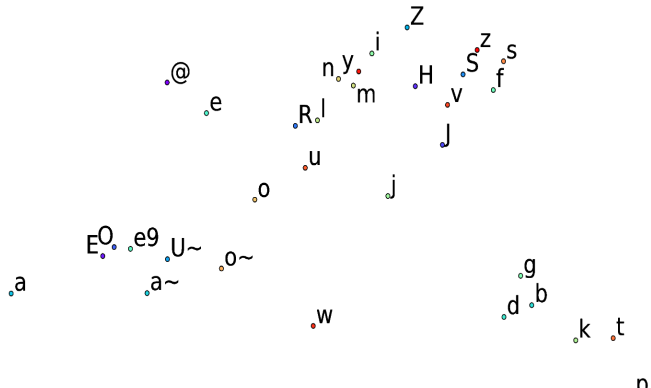
Inferring phonemic classes from CNN activation maps using clustering techniques

Thomas Pellegrini, Sandrine Mouysset

Université de Toulouse; UPS; IRIT; Toulouse, France
thomas.pellegrini@irit.fr, sandrine.mouysset@irit.fr



Motivation

Trainability: if a good network solution exists with small training error, how do we find it? And what makes a learning problem difficult?

Expressivity: what kinds of functions can a deep network express that shallow networks cannot?

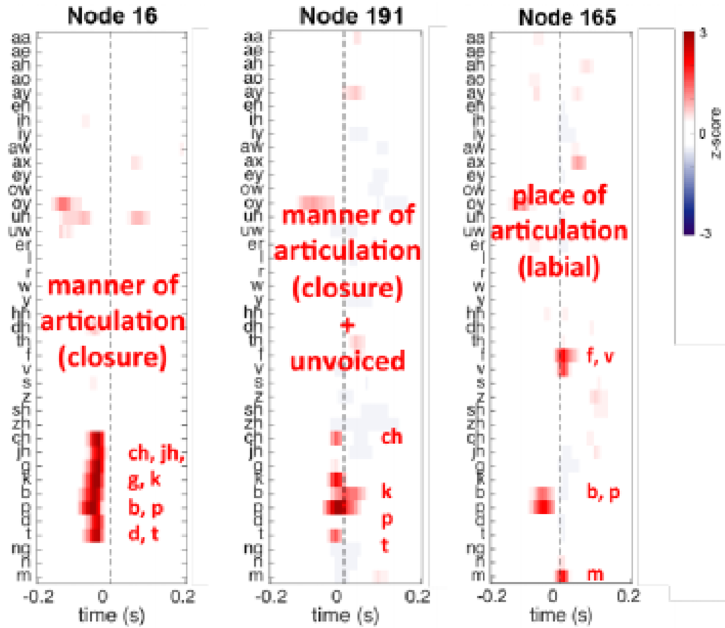
Generalizability: what principles do deep networks use to place probability / make decisions in regions of input space with little data?

→ **Interpretability** : once we have a trained network, how do we understand what it does? How is the training data embedded in the weights?

Biological Plausibility: how can we do what we do within the constraints of neurobiology? How can we interpret specific architectures used by the brain?

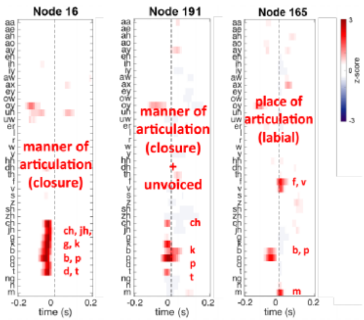
Slide from Surya Ganguli, `http://goo.gl/YmmqCg`

Related work in speech: with DNNs



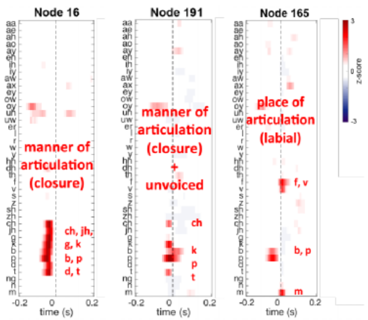
Source : Nagamine et al. Exploring How Deep Neural Networks Form Phonemic Categories. INTERSPEECH 2015

Related work in speech: with DNNs



- Single nodes and populations of nodes in a layer are selective to phonetic features
- Node selectivity to phonetic features becomes more explicit in deeper layers

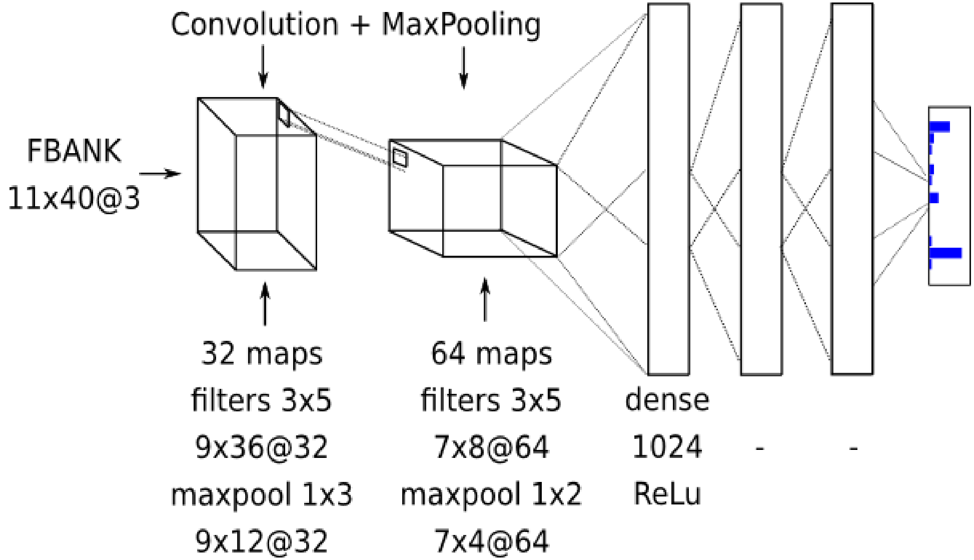
Related work in speech: with DNNs



- Single nodes and populations of nodes in a layer are selective to phonetic features
- Node selectivity to phonetic features becomes more explicit in deeper layers

- Do these findings still hold with convolutional neural networks?

CNN Model used in this study



- BREF corpus: 100 hours, 120 native French speakers
- Train / Dev sets: 90%/10%, 1.8M/150K samples
- PER: 20% → good accuracy, allows the analysis of the model

Study workflow

Does a CNN encode phonemic categories such as a DNN does?

- 100 input samples per phone feed-forwarded through the network
- The outputs of each layer extracted and fed to either k-means or spectral clustering, with optional front-end dimension reduction
- Remark: 4-d tensors reshaped into 2-d matrices

Study workflow

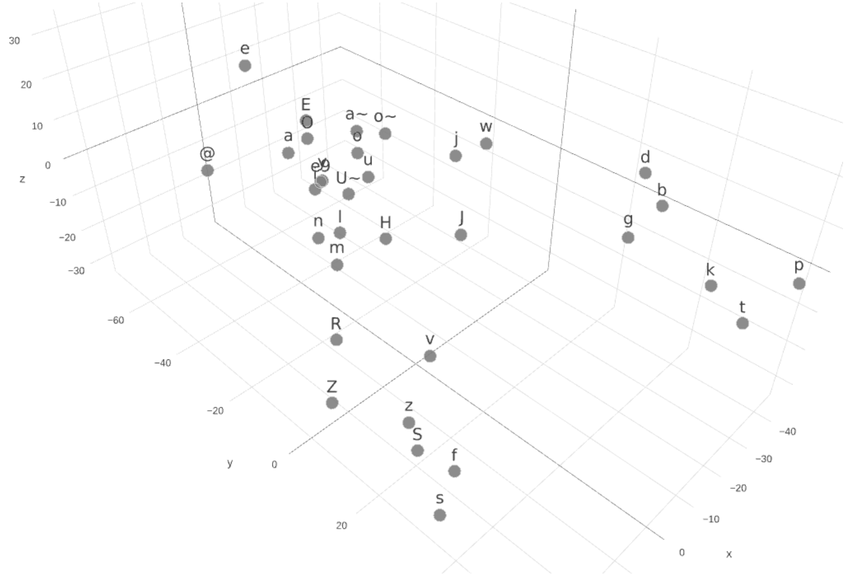
Does a CNN encode phonemic categories such as a DNN does?

- 100 input samples per phone feed-forwarded through the network
- The outputs of each layer extracted and fed to either k-means or spectral clustering, with optional front-end dimension reduction
- Remark: 4-d tensors reshaped into 2-d matrices

- Experiment 1: fixed number of 33 clusters (French phone set size)
- Experiment 2: optimal number of clusters determined automatically

Dimension reduction

- Principal Component Analysis (PCA) processed on the whole activation maps: the number of principal components that keeps at least 90% of the covariance matrix spectrum

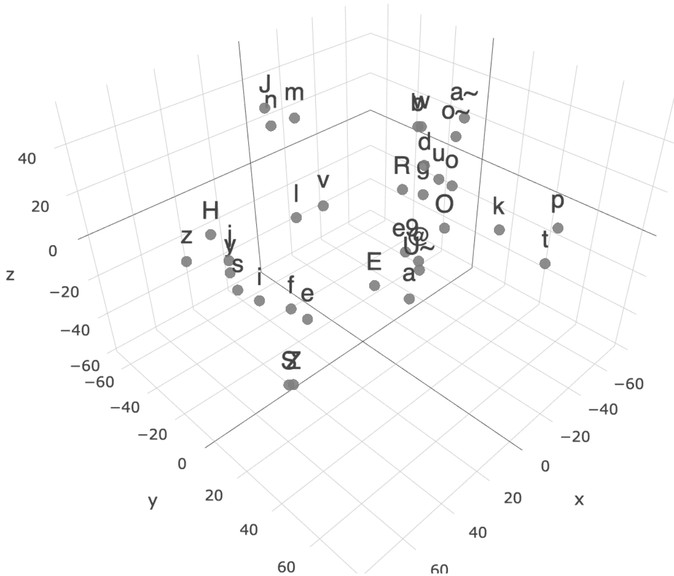


PCA projections of averaged activations

http://goo.gl/bbuZn9

Dimension reduction

- t-Distributed Stochastic Neighbor Embedding (t-SNE):
  relies on random walks on neighborhood graphs to extract the local structure of the data and also reveal important global structure



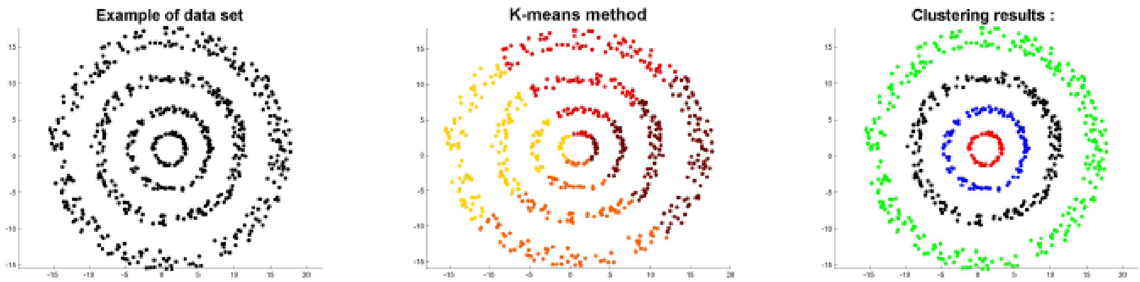
t-SNE projections of averaged activations

http://goo.gl/4f3nZ3

Clustering methods

Consider the two most popular clustering techniques based on either linear separation or non-linear separation:

- Kmeans computed with the Manhattan distance
- Spectral Clustering selects dominant eigenvectors of the Gaussian affinity matrix in order to build a low-dimensional data space wherein data points are grouped into clusters



Clustering methods

Consider the two most popular clustering techniques based on either linear separation or non-linear separation:

- Kmeans computed with the Manhattan distance
- Spectral Clustering selects dominant eigenvectors of the Gaussian affinity matrix in order to build a low-dimensional data space wherein data points are grouped into clusters

Choice of the number of clusters:

- Kmeans: within- and between-cluster sums of point-to-centroïd distances
- Spectral Clustering: within- and between-cluster affinity measure

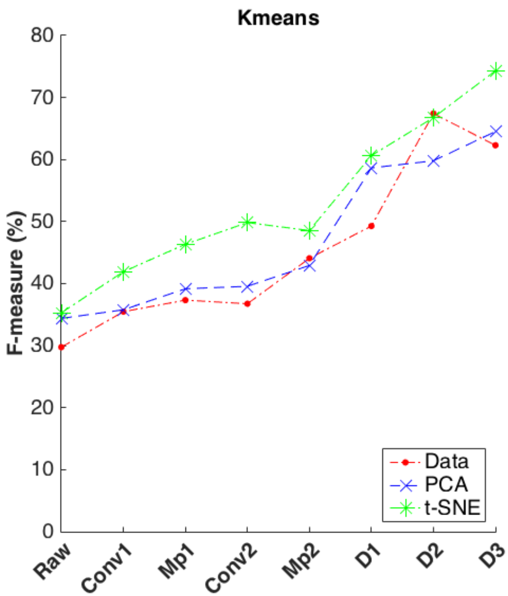
Evaluation for experiment 1

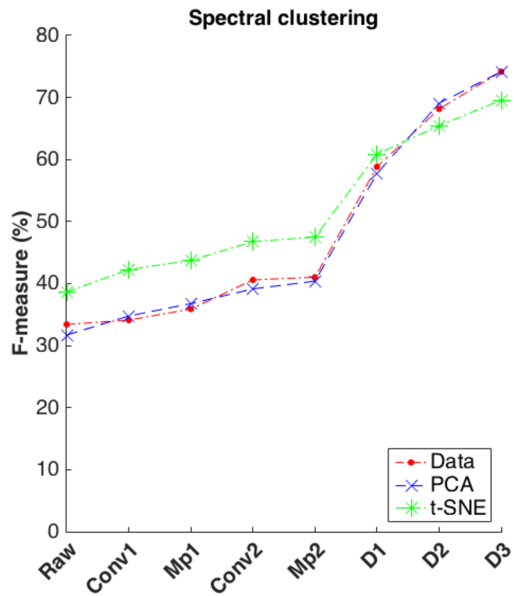
Evaluate the resulting clusters with a fixed number of 33 clusters:

$$P = \frac{tp}{tp + fp}, \; R = \frac{tp}{tp + fn}, \; F = 2\frac{P.R}{P + R}$$

where tp, fp and fn respectively represent the number of true positives, false positives and false negatives

Experiment 1: 33 clusters





→ Phone-specific clusters become more explicit with layer depth

Experiment 2: optimal number of clusters

7 clusters with SC

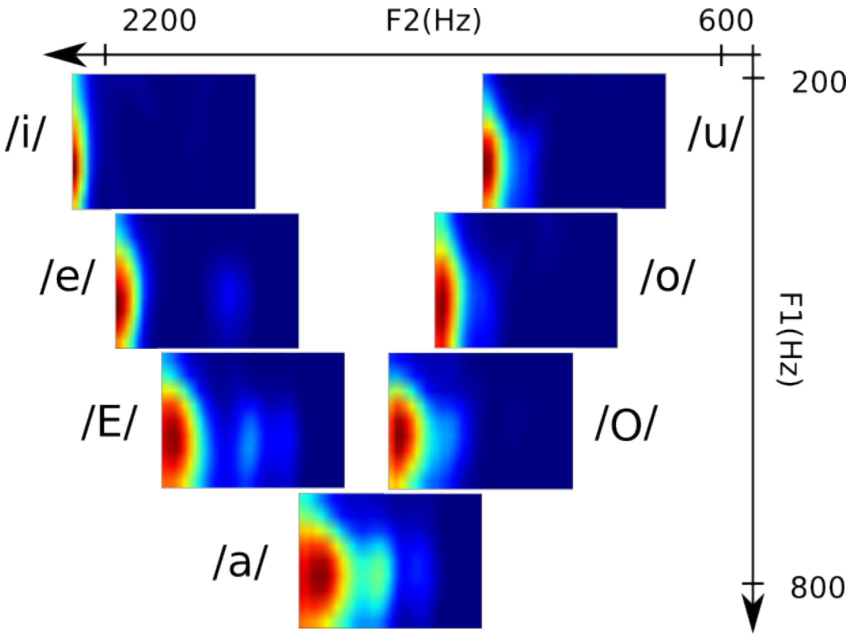
- 3 clusters for the vowels:
  1. 93% of the medium to open vowels [a], [E], [9]
  2. 83% of the closed vowels: [y], [i], [e]
  3. 60% of the nasal vowels /a~/, /o~/, /U~/
- 4 clusters for the consonants:
  1. 92% of the nasal consonants: /n/, /m/ and /J/
  2. 81% of the fricatives: /S/, /s/, /f/, /Z/
  3. 76% of the rounded vowels /o/, /u/, /O/, /w/
  4. 68% of the plosives consonants: /p/, /t/, /k/, /b/, /d/, /g/

k-means: similar clusters

→ Broad phonetic classes are learned by the network

Average activation map example of layer "conv1"

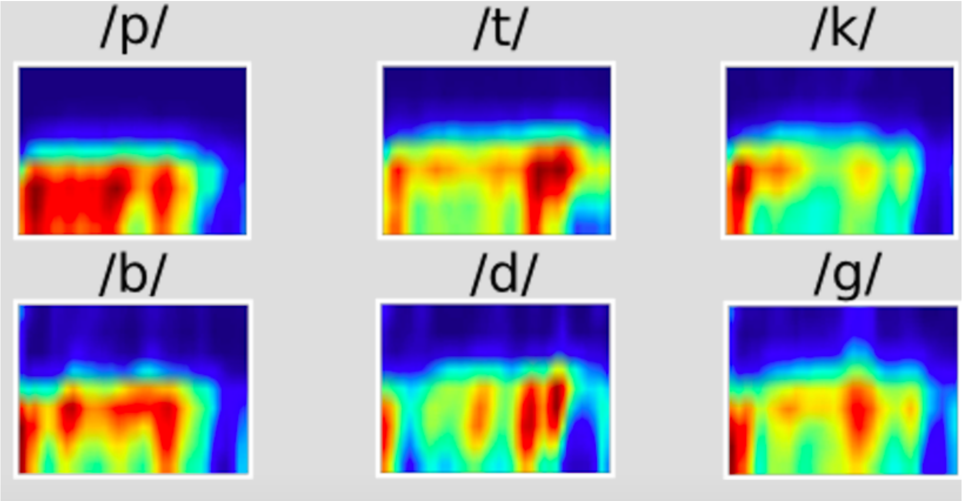
- Vowels



- This map encodes the mouth aperture (F1) but not the vowel anteriority (F2)

Average activation map example of layer "conv1"

- Plosives



Conclusions and future work

Findings with CNNs similar to previous work by Nagamine with DNNs:

1. Phone-specific clusters become more explicit with layer depth
2. Broad phonetic classes are learned by the network

Ongoing/future work:

- Studying the maps that do not correspond to phonemic categories
- What is the "gist" of the phone representations for a CNN?

Thank you!

Q&A

thomas.pellegrini@irit.fr