Artificial and Biological Neural Systems

Theory Notes

Giovanni Valer

a.y. 2023/24

These notes are for the course "Artificial and Biological Neural Systems" held by professor Uri Hasson.
I have included all the topics seen during the lectures and tried my best to avoid mistakes. Mandatory papers are underlined in the first citation. I have also included some digressions (i.e. non-mandatory topics), marking them as such and with a yellow background. That being said, a very broad range of topics was to be covered, hence I do not take responsibility for any mistake or imperfection that might be present; still, it would be much appreciated if you reported any of those, so that I can fix the notes accordingly. In the GitHub repository[1] *you can find the LaTex source code; feel free to create issues and pull requests to include topics and corrections.*
I hope that this document will help in the study of this beautiful and exciting subject.

If you consider this material valuable for you in order to be prepared for the exam, consider offering me a coffee :)
Paypal: @GiovanniValer (`https://www.paypal.me/GiovanniValer`)

XOXO - jo

[1] `https://github.com/jo-valer/ABNS-Hasson`

Contents

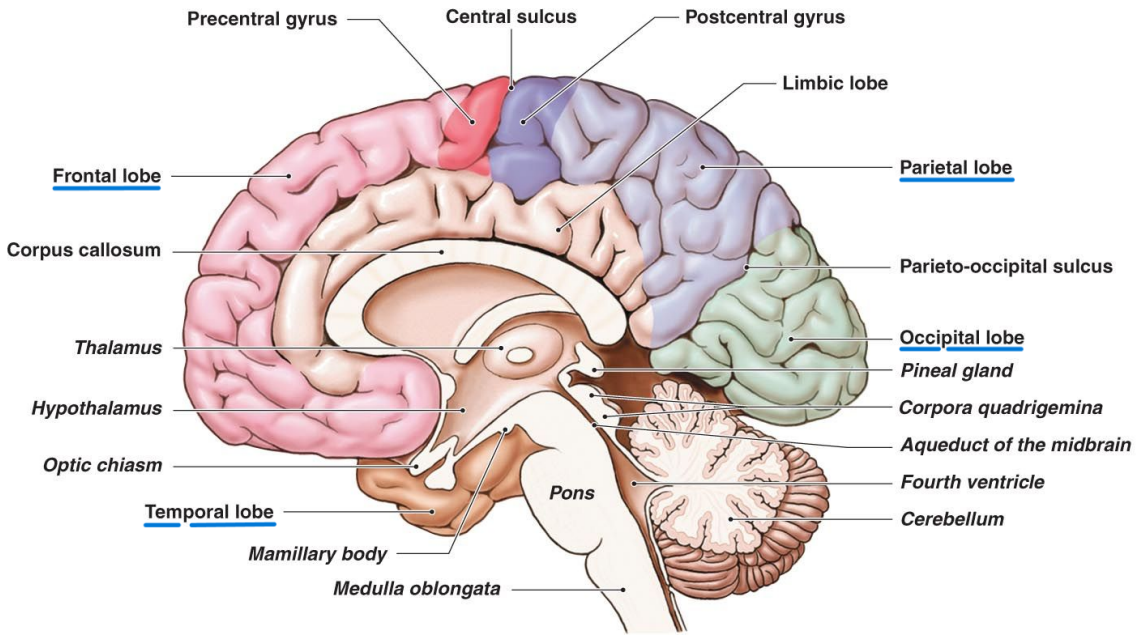
Chapter 1

Introduction to brain physiology and research methods

1.1 Basic facts about the human brain

The human brain is partitioned in many structures. The first one is the **cortex**, and in the following we provide the *traditional function specializations* ▲:

- **Temporal** lobe: **Auditory processing** (hearing and language)
- **Parietal** lobe: **Attention**, touch, saccade planning
- **Frontal** lobe: Planning, execution, **higher level cognition**, high level language processing and language production
- **Occipital** lobe: **Vision**: perception of visual features, categories and location



▲ Here we have the traditional function specializations, even though nowadays we know that all areas of the brain take part in more functions: a function is computed by the whole network. The auditory processing is a sort of exception: audio is processed only in the temporal lobe; however, it is still part of the network, as its functions can be *modified on demand*.

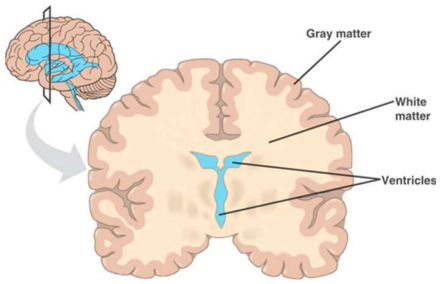
We then have the **subcortical structures**:

- **Cerebellum**: Fine motor control, implicated in emotional responses
- **Thalamus**: Major gateway for sensory processing. One of the final stops before sensory information arrives at the cortex

- **Hippocampus**: implicated in construction of memories; these are later transferred to other cortical regions
- **Corpus Callosum**: A main "highway" of white matter tracks that connects the two hemispheres

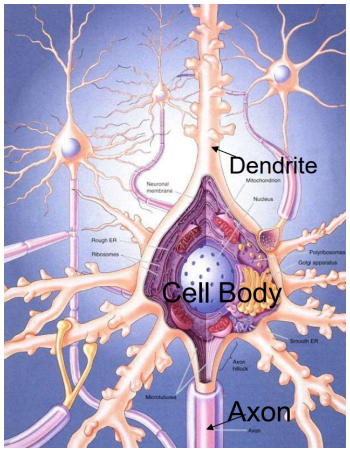
1.1.1 Gray matter

Gray matter (GM), which gets its name from its color, is the brain part that **performs computation**. The gray matter includes not only the cerebral cortex but also the cerebellum, basal ganglia, thalamus, and several other regions. Techniques for measuring brain activity often reflect the function of gray matter regions, which are responsible for the majority of cognitive processing. As a result, understanding the role and function of gray matter in the brain is essential for gaining insight into various neurological and psychiatric conditions.
The gray matter is on the perimeter of the brain, and its folded structure (with *gyri* and *sulci*, determined by the DNA) allows for higher surface. This also causes parts that are located close to each other, in physical space, to be far away in cortical space.



1.1.2 White matter

White matter consists mainly of **long-range axon pathways**, or *tracts*, that **connect different regions of the brain**. Unlike gray matter, there is **no direct information processing** within white matter itself. The structure of white matter can change with learning because it reflects the long-range neural pathways that carry nerve impulses and facilitate the communication between different regions of the brain.



Damage to white matter, such as the loss of myelin in conditions like multiple sclerosis, can have significant impacts on communication between different regions of the brain, leading to various neurological and psychiatric symptoms. Therefore, understanding the role and function of white matter is critical for studying brain function and identifying potential targets for interventions in various neurological and psychiatric disorders.

1.1.3 The neuron

A neuron consists of **dendrites**, a **cell body** (aka **soma**) and an **axon**. Connections between neurons are called **synapses**, and typically occur on a neuron's dendrite (but in some cases also on soma), which receives synaptic signals. Synapses are **chemical**, **not electrical**. A neuron will *fire* (generate an action potential) depending on the number of signals it receives on its dendrites and their strength, which are *summed* in the neuron's body.

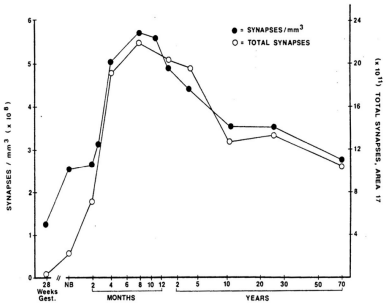
Digression: Synaptic communication

Synaptic communication is the process by which neurons communicate with each other through the release and reception of chemical signals called neurotransmitters. The postsynaptic neuron receives the signal, which is either excitatory (the neuron is depolarized and fires an action potential), or inhibitory (causing the neuron to hyperpolarize and become less likely to fire an action potential). Depolarization produces an *all-or-nothing* signal: **no partial firing**.

1.1.4 Neuroplasticity

Neuroplasticity refers to *network changes* over time, to adapt to the environment. Synapses can be *strengthened* or *pruned*.

Connections between neurons are consistently removed (or created) depending on use. A large percentage of neurons that develop die. A *hub* neuron, which receives inputs from many neurons is more likely to survive. When many neurons connect to a target neuron, this decreases their survival rate. This calibration is thought to be associated with generating an optimal degree of synaptic connection. Brain volume triples between birth and adulthood; this is mostly not due to addition of neurons, but to an increased number of connections (synapses), myelination of existing axons and greater dendritic branching.



Synaptic density and total synapses in visual cortex as a function of age.

1.2 Studying the human brain: tools and basics of experimentation

The human brain can be studied at different levels of organization, from systems and pathways to synapses and membranes.

- Systems and pathways: large-scale neural networks responsible for specific functions, such as sensory perception, motor control, and cognition. These may be topographically distributed.
- Circuits and neurons: networks of interconnected neurons that underlie information processing within the brain.
- Synapses and membranes: molecular and cellular mechanisms that govern the transmission of signals between neurons, such as the release of neurotransmitters and activation of ion channels.

Studying the brain at different scales provides insights into organization of function from the macroscopic to the microscopic level.

1.2.1 Basics of experimentation

There are several **non-invasive tools** available for studying brain activity and structure, including electroencephalography (EEG), structural imaging, functional magnetic resonance imaging (fMRI), and diffusion-weighted imaging. These tools provide insights into different aspects of brain function and structure: electrical activity of neurons, structural connectivity of brain regions, and metabolic activity (i.e. energy consumption) associated with specific tasks or behaviors.

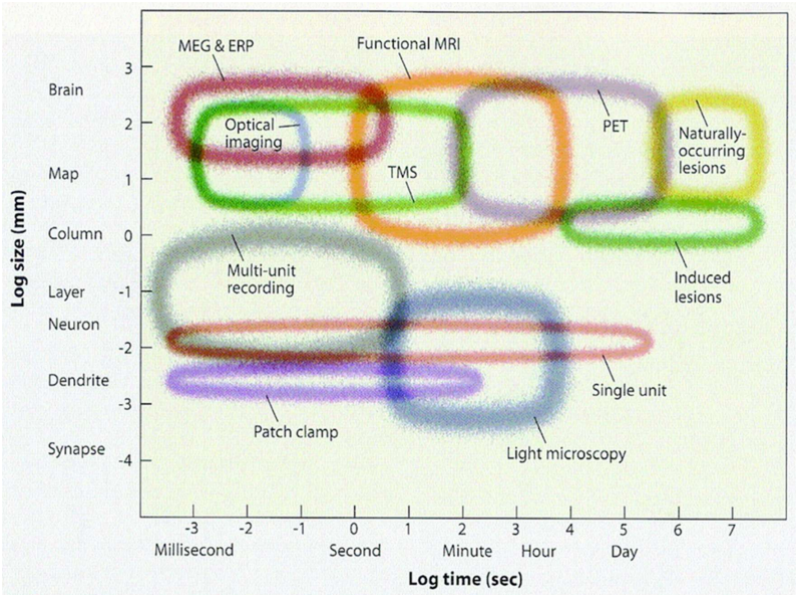


Figure 1.1: **Temporal** and **spatial resolution** of common tools. An additional dimension is **coverage**: how much of the brain is simultaneously observable by the tool.

Experimental procedures are used to analyze the data collected by these tools; they typically consist in conducting studies that involve manipulating variables of interest and collecting and analyzing data from participants. In neuroscience, an **experiment** or study is a systematic **investigation of a research hypothesis** that involves **manipulating variables and measuring their effects on some outcome of interest**. Conclusions from experiments are drawn by analyzing the data collected from participants and testing whether there are statistically significant differences between groups or conditions, typically using statistical methods to quantify the strength and direction of effects and assessing the probability that the observed effects are due to chance.

In the following sections we will discuss the most important tools (namely EEG, structural imaging, fMRI), however there are many others:

- TMS (Transcranial Magnetic Stimulation): induces a virtual lesion of a part of the brain (this might be dangerous).
- MEG (Magnetoencephalography): directly measures the magnetic fields generated by neural activity.
- Patch clamp: records the current from ion channels in the cell membrane

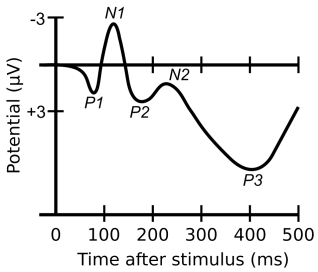
1.2.2 Electroencephalography (EEG)

EEG is a non-invasive method used to study patterns of brain activity with high temporal resolution. The principle behind EEG is that it is sensitive to very subtle changes in electric potentials below the sensors, which are propagated to the scalp. These changes reflect alterations in the electrical environment of thousands of neurons that fire in synchrony. Each EEG sensor gives one time series. It is difficult to pinpoint the brain regions causing the fluctuations, since the electromagnetic waves are dispersed by the scalp. However, the **timing of the signals is very precise**.

From EEG time series to ERP

Tasks can generate **stereotyped evoked potentials** that can be averaged (over all epochs ▲) to obtain an **Event-Related Potential**" (aka **Evoked-Response-Potential**). An ERP analysis quantifies electrical brain responses to events/stimuli based on time-locked EEG portions. This analysis can be used as the basis for more sophisticated analysis such as source localization.

▲ An epoch consists in the timespan [0ms, 500ms] after the stimulus presentation.



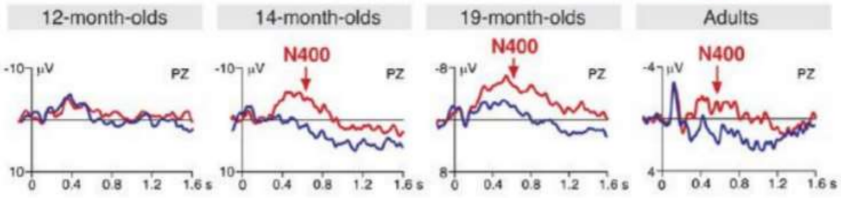
A waveform showing several ERP components. Notice the plot has negative voltages upward.

Neural activity is not the only activity causing electrical fluctuations. Muscles are also one of the main causes of fluctuations (e.g. every time we blink or move our eyes there are oscillations collected by EEG sensors). All these **noise** artifacts have to be independently measured and then removed from the data. The sensors placed near the eyes are indeed used to measure such noise. Moreover, since the interesting brain **signal is low**, while the **noise is high**, **many repetitions** of each condition are needed (the impact of noise on computing ERPs scales down as a function of the square root of the number of observations). This noise reduction applies to each timepoint measured. The need for many repetitions can be challenging and time-consuming, but it is necessary for obtaining reliable and statistically significant results. In addition to repetition, other techniques such as filtering and artifact rejection can also help to reduce noise in EEG recordings.

EEG also contains important information in form of frequency characteristics (cycling rate). The frequency differentiates sleep stages from awake, and drowsy from alert. Power plots can show the relative strength of each frequency, helping in *frequency band interpretation*. For instance, the alpha band in EEG, which has a frequency range of 8-12 Hz, is commonly observed during eyes-closed recordings and relaxed states. Alpha oscillations are associated with reduced communication between the cortex and thalamus. During externally oriented attention and stimulus processing, the alpha activity is suppressed.

ERP in practice

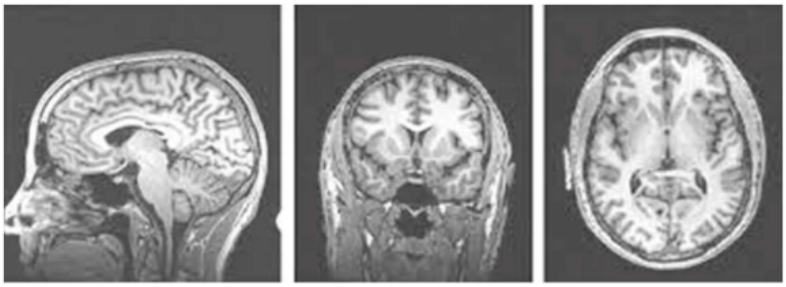
This experiment tries to understand when children develop the ability to predict language. It consists in presenting congruous (e.g. *pizza was too hot to eat*, in blue) or incongruous (e.g. *pizza was too hot to* ***sit***, in red) sentences.



Here we see the ERPs from a single sensor (the PZ one). We can notice the time point where the two functions diverge is when the word is processed by the brain (so we can understand how much it takes for a word to be processed).

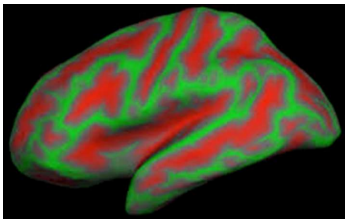
1.2.3 Structural imaging

Structural imaging involves collecting 3D images of the brain, similar to the images obtained in a medical setting.



This type of imaging can provide information about various aspects of brain structure, such as:

- Gray matter volume (i.e. the overall size of the gray matter in the brain)
- The density of gray matter in specific regions, which can approximate the concentration of neurons
- Cortical thickness at a resolution of a few millimeters
- Surface area of particular brain regions



Example of cortical thickness map.

Calculation of cortical thickness is usually done by converting the brain's 3D representation to a 2D sheet representation. We can look for correlation (covariance across different people) of structural cortical thickness between different areas of the brain.

1.2.4 Functional magnetic resonance imaging (fMRI)

Functional MRI is the method that helped most in uderstanding how the brain works. It consists in observing which parts of the brain are involved in **metabolic activity** when we do things like thinking or perceiving ▲.
FMRI uses a big magnet to affect protons in the brain and then measures how they behave as they return to their original state (areas that are involved in oxygen consumption have a different relaxation profile). By analyzing the patterns of proton behavior, we can identify which brain areas are more active during certain tasks. This method gives us very detailed information about where activity is happening in the brain and can help us understand how different regions contribute to complex processes like thinking and perception. With fMRI we get a time series for each **voxel** (typically a $3 \times 3 \times 3$ mm brain region).

▲ In neurology, metabolic activity is often used as a proxy for neural activity: active neurons require more energy to function and can increase their metabolic rate.

In our brain there are 60 to 70 thousands voxels. By chance (statistics) we will surely get some false positives. For this reason we undertake many experiments.

fMRI in an experimental context

The **signal** is defined as the measurable response to a stimulus.
In statistical detection theory we:

- understand relationship between stimulus and signal;
- describe noise properties statistically;
- devise methods to distinguish noise-only measurements from signal+noise measurements, and assess the methods' reliability.

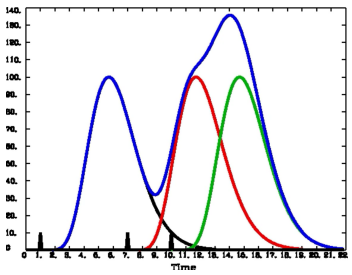
Stimulus-signal connection and *noise statistics* are complex and poorly characterized. We devise, a priori, a **mathematical model connecting stimulus** (or "activation") **to signal** (typically a regression model). We make an estimation of the **statistical model for the noise**, taking into account whether it is random, structured, oscillatory, etc. These two models are then combined to produce an **equation for measurements given signal+noise** (also often a regression model). The final result is an equation with few **free parameters that can be fitted to the data**.

FMRI analysis often fits a (convolved) activation model to **each voxel's time series separately** (*massively univariate analysis*). Pre-processing techniques are applied to reduce noise, including spatial smoothing across nearby voxels. The outcome of model fitting is a collection of parameters estimated from each voxel's data. The Activation Amplitude (Beta) is the most critical parameter, and it is related to the correlation between the model and activity. At the group level, the **voxel-level estimates are pooled together to reach a group-level conclusion per voxel**.

FMRI measures changes in neural activity, there are no absolute magnitudes. The baseline signal level in a voxel does not provide information about neural activity. An experiment using fMRI requires at least two conditions to detect changes in neural activity. Minimally, experiments use a "task" and "rest" condition. In such a two-task experiment intermixed with rest, a Beta value is estimated per task, and their values are contrasted to determine the difference between the two tasks.

Hemodynamic response function (HRF)

The response measured by fMRI (called "Hemodynamic" because it is related to blood) is delayed since the blood requires time to flow. Notice this is not a simply shifted impulse response, as it is a smooth function. For this reason, fMRI tells us **where** the process takes place, but not **when**. Combining fMRI and EEG can be a solution.



1.2.5 Diffusion weighted imaging (DTI)

Diffusion weighted imaging (also known as **Diffusion MRI** or **Diffusion tensor imaging**), is used to examine the structure of **white matter fibers**. For each voxel, the preferred **direction** of diffusion and the **strength** of diffusion are estimated to determine white matter tracts. These connections are considered *hardwired* connections. They can be cross-referenced against functional connectivity.

Chapter 2

Overview of ML approaches to modeling cognitive neuroscience data

In this Chapter we have two papers on the topic of **cognitive neuroscience models**.

2.1 *Analyzing biological and artificial neural networks: challenges with opportunities for synergy?* Barrett et al. (2019)

Deep neural networks brought a revolution in the area of ML, with millions of parameters, no engineered features, and very high performance. We can see an **analogy with neuroscience**, as both fields need to:

- understand how neural networks, consisting of large numbers of interconnected elements, transform representations of stimuli across multiple processing stages to implement a wide range of complex computations and behaviours;
- describe and analyze very high dimensional data.

The analogies appear in four areas: Receptive fields, Ablation, Dimensionality reduction, and Representational geometries.

2.1.1 Receptive fields

Neurons in the human visual cortex are **specialized to process stimuli in specific spatial areas** (see ◇ Retinotopy map) **or certain types of features**. The neurons in the **initial processing** regions of the visual cortex have **small receptive fields**; sensitive to stimuli in small areas of visual space. As information is transmitted to **higher level** areas of visual processing, **receptive fields become larger**, enabling sensitivity to larger areas of space. These regions also encode more complex features, and there is evidence of "abstract" coding with invariance to small transformations. There are "**concept cells**" **sensitive to identity** of objects but **not to appearance**. For example, simple "repetition priming ▲" effect repeated exactly with same face but different orientation.

▲ Repetition priming refers to the change in responding to a word or an object as a result of a previous encounter with that same item, either in the same task or in a different task.

(Some) AI researchers also think that **DNN neurons may code for specific information**, which can be studied via receptive field analysis.
Some experiments investigate which types of images maximally activate a neuron. Other studies examine how receptive fields change in deeper layers (as we go deeper in the network, each element in the feature map is occupying more space in the visual field, through pooling, but *in what way are larger receptive fields more complex?*).

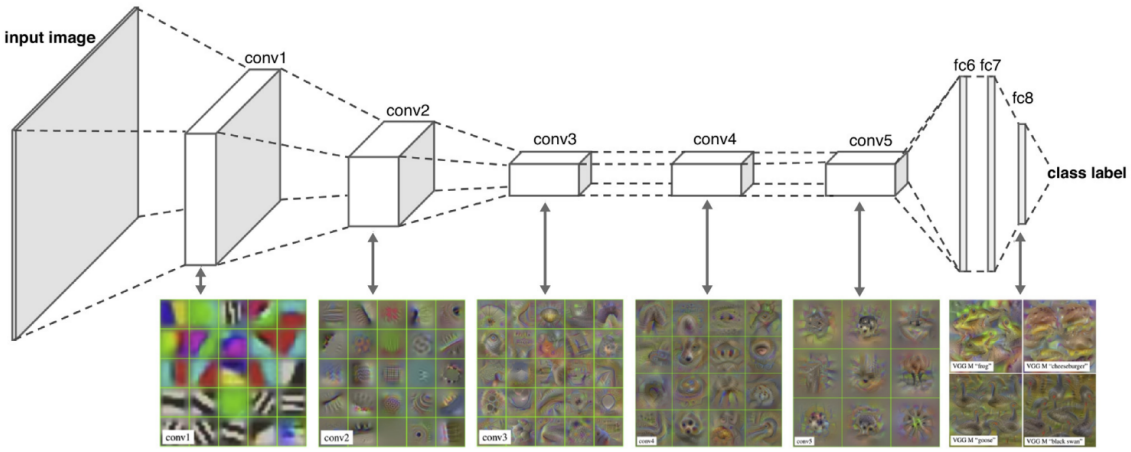
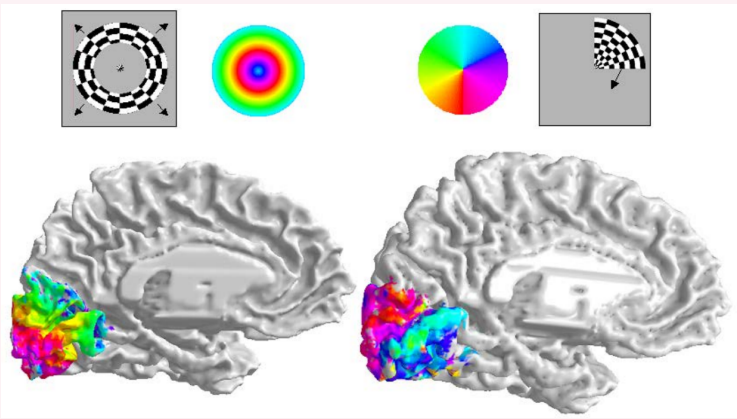


Figure 2.1: Top: how an input image is progressively processed by a CNN. Bottom: receptive fields for each layer are calculated using activation maximization (i.e. take an already trained CNN and create stimuli that produce the maximal activation in a given feature mapping in the network). For each layer, a single square corresponds to a feature map. We can notice `conv1` (the first layer) encodes large scale information; some shapes (e.g. circles) start appearing as early as `conv2`.

◆ **Retinotopy map**

The **occipital cortex** (**visual cortex**) is devoted to image processing. They found out that the coding in the brain is along two dimensions: **eccentricity** and **radial degree**. It is possible to get a map of such coding by applying fMRI to a relatively simple experiment:



Focus first on the experiment on the left. The person is looking at the center of the shape. While looking, the shape changes according to a defined temporal pattern (shrinking and expanding with a certain period). If there are neurons caring for a certain distance from the center, then they should fire at determinate time points. This is indeed the case, and we can track which neurons code for a certain eccentricity (we use colors to represent eccentricity). The experiment on the right is quite similar: while the person is looking at the center, the shape rotates around it.
Different people have almost all the same coding, and it is interesting to note how there are no "jumps" in the coding.

A similar thing can be done with the **primary auditory cortex**, in the **temporal lobe**, to get a **tonotopy map**.

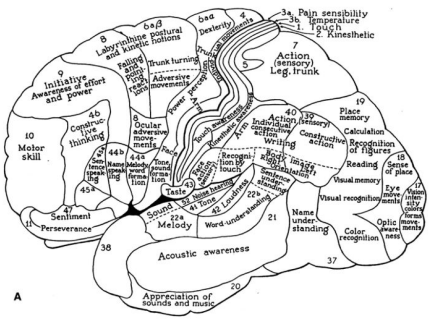
2.1.2 Ablation

Brain lesions (ablations) offer much information about potential function of brain areas. By mapping lesions to symptoms, we can understand which brain areas are important for given tasks.

Pruning and ablation causes performance deficits. However, thanks to neuroplasticity, the human brain can achieve again the same performance.

Speaking of artificial neural networks, the debate in the last years has seen three thesis on how DNNs encode information:

- distributed information (meaning we can remove some parts without hardly impacting a single task),
- local encoding (meaning some neurons are super specialized and don't share encoding information),
- Modularity theory: not each single neuron is necessary, but some modules (groups of neurons) are necessary for specific tasks.



Kleist's functional brain map. This map is old and partially incorrect, but still interesting.

We ask ourselves, how do artificial neural networks change after ablation? The ablation (lesioning) analysis is applicable to DNNs. We can *silence* neurons and observe how this impacts the network output. Silencing of neurons is done via **structural pruning** (entirely removing a neuron with all its outgoing weights). Networks trained for generalization (out of sample prediction) are more robust to ablation than those trained on memorizing labels. Pruning and fine-tuning are active areas of research.

2.1.3 Dimensionality reduction

The brain codes information in a distributed manner, necessitating multivariate analysis:

- Multiple units encode information in the brain, leading to **redundancy** where two neurons may fire almost identically; or
- Information is coded in a **distributed** manner among multiple units (e.g. coding for 4 classes among 2 neurons, each coding $\{0, 1\}$); or
- **Correlation** among units could indicate that the **activity can be described in a lower dimensional space**.

It has been shown that, in DNNs, an object-by-feature matrix from fully connected layer can be **compressed by more than 80%** (e.g. from 4096 dimensions to 512) while maintaining almost all variance. This means few dimensions are enough to explain differences between images.

2.1.4 Representational geometries

To **understand representations**, we study how they are represented in different layers, how they change over time, and how are different embedding spaces related to each other.

Matrix factorization measures: they are a good starting point to compare matrices and understand if there is some sort of covariance between the two. We compare representations across networks by comparing **object-by-feature matrices**. *Canonical Correlation Analysis* and *PLScorrelation* are two different factorization measures, they identify lower-level factors that capture and maximize the correlations/covariance between the datasets (note: these methods can be seen as "supervised" as they re-weight columns in both tables to maximize similarity).

Representational Similarity Analysis: it involves **comparing two similarity matrices**, often constructed from object-by-feature matrices. This yields an object-by-object similarity matrix. Representational Similarity Analysis does not factorize matrices. The principle is: describe how the objects cluster together (in the representational space) according to the matrix. To code for distances between objects, we need a distance matrix; and to get this a distance function is needed (e.g. Euclidean, Mahalanobis, etc.), indeed each row in the object-by-feature matrix is a vector. So we can **turn the object-by-feature matrix into an object-by-object distance matrix** (pairwise distance between objects). For human data (how human represent things), we usually don't get an object-by-feature matrices (we are directly provided an object-by-object matrix), so with RSA we can study the **relationship between human representation and machine representation**.

Linear regression: The machine representation in the neural network (*DNN embeddings*) can be used to predict brain activity (*neurobiological activation vectors*) using linear regression.

2.2 *What does the mind learn? A comparison of human and machine learning representations* Spicer and Sanborn (2019)

This paper reviews the modern machine learning techniques and their use in models of human mental representations, detailing three notable branches: spatial methods, logical methods and artificial neural networks.

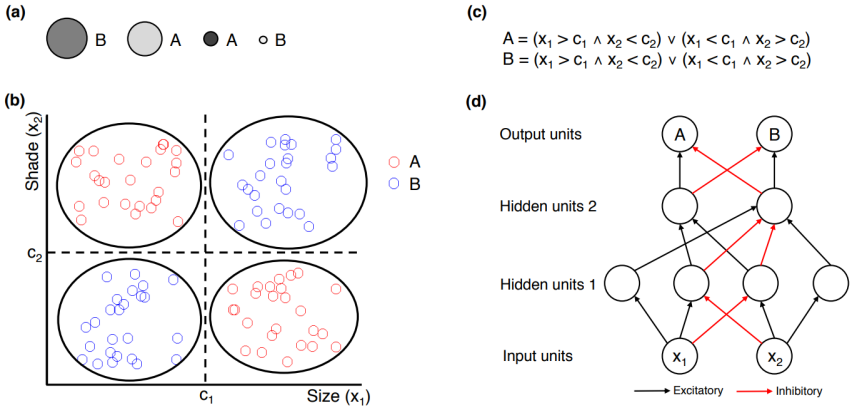


Figure 2.2: Illustration of the XOR classification task using size and shade: examples of stimuli in **(a)**. This problem can be solved by a spatial clustering method **(b)**, a logical Boolean method **(c)**, and an artificial neural network **(d)**.

2.2.1 Spatial methods

Spatial methods involve placing items in a **multidimensional space** and using their location to draw conclusions about **categorization**. Classification based on spatial methods can be determined by an item's location relative to a hyperplane or its similarity to different prototypes (means) or exemplars (centroids):

- **Prototype approaches** assume that learning is based on similarity to the center of a category (mean), which is stored after training (i.e. all examples are forgotten, the prototype only is stored).
- **Exemplar approaches** calculate similarity as a ratio between the similarity of an item to all items within a class (i...n) and the similarity of that item to all other items. This provides a fit per class and requires storing item-level information (i.e. all examples are kept).
- **Clustering** organizes items into groups (cohorts), with quality often quantified by the distance between items within and between clusters. Clustering can be either hard (each item belongs to only one cluster) or soft (items can have multiple memberships, potentially fuzzy).

Example of spatial method: Generalized Context Model (GCM)

According to the GCM, the probability that stimulus i is classified into category C_J is found by summing the similarity of i to all training exemplars of C_J and then dividing by the summed similarity of i to all training exemplars of all categories:

$$P(C_J|i) = \frac{\left(\sum_{j \in J} s_{ij}\right)^{\gamma}}{\sum_K \left(\sum_{k \in K} s_{ik}\right)^{\gamma}}$$

2.2.2 Logical methods and Artificial neural networks (ANNs)

Logical methods: concepts are based on a definition that is applied to the features of the object. One viable solution is searching for rules that maximize discrimination between stimuli; the rules can be probabilistic (i.e., the rules, if not given, can be automatically computed/learned).

ANNs: do not make assumptions about the representations involved, but offer an implementation method.

2.2.3 Thoughts

Asking which model is the most accurate might be misleading: the answer depends on what area of science you work in. Therefore it is better to focus on whether a model offers "useful explorations of the ways in which human learning operates". The value depends not just on match to human behavior, but whether there is a need to understand the underlying representations. We have to consider **not just accuracy**, but **also confusion**. Today, very large emphasis is put on **explainability**: we want to explain the model behaviors (for naturally explainable models). And if it is not naturally explainable, we need to address this issue.

Chapter 3

Psychology of concepts and categories

In this Chapter we discuss about categories in sensation/perception (non-semantic categories, Section 3.1) and conceptual structure (words and concepts, Section 3.2).

3.1 Categories and categorical perception

Categorical perception is the phenomenon in which people perceive stimuli from different categories as more different from each other than stimuli from within the same category. This is useful as it introduces invariance in response with respect to a functionally defined category, allowing for rapid prediction, efficient memory, and compression.

How we know categorical representation exists

A demonstrating experiment consists in:

- selecting a set of stimuli that uniformly covers a certain physical domain (e.g. sound freq 100Hz-8000Hz),
- select an objective distance measure so that the space is partitioned in intervals; e.g. distance in frequency space (applicable to both sounds and colors),
- select a method for operationalizing human similarity (e.g. similarity judgments, generalization, confusion [same/different]),
- in one procedure, assign all stimuli to categories (e.g. assign all stimuli to color names); in a second, obtain similarity judgments for within-category vs. between-category pairs, or ask for categorization, and evaluate if the boundary is fuzzy or not (e.g. how much objects are considered similar when belonging to the same category, and when to different categories)

3.1.1 Categorical perception in audition

In auditory stimuli, there is discrimination between speech sounds. People have a sharper discrimination boundary between sounds that are perceived as belonging to different phonetic categories than between sounds that are perceived as belonging to the same category. For experimenting, we use as objective dimension the **Voice Onset Time** (VOT) of consonants (i.e. the timespan between the start of the consonant and the start of sound emitted by vocal cords). The discrimination performance is simply a "same/different" judgment. We present consonants such as /b/ and /p/. A fixed-size physical difference in VOT, that is easily discriminated when it straddles the boundary between two categories (labeled as /b/ or /p/), produces *chance* discrimination performance when both tokens come from the same category (either both /b/ or both /p/); results are in Figure 3.1.
Within phonetic category, two **different stimuli sound the same** (see ◇ Eimas et al. (1971)).

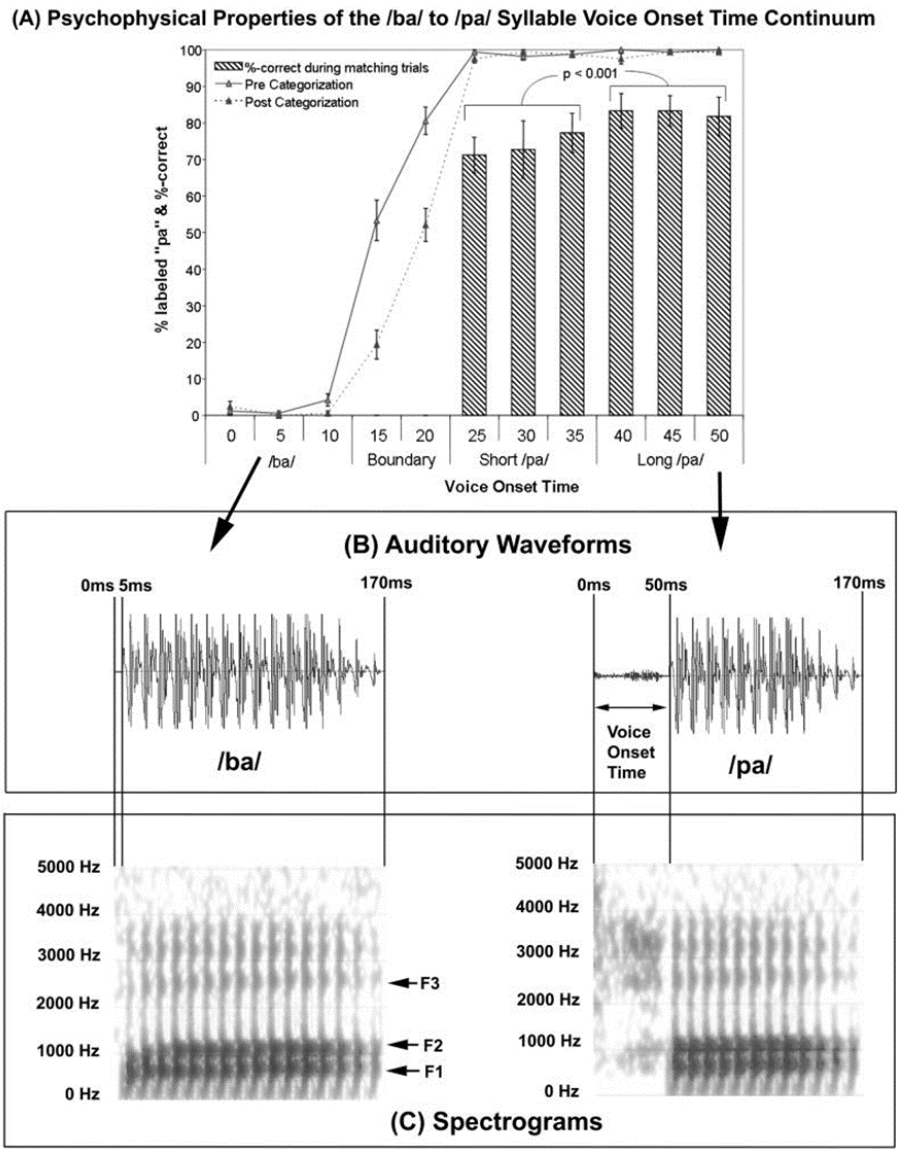


Figure 3.1: For the experiment, they created a range of stimuli with VOT between 5ms and 50ms (which are the standard values for respectively /ba/ and /pa/). In **(A)** are the results, with the percentage of responders hearing a /pa/ sound. We see the shift is between 10ms and 25ms

Prior categories aid online processing via categorical perception, in a sort of *experience-dependent learning*; related phenomena are:

- *Change deafness*, Vitevitch (2003): participants repeat words presented by a speaker. Halfway through study, the speaker changed. Only 40% of participants noticed the change.
- *Sine-wave speech*: phonetic categories/expectations can be considered *priors* on sounds, impacting whether a stimulus is perceived as speech (e.g. if you previously listen to a voice and then hear a sound which is not speech, but has some correlation to the previous voice, you can "hear" the words[1]).
- *McGurk Effect*: the categorization of sounds is not only an auditory task, since our brain combines multimodal inputs (e.g. also from vision[2]).

[1] `https://users.sussex.ac.uk/~cjd/SWS/`
[2] `https://www.youtube.com/watch?v=PWGeUztTkRA&t=49s`

◆ **Eimas et al. (1971)**

They observed 4-month old sucking rate on pacifier (notice that a higher rate is interpreted as more surprise/interest). They examined the rate as function of relation between current and previously heard stimuli, in particular they presented two stimuli with VOT differing by 20ms. In one condition (labeled "D") the difference straddled (on two sides of) the border of a phonetic boundary (stimuli perceived as "b" and "p" by adults). In another condition "S" they belonged to the same phonetic category.

See Figure 3.2 for more details.

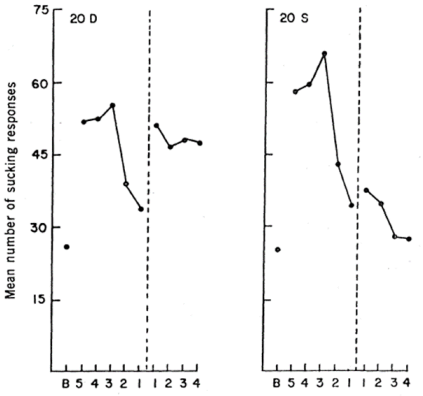
Brain areas coding for **low-level** representations are **not influenced** by such categorization, while those coding for high-level representations are.

3.1.2 Categorical perception in vision

In categorical perception for color, discrimination of items that cross category boundaries is better (faster, more accurate) than when the items are within the same color category. Notice that color category is **linguistic** ▲. For example, it is easier to distinguish between a green stimulus and a blue stimulus than between two stimuli within the same category (two shades of green), who are **spaced at the same distance**.

▲ Practical note: color differences in terms of discriminability can be equated across between-category and within-category comparisons by using the *Commision Internationale de L'Eclairage* (CIE) values.

Color categories are not universal, and thus **categorical perception depends on language** (see ◇ Robertson et al. (2000)).



(a) (b)

Figure 3.2: 5 min of habituation precede a 20ms VOT change, either within the same **(a)** or across **(b)** phonetic category; habituation consists in hearing the /ba/ sound. This proves how 4-month-old children already have the auditory **categorization** enabling them to distinguish between /b/ and /p/ sounds.

◆ **Robertson et al. (2000)**

The stone-age tribe Berinmo uses "*nol*" as the color name that in English falls under both green and blue, so they have no categorical perception at the boundary between green and blue (no boundary in their language). On the contrary, they have a category boundary between "*nol*" and "*wor*" that does not exist in English as both sides are green. Berinmo people exhibit better discrimination of 32 cross-category items than 32 within-category items at the boundary between *nol* and *wor*. English speakers do not show categorical perception at this boundary.

3.2 Conceptual structure

3.2.1 Theory on words and concepts

The typical approach is to assume that words are associated with concepts or a network of conceptual representations. We need to first have a good theory of what the conceptual structure is like, and then we can see how this structure is used to represent meaning when referred to by words. Ultimately, a word is a sound or written pattern, and it is generally assumed that a single word corresponds to a single concept (this assumption would help with word-embedding models), but there are complications.

There is evidence proving that *form-to-meaning mapping* is not *1-to-1*. We encounter **polysemy**: a phenomenon where a single word can have multiple meanings depending on the context in which it is used, such as "cinema" which can refer to different things in different contexts (e.g. *American cinema is naïve* vs *This cinema is ugly*).

According to Murphy, it is impossible for a single concept to fully capture the meaning of a polysemous word; the process of **meaning extension** is not a result of *natural change* (chaining) but a matter of **online derivation**. Online derivation does not prevent information from also being stored somewhere, at least for a short time, in order to understand the nearby context more effectively. Indeed, Klein and Murphy (2001) found evidence suggesting that the different senses of polysemous words can be stored, as they observed priming effects when a word was used twice in the same sense, and interference effects when the sense was switched.
Another possibility is that a word specifies a set of potentials that is then refined by context to determine which sense is intended.

To put it in a nutshell: **words are not concepts**. Such distinction between *meaning* and *lexical form* is also proven by *anomia*, which is a type of *aphasia* that results in the inability to retrieve the lexical form of a concept for production, even though the ability to recognize or define the term is maintained.

3.2.2 Focusing on conceptual structure

Understanding word meaning requires understanding the organization of conceptual structure in the mind-brain. Concepts represent our knowledge of things in the world and enable us to identify things, infer features, or know how to interact with them. Access to conceptual structure can be achieved through various means, including words, pictures, or music ▲. The study of conceptual structure or "semantic memory" should be independent of what governs word meaning or how word meanings are accessed through language. Much of this psychology work focuses on how people represent categories, with classical experiments focused on artificial, non-linguistic categories to see how people "summarize" those. We will not get into that, but be aware of the link. In much of the literature, concepts are studied by using words, assuming that words map onto conceptual knowledge and are a reasonable "proxy" for studying relations between concepts.

- ▲ Access to conceptual structure is more rapid when people are presented an image, than when presented a word (for the same concept). There are studies showing that music also influences the time of access to conceptual structures.

The classic view of word meaning

In the classic view *word* and *concept* are the same, and the word's meaning is a **definition**. Definitions are sets of necessary conditions that are jointly sufficient: every object is **either within a category or not** (e.g. *Bachelor = Man, unmarried*). This classical view of *concept structure* (word meaning) implies that there are no different levels of category membership (all members are equivalent to others). It has the advantage of supporting hierarchical structure and inheritance.

3.2.3 Representing the meaning of concepts in the brain

We first introduce **propositional networks**. A concept has a definition (e.g. *Dog: a member of the canine species, that is domesticated*), it can inherit the default features of the parent class (*canine*) and add a few exceptions if non-default. Then the parent classes in the taxonomy are defined (*canine*, then *mammal*, *vertebrate*, *animal*, *organism*, etc). At the end, some reference to visual features is added.
An example of representation in memory is provided in Figure 3.3, while Figure 3.4 shows a propositional network.

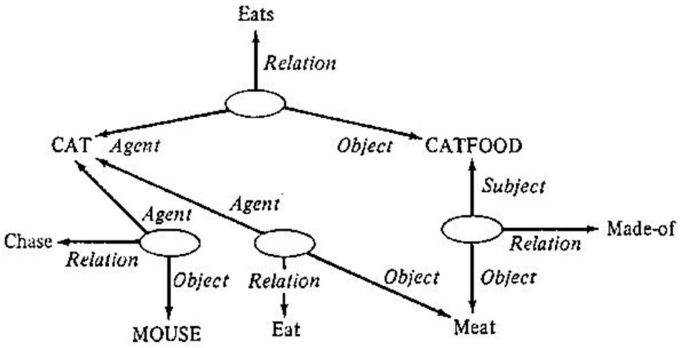


Figure 3.3: A partial representation of "cat" in memory.

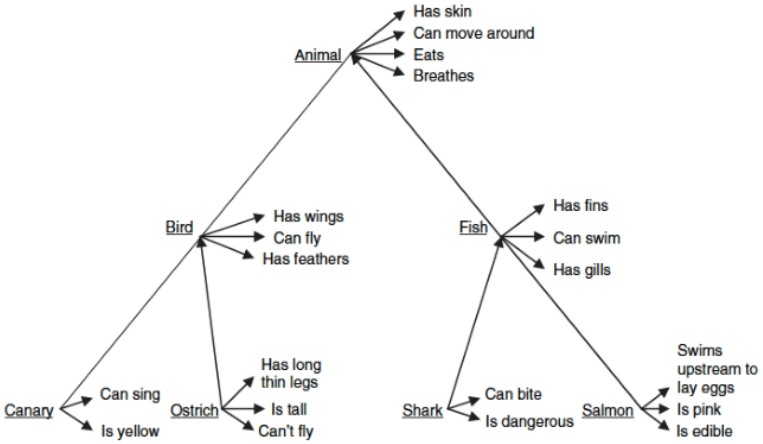


Figure 3.4: Propositional network used by Collins and Quillian (1969).

Sentence	Latency
Robins *eat worms*	1310 ms
Robins *have feathers*	1380 ms
Robins *have skin*	1470 ms

Table 3.1: Verification time of affirmative responses. Even though the difference in timing seems to be small, it actually is massive (there are brain processes taking only 5 ms, so 70 ms is a lot). To measure the latency, EEG was used.

Propositional networks as psych model

Collins and Quillian (1969) study of conceptual structure involved **measuring response latencies for statements** such as "Robins *eat worms*/*have feathers*/*have skin*". The study proposed that **some features of a concept are directly stored** while **others are inferred through inheritance**. For example, the feature "have skin" can be inferred after traversing three links in the ISA hierarchy: "is bird", "is animal", "animals have skin". On the other hand, "have feathers" can be inferred from a lower level in the hierarchy. The latencies of affirmative responses were expected to reflect this distinction (see the results in Table 3.1). One **question** that the study raises is whether these hierarchies are **pre-represented in memory** or whether they are **formed *on the fly*** when people are asked questions.

Propositional networks, however, present some problems:

- **Verification time is not solely determined by hierarchical relations**, but is **also influenced by the frequency of encountered statements**. For instance, "apples are eaten" is verified faster than "apples have dark seeds."
- The propositional network approach suggests that verification times should increase as the number of ISA links traveled increases. However, people are **faster to verify** "**dogs are animals**" **than** "**dogs are mammals**".
- The approach expects **clear logical inferences**, such as "if A is a B and B is a C, then A is a C." But this **assumption has been challenged**. For example, while a car seat is a seat and people agree that seats are furniture, car seats are not considered furniture.

Prototypes

Rosch (1973) proposed an alternative to propositional networks. He argued against the idea of "sets of necessary and sufficient features", taking the example of "game" (there is no set of necessary and sufficient conditions that is in common to all games. Team? Yes/no. Physical skill? Yes/no etc). Instead, what makes all games "games" is their **family-resemblance** to each other. The **prototypical members are highly similar to other members within the category but less similar to members of other categories** (in a sort of natural clustering, see ◇ Rosch and Mervis (1975)).

Rosch's refutation of "classic view" in psychology has empirical support, as **typicality effects** (i.e. typical reactions of people to stimuli) are evident in:

- **verification times** (e.g. *Robins are birds* faster to verify than *Chicken are birds*);
- **ratings** (people are very systematic in rating whether a certain member is a typical member of a category);
- **generation tasks** when listing members (e.g. for "birds" many more people list *robins* than *chicken*, and for "sports" many more list *soccer* than *weightlifting*).

In conclusions **categories have central and peripheral members**.

◆ Rosch and Mervis (1975)
Family resemblances: Studies in the internal structure of categories

The underlying idea of their study: a prototype of a category is a (hypothetical) member of a category for which all the values are the default or most popular. They first collected feature listing for various items (in different categories), then asked people to rate *how typical* items were of categories, and counted how many category members tended to share features listed. Finally, they analysed **correlation**, finding items that had more features in common with other items in the category were, independently, rated as more typical of the category (they cross-referenced the typicality ratings against the number of shared features). The results are in Table 3.2.

Category	Most typical members	Least typical members
Furniture	13	2
Vehicle	36	2
Fruit	16	0
Weapon	9	0
Vegetable	3	0
Clothing	21	0

Table 3.2: Number of attributes in common to five most and five least prototypical members of six categories.

Fuziness and levels of abstraction

Propositions also have another problem: **fuzzy category boundaries**. Natural **categories do not necessarily have fixed boundaries**. Mcloskey and Glucksberg (1978) showed that people's judgments about category membership differ with typicality (e.g. all agree that *cancer* is a disease and *happiness* is not, opinions differ on *stroke*). Category membership is related to typicality. Edge cases exist, and **people disagree**. Moreover, **people are unsure** (e.g. when asked the same question after one month, 11/30 reverse on "stroke is a disease").

These observations raise a question: Why do we end up with the words we have? We introduce the concept of **levels of abstraction and informativeness**.
When presented an image of a cat and asked what they see, some people will answer *a cat*, some other *a Siamese cat*, or even *an animal*. People likely represent the world at different levels of granularity:

- **Superordinate** level (*animal*)
- **Basic** level (*cat*)
- **Subordinate** level (*Siamese cat*)

People will use the level of abstraction corresponding to the level of discrimination they need. **Informativeness** (i.e. the amount of facts linked to the category) and **distinctiveness** (the extent to which a category differs from other categories at the level) is what makes them useful. The superordinate level lacks informativeness but is distinctive, the opposite of the subordinate level. The basic level has both informativeness and distinctiveness. We do not create a subordinate concept for each item due to reasons of cognitive economy: they maximize information, but without offering much distinction.

Chapter 4

Modeling conceptual organization

4.1 Modeling typicality

Deep Neural Networks Predict Category Typicality Ratings for Images

Lake et al. (2015)

They evaluate deep convolutional networks trained for classification on their ability to **predict category typicality** (human typicality ratings), and try to understand whether deep learning systems can serve as potential cognitive models.

4.1.1 Background

The motivation is that, for any task that requires relating an item to its category, **typicality will influence performance**, whether it is the speed of categorization, ease of production, ease of learning, usefulness for inductive inference, or word order in language.
CNNs learn categorization, but **perhaps they categorize by learning prototypes**, i.e., they produce representations that track categorical structure with typicality structure.

4.1.2 Methods

They asked people to rate a collection of images for category typicality (images drawn from 8 image categories), and tested different CNN architectures on their ability to predict these ratings.

Behavioral experiment

Each participant rates "how well does this picture fit your idea or image of the category". Mean typicality per image is computed across all splits. Human reliability ratings have good split-half correlation: the average reliability of human ratings across random splits is $\rho = 0.92$. This confirms that two groups of people produce similar rankings ▲.

- ▲ This is required to know if we can trust human ratings (i.e. *is human behaviour reliable enough to predict itself?*). If we could not trust human data, there would be no point in aligning the model output to human ratings.

Computational experiment

They use 3 CNN architectures (but for the sake of simplicity *OverDeat* only is described). After the convolutions, the next two layers have 3072 and 4096 fully-connected units, respectively. Finally, a 1000-way softmax layer produces a probability distribution over the $j = 1, \ldots, 1000$ classes. They get a top-five error rate of 14.2% ▲.

- ▲ Top-five error: the correct label did not appear in the top five guesses.

4.1.3 Estimating image-typicality

They assume that (human) **typicality is related to the strength of the model's classification response** to the category of interest. The **classification strength** can be estimated in two ways:

- **Raw typicality**: This is a raw category score. There is a theoretical vector (input of last layer) which **maximizes the activation** of a particular category. Images with representation very close to this theoretical vector are expected to be more typical. So we can maximize y_j to get the particular abstract representation for each category j:

$$y_j = \sum_{i=1}^{4096} w_{ij} x_i$$

- **Contrast typicality**: It measures to what extent the correct category is more active than the others. It benefits images that load on the correct category much more than on other ones. The most typical image produces y_j that is most differentiated from other categories' response to this image. This is independent from the raw value:

$$z_j = \frac{e^{y_j}}{\sum_{j=1}^{1000} e^{y_j}}$$

The values computed for each j are averaged, then they check for correlation with human ratings.

4.1.4 Results

They found that raw and contrast scores do similarly well, and that some models have significant human-machine typicality correlation. This suggests that deep **CNNs learn graded categories that can predict human typicality ratings**, at least for some types of everyday categories.

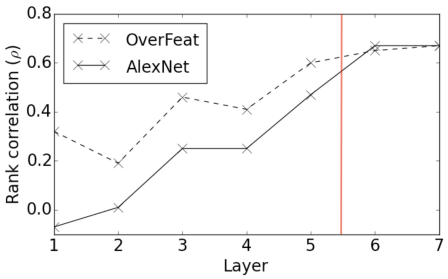


Figure 4.1: Correlation between human and convnet typicality ratings as a function of network depth. The red line indicates a transition from convolutional (1-5) to standard layers (6-7).

However, when dealing with hidden layers, one cannot have categories' activations, so they needed to redefine typicality. They used 1300 images (of the same category) as input for the network and averaged the activation of the given layer over all images to get the **category prototype** (typical activation vector). Typicality was modeled as the cosine distance between the activation vector for a new image and the stored prototype. They found better prediction in deeper convolutional layers (see Figure 4.1), i.e., by going deeper (closer to the output layer) the layer representations predict increasingly better human typicality.

4.2 Words-as-features as models of cognition

Predicting Human Brain Activity Associated with the Meanings of Nouns

Mitchell et al. (2008)

They try to answer this question: Can feature models explain behavioral and brain responses? The underlying idea is to understand how/where word meaning is stored in the brain, under the assumption that our brain represents words as features. Notice: this assumption is not necessarily true, there might be also other possibilities (not tackled in this paper).

4.2.1 Background

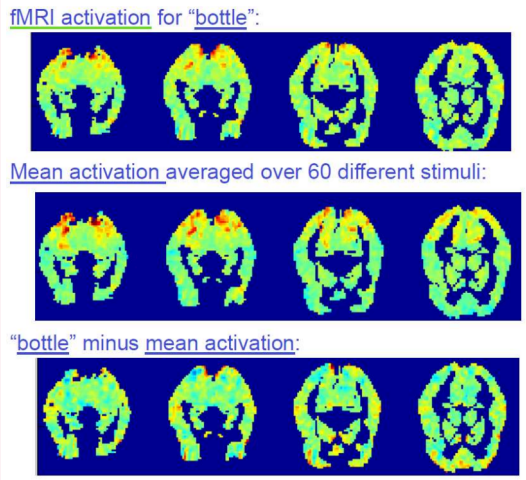
Core questions for neuroscience are:

- Are there systematic differences in neural activity as people think about different concepts?
- Is the neural representation of concepts localized in specific brain areas or is it distributed across the entire cortex?
- How meaningful are individual differences, or is the *representation of meaning* similar across people?

We ask ourselves if fMRI and neuroscience allow us to *test* or *understand* what are the **basis functions** (the **semantic feature** space) that underlie the representation of words. This can help in designing more cognitive real computational models.

Historical approach

We present multiple words sampled from several categories (e.g. tools, buildings), and then train a classifier that predicts the class (tool/building) from the brain images of the words. Brain images are taken with fMRI: see the figure, which shows activation maps from fMRI data (each of the 4 images in a row represent a slice of the brain). A classifier can be trained using a single voxel. The results of the classifier can be used as a tool for studying the semantics in the brain. For instance, we can understand which brain area contains information about particular classes. This is a pure *decoder*; there is no domain-based knowledge that is applied to predict the brain response from more basic principles.



Classifiers capture some meaning, as they show **cross-domain generalization** (train on words, guess class of image). They collected brain activity while people watched an image (instead of the word), and while Portuguese (instead of English) people watched the same words. They then took the classifier trained to discriminate categories based on brain responses to words presented in English and tested on brain activity from those other domains. Both *testing on pictures* and *testing on other language* produce above chance accuracy: **semantics generalize beyond modality used**.

The historical (decoding) approach works, but has a problem: **data is highly dimensional** (over 20 thousand features/voxels per word). At the same time, however, **data is also sparse** (only few examples of brain activity per category). This is difficult from a regression perspective and solutions to this (regularization) are mathematically valid but may lose information about the brain. These considerations sparked Mitchell et al. (2008) to propose an alternative.

4.2.2 Generative encoding model

Their proposal is to **come up with a theory of word meaning and see whether the theory predicts brain activity** (*activation*). They capture "word meaning" from corpus statistics (from mutual constraints appearing in corpora). The challenge is basically to find a "mapping function" from word meaning to brain activity.
The model works in two steps. The first step encodes the stimulus word. The second step predicts the neural fMRI activation at every voxel location in the brain, as a weighted sum of neural activations contributed by each of the intermediate semantic features. A schematic explanation can be seen in Figure 4.3.

Semantic feature values: "**celery**"
0.8368, eat
0.3461, taste
0.3153, fill
0.2430, see
0.1145, clean
0.0600, open
0.0586, smell
0.0286, touch
…
…
0.0000, drive
0.0000, wear
0.0000, lift
0.0000, break
0.0000, ride

Figure 4.2: Example of feature vector.

Define word meaning

We mentioned that they captured word meanings via corpora exploitation. They use 60 **nouns**. Each noun j is described using 25 **features**.

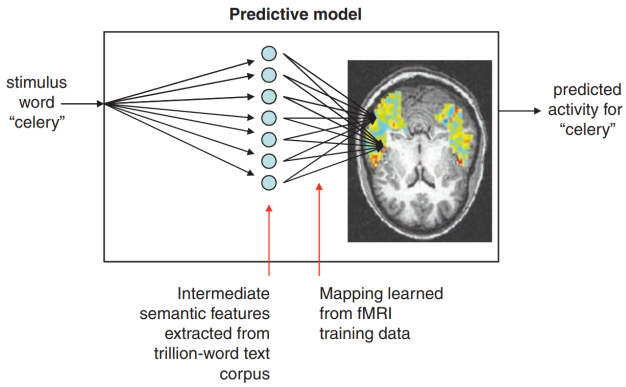


Figure 4.3: The predictive encoding model. Each word gets a vector of n features (in this simplified schema just 7); the value of n has to be chosen as it is a hyper-parameter. Note: there is no single step prediction for the whole brain. The prediction is done for a single pixel, i.e., we get a weight vector for each voxel. Then all pixels are put together to get the whole map.

Feature i is defined as the **co-occurrence frequency of the stimulus noun with the verb** i. They chose verbs which are either: *sensory*, *motor*, or *abstract* verbs. An example of the resulting semantic features is provided in Figure 4.2.

Single-voxel analysis

They perform a single-voxel analysis. For each voxel in the brain they learn the relation between voxel activity values for the 60 nouns, and the semantic features of the 60 nouns ▲.
Such analysis can tell, **for each voxel**, **what is the relative importance of each of the 25 features** when it comes to predicting brain activation. To show generalization, they train the model with 58 words and test on the remaining 2.
In matrix notation the multiple regression model is $\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$, where:

$$\mathbf{Y} = \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{bmatrix}, \quad \boldsymbol{\varepsilon} = \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_n \end{bmatrix}, \quad \boldsymbol{\beta} = \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_k \end{bmatrix}, \quad \mathbf{X} = \begin{bmatrix} 1 & X_{11} & \dots & X_{1k} \\ 1 & X_{21} & \dots & X_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & X_{n1} & \dots & X_{nk} \end{bmatrix}$$

$\mathbf{Y}$ is the activity for the 58 nouns in the voxel; $\boldsymbol{\varepsilon}$ is the bias; $\boldsymbol{\beta}$ are the 25 weights to be fit; $\mathbf{X}$ is the set of 25 feature-value per noun ($n = 25, k = 58$).

▲ To be precise, they present stimuli of noun+image together (but for the sake of simplicity we refer to them just as "nouns").

Predicting word activity in each voxel just means multiplying its semantic feature values by learned weights (Figure 4.4).

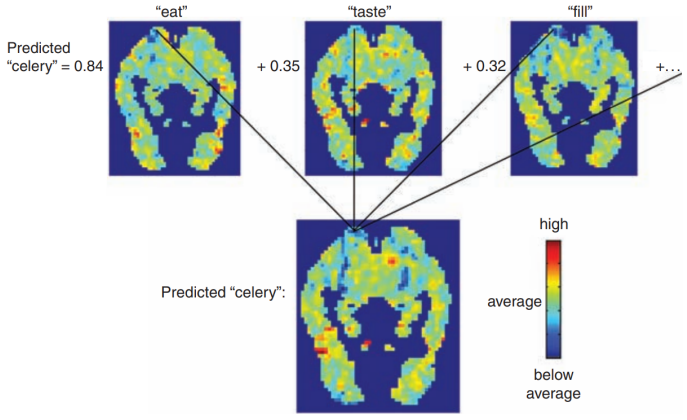
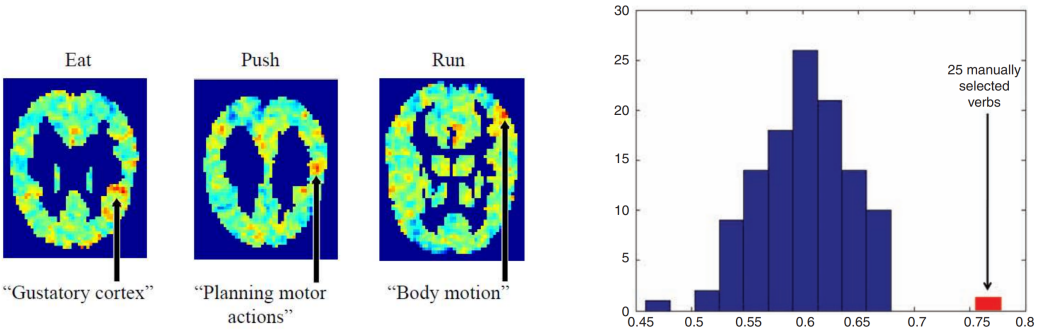


Figure 4.4: Predicting fMRI image for given word.

4.2.3 Results

They tested the model by predicting activity for word A, and checking if it is more similar to true activity of word A than it is to the other word B (where A and B are the two left over words, not used for training). They got an average accuracy of 0.79, suggesting that word meanings are indeed **represented as features in the brain**.

They also examined, for each of the 25 features (verbs), the importance across the brain, to obtain a map of which brain areas are the most important for a given verb (and therefore for a given activity, see Figure 4.5a).



(a) The three images show the weight magnitude of each voxel, given a verb (notice different fMRI slices are shown for different verbs).

(b) Mean accuracy over 9 participants, using randomly selected sets of intermediate semantic features vs. using the manually chosen set.

Figure 4.5: Results.

Furthermore, they performed a *by participant* analysis that allowed them to account for inter-individual differences. They split the brain into areas and swept through a very large number of words (10 thousands) to see **which word maximally activates that region** (or the entire brain). They found there are indeed words that are most activating for specific brain areas.

In the end, they experimented with **random** *25-feature-basis* sets, instead of a manually chosen set. They tried with 115 randomly selected sets (composed not only by verbs), finding that the results are much worse, yet with **significant accuracy** (> 0.61, see Figure 4.5b).

4.3 Studying representations via similarity spaces

Matching categorical object representations in inferior temporal cortex of man and monkey
Kriegeskorte et al. (2008)

In this section we see how they model similarity spaces expressed in human behavior and brain responses.

4.3.1 The principles of Representational Similarity Analysis (RSA)

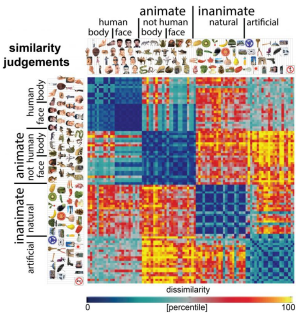
To compare if two clustering representations are similar, a possibility is to pick single elements, and check if the neighbors that are part of the same cluster in a representation, are part of the same cluster also in the other representation. This idea coming from statistics is exploited in RSA. RSA relates modalities of human behavior (or brain activity measurement) and information processing models by **comparing activity-pattern dissimilarity matrices**. A single similarity-matrix captures *first-order* similarity between stimuli (either similarity in brain response, or similarity as computed by a model). **RSA is a 2nd order similarity** because it quantifies how alike two similarity-matrices are. RSA:

- is modality independent: it allows to compare completely different modalities, provided we can measure similarity or distance between pairs of stimuli;
- can relate whatever modality of brain or behavioral measurement to information processing models;
- is based on the notion of similarity, or distance, between stimuli.

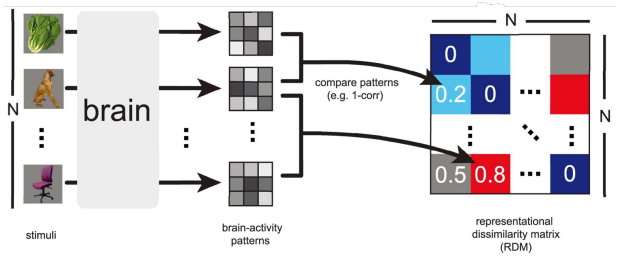
4.3.2 Applied contexts for RSA

A **Representational Dissimilarity Matrix** (**RDM**) of **human behavior** is shown in Figure 4.6a. We can see how objects in the same category are judged to be similar (as expected). Such matrix can be populated either by directly asking pairwise similarity to people, or via item sorting tasks: given randomly placed objects, people are required to sort objects in an array with similar objects close one to each other.

An RDM can also be constructed on **brain data**. Figure 4.6b shows how it is populated: we consider how much each group of voxels is related to input stimuli. We can then compare the similarity matrices as in Figure 4.7.



(a) Example of a single RDM from human similarity judgements.



(b) Construction process of an RDM from brain data. For convenience, just 9 groups of voxels are represented.

Figure 4.6: Examples of RDMs.

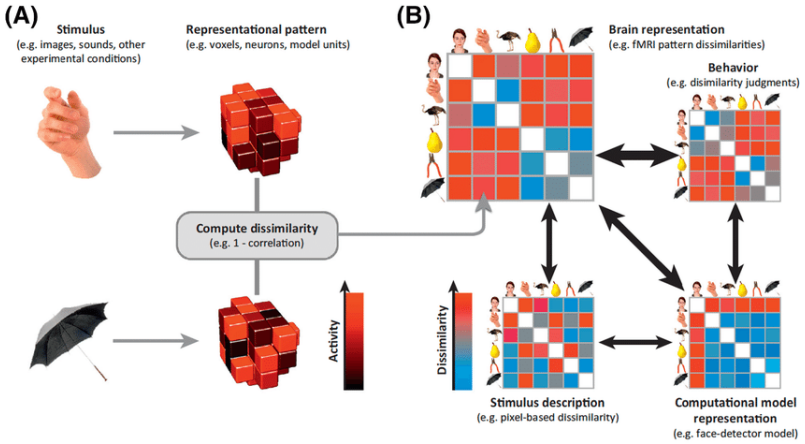


Figure 4.7: Example of comparing RDMs, with 2 stimuli and brain response. **(A)** First-order RSA: differences between patterns of activity in a chunk of tissue responding to two objects, here a hand and an umbrella, populate one cell of an RDM in (B). **(B)** A complete RDM can now be compared using second-order RSA with other RDMs constructed from behavior, input measures, or other models.

Another possibility (Figure 4.8) is to understand what information is coded over time, so that we can check if there is similarity between representations in the brain and in a neural network at the same depth (e.g. comparing shallow layers of the brain with shallow layers of a CNN). MEG data can be split along the time domain into intervals. For each interval, a similarity matrix is computed correlating the activity between images (note that the matrices are not the same, as representation in the brain changes over time). With fMRI data we can do another thing: relate different modalities (different brain areas), discovering that Extrastriate and Inferior temporal fit with the NN.

Moreover, probing for 2^{nd} order similarity across brain regions (not against a model) allows to relate brain and behavior, find areas that code similarly for different stimuli across participants or even species. It also allows to code a single set of stimulus across multiple dimensions and code RDMs at each feature level.

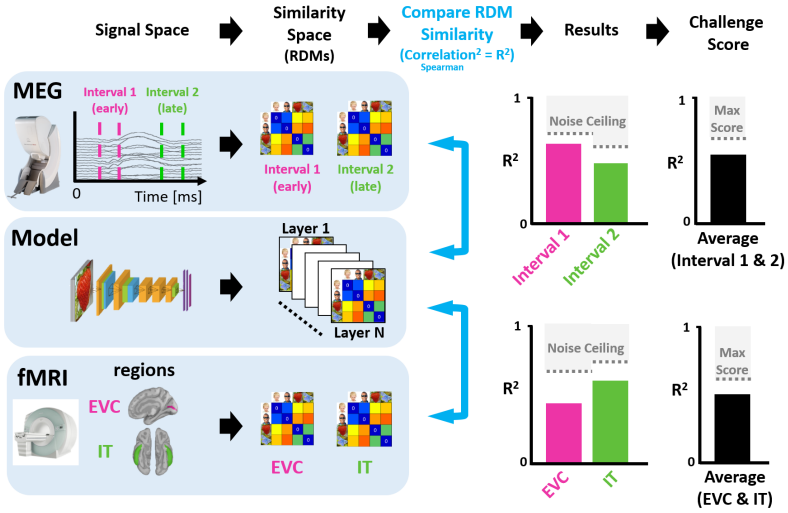


Figure 4.8: Example of comparing similarity matrices. The Noise ceiling is the maximal (ceiling) value expected given the noise in the data. Oftentimes the noise ceiling is estimated as the correlation between the estimates of the responses in two independent repetitions of the same experimental procedure. The idea is that the ability of X to predict Y cannot exceed the noise ceiling, defined as the correlation between Ys (Y1 and Y2) obtained for the same stimuli on 2 different test data.

4.3.3 RSA and loss of dimensions

When using Human Similarity Judgments, we create an RDM directly from those judgments (we do not have the features). When using NeuroBio data to produce RDMs we use an $S \times V$ (stimuli/observation $\times$ voxels/sensors/regions) matrix. When using Computational Models to produce RDMs we use an $S \times F$ (stimuli/observation $\times$ features) matrix. When relating NeuroBio and Computational models the $[S \times V]$ and $[S \times F]$ matrices are first converted to RDMs. Consider you can get the exact same RDM from different $S \times F$ matrices that differ massively on the number of F. This means that when we convert to RDMs **we do not have specific information on dimensions that produce the alignment**. **This is the main drawback of RSA**. For instance, in the case of Mitchell et al. (2008), we get an RDM with shape $[60 \times 60]$, so we lose the dimension of the 25 verbs used to compute word representations. However, the main problems of methods that keep information of the features are that they are unstable and very complicated to understand or to implement.

When dealing with 2 domains (brain, model) represented as observation $\times$ feature matrices, and when the two matrices reflect the same feature, we could evaluate the fit directly at the matrix level. There are many different techniques that probe for strength of common dimensions between two matrices.

Digression

The following techniques all probe for strength of common dimensions between two matrices:

- Procrustes rotation: it takes N objects in D features, and tries to find a transformation to map one into the other, see image;
- Principal component regression (i.e., supervised PCA);
- Partial least squares correlation: similar to PCA, but it tries to maximize the correlation on both tables;
- Canonical correlation analysis.

4.3.4 Univariate and multivariate approaches

In the following we consider already seen topics, but from a different perspective.

Information contained in multiple voxels

The typical approach of fMRI is massively multivariate, since it considers one voxel at a time. The problem is that information contained in multiple voxels is lost.
For this reason they studied brain responses during narrative comprehension: some participants were required to focus on space, others on time, others on actions, hearing the same story. They considered each IFG sub-region as a "voxel". With an univariate approach, regional activity in *pars opercularis* (see Figure 4.9) was higher for some conditions, while *pars triangularis* responded always with the same level of activation for the different dimensions. In a multivariate approach, they considered the entire set of values in each region, and quantified how similar those activity patterns were for the three conditions. Instead of considering the whole area (*pars triangularis*) by averaging, they kept voxels separated and found very different activations when focusing on different dimensions.

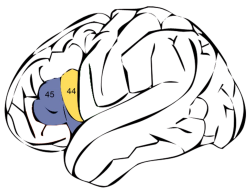


Figure 4.9: Pars opercularis (yellow) and pars triangularis (blue).

Decoding category (binary case) from brain

A multi voxel pattern analysis (MVPA) can be carried out. Two conditions are presented, which produce different distributions of activity across trials. In Case 1, each condition produces different activity levels, in both Voxel 1 and Voxel 2. Clearly, the region discriminates the classes. In Case 2, each condition produces highly similar mean activity levels in both Voxel 1 and Voxel 2. So one might conclude that the region does not discriminate, if aggregating across univariate analysis. But the multivariate analysis leads to a completely different result, allowing us to better understand: the region containing voxels 1 and 2 **contains information about conditions in the joint distribution** of Voxel 1 and Voxel 2.

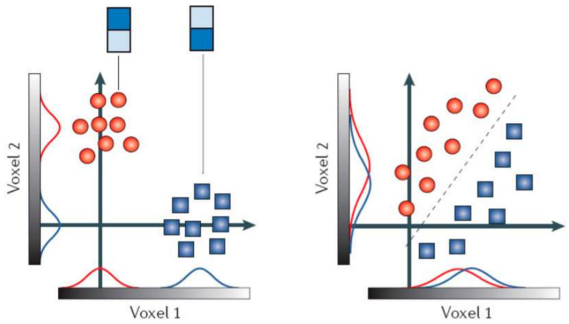


Figure 4.10: Case 1 on the left, Case 2 on the right. Each trial is represented as a circle or a square (depending on its category).

Chapter 5

Modeling human representational geometry

From a historical point of view, the success of RSA brought to many studies on ablation, plasticity, etc. Many researchers used it, but they started questioning whether it is a good model of human knowledge. In the following we are discussing different assumptions, in respect to artificial DNN.

5.1 Background

We have already seen in Chapter 3 the most accepted representation of concepts, as categories, in psychology: semantic domains or categories (e.g. mammals, animals, dogs) are organized via features or dimensions that carry the relevant variance for the category (Classical view, Rosch and Mervis (1975), Lake et al. (2015)). We have also seen how we can have typicality effects in AI: entities (e.g. images, words) are described by feature values from which representational category effects emerge.
We can use AI systems as models of semantics (Section 4.3): AI systems trained for image categorization or word embedding produce representations that reasonably approximate those of humans. Similarity between categories, as operationalized from human data, is well predicted by distances between objects in the AI model, where human similarity is quantified using brain/behavior and model similarity is quantified via Euclidean distances, inverse cosine, etc. For prediction of human similarity judgments on images, they found 20 to 60% of the judgments is modeled by the NN. While in MVPA the results are more complicated: these can predict activation in brain areas, but the correlation is just barely above the significance threshold.

5.2 AI modeling of human representations

Modeling human representations with AI is useful for many fields:

- Psychology: AI models achieve human-like competence on different tasks. Because of their competence, they offer a **model of potential knowledge organization**. They can also offer **interpretability of human behaviour**: if we have an AI model that behaves as humans, we can study it to come up with valid hypotheses.
- Engineering: better prediction of human behavior, and improved AI-human alignment. For instance, to evaluate how good generated images are, similarity metrics are used. However this does not take into account the subjectivity of human brain.
- Computer science: understanding representations in neural networks.

There are three approaches:

- **Default**: **use all DNN features** (all in the whole network, or all in a particular layer) as object-representation for modeling human data. This implicitly assumes that all features are relevant and equally important, for the alignment, for all concepts.
- **Reweighting**: **keep all features, but adjust the weights by finetuning**. It addresses mis-calibrated, human-relevant features by applying concept-specific adjustment of feature

saliency for modeling human representations. This assumes that AI learns human-relevant features, but these are mis-calibrated.

- **Pruning**: **keep the weights unchanged but remove some features**, investigating modular structures in AI models. This assumes that the network develops a modular structure where information about different categories is represented in different subspaces (subsets of latent dimensions) within the model, i.e., knowledge about particular categories or concepts is stored in particular areas ("modules") in the AI model.

5.2.1 Default approach

Assuming two objects, U and V, each with 3 features, **human similarity** can be defined using the **inner product** (or a related quantity), and approximated as follows:

$$Similarity(V, U) = V \cdot U = V_1 \cdot U_1 + V_2 \cdot U_2 + V_3 \cdot U_3$$

Note this is just an evaluation (no learning involved).

5.2.2 Reweighting

The similarity is defined as the **weighted inner product** (or related measure), where the weights ($W_{1,2,3}$) are learned via regression and evaluated on out-of-sample data.

$$Similarity(V, U) = W_1 \cdot V_1 \cdot U_1 + W_2 \cdot V_2 \cdot U_2 + W_3 \cdot V_3 \cdot U_3$$

This **involves learning** (via linear regression or other techniques). Learning the weights is far from easy, since the number of free parameters is extremely huge (regularization is needed), as weight solutions are category-specific. ◇ Peterson et al. (2018) proved that this approach generalizes well, outperforming the baseline (default approach), despite this being already high.

◆ **Peterson et al. (2018)**
Evaluating (and Improving) the Correspondence Between Deep Neural Networks and Human Representations

This study explores how well the representations discovered by DNNs align with human psychological representations of natural images, shows how they can be adjusted to increase this correspondence, and demonstrates that the resulting representations can be used to predict complex human behaviors such as learning novel categories.
Reweighting starts like RSA, by comparing the representations formed by deep neural networks to those of humans, i.e., similarity judgments (accordingly, the first experiment of the paper focuses on this comparison). Notice that a similarity function over a set of pairs of data points corresponds to an implicit representation of those points.

Firstly, they evaluate the performance of deep neural networks in predicting human similarity judgments for 6 categories of images (120 images each). They collect the human judgements as pairwise image similarity ratings (within each category) from human participants on Amazon Mechanical Turk, using a scale from 0 ("not similar at all") to 10 ("very similar"). The result is six 120×120 similarity matrices after averaging over individual judgments. They collect the activations of a DNN (they experiment with several models, here are the results of VGG only) in a feature matrix, with an image per row, that therefore has shape $[120 \times 4096]$. A similarity matrix $\mathbf{S}$, in which the entry s_{ij} gives the human similarity judgements between images i and j, can then be approximated by the matrix product $\mathbf{F}\mathbf{F}^T$:

$$\mathbf{S} = \mathbf{F}\mathbf{F}^T$$

They assess the model performance in predicting human similarity judgments by computing the correlation between $\mathbf{S}$ and $\mathbf{F}\mathbf{F}^T$, finding that the **raw deep representations provide a reasonable first approximation to human similarity judgments**.

To better understand how DNNs succeed and fail to reproduce the structure of psychological representations, they applied two classic psychological tools: *non-metric multidimensional scaling*, which converts similarities into a spatial representation (from relative distances of n elements to a map of relative distances in 2 dimensions), and *hierarchical clustering*, which produces a tree structure (dendrogram). They find that human representations exhibit highly distinguished clusters in the spatial projections and intuitive taxonomic structure in the dendrograms, neither of which is present in the DNN representations.

They procede exploring how DNN representations can be transformed to increase their alignment with psychological representations. They augment the model of similarity judgments with a set of weights on the features used to compute similarity:

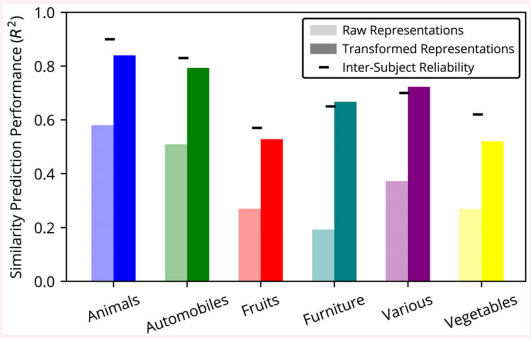
$$\mathbf{S} = \mathbf{FWF}^T$$

where $\mathbf{W}$ is a diagonal matrix of dimension weights. the diagonal of $\mathbf{W}$, the vector of weights $\mathbf{w}$, can be expressed as the solution to a linear regression problem where the predictors for each similarity s_{ij} are the (elementwise) product of the values of each feature for objects i and j (i.e., each row of the regression design matrix X can be written as $\mathbf{F}_i \circ \mathbf{F}_j$, where $\circ$ is the Hadamard product). The similarity s_{ij} between objects i and j is therefore modeled as

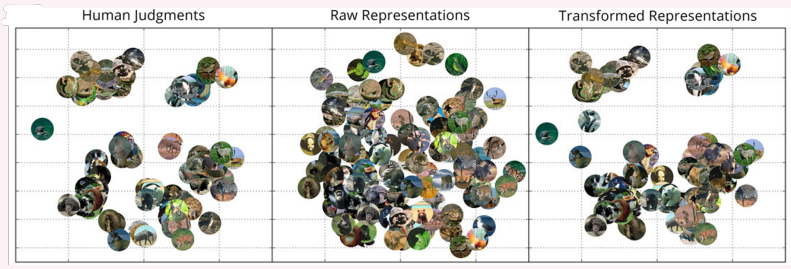
$$s_{ij} = \sum_k w_k f_{ik} f_{jk}$$

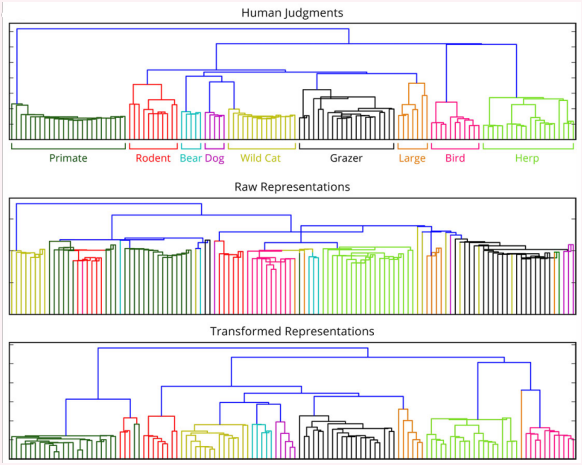
where f_{ik} is the k^{th} feature of image i and w_k is its weight.
Freely identifying the $\mathbf{w}$ that best predicts human similarity judgments runs the risk of over-fitting, since the DNN generates thousands of features. To address this, they use L2 regularization on $\mathbf{w}$, penalizing models for which the inner product $\mathbf{w}^T\mathbf{w}$ is large.

The **new representations that emerge explain nearly twice the variance compared to raw representations** on all domains:



The stricter case of cross-validation titled "CV control" explains the similarity judgements better than raw representations. In this case no single images occurred in both training folds and test folds of cross-validation. However, for "Transformed model", the exclusivity was in respect to pair of images.
The MDS and dendrogram plots for the transformed representations show a stronger resemblance to the original human judgments:





The transformations learned are highly contingent on the domain and do not generalize well to others. However, it is possible to use the same adaptation method to produce a more robust transformation of the DNN representations for the purposes of predicting human similarity judgments. To do so, they learn a transformation using all six domains at once. In conclusion, DNNs develop the correct basis set of features, just at the wrong level of saliency. Learning a reweighting of salience (transformed representation) improves prediction of human similarity judgments. The simple re-weighting of features, that the linear transformation performs, can be viewed as an analogue to dimensional attention. The ability of transformed representations to generalise for new stimuli empowers studies on cognitive processes relying on such representations in the brain.

5.2.3 Pruning

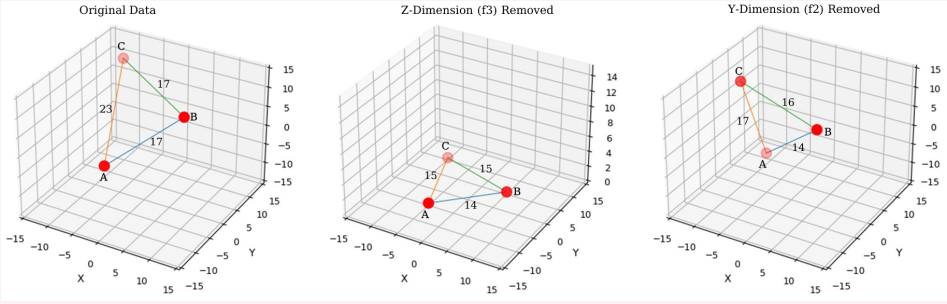
Differently than reweighting, pruning assumes that DNNs **do acquire the relevant features at appropriate levels of salience**, but the **contribution of relevant features is diluted by irrelevant ones**. Pruning aims at identifying a **subset of features**, **per category**, improving prediction of human representations (i.e., that produces object-to-object distances that best match those produced from human behavior), e.g.:

$$Similarity(V, U) = V_2 \cdot U_2 + V_3 \cdot U_3$$

By iterating over features and using *Sequential Feature Selection* algorithms, supervised pruning learns a subset of features that **better predicts human judgments and generalizes to out-of-sample data**. It is supervised as it uses human judgments to choose which features to prune.

Pruning at work: a toy example

Humans find Tigers more similar to Lions than to Pumas. The model representation is disaligned from human representation (as Tiger-Lion and Tiger-Puma distances are the same in the embedding space of the model). We see that removing Y-Dimension brings the embedding space closer to human representation:



To summarize, pruning:

- improves out-of-sample prediction accuracy for human similarity judgments of images, (higher RSA isomorphism);

- produces a more psychologically valid representational space;
- improves prediction of out-of-sample MVPA data (Brain RDMs).

Moreover, the feature-sets retained by pruning vary depending on the category guiding the pruning process, and these sets identify different subspaces (latent factors) in the feature space. Pruning improves out of sample prediction of human similarity judgments for words and allows an interpretation of latent dimensions underlying word similarity judgments.
In the following we will go through a list studies on pruning to improve representational similarity between AI and Humans.

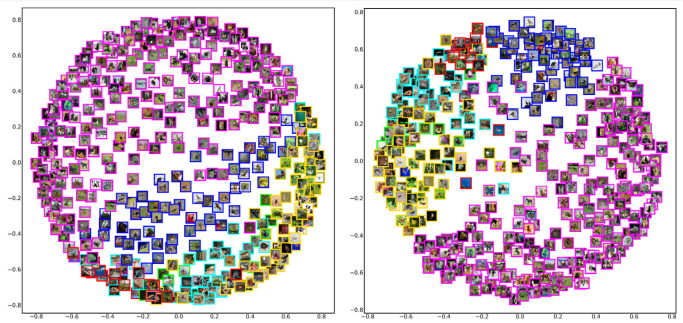
◆ Tarigopula et al. (2023)

Improved prediction of behavioral and neural similarity spaces using pruned DNNs

They prune of a model that learns to predict human similarity judgments within 6 categories, each consisting of 120 images. In particular, they prune the penultimate layer of VGG19, which has 4096 nodes (features), and show that pruning outperforms other methods, including reweighting:

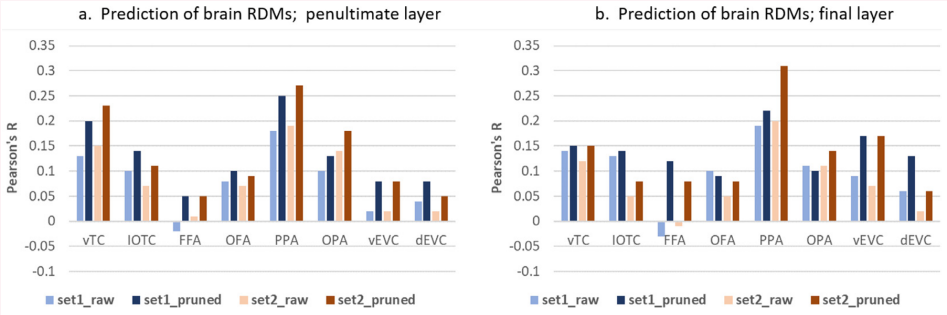
	Animals	Automobiles	Fruits	Furniture	Various	Vegetables
Baseline	0.61 (0.07)	0.51 (0.07)	0.33 (0.08)	0.29 (0.05)	0.43 (0.10)	0.32 (0.07)
PAG18	0.71 (0.09)	0.50 (0.05)	0.25 (0.15)	0.34 (0.08)	0.50 (0.13)	0.27 (0.07)
LASSO	0.64 (0.12)	0.51 (0.08)	0.38 (0.13)	0.37 (0.11)	0.47 (0.12)	0.31 (0.08)
Sim-DR	0.64	**0.57**	0.30	0.33	0.50	0.30
Pruned	**0.75** (0.05)	0.55 (0.08)	**0.39** (0.08)	**0.38** (0.07)	**0.56** (0.1)	**0.41** (0.05)
# nodes	807 (63)	647 (45)	563 (76)	557 (101)	830 (44)	605 (190)

The number of nodes refers to how many nodes are retained by the pruning algorithm (e.g. for Animals 807 out of the original 4096).
They then proceed showing how pruning improves representational space for **out-of-sample image embeddings**. They first use a **different dataset** of Animal images and apply MDS (MultiDimentional Scaling); then they repeat but using only feature indices retained from pruning against the original experimental Animals dataset. In this second case, the animal types are better separated in the MDS representation (better clustering).



(a) Original VGG-19 (b) Pruned VGG-19

Pruning also improves representational space for **out-of-sample brain data**. They use a dataset with two independent sets of 144 images. They produce RDMs from brain activity, per regions, then use RDMs to supervise pruning, and test on out-of-sample data, finding the prediction of brain-derived RDMs is improved.



◆ **Bavaresco et al. (unpublished)**
Continuation of Tarigopula et al. (2023)

They try to understand whether the different feature sets retained by pruning, for each class, are similar; and whether they code for different information. To measure the overlap between the (indices of the) features retained from the different category they use the Dice coefficient ▲:

$$DSC = 2 \cdot \frac{intersection}{union}$$

▲ The Dice coefficient is also called *Dice-Sørensen coefficient* or *Dice similarity coefficient* (*DSC*).

They find the lowest value for $\langle Fruits, Automobiles \rangle$ (0.13) and the highest for $\langle Fruits, Vegetables \rangle$ (0.28). It is not surprising that the highest DSC value is between fruits and vegetables, which are indeed similar. However, in general, the overlap is quite low. They find that just 1 out of 4096 feature is always selected for all 6 categories, meaning that there is not a "core" set of features that is always retained.

To know if the selected features identify different information, they use a different dataset of 50k images. **Images are represented using only activations on the retained features** (6 different versions: 50k×807; 50k×647, etc.). For each of these derived embeddings they **identified the top-5 images that maximize activation on that set of features** (i.e., highest sum by row (in the matrix with inputs in rows and features in columns). They find that pruning-retained features, per category, are **maximized by images that exemplify the category**. This is consistent with the prototype theory. Here is an example of the top-5 images for "transportation" category:

To have a more quantitative analysis, they apply PCA to each version (of the 6) and obtain the scores for the 50k images on the first Principal Component (this is needed since the matrices are too large). Then the 50k PC1 scores are correlated across solutions (they get a score for each image). They conclude that **different retained sets select for different latent dimensions** (i.e., PC1 of similar categories encode for similar information).

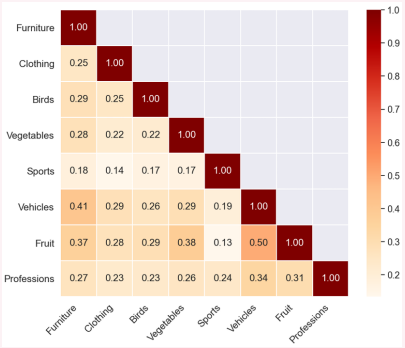
To summarize, human representations of concepts/categories are better approximated by AI models that are modified to reflect only the relevant variance dimensions, as indicated by specific subsets of features. Practically, pruning via sequential feature selection is one way to identify these dimensions. It is competitive in learning human representations and allows to interpret the relevant lower dimensions.

5.2.4 Pruning vs reweighting

Vision DNNs trained to classify already provide a moderate approximation of human representational space. Reweighting is emerging as a new technique to improve this match (useful for practical applications), and was interpreted to suggest that DNNs learn human-like filters, but at wrong levels of salience.
Pruning outperforms reweighting in learning prediction of human representational spaces, but also originates in a different perspective on the importance of DNN filters: the filters are effective at the learned levels of salience, but different datasets benefit from different combinations of filters. Pruning is also more **easily interpretable** as a regularization (data reduction) technique in context of explainable AI and provides insights into brain organization.

◆ **Manrique et al. (2023)**
Enhancing Interpretability Using Human Similarity Judgements to Prune Word Embeddings

They try to pring the same concepts of Tarigopula et al. (2023) to NLP. They define 8 categories as sets of 20-30 noun words each. They have human similarity judgments for all word pairs. The idea is to predict the test-set using all GloVe features, or GloVe feature-sets pruned from the training-set. The results show that **pruning improves out-of-sample prediction** using a **less than half of GloVe's 300 features**. Notice that the features that are important for positioning objects with respect to each other are not necessarily important for classification, and vice versa.
For each solution they identify the retained feature indices and compute the DSC between pruned sets, which often results low (few of the GloVe features are never selected, and no feature is consistently selected). Some results make sense (fruit and vegetables have high overlap) but others do not (vehicles and fruit):

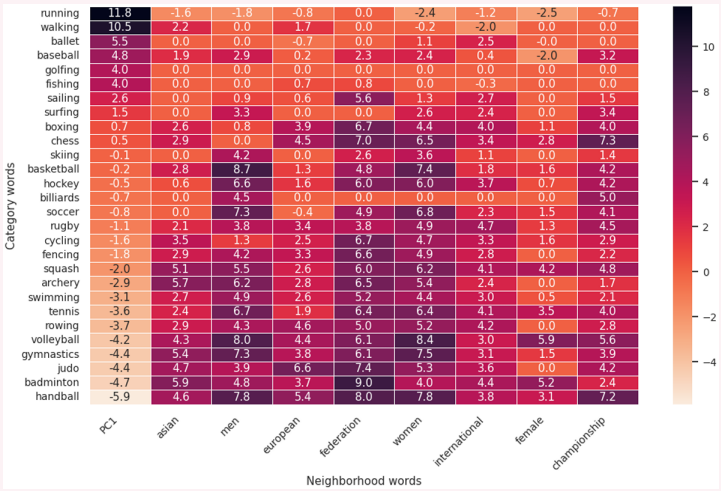


This is a problem for interpretability. So they try to understand what information is coded in a subset of features of a language model. Here an example with the "sports" category is provided. From previous pruning, 100 features out of 300 were retained, from these they take the following steps:

- further reduce the dimensionality with PCA, retaining only the first PC
- rank sports by PC1 value
- for every word in the corpus, compute the co-occurence with each sport
- select the words whose co-occurence scores (in the form of PMI ▲) are correlated to the PC1 scores for each sport

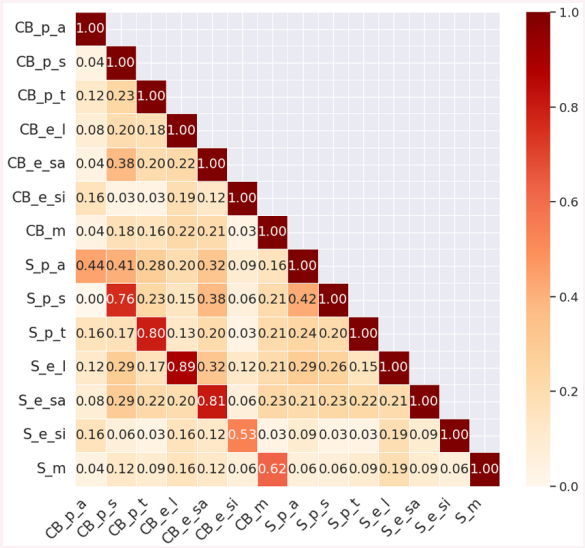
▲ PMI: Pointwise Mutual Information.

The results suggest that human similarity judgments are sensitive to gender- and location-inclusiveness, and (relatedly) international reach.



◆ **Bao et al. (2024)**
Identifying and interpreting non-aligned human conceptual representations using language modeling

This study tries to understand if **people of different groups represent words differently**. In particular they focus on English native speakers (blind vs sighted). They use 7 verb categories each containing 14 verbs and have **human similarity judgments** for all verb-pairs. The judgments **made by congenitally blind and sighted**. They train a model to prune GloVe embeddings to improve out-of-sample prediction, first for blind and then for sighted,so they can compute the DSC between the two subsets of retained features.



They found that verbs describing emission of animate sounds (`e_sa`, e.g. *whine*) and light (e.g. *blink*) have good concordance between retained features for blind (`CB`) and sighted (`S`). For others, such as perception sight (`p_s`, e.g. *see*, *look*), a lower concordance is found.

5.3 Why pruning works

Network Trimming: A Data-Driven Neuron Pruning Approach towards Efficient Deep Architectures

Hu et al. (2016)

Extremely sparse matrices produced by top layers of neural networks indicate that empirically designed networks are heavily oversized. Many neurons in a CNN have very low activations, no matter what data is presented. Such weak neurons are highly likely to be redundant and can be excluded without damaging the overall performance. Their existence can only increase the chance of overfitting and optimization difficulty.

They define the *Average Percentage of Zeros* (*APoZ*) of a single neuron as the percentage of zero activations of that neuron after the ReLU mapping. They use the layer as the unit of analysis (rather than the single neuron), so the analysis is collapsed across all neurons in a layer. The *APoZ* of the c^{th} neuron in i^{th} layer is defined as:

$$APoZ_c^{(i)} = APoZ(O_c^{(i)}) = \frac{\sum_k^N \sum_j^M f(O_{c,j}^{(i)}(k) = 0)}{N \times M}$$

where $f(\cdot) = 1$ if true, and $f(\cdot) = 0$ if false, M denotes the dimension of output feature map of $O_c^{(i)}$, and N denotes the total number of validation examples. Therefore $N \times M$ is the maximum number of activations.

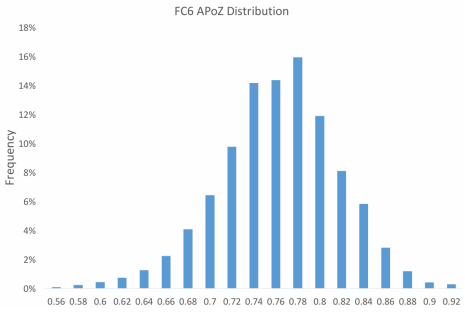


Figure 5.1: *FC6* layer *APoZ* distribution.

The results tell that the redundancy is mainly in the deeper convolutional layers and the fully connected layers. They find more than 600 neurons with $APoZ > 90\%$ (Figure 5.1). To understand if high-$APoZ$ neurons are effectively redundant, they implement a neuron-pruning approach. After training to criteria, they remove all weights to and from high-$APoZ$ nodes (i.e., they remove these nodes from the network). They then re-initialize the network with the last set of weights (prior to pruning) and retrain to criteria (i.e., they fine-tune the pruned network).

		Before Fine-tuning (%)		After Fine-tuning (%)	
Network (CONV5-3, FC6)	Compression Rate	Top-1 Accuracy	Top-5 Accuracy	Top-1 Accuracy	Top-5 Accuracy
(512, 4096)	1.00	68.36	88.44	68.36	88.44
(488, 3477)	1.19	64.09	85.90	71.17	90.28
(451, 2937)	1.45	66.77	87.57	71.08	90.44
(430, 2479)	1.71	68.67	89.17	71.06	90.34
(420, 2121)	1.96	69.53	89.49	71.05	90.30
(400, 1787)	2.28	68.58	88.92	70.64	89.97
(390, 1513)	2.59	69.29	89.07	70.44	89.79

Example of trimming nodes in *CONV5* and *FC6* layers. We have to be aware that a column that is made of all 0s (i.e., a 100%-*PoZ* node) might still impact the representational geometry (object-similarities) if specific distance measures are used (see Section 5.3.1).

5.3.1 Redundancy and representational geometry

For a given dataset, a 100%-PoZ feature does not provide discriminating information between objects. It may serve to separate objects in this dataset from others. However, these features **do contribute to pair-wise similarity** estimations, i.e., estimation of object-similarity (cosine, Pearson). Truong and Hasson (unpublished) try to answer the following questions:

- How do these features contribute to a DNN's Representational Dissimilarity Matrix (RDM)?
- Can their removal improve prediction of human similarity judgments?

Figure 5.2 shows how around 20% of nodes in MNIST and CIFAR-10 have $PoZ > 0.70$, while very few have $PoZ > 0.95$.

To answer the first question, they define the representation of the full network, and get the object-by-object similarity matrix (RDM), that will be used as reference. Then, they compute the RDM with just a small subset of features, the ones with lower PoZ, and compare it with the original RDM (using Person R^2). They progressively insert features from low to high PoZ (i.e., from the most informative features to the least ones) and repeat the steps. The results (Figure 5.3a) show that it is possible to use just a subset of the original features, using the nodes with a PoZ lower than $\sim$ 35%, to approximate the representation of the original network.

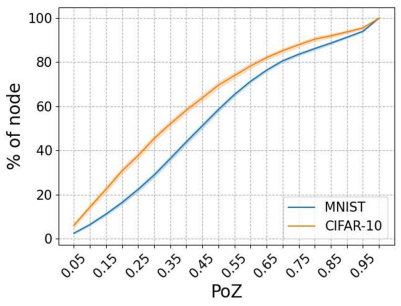
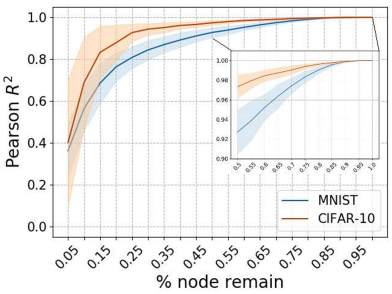


Figure 5.2: Percentage of nodes with a *PoZ* lower than the given value.

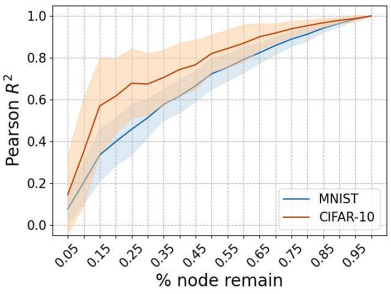
They also try to understand what other nodes (those with the highest PoZ) encode for. They experiment in the other direction, i.e., inserting features from high- to low-PoZ. The results (Figure 5.3b) show that these nodes are not random, nor they encode noise. They can indeed approximate the full network: the lowest 20% already produces $R^2 > 0.6$ for CIFAR-10. This means **relevant information is coded in the network in a redundant way**.

Adversarial attacks

When we prune a network we have to be aware that it becomes less robust to adversarial attacks, since the **embedding space becomes less sparse**, so that the **classes are closer to each other**.



(a) Subset increased progressively from **low- to high-***PoZ* features.



(b) Subset increased progressively from **high- to low-***PoZ* features.

Figure 5.3: Similarity between the RDMs form a subset of features and the RDM got from the original embedding (all features).

To answer the second question they evaluate the *APoZ*-based pruning and its impact on prediction of human similarity judgments. Features are removed sequentially from high-to-low *PoZ* and *2OI* (second-order interaction) is computed between human similarity judgments and the (pruned) model RDM. The approximation of human similarity judgements starts dropping significantly only when we use less than 6% of original features (Figure 5.4). We can notice that by removing sparse features, for the category of "furniture", the approximation of human judgements increases until we use 30% of the original features only. So, for "furniture", sparse features actually contain noise.

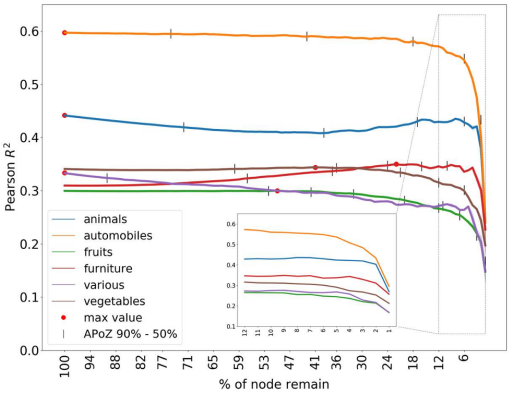


Figure 5.4: Trimming using *APoZ* and evaluating *2OI* against human judgments.

Chapter 6

Developing common representations in AI and humans

In this Chapter we study how human representation and AI representation can be combined, and how brain data can be infused into neural networks.

6.1 *Using human brain activity to guide machine learning* Fong et al. (2017)

This study tries to **improve ML performance by guiding it with brain activity**, and explore whether such guidance makes the representations themselves more "human-like". Notice that here the focus is not on performance but on **representational geometry**. The underlying idea is that, if the human brain is a natural reference point (for representation geometry) and performance, and ANNs are a good algorithm for learning structure, then we can attempt to leverage the ML algorithm with biological information.

They consider 7 brain areas (ROIs, *regions of interest*, whose partitioning is defined a-priori). Different brain areas code for different information about the stimulus, but all of them are involved in **visual processing** (either low, middle, or high level). They use two types of models: CNN and HOG (*histogram of oriented gradients*, an often-used algorithm for generating features that capture the local "slopeness" of different parts of an image).
Their idea is to train a classifier on brain data to perform binary classification. To do so (to maximize the margin between the decision boundary and the data points), they use a loss derived from the **Hinge loss** ▲. They define the **"response strength" from brain fMRI activity data** as the **distance of an object from the decision boundary** for a given binary classification task. This produces a per-stimulus activity weight (response strength) for each stimulus. As input to the classifier, each image is encoded as a vector of brain activity values sampled from a given brain area (one of the 7 mentioned before). From the Hinge loss they define the **Activation weighted loss** (AWL) as:

$$\phi(x, z) = max(0, (1 - z) \cdot M(x, z)$$

where $M(x, z) = \begin{cases} 1 + c_x, \text{if } z < 1 \\ 1 \text{ otherwise} \end{cases}$

and $c_x \geq 0$ is an activity weight derived from fMRI data corresponding to x (it is the distance of the object from the classification boundary of the binary classifier trained on brain data).
While HL penalizes misclassified examples, **AWS penalizes misclassified examples on stimuli that are easy for humans to distinguish**.

They train a classifier on brain responses (fMRI), then use this prior knowledge to train a new model on non-annotated data. They basically use the learned weights as a starting point. The process consists in the following steps:

1. **Derive per-stimulus "activity weigths" from fMRI data**:
    - collect *per-stimulus* activity vectors: use fMRI to record bold response of subject;
    - train a classifier on fMRI activity vectors: SVM classifier trained and tested;

- activity weights derived from distance boundary: use transformed classification scores.

2. **Train (calibrate) 2 image classifiers**:
   - Conventional image classifier training: Radial Basis Function SVM classifier;
   - Margins reweighted by activity data: SVM classifier with activity weighted loss function (note that not all training samples require fMRI weight).

They use images from 5 categories, and the classification problems are based on CNN or HOG features. Information from the higher-level cortical regions is combined in all possible combinations to produce feature sets.

Hinge loss

The true label is denoted as t ($t = 1$ or $t = -1$), while y is the predicted activation for an object by the classifier. $t \cdot y$ is therefore the distance from the boundary. The Hinge loss wants to push this distance so that the margin from the boundary is above 1:

$$Loss(y) = max(0, 1 - ty)$$

if $ty < 0$ (i.e., incorrect classification) $\Rightarrow 1 - ty > 1$;
if $0 < ty < 1$ (correct classification but below the margin) $\Rightarrow 1 - ty > 0$;
if $ty > 1$ (correct classification above the margin) $\Rightarrow 1 - ty < 0$.
This means that the HL is proportional to the distance from the decision boundary, and it **does not care about magnitude of correct decisions above the margin**.

However, in Fong et al. (2017) they use a different formulation:

$$\phi_h(z) = max(0, 1 - z)$$

where $z = y \cdot f(x)$, $y \in N$ is the true label, and $f(x) \in \mathbb{R}$ is the predicted output; thus, z denotes the correctness of a prediction. The HL function assigns a penalty to all misclassified data that is proportional to how erroneous the prediction is.

6.1.1 Results

They argue that brain activity compensates for poor feature representation (consider the large difference for HOG in Figure 6.1).

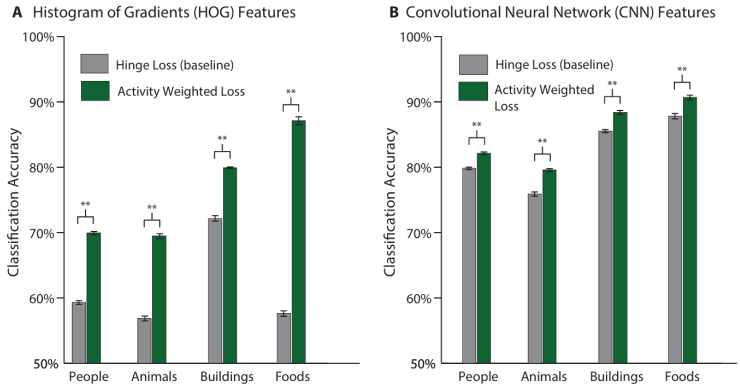


Figure 6.1: Side-by-side comparisons of the mean classification accuracy between models that were trained using either **(A)** HOG features or **(B)** CNN features and either a hinge loss (HL) or activity weighted loss (AWL) function.

They also experiment using information from one brain area only for the loss, observing how this affects classification. Figure 6.2 shows that certain areas produce significantly better accuracy for the specific categories they are selective for.

In conclusion, **information measured directly from brain can guide an ML algorithm to make better human-like decisions**. One can harness measures of the internal representations employed by the brain to guide machine learning. However, the authors ignore an important

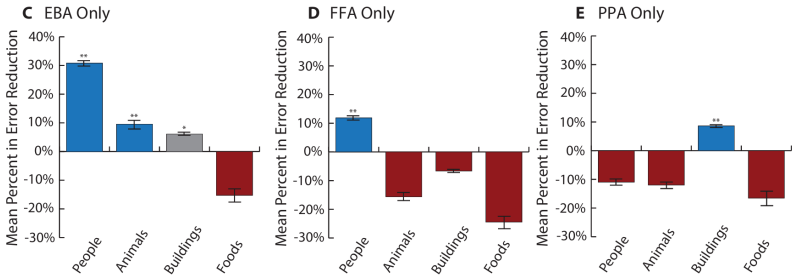


Figure 6.2: Mean error reductions gained by switching from HL to AWL loss when using conditioning classifies on brain activity from individual ROIs (i.e., EBA, FFA, or PPA).

question: Are the better results obtained thanks to **brain** data, or is it just because it is **more** data? What if we improved the HOG classifier using AWL from CNN classifier (i.e., what if we compute the C_x of the hinge loss from the CNN embeddings?). The accuracy should be much higher, so we expect the difference when using Activity Weighted Loss to be less significant.

6.2 *Interpretable Semantic Vectors from a Joint Model of Brain- and Text-Based Meaning* Fyshe et al. (2014)

This study is similar to Fong et al. (2017), but the domain is completely different: human-constraint NLP. They are computational linguists and want to improve word embeddings (they are not really interested in studying the brain).

6.2.1 Background

Vector Space Models (VSMs) represent lexical meaning by assigning each word a point in high dimensional space. The high dimensional space can be any vectorial representation associated with each word. VSMs are typically created using a large text corpora. When this is he case, the VSM represents word semantics as observed in text. In the VSM, the distance between any two words is taken to indicate their semantic similarity (matching, e.g., that observed and rated by speakers). Corpus-based VSMs have been criticized as being noisy or incomplete representations of meaning.

When a person is reading or writing, the semantic content of each word necessarily produces patterns of activity over neurons/voxels/sensors. In principle then, **brain activity could be used instead of corpus data to construct a VSM**. If brain activation data encodes semantics, including brain data in a model of semantics could result in a more effective model. They anticipate that, if it is indeed possible to create word embeddings using brain activations, then the inclusion of this data will only improve a text-based model if brain data contains semantic information not readily available in the corpus.

This study tries to:

- create a database of human-annotated word semantics and create a brain-informed VSM that better predicts this database,
- predict corpus representations of withheld words more accurately by infusing brain data into the learning model,
- map semantic concepts onto the brain by jointly learning neural representations.

6.2.2 Data and method

The corpus data are compiled from a 16 billion word subset of ClueWeb09 and contain two types of corpus features: **dependency** (between words) and **document features**. Dependency statistics were derived by dependency parsing the corpus and compiling co-occurence counts for all dependencies incident on the word. Count thresholding was applied to reduce noise, and positive

pointwise mutual-information (PPMI ▲) was applied to the counts. SVD was applied to the document. They started with a word-by-word co-occurence matrix and then get a word-by-feature matrix X, after PPMI and SVD.

- ▲ They consider just the positive PMI, i.e., words that co-occur more often that if they were independent.

The brain data comes from fMRI data and MEG data for 18 subjects (9 in each imaging modality). Each read 60 concrete nouns. The 60 words span 12 word categories.

They adopt two methods: NNSE and JNNSE.

Non-Negative Sparse Embedding (NNSE)

They want to approximate X by matrix factorization, i.e., get two matrices A and D that when multiplied together approximate X. A has the same number of rows (words) as X, but with a much lower number of latent dimensions (they are trying to compress the feature space), D instead has the same number of columns but less rows. NNSE is therefore defined as follows:

$$\underset{A,D}{\operatorname{argmin}} \sum_{i=1}^{w} \|X_{i,:} - A_{i,:} \times D\|^2 + \lambda \|A\|_1$$

subject to:

$$D_{i,:} D_{i,:}^T \leq 1 \quad (\forall\, 1 \leq i \leq l)$$

$$A_{i,j} \geq 0, \quad 1 \leq i \leq w, \quad 1 \leq j \leq l$$

The matrix A is the output of the algorithm, i.e., the sparse approximated representation we look for (note the $\lambda \|A\|_1$ term that induces sparsity).

Applying NNSE is important for interpretability: without reducing the number of dimensions, the matrix is likely to have many columns that are correlated. With fewer columns, they can study them as they are not linearly dependent. The meaning of a word is captured by finding the top-scoring dimensions for the word (i.e., the colums with highest values for the word's row), and then finding the words that score highest on those dimensions. Each of these dimension, with the set of associated words, captures a meaning of the original word. For example, the word *chair* has the following top-scoring dimensions:

1. chairs, seating, couches;
2. mattress, futon, mattresses;
3. supervisor, coordinator, advisor.

These dimensions cover two of the distinct meanings of the word chair (*furniture* and *person of power*).

Joint Non-Negative Sparse Embedding (JNNSE)

They extend NNSEs to incorporate an additional source of data for a subset of the words in X, and call the approach JNNSEs. Such algorithm allows to have a "shared semantic space" (the matrix A is an approximation of both X and Y, Y being the $word \times voxel$ matrix), and is defined as follows:

$$\underset{A,D^{(c)},D^{(b)}}{\operatorname{argmin}} \sum_{i=1}^{w} \|X_{i,:} - A_{i,:} \times D^{(c)}\|^2 + \sum_{i=1}^{w'} \|Y_{i,:} - A_{i,:} \times D^{(b)}\|^2 + \lambda \|A\|_1$$

subject to:

$$D_{i,:}^{(c)} D_{i,:}^{(c)T} \leq 1 \quad (\forall\, 1 \leq i \leq l)$$

$$D_{i,:}^{(b)} D_{i,:}^{(b)T} \leq 1 \quad (\forall\, 1 \leq i \leq l)$$

$$A_{i,j} \geq 0, \quad 1 \leq i \leq w, \quad 1 \leq j \leq l$$

Notice the difference in summation. The first expression goes over all the words in the corpus (w), while the second goes over the subset for which we have brain data (w'). A' is the subset of the A

matrix that is relevant for reconstructing Y (it is a subset of the rows in X). Both X and Y can be reconstructed by multiplying A with $D^{(c)}$ and $D^{(b)}$ respectively.

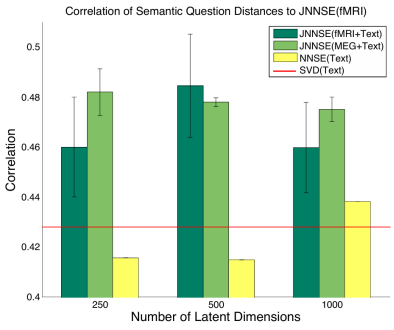
JNNSE has many advantages:

- Handle partially paired data, compared to Canonical Correlation Analysis or Partial Least Squares that require data about the same observations in both cases;
- No need to have a common average brain (can concatenate activation across subjects in the Y matrix, per word);
- Merge different brain imaging experiments adding a specific loss.

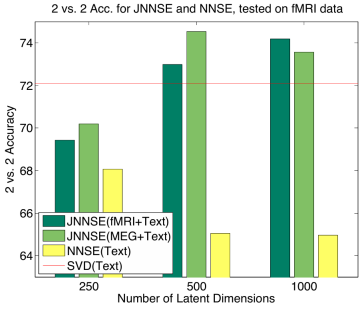
6.2.3 Experiments and results

To check whether the word-by-feature matrix encodes for specific properties, they use a **probing classifier**: they train the regression model on A and evaluate it on a human-rated-property table. More specifically, they obtain behavioral measure of semantics for 60 words, rated on 218 properties (e.g. *smell*, *emotion*, etc.), and then train the classifier to predict the $[60 \times 218]$ human behavior's matrix from the A matrix obtained using NNSE (text only) or JNNSE (brain+text). They experiment with different numbers of latent dimensions l: 250, 500 or 1000. For **evaluating the correlation** of the latent representation (A) with behavioral data (Y), they do not compare the tables directly, but use in-matrix pairwise (Euclidean) distance instead, i.e., they are somehow checking if the two vector spaces are isomorphic.
Figure 6.3a shows that joint embeddings improve prediction of human similarity space. However, as in Fong et al. (2017), the higher performance might not be caused by infusing brain data, but just by using a multiple data sources, which allow to remove noise.

Another experiment they deal with consists in **word prediction from brain data**. The matrix A from NNSE or JNNSE is used as outcome variable: they predict it (for left out words) one column at a time. For each lower-dimension, there is a set of values over words (Y). Regression is used to predict the value of that dimension l over the Y words. This is repeated for all l dimensions, which produces a predictive l-dimensional vector per word. They train the regression on A (NNSE or JNNSE) consisting of 58 words. They then produce predictive vectors for the 2 left out words and evaluate their similarity to the ground truth of those embeddings in A.
The results presented in Figure 6.3b show how word prediction from brain data improves when the embedding space itself (A) is constrained by brain data.



(a) Correlation of NNSE and JNNSE models with the distances in a semantic space constructed from behavioral data.



(b) Average 2 vs. 2 accuracy for NNSE and JNNSE, tested on fMRI data. Models created with one subject's fMRI data were not used to compute 2 vs. 2 accuracy for that same subject.

Figure 6.3: Comparison between NNSE and JNNSE.

Here they ask: can an accurate latent representation of a word be constructed using only brain activation data? They try to produce X-matrix ("raw") entries for words for which there is not enough corpus data required for creation of corpus statistics (i.e., they present rare words to people to get fMRI/MEG data). This experiment practically consists in **predicting corpus data**. They use JNNSE to obtain A-entries for words appearing in X and Y (corpus and brain), but also some words appearing only in Y (brain). Notice that JNNSE allows to have brain activation entries without the corresponding corpus entries. Then, $D^{(c)}$ in used to recreate the entries for those words.

They compute a rank accuracy measure and the mean rank accuracy is as high as 67% for $l = 500$.

The last experiment consists in **mapping semantics onto the brain**. $D^{(b)}$ is a $[l \times v]$ matrix (*latent dimensions* $\times$ *voxels*). This allows to obtain a brain map for each dimension l in that matrix. They tweak the importance of the perceptual features (Y) by scaling their values (details missing; but we can consider weighting schemes). They plot these dimensions on brain slices. They can find a brain area, link it to the dimension that predicted it, and see which words load strongly on that dimension. This allows to study how different groups (age, culture, etc.) have different representations of language.
In Figure 6.4 there is an example (through fMRI slices) of mapping ($D^{(b)}$) from latent semantic space (A) to brain space (Y) for fMRI and words from three semantic categories.

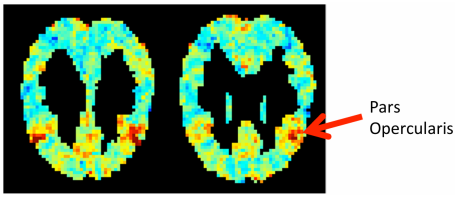


Figure 6.4: $D^{(b)}$ matrix; dimension with top scoring words *buffet*, *brunch*, *lunch*. Pars opercularis is believed to be part of the gustatory cortex, which responds to food related words.

In conclusion, VSMs can be extended or even substituted using brain data. Addition of brain data strongly improves the prediction of human annotations (perhaps can substitute them) and the prediction of latent dimension scores produced in the joint embedding. It is possible to use brain data to synthesize raw corpus data for those words. And finally, solutions of joint embeddings can be mapped onto the brain space.

6.3 *Decoding Brain Representations by Multimodal Learning of Neural Activity and Visual Features* Palazzo et al. (2020)

The idea of this paper we are interested in is: studying attention and saliency on images by infusing brain data into a neural network. In particular, they aim at:

- achieving multimodal learning that projects brain data and image data into the same latent space;
- using brain activity to guide machine learning tasks (**visual saliency modeling**).

Most prior work in multimodal learning tries to learn a joint embedding space for images and text. Here instead, they try to learn a joint embedding of EEG and images. The first problem when creating a joint embedding of EEG and images is that the image does not change over time, while the EEG data does. They need an architecture able to encode time.

6.3.1 Data and method

They have EEG data of 6 people watching 40 categories from ImageNet (50 images in each class). The EEG data is at high resolution: 128 channels, 1kHz recording, 0.5sec each image (so that they collect 500 time-points for each image).

They use a Siamese network with the triplet loss to maximize the similarity between modalities. EEG (e) and visual (v) features of the same image (e_1, v_1) should be mapped to be nearby in image space, whereas EEG features of image 1 and visual features of image 2 (e_1, v_2) should be pushed apart. As **compatibility** (the measure of how much two encodings are similar) between features, they just take the **dot product of the embeddings**. Similarity of mismatching EEG/vision should be lower than the matching case. The loss is defined as follows:

$$L(e_1, v_1, v_2) = max0, F(e_1, v_2) - F(e_1, v_1)$$

where F is a similarity measure.
The triplet loss is positive if the similarity of e_1 with v_2 (negative item) is greater than its similarity with the positive item (v_1).

The architecture receives more than one input at a time (in particular there are two networks, and in total a triplet of inputs is presented). The two networks have the same exact architecture

(FIgure 6.5). The EEG data are pushed through four 1D-convolutional layers (learning filters at different temporal scales), and these are fed into a recurrent layer. The hidden state of the recurrent layer is fed to a fully connected layer from which embeddings are extracted for the compatibility function. They don't use directly the embeddings of VGG (or other), but rather put a layer in between (to learn more "brain related" representations: they train different architectures, whose representations are fine-tuned during the Siamese training.

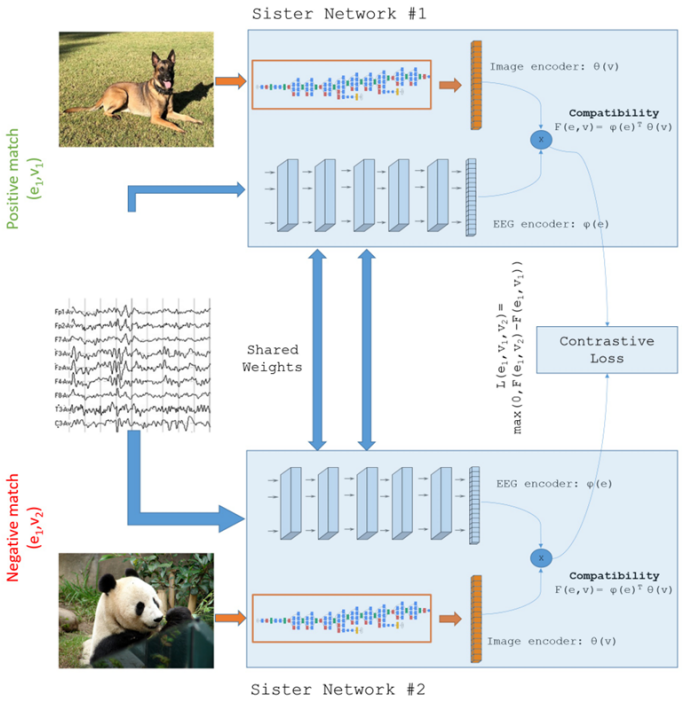


Figure 6.5: Siamese network for learning a joint brain-image representation. The idea is to learn a space by maximizing a compatibility function between two embeddings of each input representation. Given a positive match between an image and the related EEG from one subject, and a negative match between the same EEG and a different image, the network is trained to ensure a closer similarity (higher compatibility) between related EEG/image pairs than unrelated ones.

After training, they use the trained EEG and image encoders as feature extractors in the joint embedding space, followed by a softmax layer for image classification, where they classify into 40 classes. They test different configurations of image and EEG encoding in joint embedding, and then quantify each encoder separately as a feature extractor for classification. The results are provided in Table 6.1.

Image encoder	EEG	EEG Acc.	Image Acc.	Avg Acc.
Inception-v3	LSTM	90.1%	93.6%	91.9
Inception-v3	**GRU**	**90.4%**	**94.7%**	**93.0**
ResNet-101	LSTM	90.7%	91.2%	91.0
ResNet-101	GRU	92.3%	91.5%	91.9
DenseNet-161	LSTM	92.4%	92.3%	92.4
DenseNet-161	GRU	93.7%	91.8%	92.8
AlexNet	LSTM	85.6%	70.1%	77.8
AlexNet	GRU	77.2%	69.9%	73.6

Table 6.1: EEG and Image Accuracies refer to accuracy based on EEG encoder and image encoder alone, respectively.

In Figure 6.6 is shown how joint embeddings increase performance; merging visual features into EEG features of course boosts EEG classification performance, but is less interesting.

Moreover, they **use the joint embedding for saliency detection** ▲. **After training for joint embedding**, they employ a masking process. This has to be done after training so that they can

Image classification performance		
Model	**Visual Learning**	**Joint Learning**
Inception-v3	93.1 %	94.4 %
ResNet-101	90.3 %	90.5 %
DenseNet-161	91.4 %	92.1 %
AlexNet	65.5 %	69.4 %

EEG classification performance		
EEG encoder	**EEG Learning**	**Joint Learning**
EEG-ChannelNet	48.1%	60.4%

Figure 6.6: Comparison of image and EEG classification performance when using only one modality (either image or EEG) relative to when the joint neural-visual features are used. For each model, the best performance according to Tab. 6.1 is reported.

trust the compatibility in output of the model. The masking process follows these steps (notice that no learning is involved):

1. a mask is applied to a part of the image,
2. the compatibility between the brain vector (to original image) and the image vector (of the masked image) is computed,
3. the decrease in match vs. the original compatibility score is computed.

In other words, the saliency value at pixel (x, y) is obtained by removing the $\sigma \times \sigma$ image region around (x, y) and computing the difference between the original compatibility score and the one after suppressing that patch. This way they understand how much salient an image part is: the higher the decrease in compatibility, the more the saliency. Note that if non-salient parts are removed from the image, the compatibility might increase.

▲Saliency detection

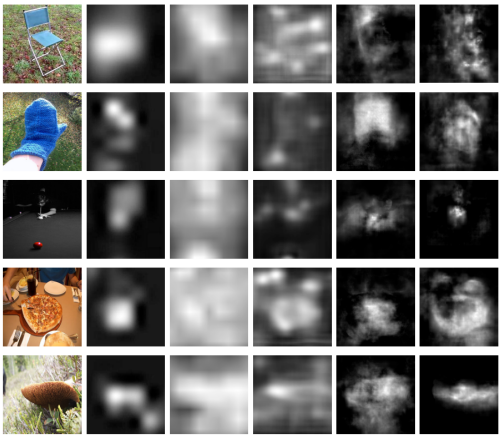
Saliency detection is set of algorithms that can process an image to identify which parts of the image are particularly important (outputting a sort of heat map). These are evaluated by comparing the saliency map generated by the algorithm to the ground truth using either:

- Correlation between the two maps;
- Normalized Scanpath Saliency (NSS), which computes the mean normalized saliency value at fixated locations (more is better).

The ground truth can be obtained using an eye tracker and measuring the "density" of where people look more.

Huang et al. (2015) proposed SALICON, a DNN trained to produce saliency maps. It was created by training an off-the-shelf DNN to predict which parts of the image space were salient. Specifically, a layer is added on top of the last convolutional layer. It contains a single feature, which learns which combinations of feature maps (at that depth) predict pixel saliency (the kernel is 1×1). Basically, the last layer learns a single kernel that is applied to all pixels. The network is trained with objective functions that maximize the fit to a human saliency map. SALICON can predict saliency at many different image scales by combining information from the original image and its downsampled versions.

The experiment consists in participants freely observing the same 2000 images for which EEG data were collected. They test competitive saliency-detection algorithms (SALICON and SALNET), in addition to the baseline (effect of masking on simple visual classification, i.e., how much worse a classifier performs when the input image has a masked patch). The results are presented in Fig. 6.7.



(a) Qualitative comparison of generated saliency maps. From left to right: input image, human gaze data (ground truth), SALICON, SalNet, visual classifier–driven detector, and visual/EEG–driven detector (current method).

Method	s-AUC	NSS	CC
SalNet	0.637	0.618	0.271
SALICON	0.678	0.728	0.348
Visual classifier–driven detector	0.532	0.495	0.173
Our neural-driven detector	**0.643**	**0.942**	**0.357**
Human Baseline	0.939	3.042	1

(b) Saliency performance comparison in terms of shuffled area under curve (s-AUC), normalized scanpath saliency (NSS) and correlation coefficient (CC) between the compatibility-driven saliency detector and the baseline models. The human baseline consists in the scores computed using the ground truth maps. Since they adopt a leave-out-one setup, the reported values for their approach are averaged over all the 40 experiments.

Figure 6.7: Results of Palazzo et al. (2020).

6.4 *Human uncertainty makes classification more robust* Peterson et al. (2019)

The authors think at accomodating human uncertainty into the network. We are more interested in modifying the training objective: Should we use a distribution for ground truths, instead of a one-hot encoding?
The main idea is to **introduce a soft-labelling scheme** that is informed by human uncertainty, and evaluate whether soft labels makes categorization more generalizable to out-of-sample data.

6.4.1 Background

In a classification-learning context, soft labels are ground truth labels for an observation where the mass of the observation is not entirely located at a single correct category.
Many soft labeling schemes have been described:

- split mass uniformly among non-target item,
- split mass as a function of nearness of an observation to a classification boundary (prevents overfitting),
- split mass in relation to types of objects potentially recognized in the scene (categorization and object detection).

Human knowledge is inconsistent with the notion of "hard labels" because in some cases humans are not confident in category assignment, whereas CNNs are. In some cases humans may be not confident, and so will the CNN, but in different ways. **They propose a method to estimate the soft-label probability distribution from human judgments**. Knowing the human classifications is useful to understand the "image quality". It is useful for building NNs that reproduce the same sort of vagueness/fuziness.

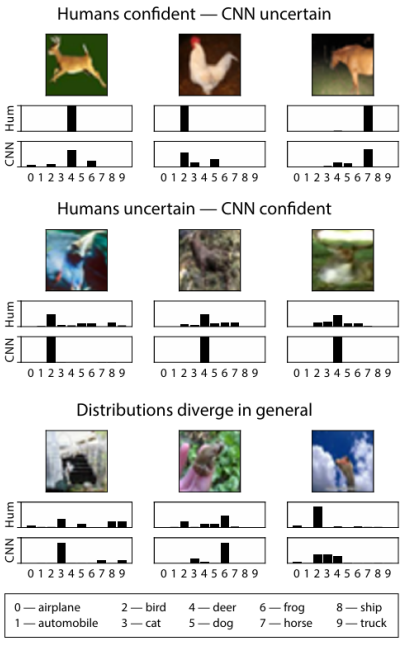


Figure 6.8: CIFAR10 images for which humans and the best traditionally-trained CNN agree in their top guess, but systematically differ over other choices.

6.4.2 Data and method

They produce a curated dataset, CIFAR10H: a behavioral dataset consisting of $\tilde{5}00$k human categorization decisions over the 10k-image testing subset of CIFAR10 (approximately 50

judgments per image). One of the reason for using CIFAR10 is that it contains observations close to the category boundaries. They use Amazon Mechanical Turk: on each trial, a person categorizes each image by clicking one of the 10 labels surrounding it as quickly and accurately as possible (but with no time limit). Note: no confidence is obtained for the judgments.

They expect the human image label distribution $p_{hum}(y|x)$ to better reflect the natural distribution over categories given an image, so they use it as an improved estimator for $p(y|x)$, where y is the distribution of activity assigned to all labels for a given image x. They then simply use the usual cross-entropy loss to minimize the divergence between the human distribution and the post-softmax activity distribution.

6.4.3 Results

They train multiple architectures using the human labeling data (9k images for training, 1k images for testing). This is simply to show accuracy for the homogenous dataset. They then apply the learned model (weights) to several other CIFAR10 variants. As generalization measures they evaluate accuracy and cross-entropy loss.

As shown in Fig. 6.9, **generalization improves with human soft labels**. Accuracy is higher and loss is lower using human labels for every individual CNN and dataset.

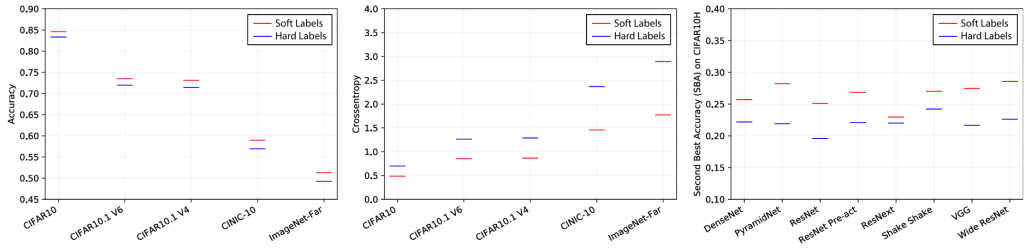


Figure 6.9: Generalization results. Left: accuracy against ground-truth labels, for increasingly out-of-training-sample distributions, averaged across CNNs. Center: cross-entropy against ground-truth labels, averaged across CNNs. Right: Second best accuracy (SBA) for all models using CIFAR10H held out set, averaged across folds.

Soft labeling also produces a better calibrated model (*calibration* refers to the relation between confidence of the model and correct/incorrect output). Consider Fig. 6.10: on correct predictions, both hard- and soft-labeling show similar high confidence. Instead, on incorrect predictions, the soft-label training results in significantly lower confidence.

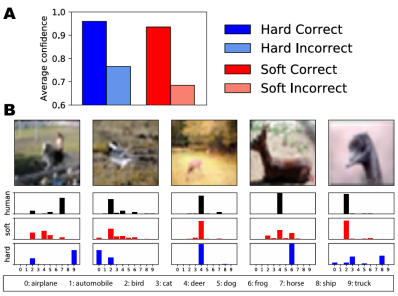


Figure 6.10: **(A)** Mean confidence for correctly/incorrectly classified examples after hard/soft label training. Soft-label models are far less confident when incorrect than hard-label controls, and only slightly less confident when correct. **(B)** Soft label training yields predictions that distribute probability mass more like people, with the same top choice.

They consider several other alternatives to soft labels training. Here are 4:

- **Category soft targets**: approximate category-level confusions by averaging ratings across images within each category. They apply then the same soft label for all images in a category (i.e., all the samples from a category share the same target distribution);
- **Knowledge distillations** an alternative way for simulating confusion (instead of infusing human knowledge): they simulate the confusion by using different models. They take the post-softmax profile of each image from 8 different classification models, averaging.
- **Mixup**: a virtual training mechanism that generates merged images ("virtual training examples") and assigns them merged labels.
- **Sampled hard targets**: use human uncertainty in different a way. Based on the human confusion, assign all mass to a single category, but during training "swap" the labels: for single image train on more than single 1-hot model.

Probabilistic soft labels outperform their alternative methods for soft-label construction and also generalize better to new test sets.

6.4.4 Discusssion

Speaking of adversarial attacks, which aim to produce a wrong label for a minimally changed image, the soft-labeling approach provides protection. The accurate behavior here is the maintenance of the original label. Human training produces a network that is much more robust against Fast Gradient Sign Method (FGSM) attacks (one of the easiest ways to create an adversarial image). Also, when exposed to an adversarial example, the Cross Entropy between the network and the model is lower after being trained with human-trained soft labels. This means that the adversarial example is not shifting the decision as much as it does on a non-soft-label architecture.

However, this approach has some potential weaknesses that might be tackled:

- the cross-entropy **loss term for training instance x might be down-weighted if the label for x is determined as not trustworthy or ambiguous**. This does not require setting up a soft label for training, but simply determining which observations are less important than others.
- Li et al. (2020) prevent overfitting by **down-weighting images that are near the decision boundary** of two classes.
- In the area of NLP there is a substantial literature on how to incorporate annotator disagreement into models. Fornaciari et al. (2021) find that incorporating soft-label information based on annotator disagreement improves performance on NLP categorization tasks. They add prediction of soft labels as an auxiliary task in Multi-Task Learning: NLP categorization and prediction of soft label distribution (annotators disagreement).

Bibliography

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. ISSN 0959-4388. doi: https://doi.org/10.1016/j.conb.2019.01.007. URL https://www.sciencedirect.com/science/article/pii/S0959438818301569. Machine Learning, Big Data, and Neuroscience.

Ruth Fong, Walter Scheirer, and David Cox. Using human brain activity to guide machine learning, 2017.

Alona Fyshe, Partha P. Talukdar, Brian Murphy, and Tom M. Mitchell. Interpretable semantic vectors from a joint model of brain- and text- based meaning. In Kristina Toutanova and Hua Wu, editors, *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 489–499, Baltimore, Maryland, June 2014. Association for Computational Linguistics. doi: 10.3115/v1/P14-1046. URL https://aclanthology.org/P14-1046.

Hengyuan Hu, Rui Peng, Yu-Wing Tai, and Chi-Keung Tang. Network trimming: A data-driven neuron pruning approach towards efficient deep architectures, 2016.

Brenden M. Lake, Wojciech Zaremba, Rob Fergus, and Todd M. Gureckis. Deep neural networks predict category typicality ratings for images. In David C. Noelle, Rick Dale, Anne Warlaumont, Jeff Yoshimi, Teenie Matlock, Carolyn D. Jennings, and Paul P. Maglio, editors, *Proceedings of the 37th Annual Meeting of the Cognitive Science Society, CogSci 2015*, Proceedings of the 37th Annual Meeting of the Cognitive Science Society, CogSci 2015, pages 1243–1248. The Cognitive Science Society, 2015. Publisher Copyright: © Cognitive Science Society, CogSci 2015.All rights reserved.; 37th Annual Meeting of the Cognitive Science Society: Mind, Technology, and Society, CogSci 2015 ; Conference date: 23-07-2015 Through 25-07-2015.

Tom Michael Mitchell, Svetlana V. Shinkareva, Andrew Carlson, Kai min Kevin Chang, Vicente L. Malave, Robert A. Mason, and Marcel Adam Just. Predicting human brain activity associated with the meanings of nouns. *Science*, 320:1191 – 1195, 2008. URL https://api.semanticscholar.org/CorpusID:6105164.

Simone Palazzo, Concetto Spampinato, Isaak Kavasidis, Daniela Giordano, Joseph Schmidt, and Mubarak Shah. Decoding brain representations by multimodal learning of neural activity and visual features, 2020.

Joshua C. Peterson, Joshua T. Abbott, and Thomas L. Griffiths. Evaluating (and improving) the correspondence between deep neural networks and human representations. *Cognitive Science*, 42 (8):2648–2669, 2018. doi: https://doi.org/10.1111/cogs.12670. URL https://onlinelibrary.wiley.com/doi/abs/10.1111/cogs.12670.

Jake Spicer and Adam N Sanborn. What does the mind learn? a comparison of human and machine learning representations. *Current Opinion in Neurobiology*, 55:97–102, 2019. ISSN 0959-4388. doi: https://doi.org/10.1016/j.conb.2019.02.004. URL https://www.sciencedirect.com/science/article/pii/S095943881830103X. Machine Learning, Big Data, and Neuroscience.

Priya Tarigopula, Scott Laurence Fairhall, Anna Bavaresco, Nhut Truong, and Uri Hasson. [improved prediction of behavioral and neural similarity spaces using pruned dnns]. *Neural Networks*, 168:89–104, 2023. ISSN 0893-6080. doi: https://doi.org/10.1016/j.neunet.2023.08.049. URL https://www.sciencedirect.com/science/article/pii/S0893608023004690.