Untangling in Invariant Speech Recognition

Cory Stephenson
Intel AI Lab
cory.stephenson@intel.com

Jenelle Feather
MIT
jfeather@mit.edu

Suchismita Padhy
Intel AI Lab
suchismita.padhy@intel.com

Oguz Elibol
Intel AI Lab
oguz.h.elibol@intel.com

Hanlin Tang
Intel AI Lab
hanlin.tang@intel.com

Josh McDermott
MIT/ Center for Brains, Minds, and Machines
jhm@mit.edu

SueYeon Chung
Columbia University/ MIT
sueyeon@mit.edu

Abstract

Encouraged by the success of deep neural networks on a variety of visual tasks, much theoretical and experimental work has been aimed at understanding and interpreting how vision networks operate. Meanwhile, deep neural networks have also achieved impressive performance in audio processing applications, both as sub-components of larger systems and as complete end-to-end systems by themselves. Despite their empirical successes, comparatively little is understood about how these audio models accomplish these tasks. In this work, we employ a recently developed statistical mechanical theory that connects geometric properties of network representations and the separability of classes to probe how information is untangled within neural networks trained to recognize speech. We observe that speaker-specific nuisance variations are discarded by the network's hierarchy, whereas task-relevant properties such as words and phonemes are untangled in later layers. Higher level concepts such as parts-of-speech and context dependence also emerge in the later layers of the network. Finally, we find that the deep representations carry out significant temporal untangling by efficiently extracting task-relevant features at each time step of the computation. Taken together, these findings shed light on how deep auditory models process time dependent input signals to achieve invariant speech recognition, and show how different concepts emerge through the layers of the network.

1 Introduction

Understanding invariant object recognition is one of the key challenges in cognitive neuroscience and artificial intelligence[1]. An accurate recognition system will predict the same class regardless of stimulus variations, such as the changes in viewing angle of an object or the differences in pronunciations of a spoken word. Although the class predicted by such a system is unchanged, the internal representations of individual objects within the class may differ. The set of representations corresponding to the same object class can then be thought of as an object manifold. In vision systems, it has been hypothesized that these "object manifolds", which are hopelessly entangled in the input, become "untangled" across the visual hierarchy, enabling the separation of different categories both in the brain [2] and in deep artificial neural networks [3]. Auditory recognition also requires the separation of highly variable inputs according to class, and could involve the untangling

33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, Canada.

of ‘auditory class manifolds’. In contrast to vision, auditory signals unfold over time, and the impact of this difference on underlying representations is poorly understood.

Speech recognition is a natural domain for analyzing auditory class manifolds not only with word and speaker classes, but also at the phonetic and semantic level. In recent years, hierarchical neural network models have achieved state of the art performance in automatic speech recognition (ASR) and speaker identification [4, 5]. Understanding how these end-to-end models represent language and speech information remains a major challenge and is an active area of research [6, 7]. Several studies on speech recognition systems have analyzed how phonetic information is encoded in acoustic models [8, 9, 10], and how it is embedded across layers by making use of classifiers [11, 12, 13, 6]. It has also been shown that deep neural networks trained on tasks such as speech and music recognition resemble human behavior and auditory cortex activity [14]. Ultimately, understanding speech-processing in deep networks may shed light on understanding how the brain processes auditory information.

Much of the prior work on characterizing how information is represented and processed in deep networks and the brain have focused on linear separability, representational similarity, and geometric measures. For instance, the representations of different objects in vision models and the macaque ventral stream become more linearly separable at deeper stages, as measured by applying a linear classifier to each intermediate layer [2, 3]. Representations have been compared across different networks, layers, and training epochs using Canonical Correlation Analysis (CCA) [15, 16, 17]. Representational similarity analysis (RSA), which evaluates the similarity of representations derived for different inputs, has been used to compare networks [18, 19]. Others have used explicit geometric measures to understand deep networks, such as curvature [3, 20], geodesics [21], and Gaussian mean width [22]. However, none of these measures make a concrete connection between the separability of object representations and their geometrical properties.

In this work, we make use of a recently developed theoretical framework[23, 24, 25] based on the replica method [26, 27, 28] that links the geometry of object manifolds to the capacity of a linear classifier as a measure of the amount of information stored about object categories per feature dimension. This method has been used in visual convolutional neural networks (CNNs) to characterize object-related information content across layers, and to relate it to the emergent representational geometry to understand how object manifolds are untangled across layers [25]. Here we apply manifold analyses[1] to auditory models for the first time, and show that neural network speech recognition systems also untangle speech objects relevant for the task. This untangling can also be an emergent property, meaning the model also learns to untangle some types of object manifolds without being trained to do so explicitly.

We present several key findings:

1. We find significant untangling of word manifolds in different model architectures trained on speech tasks. We also see emergent untangling of higher-level concepts such as words, phonemes, and parts of speech in an end-to-end ASR model (Deep Speech 2).
2. Both a CNN architecture and the end-to-end ASR model converge on remarkably similar behavior despite being trained for different tasks and built with different computational blocks. They both learn to discard nuisance acoustic variations, and exhibit untangling for task relevant information.
3. Temporal dynamics in recurrent layers reveal untangling over recurrent time steps, in the form of smaller manifold radius, lower manifold dimensionality.

In addition, we show the generality of auditory untangling with speaker manifolds in a network trained on a speaker recognition task, that are not evident in either the end-to-end ASR model or the model trained explicitly to recognize words. These results provide the first geometric evidence for untangling of manifolds, from phonemes to parts-of-speech, in deep neural networks for speech recognition.

2 Methods

To understand representation in speech models, we first train a neural network on a corpus of transcribed speech. Then, we use the trained models to extract per-layer representations at every time

[1] Our implementation of the analysis methods: https://github.com/schung039/neural_manifolds_replicaMFT

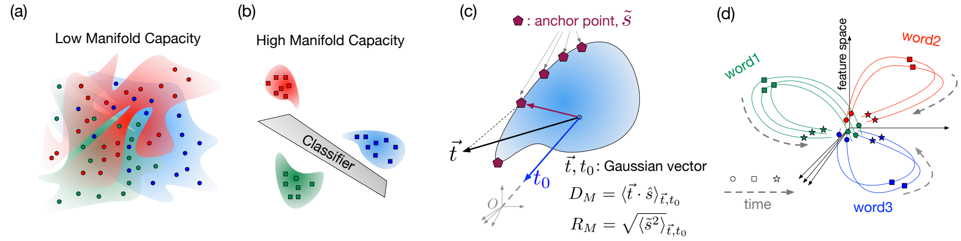
(a)
Low Manifold Capacity
(b)
High Manifold Capacity
Classifier
(c)
⬟ : anchor point, $\tilde{s}$
$\vec{t}$
t_0
O
$\vec{t}, t_0$: Gaussian vector
$D_M = \langle \vec{t} \cdot \hat{s} \rangle_{\vec{t}, t_0}$
$R_M = \sqrt{\langle \tilde{s}^2 \rangle}_{\vec{t}, t_0}$
(d)
word1
feature space
word2
○ □ ☆
time
word3

Figure 1: **Illustration of word manifolds.** (a) highly tangled manifolds, in low capacity regime (b) untangled manifolds, in high capacity regime (c) Manifold Dimension captures the projection of a Gaussian vector onto the direction of an anchor point, and Manifold Radius captures the norm of an anchor point in manifold subspace. (d) Illustration of untanglement of words over time.

step on each corpus stimulus. Finally, we apply the mean-field theoretic manifold analysis technique [24, 25] (hereafter, MFTMA technique) to measure manifold capacity and other manifold geometric properties (radius, dimension, correlation) on a subsample of the test dataset.

Formally, if we have P objects (e.g. words), we construct a dataset $\mathcal{D}$ with pairs (x_i, y_i), where x_i is the auditory input, and $y_i \in \{1, 2, ..., P\}$ is the object class. Given a neural network $N(x)$, we extract $N_t^l(x)$, which is the output of the network at time t in layer l, for all inputs x whose corresponding label is p, for each $p \in \{1, 2, ..., P\}$. The object manifold at layer l for class p is then defined as the point cloud of activations obtained from the different examples x_i of the p^{th} class. We then apply the MFTMA technique to this set of activations to compute the manifold capacity, manifold dimension, radius, and correlations for that manifold.

The manifold capacity obtained by the MFTMA technique captures the linear separability of object manifolds. Furthermore, as shown in [24, 25] and outlined in SM Sect. 1.2, the mean-field theory calculation of manifold capacity also gives a concrete connection between the measure of separability and the size and dimensionality of the manifolds. This analysis therefore gives additional insight into both the separability of object manifolds, and how this separability is achieved geometrically.

We measure these properties under different manifold types, including categories such as phonemes and words, or linguistic feature categories such as part-of-speech tags. This allows us to quantify the amount of invariant object information and the characteristics of the emergent geometry in the representations learned by the speech models.

2.1 Object Manifold Capacity and the mean-field theoretic manifold analysis (MFTMA)

In a system where P object manifolds are represented by N features, the 'load' in the system is defined by $\alpha = P/N$. When α is small, i.e. a small number of object manifolds are embedded in a high dimensional feature space, it's easy to find a separating hyperplane for a random dichotomy[2] of the manifolds. When α is large, too many categories are squeezed in a low dimensional feature space, rendering the manifolds highly inseparable. *Manifold capacity* refers to the critical load, $\alpha_C = P/N$, defined by the critical number of object manifolds, P, that can be linearly separated given N features. Above α_C, most dichotomies are inseparable, and below α_C, most are separable[24, 25]. This framework generalizes the notion of the perceptron storage capacity [26] from points to manifolds, re-defining the unit of counting to be object manifolds rather than individual points. The manifold capacity thus serves as a measure of the linearly decodable information about object identity per feature, and it can be measured from data in two ways:

1. **Empirical Manifold Capacity,** α_{SIM}: the manifold capacity can be measured empirically with a bisection search to find the critical number of features N such that the fraction of linearly separable random dichotomies is close to $1/2$.

2. **Mean Field Theoretic Manifold Capacity,** α_{MFT}: can be estimated using the replica mean field formalism with the framework introduced by [24, 25]. α_{MFT} is estimated from

[2] Here, we define a random dichotomy as an assignment of random ±1 labels to each manifold

the statistics of *anchor points* (shown in Fig. 1(c)), $\tilde{s}$, a representative point for a linear classification[3].

The manifold capacity for point-cloud manifolds is lower bounded by the case where there is no manifold structure. This lower bound is given by [29, 24]. In this work, we show α_{MFT}/α_{LB} for a comparison between datasets of different sizes and therefore different lower bounds.

Manifold capacity is closely related to the underlying geometric properties of the object manifolds. Recent work demonstrates that the manifold classification capacity can be predicted by an object manifold's *Manifold Dimension*, D_M, *Manifold Radius*, R_M, and the correlations between the centroids of the manifolds [23, 24, 25]. These geometrical properties capture the statistical properties of the anchor points, the representative support vectors of each manifold relevant for the linear classification, which change as the choice of other manifolds vary [24]. The MFTMA technique also measures these quantities, along with the manifold capacity:

Manifold Dimension, D_M: D_M captures the dimensions realized by the anchor point from the guiding Gaussian vectors shown in Fig. 1(c), and estimates the average embedding dimension of the manifold contributing to the classification. This is upper bounded by $\min(M, N)$, where M is the number of points per each manifold, and N, the feature dimension. In this work, $M < N$, and we present D_M/M for fair comparison between different datasets.

Manifold Radius, R_M: R_M is the average distance between the manifold center and the anchor points as shown in Fig. 1(c). Note that R_M is the size relative to the norm of the manifold center, reflecting the fact that the relative scale of the manifold compared to the overall distribution is what matters for linear separability, rather than the absolute scale.

Center Correlations, ρ_{center}: ρ_{center} measures how correlated the locations of these object manifolds are, and is calculated as the average of pairwise correlations between manifold centers. [25].

It has been suggested that the capacity is inversely correlated with D_M, R_M, and center correlation [24, 25]. Details for computing anchor points, $\tilde{s}$ can be found in the description of the mean-field theoretic manifold capacity algorithm, and the summary of the method is provided in the SM.

In addition to mean-field theoretic manifold properties, we measure the dimensionality of the data across different categories with two popular measures for dimensionality of complex datasets:

Participation Ratio D_{PR}: D_{PR} is defined as $\frac{(\sum_i \lambda_i)^2}{\sum_i \lambda_i^2}$, where λ_i is the i^{th} eigenvalue of the covariance of the data, measuring how many dimensions of the eigen-spectrum are active [17].

Explained Variance Dimension D_{EV}: D_{EV} is defined as the number of principal components required to explain a fixed percentage (90% in this paper) of the total variance [30].

Note that in all of the dimensional measures, i.e. D_M, D_{PR} and D_{EV} are upper bounded by the number of eigenvalues, which is $\min(\#samples, \#features)$ (these values are used in (Fig. 2-6)).

2.2 Models and datasets

We examined two speech recognition models. The first model is a CNN model based on [14], with a small modification to use batch normalization layers instead of local response normalization (full architecture can be found in Table SM1). We trained the model on two tasks: word recognition and speaker recognition. For word recognition, we trained on two second segments from a combination of the WSJ Corpus [31] and Spoken Wikipedia Corpora [32], with noise augmentation from AudioSet backgrounds [33]. For more training details, please see the SM.

The second is an end-to-end ASR model, Deep Speech 2 (DS2) [5], based on an open source implementation[4]. DS2 is trained to produce accurate character-level output with the Connectionist Temporal Classification (CTC) loss function [34]. The full architecture can be found in Table SM2. Our model was trained on the 960 hour training portion of the LibriSpeech dataset [35], achieving a word error rate (WER) of 12%, and 22.7% respectively on the clean and other partitions of the test

[3] See SM for exact relationship between $\tilde{s}$ and capacity, the outline of the code, and a demonstration that MFT manifold capacity matches the empirical capacity (given in Fig. SM1)

[4] https://github.com/SeanNaren/deepspeech.pytorch

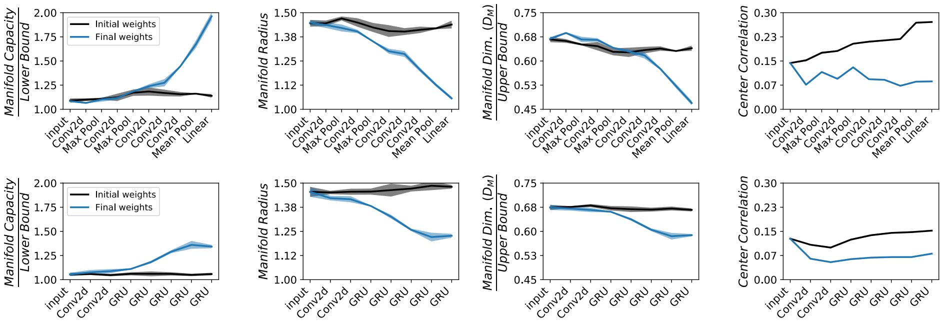

Figure 2: **Word manifold capacity emerges in both the CNN word classification model and the end to end ASR model (DS2).** Top: As expected, CNN model trained with explicit word supervision (blue lines) exhibits strong capacity in later layers, compared to the initial weights (black lines). This increase is due to reduced radius and dimension, as well as decorrelation. Bottom: A similar trend emerges in DS2 without training with explicit word supervision. In both, capacity is normalized against the theoretical lower bound (See Methods). The shaded area represents 95% confidence interval hereafter.

set without the use of a language model. The model trained on LibriSpeech also performs reasonably well on the TIMIT dataset, with a WER of 29.9% without using a language model.

We followed the procedure described in Sec. 2 to the construct manifold datasets for several different types of object categories using holdout data not seen by the models during training. The datasets used in the analysis of each model were as follows:

CNN manifolds datasets: Word manifolds from the CNN dataset were measured using data from the WSJ corpus. Each of the $P = 50$ word manifolds consist of $M = 50$ speakers saying the word, and each of the $P = 50$ speaker manifolds consist of one speaker saying $M = 50$ different words.

DS2 manifolds datasets:[5] Word and speaker manifolds were taken from the test portion of the LibriSpeech dataset. For word manifolds, $P = 50$ words with $M = 20$ examples each were selected, ensuring each example came from a different speaker. For speaker manifolds $P = 50$ speakers were selected with $M = 20$ utterances per speaker.

For the comparison between character, phoneme, word, and parts-of-speech manifolds, similar manifold datasets were also constructed from TIMIT, which includes phoneme and word alignment. $P = 50$ and $M = 50$ were used for character and phoneme manifolds, but owing to the smaller size of TIMIT, $P = 23$ and $M = 20$ were used for word manifolds. Likewise, we used a set of $P = 22$ tags, with $M = 50$ for the parts-of speech manifolds.

Feature Extraction For each layer of the CNN and DS2 models, the activations were measured for each exemplar and 5000 random projections with unit length were computed on which to measure the geometric properties. For temporal analysis in the recurrent DS2 model, full features were extracted for each time step.

3 Results

3.1 Untangling of words

We first investigated the CNN model, which was trained to identify words from a fixed vocabulary using the dataset described in 2.2. Since this model had explicit word level supervision, we observed that the word classes were more separable (higher capacity) over the course of the network layers (Figure 2 (Top) as expected. This word manifold capacity was not observed with the initial weights of the model (Figure 2, black lines). As a negative control, we trained the same CNN architecture to identify speakers instead, and did not observe increased word manifold capacity after training

[5] See Sect. SM3.2 for more details on the construction and composition of the datasets used for experiments on this model.

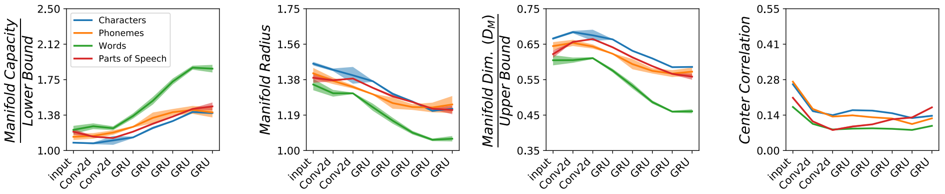

Figure 3: **Character, phoneme, word, and part-of-speech manifolds emerge in the DS2 model.**

(See Fig. SM3). These results demonstrate that word separability is task dependent and not a trivial consequence of the input, acoustic features, or model architecture. Furthermore, the MFTMA metrics reveal that this increased word capacity in later layers is due to both a reduction in the manifold radius and the manifold dimension.

Most end-to-end ASR systems are not trained to explicitly classify words in the input, owing to the difficulty in collecting and annotating large datasets with word level alignment, and the large vocabulary size of natural speech. Instead, models such as Deep Speech 2 are trained to output character-level sequences. Despite not being trained to explicitly classify words, the untangling of words was also emergent on the LibriSpeech dataset (Figure 2, bottom). In both CNN and DS2 models, we present the capacity values normalized by the lower bound and manifold dimension is normalized by the upper bound (Sec. 2.1), for easy comparison.

Surprisingly, across the CNN and recurrent DS2 architectures, the trend in the other geometric metrics were similar. The manifold capacities improve in downstream layers, and the reduction in manifold dimension and radius similarly occurs in downstream layers. Interestingly, word manifolds increases dramatically in the last layer of CNN, but only modestly in the last layers of DS2, perhaps owing to the fact that CNN model here is explicitly trained on word labels, while in the DS2, the word manifolds are emergent properties. Notably, the random weights of the initial model increase correlation across the layers in both networks, but the training significantly decreases center correlation in both models. More analysis on training epochs are given in Section 3.4 and in Sect. SM4.5.

3.2 Untangling of other speech objects

In addition to words, it is possible that auditory objects at different levels of abstraction are also untangled in the ASR task. To investigate this, we look for evidence of untangling at four levels of abstraction: characters, phonemes, words (each word class contains multiple phonemes and multiple characters), and part of speech tags (each part of speech class contains multiple words). These experiments were done on the end-to-end ASR model (DS2). Results for these four object types are shown in Fig. 3.

On the surface, we see all four of these quantities becoming untangled to some degree as the inputs are propagated through the network. However, in the final layers of the network, the untangling of words is far more prominent than that of phonemes (see the numerical values of capacity in Fig. 3). This suggests that the speech models may need to abstract over the phonetic information and character information (i.e. silent letters in words). While the higher capacity is due to a lower manifold radius and dimension, we note that the manifold dimension is significantly lower for words than it is for phonemes and parts of speech. The phoneme capacity starts to increase only after the second convolution layer, consistent with prior findings [6].

3.3 Loss of speaker information

In addition to learning to increase the separability of information relevant to the ASR task, robust ASR models should also learn to ignore unimportant differences in their inputs. For example, different instances of the same word spoken by different speakers must be recognized as the same word in spite of the variations in pronunciation, rate of speech, etc. If this is true, we expect that the separability of different utterances from the same speaker is decreased by the network, or at least not increased, as this information is not relevant to the ASR task.

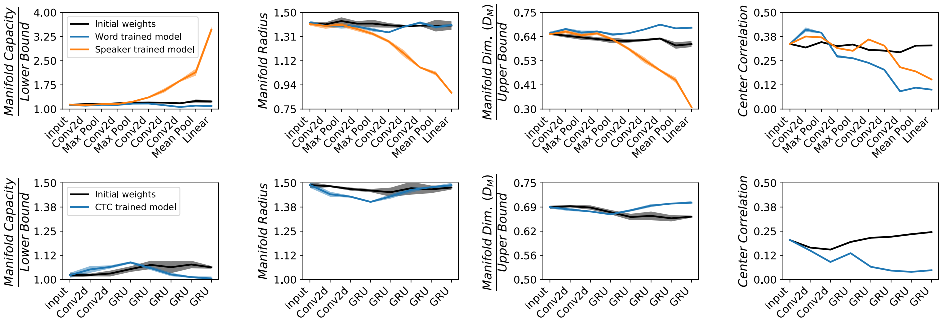

Figure 4: **Speaker Manifolds disappear in different speech models.** Top: CNN ((black) before, and initial weights, trained on (blue) words. and (orange) speakers); Bottom: DS2 ((black) initial weights, (blue) trained on ASR task).

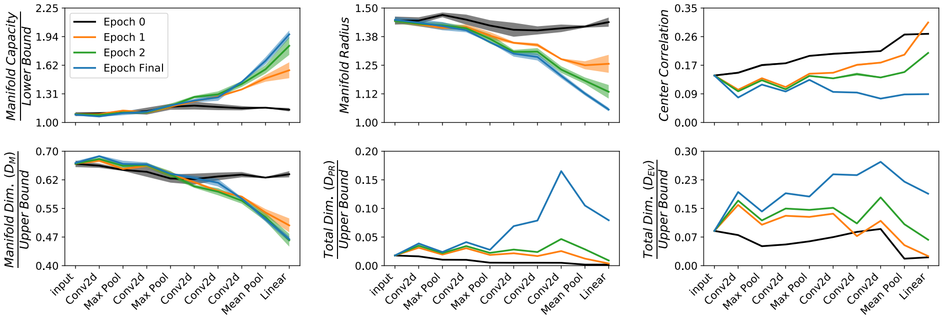

Figure 5: **Evolution of word manifolds via epochs of training, CNN.** Manifold capacity improves over epochs of training, while manifold dimension, radius, correlations decrease over training. Total data dimension (D_{PR}, D_{EV}) is inversely correlated with center correlation and increases over training epochs. Similar trends are observed in the DS2 model (SM).

Indeed, Figure 4 (Top) shows that for the CNN model trained to classify words, the separability of speaker manifolds defined by the dataset described in 2.2 decreases deeper in the network, and is even lower than in the untrained network. In contrast, when training a CNN explicitly to do speaker recognition the separability of speaker manifolds increases in later layers (while the separability of word manifolds does not, see Sect. SM4.2), demonstrating that the lack of speaker separation and the presence of word information is due to the specific task being optimized.

A similar trend also appears in the DS2 model, as shown in Fig. 4 (Bottom). In both the CNN trained to recognize words and DS2 models, speaker manifolds become more tangled after training, and in both cases we see that this happens due to an increase in the dimensionality of the speaker manifolds, as the manifold radius remains unchanged after training, and the center correlation decreases. In some sense, this mirrors the results in Sec. 3.1 and Sec. 3.2 where the model untangles word level information by decreasing the manifold dimension, and here discards information by increasing the manifold dimension instead.

For the CNN model, the decrease in speaker separability occurs mostly uniformly across the layers, where in DS2, the separability only drops in the last half of the network in the recurrent layers. Surprisingly, the early convolution layers of the DS2 model show either unchanged or slightly increased separability of speaker manifolds. We note that the decrease in speaker separability coincides with both the increase in total dimension as measured by participation ratio and explained variance, as seen in Fig. 5 and Fig. SM6, as well as a decrease in center correlation.

3.4 Trends over training epochs

In addition to evaluating the MFTMA quantities and effective dimension on fully trained networks, the analysis can also be done as the training progresses. Figure 5 shows the early stages of training

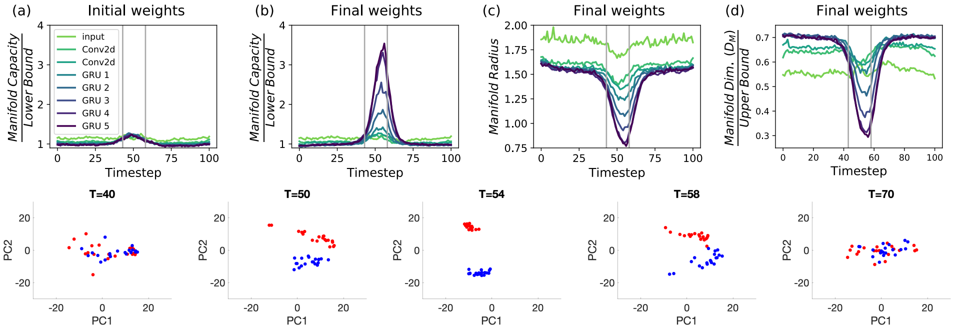

Figure 6: **Untangling of word manifolds in input timesteps.** (Top) Evolution of Librispeech word manifolds in timesteps, DS2 model (hypothesized in Fig 1). (a) Epoch 0 model, capacity (b-d) fully trained model, (b) capacity, (c) manifold radius, (d) manifold dimension. Vertical lines show the average word boundaries. (Bottom) Untanglement of two words over timesteps (T=40 to 70) in GRU 5 layer of DS2, projected to 2 PCs.

of the word recognition CNN model (See Fig. SM6 for the early stages of training for DS2, which shows similar trends). The capacity, manifold dimension, manifold radius, and center correlations quickly converge to those measured on the final epochs. Interestingly, the total data dimension (measured by D_{PR}, and D_{EV}) increases with training epochs unlike the manifold dimension, D_M, which decreases over training in both models. Intuitively, the training procedure tries to embed different categories in different directions while compressing them, resulting in a lower D_M and a lower center correlation. The increase in total D_{PR} could be related to lowered manifold center correlation. The total dimension D_{PR} could play the role of the larger 'effective' ambient dimension, in turn improving linear separability [36].

3.5 Untanglement of words over time

The above experiments were performed without considering the sequential nature of the inputs. Here, we compute these measures of untangling on each time step separately. This approach can interrogate the role of time in the computation, especially in recurrent models processing arbitrary length inputs.

Figure 6 shows the behavior of capacity, manifold radius, and manifold dimension over the different time steps in the recurrent layers of the end-to-end ASR model (DS2) for the word inputs used in Sec. 3.1. As is perhaps expected, the separability is at the theoretical lower bound for times far away from the word of interest, and peaks near the location of the word. This behavior arises due to the decrease in radius and dimension. However, the peak does not occur at the center of the word, owing to the arbitrary time alignment in the CTC cost function, as noted in [37]. Do the inputs far in time from the word window play a significant role in the untangling of words? While we omit further investigation of this here due to space constraints, an experiment on varying length inputs can be found in the SM.

Interestingly, the capacity measured at each input time step has a peak relative capacity of 3.6 (Fig. 6), much larger than the capacity measured from a random projection across all features, at 1.4 (Fig. 2). This is despite the lower feature dimension in the time step analysis (due to considering the features at only one time step, rather than the full activations). This implies that sequential processing also massages the representation in a meaningful way, such that a snapshot at a peak time step has a well separated, compressed representation, captured by the small value of D_M and R_M. Analagous to 1(d), the efficiency of temporal separation is illustrated in Fig. 6, bottom.

4 Conclusion

In this paper we studied the emergent geometric properties of speech objects and their linear separability, measured by manifold capacity. Across different networks and datasets, we find that linear separability of auditory class objects improves across the deep network layers, consistent with the untangling hypothesis in vision [2]. Word manifold's capacity arises across the deep layers, due to emergent geometric properties, reducing manifold dimension, radius and center correlations. Characterization of manifolds across training epochs suggests that word untangling is a result of training, as random weights do not untangle word information in the CNN or DS2. As representations in ASR systems evolve on the timescale of input sequences, we find that separation between different words emerges temporally, by measuring capacity and geometric properties at every frame.

Speech data naturally has auditory objects of different scales embedded in the same sequence of sound and we observed here an emergence and untangling of other speech objects such as phonemes, and part-of-speech manifolds. Interestingly, speaker information (measured by speaker manifolds) dissipates across layers. This effect is due to the network being trained for word classification, since CNNs trained with speaker classification are observed to untangle speaker manifolds. Interestingly, the transfer between speaker ID and word recognition was not very good, and a network trained for both speaker ID and words showed emergence of both manifolds, but these tasks were not synergistic (see Fig. SM4). These results suggest that the task transfer performance is closely related to the task structure, and can be captured by representation geometry.

Our methodology and results suggest many interesting future directions. Among other things, we hope that our work will motivate: (1) the theory-driven geometric analysis of representation untangling in tasks with temporal structure; (2) the search for the mechanistic relation between the network architecture, learned parameters, and structure of the stimuli via the lens of geometry; (3) the future study of competing vs. synergistic tasks enabled by the powerful geometric analysis tool.

Acknowledgments

We thank Yonatan Belinkov, Haim Sompolinsky, Larry Abbott, Tyler Lee, Anthony Ndirango, Gokce Keskin, and Ting Gong for helpful discussions. This work was funded by NSF grant BCS-1634050 to J.H.M. and a DOE CSGF Fellowship to J.J.F. S.C acknowledges support by Intel Corporate Research Grant, NSF NeuroNex Award DBI-1707398, and The Gatsby Charitable Foundation.

References

[1] Tatyana O Sharpee, Craig A Atencio, and Christoph E Schreiner. Hierarchical representations in the auditory cortex. *Current opinion in neurobiology*, 21(5):761–767, 2011.

[2] James J DiCarlo and David D Cox. Untangling invariant object recognition. *Trends in cognitive sciences*, 11(8):333–341, 2007.

[3] Ben Poole, Subhaneil Lahiri, Maithra Raghu, Jascha Sohl-Dickstein, and Surya Ganguli. Exponential expressivity in deep neural networks through transient chaos. In *Advances in neural information processing systems*, pages 3360–3368, 2016.

[4] Chao Li, Xiaokong Ma, Bing Jiang, Xiangang Li, Xuewei Zhang, Xiao Liu, Ying Cao, Ajay Kannan, and Zhenyao Zhu. Deep speaker: an end-to-end neural speaker embedding system. *arXiv preprint arXiv:1705.02304*, 2017.

[5] Dario Amodei, Sundaram Ananthanarayanan, Rishita Anubhai, Jingliang Bai, Eric Battenberg, Carl Case, Jared Casper, Bryan Catanzaro, Qiang Cheng, Guoliang Chen, et al. Deep speech 2: End-to-end speech recognition in english and mandarin. In *International conference on machine learning*, pages 173–182, 2016.

[6] Yonatan Belinkov and James Glass. Analyzing hidden representations in end-to-end automatic speech recognition systems. In *Advances in Neural Information Processing Systems*, pages 2441–2451, 2017.

[7] Yonatan Belinkov. *On internal language representations in deep learning: An analysis of machine translation and speech recognition*. PhD thesis, Massachusetts Institute of Technology, 2018.

[8] Tasha Nagamine, Michael L Seltzer, and Nima Mesgarani. Exploring how deep neural networks form phonemic categories. In *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[9] Tasha Nagamine, Michael L Seltzer, and Nima Mesgarani. On the role of nonlinear transformations in deep neural network acoustic models. In *Interspeech*, pages 803–807, 2016.

[10] Yu-Hsuan Wang, Cheng-Tao Chung, and Hung-yi Lee. Gate activation signal analysis for gated recurrent neural networks and its correlation with phoneme boundaries. *Interspeech*, 2017.

[11] Shuai Wang, Yanmin Qian, and Kai Yu. What does the speaker embedding encode? In *Interspeech*, pages 1497–1501, 2017.

[12] Zied Elloumi, Laurent Besacier, Olivier Galibert, and Benjamin Lecouteux. Analyzing learned representations of a deep asr performance prediction model. *arXiv preprint arXiv:1808.08573*, 2018.

[13] Andreas Krug, René Knaebel, and Sebastian Stober. Neuron activation profiles for interpreting convolutional speech recognition models. 2018.

[14] Alexander JE Kell, Daniel LK Yamins, Erica N Shook, Sam V Norman-Haignere, and Josh H McDermott. A task-optimized neural network replicates human auditory behavior, predicts brain responses, and reveals a cortical processing hierarchy. *Neuron*, 98(3):630–644, 2018.

[15] Maithra Raghu, Justin Gilmer, Jason Yosinski, and Jascha Sohl-Dickstein. Svcca: Singular vector canonical correlation analysis for deep learning dynamics and interpretability. In *Advances in Neural Information Processing Systems*, pages 6076–6085, 2017.

[16] Yixuan Li, Jason Yosinski, Jeff Clune, Hod Lipson, and John E Hopcroft. Convergent learning: Do different neural networks learn the same representations? In *International Conference on Learning Representations*, 2016.

[17] Peiran Gao, Eric Trautmann, M Yu Byron, Gopal Santhanam, Stephen Ryu, Krishna Shenoy, and Surya Ganguli. A theory of multineuronal dimensionality, dynamics and measurement. *bioRxiv*, page 214262, 2017.

[18] Seyed-Mahdi Khaligh-Razavi and Nikolaus Kriegeskorte. Deep supervised, but not unsupervised, models may explain it cortical representation. *PLoS computational biology*, 10(11):e1003915, 2014.

[19] David GT Barrett, Ari S Morcos, and Jakob H Macke. Analyzing biological and artificial neural networks: challenges with opportunities for synergy? *Current opinion in neurobiology*, 55:55–64, 2019.

[20] Olivier J Hénaff, Robbe LT Goris, and Eero P Simoncelli. Perceptual straightening of natural videos. *Nature neuroscience*, page 1, 2019.

[21] Olivier J Hénaff and Eero P Simoncelli. Geodesics of learned representations. 2016.

[22] Raja Giryes, Guillermo Sapiro, and Alexander M Bronstein. Deep neural networks with random gaussian weights: A universal classification strategy? *IEEE Trans. Signal Processing*, 64(13):3444–3457, 2016.

[23] SueYeon Chung, Daniel D Lee, and Haim Sompolinsky. Linear readout of object manifolds. *Physical Review E*, 93(6):060301, 2016.

[24] SueYeon Chung, Daniel D Lee, and Haim Sompolinsky. Classification and geometry of general perceptual manifolds. *Physical Review X*, 8(3):031003, 2018.

[25] Uri Cohen, SueYeon Chung, Daniel D. Lee Lee, and Haim Sompolinsky. Separability and geometry of object manifolds in deep neural networks. *bioRxiv*, page 10.1101/644658, 2019.

[26] Elizabeth Gardner. The space of interactions in neural network models. *Journal of physics A: Mathematical and general*, 21(1):257, 1988.

[27] HS Seung, Haim Sompolinsky, and Naftali Tishby. Statistical mechanics of learning from examples. *Physical review A*, 45(8):6056, 1992.

[28] Madhu Advani, Subhaneil Lahiri, and Surya Ganguli. Statistical mechanics of complex neural systems and high dimensional data. *Journal of Statistical Mechanics: Theory and Experiment*, 2013(03):P03014, 2013.

[29] Thomas M Cover. Geometrical and statistical properties of systems of linear inequalities with applications in pattern recognition. *IEEE transactions on electronic computers*, (3):326–334, 1965.

[30] Peiran Gao and Surya Ganguli. On simplicity and complexity in the brave new world of large-scale neuroscience. *Current opinion in neurobiology*, 32:148–155, 2015.

[31] Douglas B Paul and Janet M Baker. The design for the wall street journal-based csr corpus. In *Proceedings of the workshop on Speech and Natural Language*, pages 357–362. Association for Computational Linguistics, 1992.

[32] Arne Köhn, Florian Stegen, and Timo Baumann. Mining the spoken wikipedia for speech data and beyond. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Marko Grobelnik, Bente Maegaard, Joseph Mariani, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may 2016. European Language Resources Association (ELRA).

[33] Jort F. Gemmeke, Daniel P. W. Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R. Channing Moore, Manoj Plakal, and Marvin Ritter. Audio set: An ontology and human-labeled dataset for audio events. In *Proc. IEEE ICASSP 2017*, New Orleans, LA, 2017.

[34] Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd international conference on Machine learning*, pages 369–376. ACM, 2006.

[35] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. Librispeech: an asr corpus based on public domain audio books. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5206–5210. IEEE, 2015.

[36] Ashok Litwin-Kumar, Kameron Decker Harris, Richard Axel, Haim Sompolinsky, and LF Abbott. Optimal degrees of synaptic connectivity. *Neuron*, 93(5):1153–1164, 2017.

[37] Haşim Sak, Félix de Chaumont Quitry, Tara Sainath, Kanishka Rao, et al. Acoustic modelling with cd-ctc-smbr lstm rnns. In *2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, pages 604–609. IEEE, 2015.

Untangling in Invariant Speech Recognition - Supplemental Material

1 Details on measuring empirical and theoretical manifold capacity

1.1 Empirical Manifold Capacity

Here we provide a detailed description for empirically finding a manifold capacity, for a given number of object class manifolds , P, by finding a critical number of feature dimensions, N_c, such that the fraction of separable dichotomies of random assignment of +/- labels to a given manifolds is at 0.5 on average (Fig. SM1). If the feature dimension N is larger than critical N_c, the fraction of separable dichotomies will be close to 1 (hence, the system is in a linearly separable regime, Fig. SM1). If the feature dimension N is smaller than critical N_c, the fraction of separable dichotomies will be close to 0 (the system is in a linearly in-separable regime, Fig. SM1). The algorithm finds N_c by doing a bisection search on N, such that "fraction of linearly separable dichotomies" for N_c is 0.5, midpoint between 1 (separable) and 0 (inseparable) on average. At the cricical N_c, the capacity is defined to be P/N_c. In our experiments shown in Fig. SM1, we used randomly sampled 101 dichotomies, to compute fraction of linear separability.

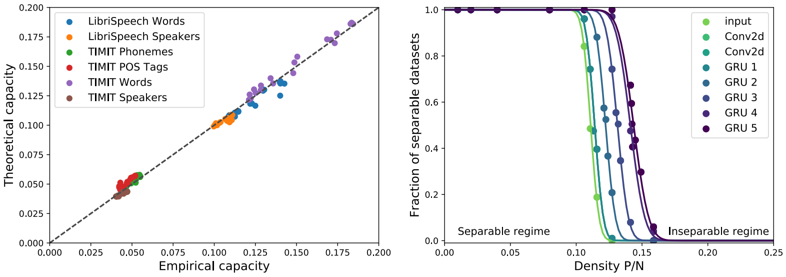

Figure SM1: **Measured capacity matches theoretical prediction.** (Left) Across multiple datasets (TIMIT, Librispeech) and manifolds (words, speakers, phonemes, part-of-speech tags) for the DS2 model, the measured capacity closely matches the theoretical capacity. Dotted line indicates unity. (Right) Empirical capacity is measured by a bisection search for critical N s.t. fraction of separable datasets cross 0.5

1.2 Mean-Field Theoretic (MFT) Manifold Capacity and Geometry

Here, we provide a summmary for finding a theoretical estimation of manifold capacity using mean-field theoretic approach. It has been proven that the general form of the inverse MFT capacity, exact in the thermodynamic limit, is given by:

$$\alpha_{MFT}^{-1} = \left\langle \frac{\left[t_0 + \vec{t} \cdot \tilde{s}(\vec{t})\right]_+^2}{1 + \left\|\tilde{s}(\vec{t})\right\|^2} \right\rangle_{\vec{t},t_0}$$

where $\langle \ldots \rangle_{\vec{t},t_0}$ is an average over random D- and 1- dimensional vectors $\vec{t}, t_0$ whose components are i.i.d. normally distributed $t_i \sim \mathcal{N}(0,1)$.

Central to this framework is the notion of *anchor points*, $\tilde{s}$ (section 2.1 in the main text), uniquely given by each $\vec{t}, t_0$, representing contributions from all other object manifolds, in their random orientations. For each $\vec{t}, t_0$, $\tilde{s}$ is uniquely defined as a subgradient that obeys the KKT conditions, hence, $\tilde{s}$ in KKT interpretation, represents a weighted sum of support vectors contributing to the linearly separating solution.

These anchor points play a key role in estimating the manifold's geometric properties, given as: $R_{\mathrm{M}}^2 = \left\langle \left\| \tilde{s}(\vec{T}) \right\|^2 \right\rangle_{\vec{T}}$ and $D_{\mathrm{M}} = \left\langle \left(\vec{t} \cdot \hat{s}(\vec{T}) \right)^2 \right\rangle_{\vec{T}}$ where $\hat{s}$ is a unit vector in the direction of $\tilde{s}$, and $\vec{T} = (\vec{t}, t_0)$, which is a combined coordinate for manifold's embedded space, and manifold's center direction (in general, if we compute the geometric properties in the ambient dimension, it includes both the embedded space and center direction).

The manifold dimension measures the dimensional spread between $\vec{t}$ and its unique anchor point $\tilde{s}$ in D dimensions (the coordinates in which each manifold is embedded).

In high dimension, the geometric properties predict the MFT manifold capacity, by

$$\alpha_{\mathrm{MFT}} \approx \alpha_{\mathrm{Ball}} \left(R_{\mathrm{M}},\, D_{\mathrm{M}} \right) \tag{1}$$

where,

$$\alpha_{\mathrm{Ball}}^{-1}(R, D) = \int_{-\infty}^{R\sqrt{D}} Dt_0 \frac{(R\sqrt{D} - t_0)^2}{R^2 + 1} \tag{2}$$

Note that the above formalism is assuming that the manifolds are in random locations and orientations, and in real data, the manifolds have various correlations. So, we apply the above formalism onto the data that has been projected into the null spaces of centers, using the method proposed by [1].

The validity of this method is shown in Fig. SM1, where we demonstrate the good match between the empirical manifold capacity (computed using a method in Section. 1.1) and the mean-field theoretical estimation of manifold capacity (using the algorithm provided in this section).

For more details on the theoretical derivations and interpretations for the mean-field theoretic algorithm, see [1][2].

Algorithm 1 compute_geometric_properties: Mean-field theoretic capacity and geometry

```
Function compute_geometric_properties($\{X^\mu\}$)
Input: Categorical data $\left\{X_i^\mu \in \mathbb{R}^N\right\}_{i\in[1..M_\mu]}^{\mu=1..P}$ ($P$=#Manifolds, $M_\mu$=#Samples per $\mu$th Manifold)
    1. Subtract global mean and update $\left\{X_i^\mu \in \mathbb{R}^N\right\}_{i\in[1..M_\mu]}^{\mu=1..P}$
    2. Compute centers of each manifold $\{\vec{c}^\mu \in \mathbb{R}^N\}^{\mu=1,\ldots,P}$
    3. Compute center correlations $\delta_{CC}$
    4. Find subspace shared by manifold centers* : $\{C^\mu\} = \text{find\_center\_subspace}\left(\{X^\mu\}\right)$
    5. Project original data into null space of center subspaces*: $\left\{X^{\perp\mu}\right\} = \text{find\_residual\_data}\left(\{X^\mu, C^\mu\}\right)$
    6. Normalize data s.t. center norms are 1**: $X^{0\perp\mu} = \text{manifold\_normalize}\left(X^{\perp\mu}\right)$
    7. For $\mu = 1..P$, calculate geometry**: $D_M^\mu, R_M^\mu, \alpha_c^\mu = \text{manifold\_geometry}\left(X^{0\perp\mu}\right)$
Output: $\{D_M^\mu\}_{\mu=1}^P, \{R_M^\mu\}_{\mu=1}^P, \{\alpha_c^\mu\}_{\mu=1}^P$
```

* is based on [1], and ** is based on [2].

2 Details of models used in experiments

2.1 CNN Model

Table 1: Word (and Speaker) CNN Model Architecture

Layer	Type	Size
0	Input	256×256 cochleagram
1	2D Convolution	96 filters of shape 9×9, stride 3
2	ReLu	-
3	MaxPool	window 3×3, stride 2
4	2D BatchNorm	-
5	2D Convolution	256 filters of shape 5×5, stride 2
6	ReLu	-
7	MaxPool	window 3×3, stride 2
8	2D BatchNorm	-
9	2D Convolution	512 filters of shape 3×3, stride 1
10	ReLu	-
11	2D Convolution	1024 filters of shape 3×3, stride 1
12	ReLu	-
13	2D Convolution	512 filters of shape 3×3, stride 1
14	ReLu	-
15	AveragePool	window 3×3, stride 2
16	Linear	4096 units
17	ReLu	-
18	Dropout	0.5 prob during training
19	Linear	Num Classes

For word recognition, we trained on two second segments from a combination of the WSJ Corpus [3] and Spoken Wikipedia Corpus [4], with noise augmentation from audio set backgrounds [5]. We selected two second sound segments such that a single word occurs at one second. For the training set, we selected words and speaker classes such that each class contained 50 unique cross class labels (ie 50 unique speakers had to say each of the word classes). We also selected words and speaker classes that each contained at least 200 unique utterances, and ensured that each category could contain a maximum of 25% of a single cross category label (ie for a given word class, a maximum of 25% of the utterances could come from a single speaker), the maximum number of utterances in any word category was less than 2000, and the maximum number of utterances within any speaker category was less than 2000. Data augmentation during training consisted jittering the input in time and placing the exemplars on different audioset backgrounds.

The resulting training dataset contained 230,357 segments in 433 speaker classes and 793 word classes. The word recognition network achieved a WER of 22.7% on the test set, and the speaker recognition network achieved an error rate of 1% on the test set.

2.2 Deep Speech 2 Model

The ASR model used in experiments is based on the Deep Speech 2 architecture[6]. A complete specification of the model is given in Table 2. The version we used is based on a popular open source implementation available at https://github.com/SeanNaren/deepspeech.pytorch

Table 2: End-to-end ASR model architecture

Layer	Type	Size
0	Input	$T \times 161$ spectral features
1	2D Convolution	32 filters of shape 41×11, stride 2
2	2D BatchNorm	-
3	HardTanh	-
4	2D Convolution	32 filters of shape 21×11, stride 2 in time only
5	2D BatchNorm	-
6	HardTanh	-
7	Bidirectional GRU	800
8	1D BatchNorm[6]	-
9	Bidirectional GRU	800
10	1D BatchNorm[6]	-
11	Bidirectional GRU	800
12	1D BatchNorm[6]	-
13	Bidirectional GRU	800
14	1D BatchNorm[6]	-
15	Bidirectional GRU	800
16	1D BatchNorm[6]	-
17	Linear	800×29

This model was trained on the 960 hour training portion of the LibriSpeech dataset [7] for 68 epochs with an initial learning rate of 0.0003 and a learning rate annealing of 1.1. The trained model has a word error rate (WER) of 12%, 22.7% respectively on the clean and other partitions of the test set without the use of a language model. The WER for different training epochs is shown in Figure **??**. The model also performs reasonably well on the TIMIT dataset, with a WER of 29.9% without using a language model.

3 Details on the manifold datasets

3.1 Manifold datasets used for experiments on the CNN Model

The words used in the CNN model word manifold dataset were selected from the WSJ corpus. The selected words were: whether, likely, large, found, declined, common, these, number, after, through, don't, workers, began, street, during, other, agreement, management, trying, office, often, members, local, without, before, continue, money, personal, first, almost, didn't, enough, results, about, their, party, would, could, people, should, military, former, january, selling, reported, times, called, holding, small, increased.

The midpoint of each word was used as the midpoint of 2s of audio, such that the amount of context was equal before and after the word of interest.

3.2 Manifold datasets used for experiments on the DS2 model

3.2.1 Words from LibriSpeech

Words were selected from a subset of the dev-clean (holdout data) portion of the LibriSpeech corpus. Word boundaries were found using forced alignment[1]. Words were selected according to the following criteria:

- Words must be at least four characters long
- Words must occur in the center of at least two seconds of audio
- Words must have at least 20 examples each from different speakers.

[1] The tool used was the Montreal Forced Aligner https://montrealcorpustools.github.io/Montreal-Forced-Aligner/

The resulting dataset consists of 50 different words, with 20 examples of each. The selected words were: made, have, while, still, being, upon, said, much, into, never, good, eyes, what, little, which, more, away, before, back, always, himself, like, from, down, such, when, under, other, should, after, another, thought, would, time, same, first, about, than, think, even, only, make, great, most, might, where, both, know, life, through.

The resulting examples were two seconds long, with the center of the word occurring at the midpoint of the signal as in the CNN word manifolds dataset. The lower bound on capacity in this experiment is $0.1 \leq \alpha$ as there are $M = 20$ samples per word.

3.2.2 Words from TIMIT

Words from TIMIT were selected according to the same criteria as the LibriSpeech words, with the exceptions that the input length was reduced to $1s$, to ensure enough samples could be obtained, and the number of samples per word was increased to 50. This resulted in the selection of 23 words. The selected words were: carry, dark, every, from, greasy, have, into, like, more, oily, suit, that, their, they, this, through, wash, water, were, will, with, would, your.

We used the same alignment as in the CNN word and LibriSpeech word manifolds dataset, with the exception of the shorter $1s$ overall length. The lower bound on capacity in this experiment is $0.04 \leq \alpha$ as there are $M = 50$ samples per word.

3.2.3 Phonemes from TIMIT

Phonemes were selected from the TIMIT dataset according to the following criteria:

- Phonemes must occur in the center of at least one second of audio
- Phonemes must have at least 50 examples each from different speakers

The resulting dataset consists of $P = 59$ phonemes, with $M = 50$ examples each. The selected phonemes are aa, ao, ax-h, b, d, dx, em, er, g, hv, iy, kcl, n, ow, pau, q, sh, th, ux, y, ae, aw, axr, bcl, dcl, eh, en, ey, gcl, ih, jh, l, ng, oy, pcl, r, t, uh, v, z, ah, ax, ay, ch, dh, el, epi, f, hh, ix, k, m, nx, p, s, tcl, uw, w, zh.

The center of the phoneme (defined as the midpoint between the boundaries) was used as the center of $1s$ of audio, similar to the construction of the word manifolds datasets. The lower bound on capacity in this experiment is $0.04 \leq \alpha$ as there are $M = 50$ samples per phoneme.

4 Additional experiments

4.1 Evolution of word error rate (WER) and manifold capacity during training

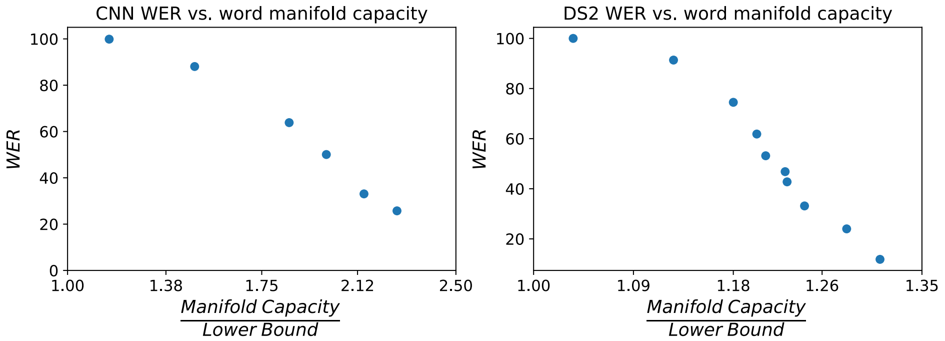

Figure SM2: **WER vs. word manifold capacity** In both the CNN (left) and DS2 (right) models, the decrease in WER during training occurs simultaneously with the rise in word manifold capacity in the final nonlinear layer. Each data points represents a different epoch of training.

During training, the word error rate (WER) decreases for both the CNN (when trained to recognize words) and DS2. At the same time, the separability of words in the later layers of both networks increases as measured by word manifold capacity. Figure SM2 shows that in the final nonlinear layer of the networks, the change in these two quantities occurs simultaneously.

4.2 Word manifolds on Speaker recognition CNN

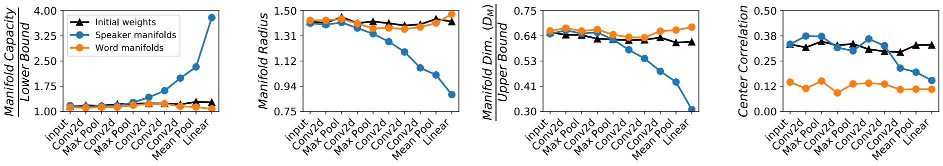

Figure SM3: **Word manifolds disappear in a model trained on a speaker recognition task** Here, the CNN model was trained to recognize individual speakers, resulting in no untanglement of word manifolds in contrast to the word trained model.

As a further negative control, we also trained the CNN model on a speaker identification task, and measured the MFTMA metrics on both speaker and word manifolds. In this scenario, we see untanglement of speaker manifolds. However, we do not see untanglement of word manifolds, which are more tangled in the final layers than with random (initial) weights of the network.

4.3 Word and speaker manifolds on a multitask trained CNN

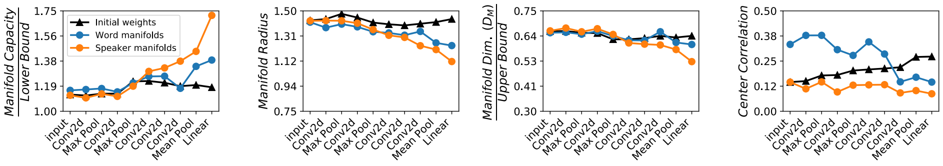

Figure SM4: **Word and speaker manifolds in a model trained simultaneously on word recognition and speaker recognition tasks**. While word and speaker manifolds emerge, these two tasks are not synergistic

When trained on both the word recognition and speaker identification tasks, the MFTMA metrics reveal some untangling of both word and speaker manifolds as shown in Fig. SM4. However, the amount of untanglement is significantly lower than when the model is trained on a single task, indicating that the two tasks are not synergistic.

4.4 Analysis across time in the DS2 model before training

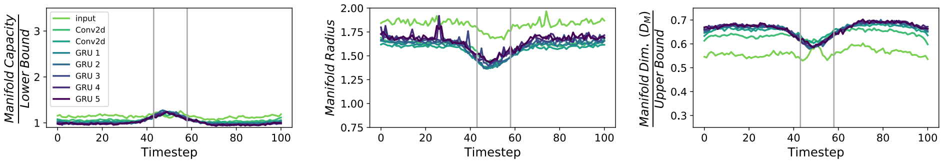

Figure SM5: **Untangling of word manifolds in input timesteps Before training DS2.** No improvement over the layers, and slightly more information in the midpoint.

When the analysis is run over timesteps in DS2, we do not see evidence for increasing untanglement of word manifolds with depth before the network is trained as shown in Fig. SM5. While we do see a slight increase in capacity along with a corresponding decrease in manifold radius an dimension near the location of the center words, these metrics remain constant through the layers of the network in contrast to the behavior seen after the network has been trained.

4.5 Analysis on DS2 training epochs

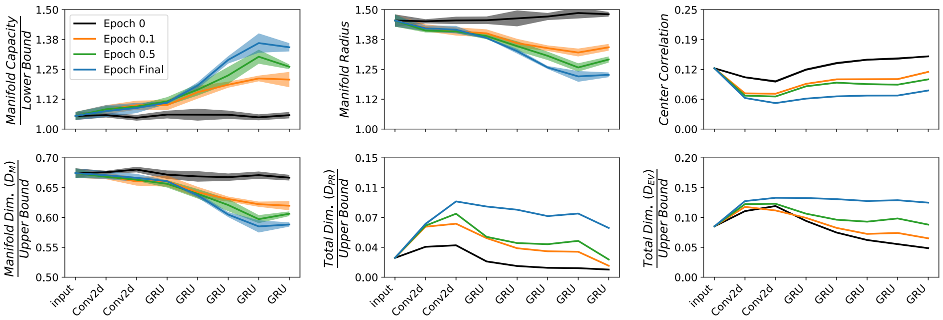

Figure SM6: **Evolution of word manifolds via epochs of training, DS2** Manifold capacity improves over epochs of training, while manifold dimension, radius, correlations decrease over training. Total data dimension (D_{PR}, D_{EV}) is inversely correlated with center correlation and increases over training epochs. Error bars show 95% confidence intervals of the mean.

Figure SM6 shows the MFTMA metrics over the course of training DS2. As a single epoch is quite large (960 hours of audio) we plot the metrics at two points within the first epoch. Here, the trends are similar to those observed in the CNN model in that the manifold dimension shrinks, while the total data dimension as measured by D_{PR} and D_{EV} rises. As in the CNN model, the manifolds are compressed (smaller manifold radius, dimension) while their centers become more spread out (larger D_{PR}, D_{EV}).

4.6 Experiment on DS2 with different length contexts

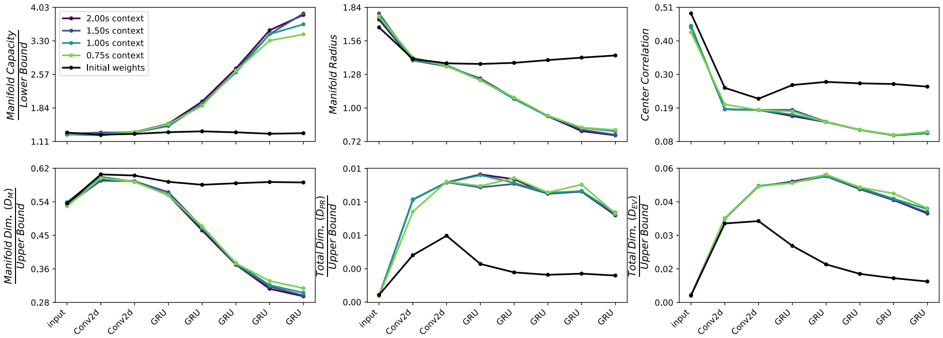

Figure SM7: **Experiment on varying length contexts (Librispeech words)**

We also carried out an exploratory analysis on the role of context in the recognition of words in the DS2 model. Figure SM7 shows the results of the MFTMA on word manifolds with different length context windows. The results shown here were measured at the center time step. The manifold datasets were constructed by taking the LibriSpeech word manifolds dataset (consisting of words centered in 2s of audio) and reducing the context symmetrically on either side of the center word. While some differences do emerge for shorter contexts in the final two layers of the network, indicating that perhaps context is important in these layers, the differences are relatively small, and so the role of context in this model remains to be understood.

4.7 Comparison with linear classifier accuracy

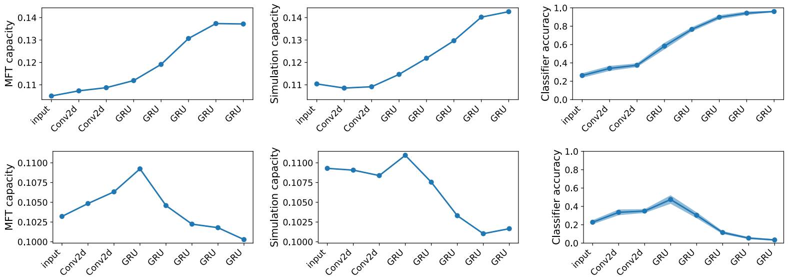

Figure SM8: Theoretical capacity (left) matches empirical capacity (center), and trends are also reflected in the generalization error of a linear classifier (right) in DS2. The top row of plots shows results on the LibriSpeech word manifolds dataset, while the bottom shows results on the LibriSpeech speaker manifolds dataset.

As further verification of the MFTMA technique, we also compared the trends in capacity, both theoretical α_{MFT} and empirical α_{SIM} to trends observed in the generalization performance of a linear classifier. Figure SM8 show this comparison to the generalization accuracy of a softmax classifier trained on 80% of the manifold data and tested on the remaining 20%. Here, the classifier was trained on 10 random train/tests splits of the manifold data, and the average over manifolds and trials is reported here along with the standard deviation of the mean. The trends observed in the the classifier performance follow those seen in the measures of capacity.

4.8 Control experiments with permuted labels

As a check that the effects we observe are indeed due to the structure of the data within the category label used to define the manifold, we conducted control experiments in which the manifold data remained unchanged, but the catergory labels were permuted. This destroys the manifold structure of data, and so the MFTMA method should give manifold capacity values near the lower bound. Figures SM9 through SM18 demonstrate that this is indeed the case.

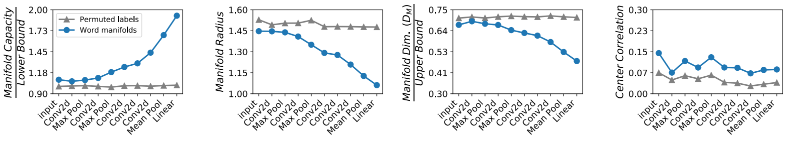

Figure SM9: Permuted vs. true manifold labels, CNN word manifolds on word trained network

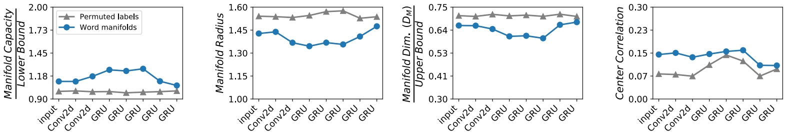

Figure SM10: Permuted vs. true manifold labels, CNN word manifolds on speaker trained network

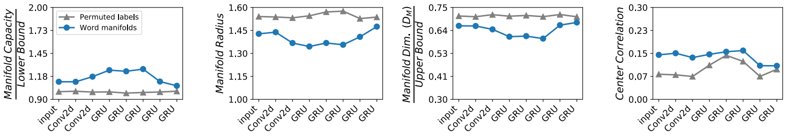

Figure SM11: Permuted vs. true manifold labels, CNN word manifolds on multitask trained network

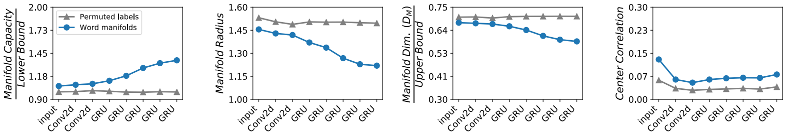

Figure SM12: Permuted vs. true manifold labels, Librispeech words experiment, DS2

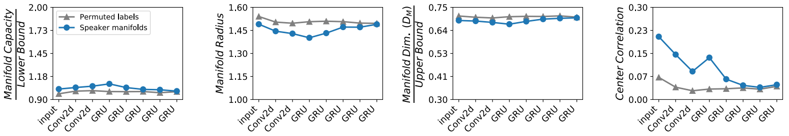

Figure SM13: Permuted vs. true manifold labels, Librispeech speakers experiment, DS2

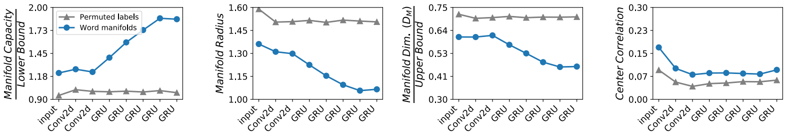

Figure SM14: Permuted vs. true manifold labels, TIMIT words experiment, DS2

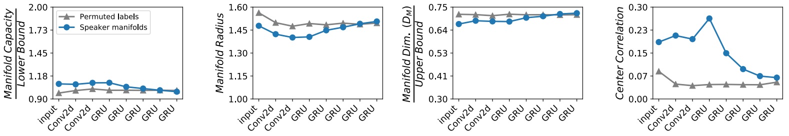

Figure SM15: Permuted vs. true manifold labels, TIMIT speakers experiment, DS2

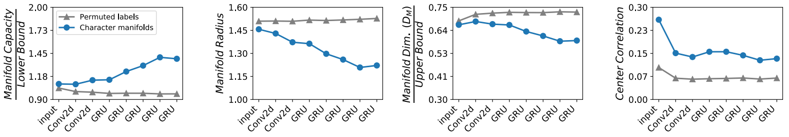

Figure SM16: Permuted vs. true manifold labels, TIMIT characters experiment, DS2

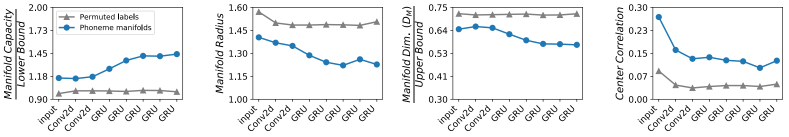

Figure SM17: Permuted vs. true manifold labels, TIMIT phonemes experiment, DS2

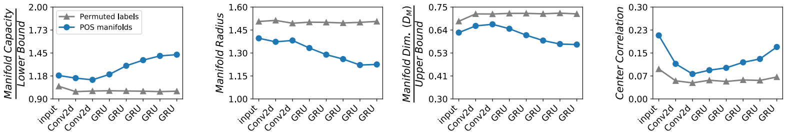

Figure SM18: Permuted vs. true manifold labels, TIMIT parts of speech experiment, DS2

References

[1] Uri Cohen, SueYeon Chung, Daniel D. Lee Lee, and Haim Sompolinsky. Separability and geometry of object manifolds in deep neural networks. *bioRxiv*, page 10.1101/644658, 2019.

[2] SueYeon Chung, Daniel D Lee, and Haim Sompolinsky. Classification and geometry of general perceptual manifolds. *Physical Review X*, 8(3):031003, 2018.

[3] Douglas B Paul and Janet M Baker. The design for the wall street journal-based csr corpus. In *Proceedings of the workshop on Speech and Natural Language*, pages 357–362. Association for Computational Linguistics, 1992.

[4] Arne Köhn, Florian Stegen, and Timo Baumann. Mining the spoken wikipedia for speech data and beyond. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Marko Grobelnik, Bente Maegaard, Joseph Mariani, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may 2016. European Language Resources Association (ELRA).

[5] Jort F. Gemmeke, Daniel P. W. Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R. Channing Moore, Manoj Plakal, and Marvin Ritter. Audio set: An ontology and human-labeled dataset for audio events. In *Proc. IEEE ICASSP 2017*, New Orleans, LA, 2017.

[6] Dario Amodei, Sundaram Ananthanarayanan, Rishita Anubhai, Jingliang Bai, Eric Battenberg, Carl Case, Jared Casper, Bryan Catanzaro, Qiang Cheng, Guoliang Chen, et al. Deep speech 2: End-to-end speech recognition in english and mandarin. In *International conference on machine learning*, pages 173–182, 2016.

[7] Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. Librispeech: an asr corpus based on public domain audio books. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5206–5210. IEEE, 2015.