse time series [52, 53, 54, 55, 22, 56, 27], the determination of parameters for TDE emerges as a pivotal aspect. This stems from the significant impact that the selection of parameters has on the resulting topological spaces and their corresponding PDs. There exist several convenient algorithms for parameter selection. For example, the False Nearest Neighbours algorithm (FNN), a widely utilised tool, provides a method for deciding the minimal embedded dimension [86]. However, in the context of PH, usually the objective is not to achieve a *minimal* dimension. Contrarily, a dimension of substantial magnitude may be desirable due to certain advantages it offers.

In this section, as a main novel feature of TopCap, we reveal and leverage the relationship between embedded dimension and maximal persistence. We relegate further aspects of parameter selection to Sec. S.3.

In the TDE–PH approach, the determination of dimension in a TDE can be complex. However, it plays a pivotal role in the extraction of topological descriptors such as MP. It is observed that a larger dimension can significantly enhance the theoretically optimal MP of a time series. In TopCap, the dimension of TDE is set to be 100, a relatively large dimension for the experiment. On the other hand, several factors also constrain this choice. These include the length of the sampled time series, since the dimension cannot exceed the length (otherwise it would render the resulting point cloud literally pointless). The constraints also include the periodicity of the time series, as the time-delay window size should be compatible with the approximate period of the time series, which is to be elaborated below.

According to Perea and Harer [71, Proposition 5.1], there is no information loss for trigonometric polynomials if and only if the dimension of TDE exceeds twice the maximal frequency. Here, no information loss implies that the original time series can be fully reconstructed from the embedded point cloud. In general, for a periodic function, a higher dimension of TDE can yield a more precise approximation by trigonometric polynomials. Although there are no absolutely periodic functions in real data, each time series exhibits its own pattern of vibration, as discussed in Sec. 2.1, and a higher dimension of embedding may be employed to capture a more accurate vibration pattern in the time series. Furthermore, an increased embedded dimension may result in reduced computation time for PD. For instance, computation times for a voiced consonant [ŋ] are 0.2671, 0.2473, and 0.2375 seconds, corresponding to embedded dimensions 10, 100, and 1000 (see Fig. 7a). This is attributed to the reduction due to a higher dimension on the number of points in the embedded point cloud. While this reduction in computation time may not be considered substantial compared to the impact of changing skip (see Fig. 7d), it may become significant when handling large datasets. More importantly, an increased embedded dimension can yield benefits such as enhanced MP, which serves as a major motivation for higher dimensions, as well as a smoother shape of resulting point clouds obtained through TDE, which makes the embedding visibly reasonable. Typically, for most algorithms, a lower dimension is preferred due to factors such as those associated with curse of dimensionality and computation cost. By contrast, in TopCap, we opt instead for a higher dimension.

However, the embedded dimension cannot be arbitrarily large. As illustrated in Fig. 7c, when the embedded dimension escalates to 1280, it becomes unfeasible to capture a significant MP in the phonetic time series. This results from a break of the point cloud. When the embedded dimension

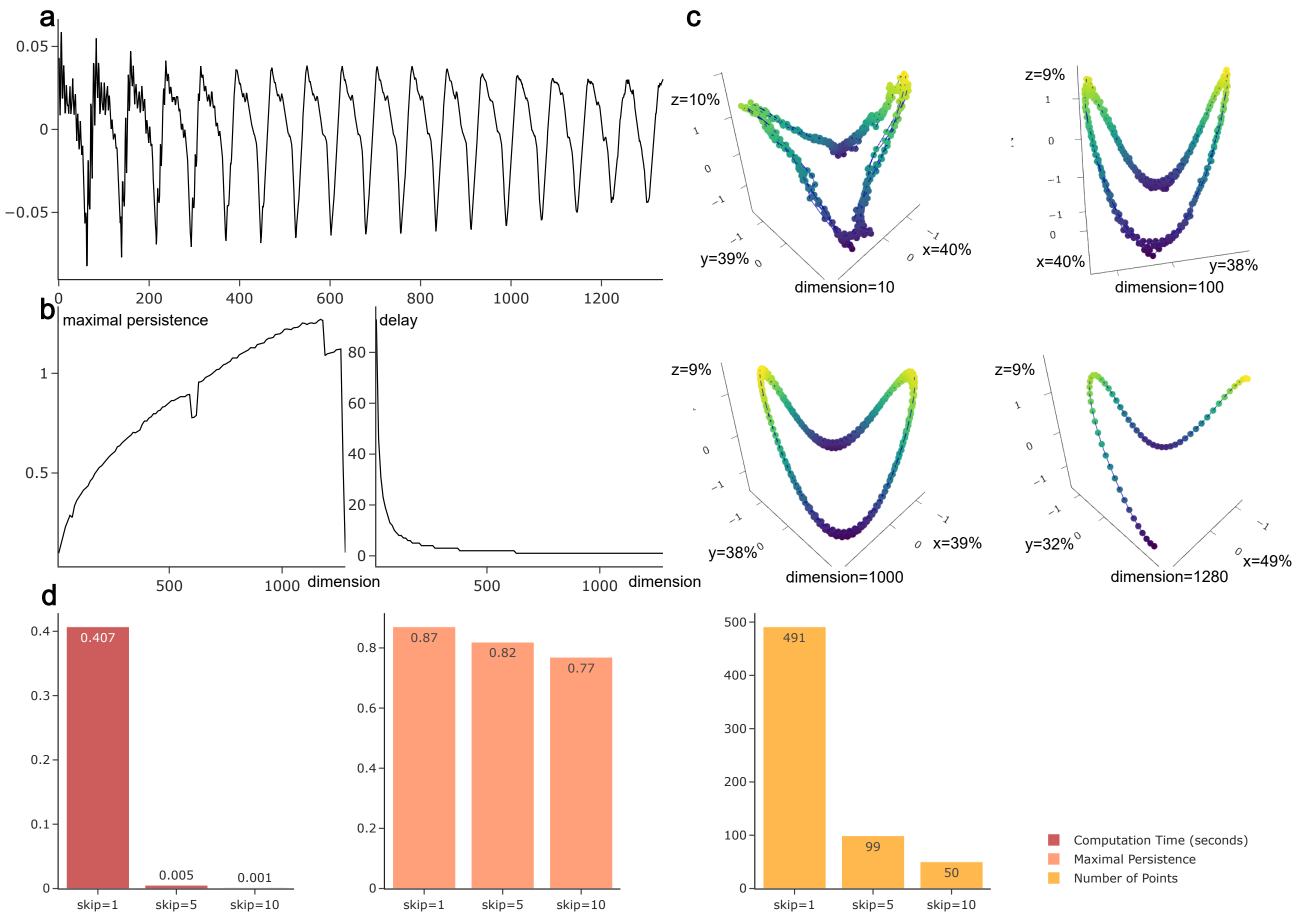

Fig. 7: Point-cloud behaviour with increasing embedded dimension. **a**, Original .wav file of a record of [ŋ] (voiced consonant). **b**, MP of the series after TDE as dimension increases (left) and the corresponding delay that ensures the time series to reach theoretically optimal MP (right). Skip equals 5 when computing PD. **c**, Visualisation of the embedded point clouds, which shows principal component analysis (PCA) of the embedded point clouds in 3D as projected from various dimensions. Skip equals 1 when performing PCA. The percentage along each axis indicates the PCA explained variance ratio. **d**, Given a sound record of the voiced consonant [m], computation time, MP, and the size of point clouds as skip increases (see Sec. S.3.1 for details). An increase in skip can lead to a significant reduction in computation time, owing to the reduced size of the point cloud. However, MP remains resilient to an increase in the skip parameter.

further reaches 1290, an empty 1-dimensional barcode is obtained due to the lack of points necessary to form even a single cycle. In this way, the dimension of TDE is related to the length of the time series.

Using a sound record of the voiced consonant [ŋ] as an exemplar, we delineate the correlation between MP and embedded dimension in Fig. 7a–c. As depicted in Fig. 7b, MP tends to escalate rapidly and nonlinearly with the increase in dimension, signifying that a more substantial MP is captured in higher-dimensional TDE. Notably, two precipitous drops in MP are observed, corresponding to embedded dimensions 600 and 1190. When $d = 600$, this time series can theoretically attain its optimal MP when $\tau = 2$ (see Sec. S.2.1). However, given the length of the series is 1337 and the window size is $d \cdot \tau = 1200$, with the skip set as 5, only 28 points are in the resulting point cloud for PD computation. The sparse point cloud fails to represent the original series adequately, leading to a decrease in MP. A similar phenomenon occurs when the dimension reaches 1190. The principal component analysis for dimension 1280 is shown in Fig. 7c. In this scenario, as observed above, the hypothetical cycle fails to form as there is a break in the point cloud, resulting in a free-fall in MP. In contrast, when $d = 630$, this series has a significant MP when $\tau = 1$, resulting in a window size of $d{\cdot}\tau = 630$. There are 142 points in the point cloud for the persistence diagram if skip equals 5, ensuring that the MP rises again without any breakdown. The embedded dimension also contributes significantly to the geometric property of time-delay embedding, as the shape becomes smoother in higher dimensions and the point cloud more structural.

As mentioned above, there are three crucial parameters in TDE, namely, d, τ, and skip. However, it is worth noting that the TDE–PH approach encompasses many other significant variables and choices. These include the construction of underlying topological space of the point clouds (i.e., the distance function for pairwise points), and the type of complexes utilised in filtering PH, among others. Some of these choices, despite their importance, were seldom addressed in the literature. Here, we propose a method for determining delay in order to capture the theoretically optimal MP of a time series in high-dimensional TDE. In future research, we aim at more systematic approaches for determining other parameters, particularly dimension of the TDE.

5 Data and code availability

The data that support the findings of this study are openly available in SpeechBox [87], ALLSSTAR Corpora, at https://speechbox.linguistics.northwestern.edu, as well as LJSpeech [88], TIMIT [89], and LibriSpeech [90].

The source code and supplementary materials for TopCap can be accessed on the GitHub page at https://github.com/sustech-topology/TopCap.

References

[1] Gunnar Carlsson. "Topology and data". In: *Bulletin of The American Mathematical Society* 46 (Apr. 2009), pp. 255–308. DOI: 10.1090/S0273-0979-09-01249-X.

[2] Ephy R. Love et al. "Topological convolutional layers for deep learning". In: *Journal of Machine Learning Research* 24.59 (2023), pp. 1–35.

[3] Gunnar Carlsson and Rickard Brüel Gabrielsson. "Topological approaches to deep learning". In: *Topological Data Analysis: The Abel Symposium 2018*. Springer. 2020, pp. 119–146.

[4] Ken W Grant, Brian E Walden, and Philip F Seitz. "Auditory-visual speech recognition by hearing-impaired subjects: Consonant recognition, sentence recognition, and auditory-visual integration". In: *The Journal of the Acoustical Society of America* 103.5 (1998), pp. 2677–2690. ISSN: 0001-4966.

[5] Yichen Shen et al. "Deep learning with coherent nanophotonic circuits". In: *Nature Photonics* 11.7 (2017), pp. 441–446. ISSN: 1749-4893.

[6] Eric W Healy et al. "Speech-cue transmission by an algorithm to increase consonant recognition in noise for hearing-impaired listeners". In: *The Journal of the Acoustical Society of America* 136.6 (2014), pp. 3325–3336. ISSN: 0001-4966.

[7] Thierry Nazzi and Anne Cutler. "How consonants and vowels shape spoken-language recognition". In: *Annual Review of Linguistics* 5 (2019), pp. 25–47. ISSN: 2333-9683.

[8] Dianne J Van Tasell et al. "Speech waveform envelope cues for consonant recognition". In: *The Journal of the Acoustical Society of America* 82.4 (1987), pp. 1152–1161. ISSN: 0001-4966.

[9] DeLiang Wang and Guoning Hu. "Unvoiced speech segregation". In: *2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings*. Vol. 5. 2006, pp. V–V. DOI: 10.1109/ICASSP.2006.1661435.

[10] Philip Weber et al. "Consonant recognition with continuous-state hidden Markov models and perceptually-motivated features". In: *Sixteenth Annual Conference of the International Speech Communication Association*. 2015.

[11] Gunnar Carlsson et al. "Persistence barcodes for shapes". In: *International Journal of Shape Modeling* 11.02 (2005), pp. 149–187. DOI: 10.1142/s0218654305000761.

[12] Oleksandr Balabanov and Mats Granath. "Unsupervised learning using topological data augmentation". In: *Physical Review Research* 2.1 (2020), p. 013354.

[13] Azadeh Hadadi et al. "Prediction of cybersickness in virtual environments using topological data analysis and machine learning". In: *Frontiers in Virtual Reality* 3 (2022), p. 973236. ISSN: 2673-4192.

[14] Firas A. Khasawneh, Elizabeth Munch, and Jose A. Perea. "Chatter classification in turning using machine learning and topological data analysis". In: *IFAC-PapersOnLine* 51.14 (2018), pp. 195–200. ISSN: 2405-8963. DOI: 10.1016/j.ifacol.2018.07.222.

[15] Daniel Leykam and Dimitris G. Angelakis. "Topological data analysis and machine learning". In: *Advances in Physics: X* 8.1 (2023). ISSN: 2374-6149. DOI: 10.1080/23746149.2023.2202331.

[16] Grzegorz Muszynski et al. "Topological data analysis and machine learning for recognizing atmospheric river patterns in large climate datasets". In: *Geoscientific Model Development* 12.2 (2019), pp. 613–628. ISSN: 1991-9603. DOI: 10.5194/gmd-12-613-2019.

[17] Frédéric Chazal and Bertrand Michel. "An introduction to topological data analysis: Fundamental and practical aspects for data scientists". In: *Frontiers in Artificial Intelligence* 4 (2021), p. 108. ISSN: 2624-8212.

[18] Aras Asaad and Sabah Jassim. "Topological data analysis for image tampering detection". In: *Digital Forensics and Watermarking: 16th International Workshop, IWDW 2017, Magdeburg, Germany, August 23-25, 2017, Proceedings 16*. Springer. 2017, pp. 136–146.

[19] Lander Ver Hoef et al. "A primer on topological data analysis to support image analysis tasks in environmental science". In: *Artificial Intelligence for the Earth Systems* 2.1 (2023), e220039.

[20] Gunnar Carlsson et al. "On the local behavior of spaces of natural images". In: *International Journal of Computer Vision* 76 (Jan. 2008), pp. 1–12. DOI: 10.1007/s11263-007-0056-x.

[21] Sebastian Zeng et al. "Topological attention for time series forecasting". In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 24871–24882.

[22] Yuhei Umeda. "Time series classification via topological data analysis". In: *Information and Media Technologies* 12 (2017), pp. 228–239. ISSN: 1881-0896.

[23] Manish Saggar et al. "Towards a new approach to reveal dynamical organization of the brain using topological data analysis". In: *Nature Communications* 9.1 (2018), p. 1399. ISSN: 2041-1723. DOI: 10.1038/s41467-018-03664-4.

[24] Jessica L Nielson et al. "Uncovering precision phenotype-biomarker associations in traumatic brain injury using topological data analysis". In: *PLOS One* 12.3 (2017), e0169490. ISSN: 1932-6203.

[25] Tamal K Dey and Sayan Mandal. "Protein classification with improved topological data analysis". In: *18th International Workshop on Algorithms in Bioinformatics*

(WABI 2018). Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik. 2018.

[26] Alessio Martino, Antonello Rizzi, and Fabio Massimo Frattale Mascioli. "Supervised approaches for protein function prediction by topological data analysis". In: *2018 International Joint Conference on Neural Networks (IJCNN)*. IEEE. 2018, pp. 1–8.

[27] Kenneth Brown and Kevin Knudson. "Nonlinear statistics of human speech data". In: *International Journal of Bifurcation and Chaos* 19 (July 2009), pp. 2307–2319. DOI: 10.1142/S0218127409024086.

[28] Sergio Barbarossa and Stefania Sardellitti. "Topological signal processing over simplicial complexes". In: *IEEE Transactions on Signal Processing* 68 (2020), pp. 2992–3007.

[29] Eduard Tulchinskii et al. "Topological data analysis for speech processing". In: *Proc. Interspeech 2023*, pp. 311–315. DOI: 10.21437/Interspeech.2023-1861.

[30] Deli Chen et al. "Measuring and relieving the oversmoothing problem for graph neural networks from the topological view". In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 34. 2020, pp. 3438–3445.

[31] Woong Bae, Jaejun Yoo, and Jong Chul Ye. "Beyond deep residual learning for image restoration: Persistent homology-guided manifold simplification". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*. 2017, pp. 145–153.

[32] Christoph Hofer et al. "Deep learning with topological signatures". In: *Advances in Neural Information Processing Systems* 30 (2017).

[33] Emilie Gerardin et al. "Multidimensional classification of hippocampal shape features discriminates Alzheimer's disease and mild cognitive impairment from normal aging". In: *Neuroimage* 47.4 (2009), pp. 1476–1486. ISSN: 1053-8119.

[34] Mingqiang Yang, Kidiyo Kpalma, and Joseph Ronsin. "A survey of shape feature extraction techniques". In: *Pattern Recognition* 15.7 (2008), pp. 43–90.

[35] Zenggang Xiong et al. "Research on image retrieval algorithm based on combination of color and shape features". In: *Journal of Signal Processing Systems* 93 (2021), pp. 139–146. ISSN: 1939-8018.

[36] Dengsheng Zhang and Guojun Lu. "Review of shape representation and description techniques". In: *Pattern Recognition* 37.1 (2004), pp. 1–19. ISSN: 0031-3203. DOI: 10.1016/j.patcog.2003.07.008.

[37] John Lee. *Introduction to topological manifolds*. Vol. 202. Springer Science & Business Media, 2010. ISBN: 1441979409.

[38] Allen Hatcher. *Algebraic topology*. Cambridge: Cambridge University Press, 2002.

[39] Afra Zomorodian and Gunnar Carlsson. "Computing persistent homology". In: *Discrete & Computational Geometry* 33.2 (2005), pp. 249–274. ISSN: 1432-0444. DOI: 10.1007/s00454-004-1146-y.

[40] Herbert Edelsbrunner and John Harer. "Persistent homology – a survey". In: *Contemporary Mathematics* 453.26 (2008), pp. 257–282.

[41] Robert Ghrist. "Barcodes: The persistent topology of data". In: *Bulletin of the American Mathematical Society* 45.1 (2008), pp. 61–75. ISSN: 0273-0979.

[42] David Cohen-Steiner, Herbert Edelsbrunner, and John Harer. "Stability of persistence diagrams". In: *Proceedings of the Twenty-First Annual Symposium on Computational geometry*. 2005, pp. 263–271.

[43] Zixuan Cang, Lin Mu, and Guo-Wei Wei. "Representability of algebraic topology for biomolecules in machine learning based scoring and virtual screening". In: *PLOS Computational Biology* 14.1 (2018), e1005929.

[44] Guo-Wei Wei. "Persistent homology analysis of biomolecular data". In: *SIAM News* 50.10 (2017).

[45] Yasuaki Hiraoka et al. "Hierarchical structures of amorphous solids characterized by persistent homology". In: *Proceedings of the National Academy of Sciences* 113.26 (2016), pp. 7035–7040.

[46] Kelin Xia and Guo-Wei Wei. "Persistent homology analysis of protein structure, flexibility, and folding". In: *International Journal for Numerical Methods in Biomedical Engineering* 30.8 (2014), pp. 814–844.

[47] Zixuan Cang and Guo-Wei Wei. "TopologyNet: Topology based deep convolutional and multi-task neural networks for biomolecular property predictions". In: *PLOS Computational Biology* 13.7 (2017), e1005690.

[48] Charu C Aggarwal, Alexander Hinneburg, and Daniel A Keim. "On the surprising behavior of distance metrics in high dimensional space". In: *Database Theory—ICDT 2001: 8th International Conference London, UK, January 4–6, 2001 Proceedings 8*. Springer. 2001, pp. 420–434.

[49] Vladislav Polianskii and Florian T. Pokorny. "Voronoi graph traversal in high dimensions with applications to topological data analysis and piecewise linear interpolation". In: *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. KDD '20. Virtual Event, CA, USA: Association for Computing Machinery, 2020, pp. 2154–2164. ISBN: 9781450379984. DOI: 10.1145/3394486.3403266.

[50] Matthew B Kennel, Reggie Brown, and Henry DI Abarbanel. "Determining embedding dimension for phase-space reconstruction using a geometrical construction". In: *Physical Review A* 45.6 (1992), p. 3403.

[51] Baihan Lin. "Topological data analysis in time series: Temporal filtration and application to single-cell genomics". In: *Algorithms* 15.10 (2022), p. 371. ISSN: 1999-4893.

[52] Saba Emrani, Thanos Gentimis, and Hamid Krim. "Persistent homology of delay embeddings and its application to wheeze detection". In: *IEEE Signal Processing Letters* 21.4 (2014), pp. 459–463. ISSN: 1070-9908.

[53] Cássio MM Pereira and Rodrigo F de Mello. "Persistent homology for time series and spatial data clustering". In: *Expert Systems with Applications* 42.15-16 (2015), pp. 6026–6038. ISSN: 0957-4174.

[54] Firas A Khasawneh and Elizabeth Munch. "Chatter detection in turning using persistent homology". In: *Mechanical Systems and Signal Processing* 70 (2016), pp. 527–541. ISSN: 0888-3270.

[55] Lee M Seversky, Shelby Davis, and Matthew Berger. "On time-series topological data analysis: New data and opportunities". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*. 2016, pp. 59–67.

[56] Marian Gidea and Yuri Katz. "Topological data analysis of financial time series: Landscapes of crashes". In: *Physica A: Statistical Mechanics and its Applications* 491 (2018), pp. 820–834. ISSN: 0378-4371.

[57] Richard Bellman. "Dynamic programming". In: *Science* 153.3731 (1966), pp. 34–37.

[58] Afra Zomorodian and Gunnar Carlsson. "Computing persistent homology". In: *Discrete & Computational Geometry* 33.2 (Feb. 2005), pp. 249–274. ISSN: 1432-0444. DOI: 10.1007/s00454-004-1146-y.

[59] Nikola Milosavljević, Dmitriy Morozov, and Primoz Skraba. "Zigzag persistent homology in matrix multiplication time". In: *Proceedings of the Twenty-Seventh Annual Symposium on Computational Geometry*. 2011, pp. 216–225.

[60] N. Atienza, R. Gonzalez-Diaz, and M. Rucco. "Persistent entropy for separating topological features from noise in vietoris-rips complexes". In: *Journal of Intelligent Information Systems* 52 (2019), pp. 637–655. DOI: 10.1007/s10844-017-0473-4.

[61] Yu-Min Chung and Austin Lawson. "Persistence curves: A canonical framework for summarizing persistence diagrams". In: *Advances in Computational Mathematics* 48.6 (2022). DOI: 10.1007/s10444-021-09893-4.

[62] Peter Bubenik. "Statistical topological data analysis using persistence landscapes". In: *Journal of Machine Learning Research* 16.1 (2015), pp. 77–102.

[63] Henry Adams et al. "Persistence images: A stable vector representation of persistent homology". In: *Journal of Machine Learning Research* 18 (2017).

[64] D. Ali et al. "A survey of vectorization methods in topological data analysis". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2023), pp. 1–14. ISSN: 1939-3539. DOI: 10.1109/TPAMI.2023.3308391.

[65] Kelin Xia et al. "Persistent homology for the quantitative prediction of fullerene stability". In: *Journal of Computational Chemistry* 36.6 (2015), pp. 408–422.

[66] D Vijay Anand et al. "Weighted persistent homology for osmolyte molecular aggregation and hydrogen-bonding network analysis". In: *Scientific Reports* 10.1 (2020), p. 9685.

[67] K. Xia, Z. Li, and L. Mu. "Multiscale persistent functions for biomolecular structure characterization". In: *Bulletin of Mathematical Biology* 80 (2018), pp. 1–31. DOI: 10.1007/s11538-017-0362-6.

[68] James D Hamilton. *Time series analysis*. Princeton University Press, 2020.

[69] Yu-Min Chung et al. "A persistent homology approach to heart rate variability analysis with an application to sleep-wake classification". In: *Frontiers in Physiology* 12 (2021), p. 637684. ISSN: 1664-042X.

[70] Jose A Perea. "Topological time series analysis". In: *Notices of the American Mathematical Society* 66.5 (2019), pp. 686–694.

[71] Jose A. Perea and John Harer. "Sliding windows and persistence: An application of topological methods to signal analysis". In: *Foundations of Computational Mathematics* 15.3 (June 2015), pp. 799–838. ISSN: 1615-3383. DOI: 10.1007/s10208-014-9206-z.

[72] Jose Perea et al. "SW1PerS: Sliding windows and 1-persistence scoring; discovering periodicity in gene expression time series data". In: *BMC bioinformatics* 16 (Aug. 2015), p. 257. DOI: 10.1186/s12859-015-0645-6.

[73] Christopher Tralie and Jose Perea. "(Quasi) periodicity quantification in video data, using topology". In: *SIAM Journal on Imaging Sciences* 11 (Apr. 2017). DOI: 10.1137/17M1150736.

[74] Hitesh Gakhar and Jose A. Perea. "Sliding window persistence of quasiperiodic functions". In: *Journal of Applied and Computational Topology* (2023). DOI: 10.1007/s41468-023-00136-7.

[75] NOAA. *Climate at a glance: National time series*. National Centers for Environmental Information, retrieved 28th January 2023 from https://www.citedrive.com/overleaf.

[76] L.R. Rabiner and B.H. Juang. *Fundamentals of speech recognition*. Prentice-Hall Signal Processing Series: Advanced monographs. PTR Prentice Hall, 1993.

[77] Yi-Ying Kao et al. "Automatic detection of speech under cold using discriminative autoencoders and strength modeling with multiple sub-dictionary generation". In: *2018 16th International Workshop on Acoustic Signal Enhancement (IWAENC)*. 2018, pp. 416–420. DOI: 10.1109/IWAENC.2018.8521319.

[78] P. Warule, S.P. Mishra, and S. Deb. "Significance of voiced and unvoiced speech segments for the detection of common cold". In: *Signal, Image and Video Processing* 17 (2023), pp. 1785–1792.

[79] Cyrill G. Ott et al. "Processing of voiced and unvoiced acoustic stimuli in musicians". In: *Frontiers in Psychology* 2 (2011). DOI: 10.3389/fpsyg.2011.00195.

[80] John Bancroft. "Separation of voiced and unvoiced speech, and silence, using energy and periodicity". In: *The Journal of the Acoustical Society of America* 68.S1 (Aug. 2005), S70–S70. DOI: 10.1121/1.2004878.

[81] R.G. Bachu et al. "Voiced/unvoiced decision for speech signals based on zero-crossing rate and energy". In: *Advanced Techniques in Computing Sciences and Software Engineering*. Ed. by Khaled Elleithy. Dordrecht: Springer Netherlands, 2010, pp. 279–282.

[82] Zhaoting Liu et al. "Unvoiced and voiced speech segmentation based on the dimension of signal local linear manifold". In: *WSEAS Transactions on Signal Processing* 18 (2022), pp. 64–69. DOI: 10.37394/232014.2022.18.9.

[83] Huiqun Deng and Douglas O'Shaughnessy. "Voiced-unvoiced-silence speech sound classification based on unsupervised learning". In: *2007 IEEE International Conference on Multimedia and Expo*. 2007, pp. 176–179. DOI: 10.1109/ICME.2007.4284615.

[84] Juan Xu and Heming Zhao. "Speaker identification with whispered speech using unvoiced-consonant phonemes". In: *2012 International Conference on Image Analysis and Signal Processing*. 2012, pp. 1–4. DOI: 10.1109/IASP.2012.6425009.

[85] Michael McAuliffe et al. "Montreal Forced Aligner: Trainable text-speech alignment using Kaldi". In: *Proc. Interspeech 2017*, pp. 498–502. DOI: 10.21437/Interspeech.2017-1386.

[86] Holger Kantz and Thomas Schreiber. *Nonlinear time series analysis*. Vol. 7. Cambridge University Press, 2004.

[87] A. R. Bradlow. SpeechBox, retrieved from https://speechbox.linguistics.northwestern.edu.

[88] Keith Ito and Linda Johnson. *The LJ Speech dataset*. https://keithito.com/LJ-Speech-Dataset/. 2017.

[89] John S. Garofolo et al. *TIMIT acoustic-phonetic continuous speech corpus LDC93S1*. Web download. Philadelphia: Linguistic Data Consortium. 1993.

[90] Vassil Panayotov et al. LibriSpeech, retrieved from https://www.openslr.org/12.

[91] V. Marx. "Seeing data as t-SNE and UMAP do". In: *Nature Methods* 21 (2024), pp. 930–933.

[92] Paul Boersma and David Weenink. *Praat: Doing phonetics by computer*. Version 6.3.09, retrieved 2nd March 2023 from http://www.praat.org/. 2023.

[93] Christopher Tralie, Nathaniel Saul, and Rann Bar-On. "Ripser.py: A Lean persistent homology library for Python". In: *The Journal of Open Source Software* 3.29 (Sept. 2018), p. 925. DOI: 10.21105/joss.00925.

[94] Ulrich Bauer. "Ripser: Efficient computation of Vietoris-Rips persistence barcodes". In: *Journal of Applied and Computational Topology* 5.3 (2021), pp. 391–423. DOI: 10.1007/s41468-021-00071-5.

ACKNOWLEDGEMENTS

The authors would like to thank Meng Yu for his invaluable mentorship on audio and speech signal processing, on neural networks and deep learning, as well as for his comments and suggestions on an earlier draft of this manuscript. The authors would also like to thank Andrew Blumberg, Gunnar Carlsson, Fangyi Chen, Zeyang Ding, Haibao Duan, Houhong Fan, Fuquan Fang, Yunan He, Wenfei Jin, Fengchun Lei, Jingyan Li, Yanlin Li, Hongwei Lin, Andy Luchuan Liu, Tao Luo, Zhi Lü, Jianxin Pan, Yuhe Qin, Qi Sun, Jie Wu, Kelin Xia, Tony Xiaochen Xiao, Jiang Yang, Jin Zhang and Zhen Zhang for helpful discussions and encouragement. The authors are grateful to the editor and anonymous reviewers for making extensive, constructive, and often enlightening comments and suggestions, which helped to improve the paper. This work was partly supported by the National Natural Science Foundation of China grant 12371069 and the Guangdong Provincial Key Laboratory of Interdisciplinary Research and Application for Data Science.

AUTHOR INFORMATION

These authors contributed equally: Pingyao Feng, Qingrui Qu.

Authors and Affiliations

Department of Mathematics, Southern University of Science and Technology, Shenzhen, China
Pingyao Feng, Qingrui Qu, Siheng Yi, Zhiwang Yu, Haiyu Zhang, Yifei Zhu

Contributions

Y.Z. planned the project. P.F. and S.Y. constructed the theoretical framework. P.F. designed the sample, built the algorithms, and analysed the data. Q.Q., S.Y., Z.Y, and H.Z. assisted with the algorithms and data analysis. P.F., Y.Z., Q.Q., S.Y., H.Z., and Z.Y. wrote the paper and contributed to the discussion.

Corresponding author

Correspondence to Yifei Zhu.

Supplementary information

S.1 Phonetic data, aural perception, and learning topologically

As a research field of linguistics, phonetics studies the production as well as the classification of human speech sounds from the world's languages. In phonetics, a *phoneme* is the smallest basic unit of human speech sounds.[1] It is a short speech segment possessing distinct physical or perceptual properties. Phonemes are generally classified into two principal categories: vowels and consonants. A *vowel* is defined as a speech sound pronounced by an open vocal tract with no significant build-up of air pressure at any point above the glottis, and at least making some airflow escape through the mouth. In contrast, a *consonant* is a speech sound that is articulated with a complete or partial closure of the vocal tract and usually forces air through a narrow channel in one's mouth or nose.

Unlike vowels which must be pronounced by vibrated vocal cords, consonants can be further categorised into two classes according to whether the vocal cords vibrate or not during articulation. If the vocal cords vibrate, the consonant is known as a *voiced* consonant. Otherwise, the consonant is *voiceless*. Since vocal cord vibration can produce a stable periodic signal of air pressure, voiced consonants tend to have more periodic components than voiceless consonants, which can in turn be detected by PH as topological characteristics from phonetic time series data.

Indeed, one of the more heuristic motivations for our research project is to reexamine (and even revise) the linguistic classifications of phonemes through the mathematical lens of topological patterns and shape of speech data, analogous to Carlsson and his collaborators' seminal work [S1] on the distribution of image data (cf. Fig. S1).

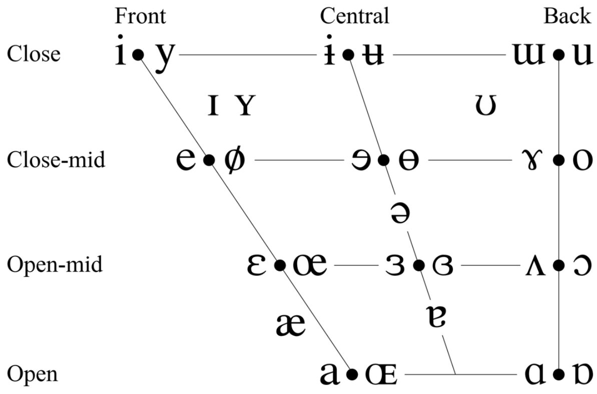

Fig. S1: A charted "distribution space" of vowels created by linguists [S2]. The vertical axis of the chart denotes vowel height. Vowels pronounced with the tongue lowered are located at the bottom and those raised are at the top. The horizontal axis of this chart denotes vowel backness. Vowels with the tongue moved towards the front of the mouth are in the left of the chart, while those with to the back are placed in the 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.

The transmission of sound to the human auditory system is a marvel of biological engineering, wherein acoustic waves are progressively transformed into neural signals. This process commences with the external ear channeling sound waves to the tympanic membrane, which subsequently induces vibrations in the ossicles of the middle ear—the malleus, incus, and stapes, constituting the smallest bones in the human body. These minute oscillations are then conveyed to one of the most critical structures in auditory perception: the cochlea.

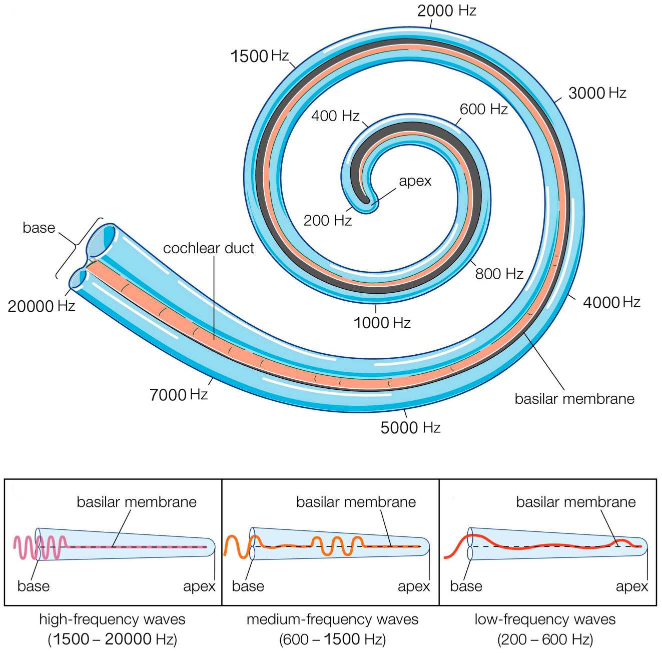

Fig. S2: Illustration depicting the distribution of frequencies along the basilar membrane of the cochlea, which functions as a natural Fourier analysis device, adapted from Encyclopædia Britannica [S3].

The cochlea, in essence, functions as a biological Fourier analysis apparatus (see Fig. S2). This spiral-shaped, fluid-filled organ amplifies the incoming sound waves and performs a spectral decomposition of complex acoustic signals. The cochlea's architecture is characterised by a gradual variation in the radius of its spiral and the mechanical properties of the basilar membrane that runs along its length. The basal end of the cochlea, with its rigid basilar membrane and narrow duct, is optimally tuned to high-frequency vibrations. In contrast, the apical region, featuring a more flexible membrane and wider duct, is more responsive to lower frequencies.

This structural gradient creates a tonotopic organisation within the cochlea, analogous to the varying tensions of musical strings producing different pitches. The basilar membrane's varying mechanical properties result in different regions having distinct resonant frequencies, each maximally sensitive to a specific range of sound frequencies. Atop this membrane reside the hair cells, specialised mechanoreceptors that transduce mechanical vibrations into electrical signals, thereby enabling auditory perception. The cochlea's spiral configuration, in conjunction with the basilar membrane's properties, constitutes a natural, passive mechani-

[1] In the main text and supplementary information, we reserve *phone* for a phoneme segmented from a recording of human speech.

cal Fourier analyser. This biological mechanism effectively distributes frequency components of sound waves along the length of the cochlea. Consequently, the neural signals generated by hair cells at different locations along the basilar membrane correspond to distinct frequency bands of the original acoustic input.

It is noteworthy that contemporary industrial approaches to speech signal processing, such as STFT and MFCC as in Sec. 2.3.2, employ analytical methods that parallel the cochlea's function. These techniques decompose signals into linear combinations of basis functions, mirroring the cochlea's spectral analysis. This convergence of biological design and signal processing methodology can be viewed as a triumph of biomimetic engineering.

Intriguingly, our experimental findings have demonstrated that topological principles can also be leveraged to extract certain acoustic information. This novel approach lacks a direct physiological counterpart in current auditory research and established theoretical frameworks. The potential for topological methods in auditory signal processing opens up an exciting new frontier for exploration, potentially bridging the gap between abstract mathematics and biological sensory systems. Future investigations in this domain may yield insights that could revolutionise our understanding of auditory perception and inspire innovative signal processing techniques (cf. [S4]).

S.2 Mathematical generalities of the TDE–PH approach to time series data

S.2.1 Time-delay embedding

Time-delay embedding (TDE) is also known as sliding window embedding, delay embedding, and delay coordinate embedding. For simplicity, we focus on 1-dimensional time series. TDE of a real-valued function $f\colon \mathbb{R} \to \mathbb{R}$, with parameters positive integer d and positive real number τ, is defined to be the vector-valued function

$$\begin{aligned} SW_{d,\tau} f\colon \mathbb{R} &\to \mathbb{R}^d \\ t &\mapsto \Big(f(t), f(t+\tau), \ldots, f\big(t+(d-1)\tau\big)\Big) \end{aligned}$$

Here, d is the *dimension* of the target space for the embedding, τ is the *delay*, and their product $d \cdot \tau$ is called the *window size*. According to the Manifold Hypothesis, a time series lies on a manifold. The method then reconstructs this topological space from the input time series, when d is at least twice the dimension of the latent manifold M. Given a trajectory $\gamma\colon \mathbb{R} \to M$ whose image is dense in M, the embedding property holds for the time series $f(t_n)$ (generically, in a technical sense we omit here) via an "observation" function $G\colon M \to \mathbb{R}$, i.e., $f(t_n) = G\big(\gamma(t_n)\big)$.

In [S5, Sec. 5], Perea and Harer established that the N-truncated Fourier series expansion

$$S_N f(t) = \sum_{n=0}^{N} a_k \cos(kt) + b_k \sin(kt)$$

of a periodic time series f can be reconstructed into a circle when $d \geq 2N$, i.e.,

$$SW_{d,\tau} f(\mathbb{R}) \cong \mathbb{S}^1$$

Moreover, let L be a constant such that

$$f\Big(t + \frac{2\pi}{L}\Big) = f(t)$$

Then the 1-dimensional MP of the resulting point cloud is the largest when the window size $d \cdot \tau$ is integrally proportional to $2\pi/L$, i.e.,

$$d \cdot \tau = m\frac{2\pi}{L}$$

for a positive integer m. Intuitively, an increase in the dimension of TDE results in a better approximation when truncating the Fourier series, and the MP of the point cloud becomes the most significant when the window size equals a period.

This methodology also proves particularly advantageous in scenarios where the system under investigation exhibits nonlinear dynamics, precluding straightforward analysis of the time series data. Via a suitable embedding, the inherent geometric configuration of the system emerges, enabling deeper comprehension and refined analysis.

S.2.2 Persistent homology

Topology is a subject area that studies the properties of geometric objects that remain unchanged under continuous transformations or smooth perturbations. It focuses on the intrinsic features of a space regardless of its rigid shape or size. Algebraic topology (AT) provides a quantitative description of these topological properties.

A simplicial complex (and its numerous variants and analogues) is a powerful tool in AT which enables us to represent a topological space using discrete data. Unlike the original space, which can be challenging to compute and analyse, a simplicial complex provides a combinatorial description that is much more amenable to computation. We can use algebraic techniques to study the properties of a simplicial complex, such as its homology and cohomology groups, which encode and reveal information about the topology of the underlying space.

Formally, a *simplicial complex* with *vertices* in a set V is a collection K of nonempty finite subsets $\sigma \subset V$ such that any nonempty subset τ of σ always implies $\tau \in K$ (called a *face* of σ) and that σ intersecting σ' implies their intersection $\sigma \cap \sigma' \in K$. A set $\sigma \in K$ with $(i+1)$ elements is called an *i-simplex* of the simplicial complex K. For instance, consider $\mathbb{S}^1 \vee \mathbb{S}^2$, a circle kissing a sphere at a single point, as a topology space. It can be approximated by the simplicial complex K with 6 vertices a, b, c, d, e, f. This simplicial complex can be enumerated as

$$\begin{aligned} K = \{&\{a\}, \{b\}, \{c\}, \{d\}, \{e\}, \{f\}, \\ &\{a,b\}, \{a,c\}, \{b,c\}, \{c,d\}, \{c,f\}, \{d,f\}, \{c,e\}, \\ &\{d,e\}, \{f,e\}, \\ &\{c,d,f\}, \{c,e,f\}, \{c,d,e\}, \{d,e,f\}\} \end{aligned}$$

which is a combinatorial avatar for $\mathbb{S}^1 \vee \mathbb{S}^2$ via a "triangulation" operation on the latter. See Fig. S3.

Given a simplicial complex K, let p be a prime number and $\mathbb{F}_p$ be the finite field with p elements. Define $C_i(K; \mathbb{F}_p)$ to be the $\mathbb{F}_p$-vector space with basis the set of i-simplices in K. To keep track of the order of vertices within a simplex,

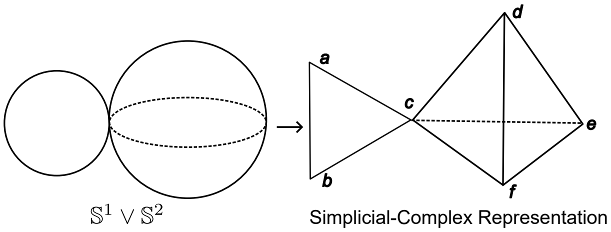
$\mathbb{S}^1 \vee \mathbb{S}^2$
a
b
c
d
e
f
Simplicial-Complex Representation

Fig. S3: From a topological space to its triangulation.

we use the alternative notation with square brackets in the following. If $\sigma = [v_0, v_1, \ldots, v_i]$ is an i-simplex, define the *boundary* of σ, denoted by $\partial\sigma$, to be the alternating sum of the $(i-1)$-dimensional faces of σ given by

$$\partial\sigma := \sum_{k=0}^{i} (-1)^k [v_0, \ldots, \hat{v}_k, \ldots, v_i]$$

where $[v_0, \ldots, \hat{v}_k, \ldots, v_i]$ is the k-th $(i-1)$-dimensional face of σ missing the vertex v_k. We can extend ∂ to $C_i(K; \mathbb{F}_p)$ as an $\mathbb{F}_p$-linear operator so that $\partial\colon C_i(K; \mathbb{F}_p) \to C_{i-1}(K; \mathbb{F}_p)$. The composition of boundary operators satisfies $\partial \circ \partial = 0$. The elements in $C_i(K; \mathbb{F}_p)$ with boundary 0 are called *i-cycles*. They form a subspace of $C_i(K; \mathbb{F}_p)$, denoted by $Z_i(K; \mathbb{F}_p)$. The elements in $C_i(K; \mathbb{F}_p)$ that are the images of elements of $C_{i+1}(K; \mathbb{F}_p)$ under ∂ are called *i-boundaries*. They form a subspace too, denoted by $B_i(K; \mathbb{F}_p)$. It follows from $\partial \circ \partial = 0$ that

$$B_i(K; \mathbb{F}_p) \subset Z_i(K; \mathbb{F}_p)$$

Then define the quotient space

$$H_i(K; \mathbb{F}_p) := Z_i(K; \mathbb{F}_p)/B_i(K; \mathbb{F}_p)$$

to be the *i-th homology group of K with $\mathbb{F}_p$-coefficients*. We call $\dim\big(H_i(K; \mathbb{F}_p)\big)$ the *i-th Betti number*, denoted by $\beta_i(K)$, which counts the number of i-dimensional holes in the corresponding topological space. As such, these homology groups are also called the homology groups of the space (it can be shown that they are independent of the particular ways in which the space is triangulated). For example, the Betti numbers of $\mathbb{S}^1 \vee \mathbb{S}^2$ from above are $\beta_1 = 1$, $\beta_2 = 1$, and $\beta_i = 0$ when $i \geqslant 3$.

The usefulness of these invariants, besides their computability (essentially Gaussian elimination in linear algebra), lies in their tractability along deformations. Given two simplicial complexes K and L, a simplicial map $f\colon K \to L$ (that preserves the simplicial structure) induces an $\mathbb{F}_p$-linear map $H_i(f; \mathbb{F}_p)\colon H_i(K; \mathbb{F}_p) \to H_i(L; \mathbb{F}_p)$. Thus, if two spaces are topologically equivalent (in fact, "homotopy equivalent" suffices), their homology groups must be isomorphic and the Betti numbers match up.

Let (X, d) be a finite point cloud with metric d. Define a family of simplicial complexes, called *Rips complexes*, by

$$R_\epsilon(X) := \{\sigma \subset X \mid d(x, x') \leq \epsilon \text{ for all } x, x' \in \sigma\}$$

The family

$$\mathcal{R}(X) := \{R_\epsilon(X)\}_{\epsilon \geq 0}$$

is known as the Rips filtration of X. Clearly, if $\epsilon_1 \leq \epsilon_2$, then $R_{\epsilon_1}(X) \hookrightarrow R_{\epsilon_2}(X)$. Thus, for each i we obtain a sequence

$$H_i\big(R_{\epsilon_0}(X); \mathbb{F}_p\big) \to H_i\big(R_{\epsilon_1}(X); \mathbb{F}_p\big) \to \cdots \\ \to H_i\big(R_{\epsilon_m}(X); \mathbb{F}_p\big)$$

where $0 = \epsilon_0 < \epsilon_1 < \cdots < \epsilon_m < \infty$. As ϵ varies, the topological features in the simplicial complexes $R_\epsilon(X)$ vary, resulting in the emergence and disappearance of holes (cf. Fig. 1d).

Given the values of ϵ, record the instances of emergence and disappearance of holes, which correspond to cycle classes in the homology groups along the above sequence. Each class has a descriptor $(b, d) \in \mathbb{R}^2$, where b represents the *birth time*, d represents the *death time*, and $b - d$ represents the *lifetime* of the holes. In this way, we obtain a multiset

$$\{(b_j, d_j)\}_{j \in J} =: \operatorname{dgm}_i\big(\mathcal{R}(X)\big)$$

which encodes the "persistence" of topological features of X. This multiset can be represented as a multiset of points in the 2-dimensional coordinate system called a *persistence diagram for the i-th PH* or as an array of interval segments called a *persistence barcode*. In particular, we use *maximal persistence* to refer to the maximal lifetime among all the points in a persistence diagram.

S.3 More specifics on parameter selection with TopCap

S.3.1 Skip, maximal persistence, and persistence execution time

Computation time assumes a critical role when processing a substantial volume of data. In this context, the parameter skip in TDE is considered, as it significantly influences the number of points within the point clouds, thereby directly impacting the number of simplices during persistent filtration and thus the computation time for PD. In this subsection, we demonstrate that an appropriate increment in the skip parameter can markedly reduce computation time. However, it is noteworthy that MP exhibits resilience to an increase in skip to a certain extent. Consequently, in this case, it is feasible to augment skip in TDE to expedite the computation of PD. For details on the complexity of computing persistent homology, the interested reader may refer to Zomorodian and Carlsson [S6, Sec. 4.3] as well as Edelsbrunner et al. [S7, Sec. 4].

Using an example of a sound record of the voiced consonant [m], we elucidate the relationship between skip, computation duration, and size of the resulting point clouds obtained via TDE in Fig. 7d. Computation duration is measured each time after restarting the Jupyter notebook, on Dell Precision 3581, with CPU Intel® Core™ i7-13800H of basic frequency 2.50 GHz and 14 cores. Computation time means the time for executing the code `ripser(Points,maxdim=1)`. As depicted in Fig. 7d, a substantial reduction in computation time is observed with an increase in the skip parameter. In contrast, our computation's output MP appears stable.

S.3.2 Multiple dependency of maximal persistence

As mentioned in the main text, there are three crucial parameters in TDE, namely, d, τ, and skip. In this subsection,

dimension = 10 desired delay = 40			dimension = 50 desired delay = 8			dimension = 100 desired delay = 4		
delay	skip	MP	delay	skip	MP	delay	skip	MP
1	1	0.0610	1	1	0.2834	1	1	0.4270
10	1	0.1299	3	1	0.3021	2	1	0.4337
20	1	0.1312	4	1	0.3054	2	5	0.4146
30	1	0.1281	5	1	0.3058	3	1	0.4357
39	1	0.1229	6	1	0.3042	3	5	0.4120
39	5	0.1134	7	1	0.3052	4	1	0.4381
40	1	0.1290	7	5	0.2886	4	5	0.4139
40	5	0.1195	8	1	0.3093	5	1	0.4375
41	1	0.1200	8	5	0.2928	5	5	0.4105
41	5	0.1153	9	1	0.3091	6	1	0.4347
45	1	0.0940	9	5	0.2913	6	5	0.4114
50	1	0.1226	10	1	0.3069	7	1	0.4380
60	1	0.1315	15	1	0.3070	8	1	0.4378
94	1	empty	18	1	empty	9	1	empty

Tab. S1: MP for choices of dimension, delay, and skip in TDE. The desired delay is computed by the algorithm in Sec. 3 of Methods. Empty in MP means the delay is too large to obtain point-cloud data.

we present a table that delineates the topological descriptor MP in relation to these from TopCap.

The experiment is executed on a record of the voiced consonant [ŋ], which comprises 887 sampled points as the length of this time series. Theoretically, given a periodic function, one obtains the optimal MP of the function in a fixed dimension under the condition that the TDE window size (i.e., the product of dimension and delay) equals a period (cf. Sec. S.2.1). However, the phonetic time series that we typically handle deviate far from being periodic. Despite our approach to calculating the period of time series by ACL functions, we cannot assure that the (theoretically derived) desired delay will indeed yield the optimal MP of a time series in general. Nevertheless, this desired delay usually gives relatively good MP. For instance, as illustrated in Tab. S1, when the dimension is 10, the desired delay is 40. This corresponds to an MP of 0.1290, which is marginally lower than the MP of 0.1315 achieved at a delay of 60. However, as the dimension rises, the point clouds from TDE become more regular. It becomes increasingly probable that at the desired delay, one can indeed obtain the optimal MP of the time series. For example, when the dimension is either 50 or 100, the MP of the time series is achieved at the desired delay. This provides additional justification for preferring higher dimensions: The table reveals that an augmentation in dimension may lead to a more substantial enhancement in the MP of a time series than simply tuning delay.

References for supplementary information

[S1] Gunnar Carlsson et al. "On the local behavior of spaces of natural images". In: *International Journal of Computer Vision* 76 (Jan. 2008), pp. 1–12. DOI: 10.1007/s11263-007-0056-x.

[S2] IPA Chart. *The international phonetic alphabet (revised to 2020).* International Phonetic Association, retrieved 16th January 2024 from https://www.internationalphoneticassociation.org/content/ipa-chart.

[S3] Encyclopædia Britannica. *Model showing the distribution of frequencies along the basilar membrane of the cochlea*. Online. Accessed: 2024-10-09, https://www.britannica.com/science/inner-ear#/media/1/288499/18100.

[S4] Yu Chen, Hongwei Lin, and Jiacong Yan. *The Gestalt computational model*. 2024. arXiv: 2405.20583 [cs.CG].

[S5] Jose A. Perea and John Harer. "Sliding windows and persistence: An application of topological methods to signal analysis". In: *Foundations of Computational Mathematics* 15.3 (June 2015), pp. 799–838. ISSN: 1615-3383. DOI: 10.1007/s10208-014-9206-z.

[S6] Afra Zomorodian and Gunnar Carlsson. "Computing persistent homology". In: *Discrete & Computational Geometry* 33.2 (Feb. 2005), pp. 249–274. ISSN: 1432-0444. DOI: 10.1007/s00454-004-1146-y.

[S7] Edelsbrunner, Letscher, and Zomorodian. "Topological persistence and simplification". In: *Discrete & Computational Geometry* 28.4 (Nov. 2002), pp. 511–533. ISSN: 1432-0444. DOI: 10.1007/s00454-002-2885-2.