VRIJE
UNIVERSITEIT
BRUSSEL

Thesis submitted in partial fulfilment of the requirements for the degree of Master of Science in Applied Sciences and Engineering: Computer Science

LEARNING HIERARCHICAL SPECTRAL REPRESENTATIONS OF HUMAN SPEECH WITH THE INFORMATION DYNAMICS OF THINKING

Steven Homer

2018-2019

Promotor & Supervisor: Prof. Dr. Dr. Geraint A. Wiggins

Sciences & Bioengineering Sciences

LEARNING HIERARCHICAL SPECTRAL REPRESENTATIONS OF HUMAN SPEECH WITH THE INFORMATION DYNAMICS OF THINKING

Steven Homer

2018-2019

Promotor & Supervisor: Prof. Dr. Dr. Geraint A. Wiggins

Wetenschappen & Bio-ingenieurswetenschappen

Abstract

The Information Dynamics of Thinking (IDyOT) is a cognitive architecture based around the principle that the human mind seeks to represent data in the most information-efficient way possible. By blending conceptual spaces theory and information theory, and making a minimal set of assumptions, IDyOT specifies three main processes – segmentation, categorization, and abstraction – that hierarchically generate spectral representations of perceptual data in an information-efficient manner.

This thesis is primarily an explication of the propositions and claims introduced in IDyOT by applying the mathematical formalisms of Hilbert spaces and the Fourier transform as implementations of the aforementioned theoretical processes. After deriving the necessary mathematics in order to implement the architecture, an implemented system is empirically tested on a corpus of human speech, where we expected to see the emergence of distinct semantic categories corresponding to human speech sound syllables.

After running the implemented system on human speech audio, we found that the hierarchical process of abstracting spectral representations of the audio signals was consistent with a few claims of the theory. First, spectral representations of similar trajectories through conceptual spaces were categorized together, indicating that hierarchical abstraction produces consistent categories as expected. Second, the duration of categories at higher levels demarcated boundaries that approximiately corresponded to annotated boundaries of syllables, indicating the segmentation process finds reasonable boundaries at which to chunk information. Finally, we saw the emergence of similar categories representing distinct human syllables at higher levels of abstraction, indicating that the spectral representations produced at these higher levels have semantic content. These results provide the first empirical evidence that the IDyOT cognitive architecture is a potentially viable model of human cognition.

Contents

Chapter 1

Introduction

1.1 Context, Problems, & Justifications

In the past few years, Artificial Intelligence has experienced an unparalleled resurgence in academia and popular culture. Due to the massive scale of data and processing power made available by the internet, the problems of sparse data and slow CPUs that once plagued Machine Learning techniques like Deep Learning (LeCun et al., 2015) and Reinforcement Learning (Sutton et al., 1998) have disappeared, yielding highly effective methods of analyzing data. However, with all of this recent progress, fundamental questions have reemerged as well. The "interpretability problem" (Chalmers, 1996) – that the internal workings of a neural network or other ML technique are so unintelligible as to be meaningless to a human observer – has resulted in a lack of trust and understanding in AI systems. Though neural networks are effective in generating accurate results, explanations of their inner parametrization are at best ad-hoc. This problem of AI Explainability yields the question "Can popular AI methods such as Neural Networks and Reinforcement Learning be embedded with semantics? Do these systems have a sort of natural semantics, and hence explainability?"

When examining this question of explainability, and therefore semantics, the natural first place to look is how humans go about explaining. Though the process of explaining a given concept is simple, usually through a series of interdependent reasons or set of examples, the way that these reasons and examples exist in the mind is a mystery. It seems that our internal voice asks for an explanation, and without much effort, our non-conscious mind searches our memory for representations and relations that solve that question (Baars, 1993). So though we generally know when an explanation is satisfactory, intuitive, or even deep, we're not sure of the manner in which the content of that explanation is represented and recruited. "How does the mind represent information and relations? How do we learn these deep, abstract representations from a relatively paltry amount of perceptual data?" (Quine & Quine, 1969)

The Information Dynamics of Thinking theory (IDyOT) (Wiggins, 2018) is a cognitive architecture that seeks to answer these questions with a minimal set of assumptions and processes, drawing inspiration from information theory and conceptual space theory. This thesis is an explication and implementation of IDyOT, with the goal of providing evidence for its theoretical processes and claims.

1.2 Objectives, Hypotheses, & Methods

The objectives of this thesis are three-fold. The first goal is to determine how utilizing Hilbert spaces and the Fourier transform can be used to implement the processes of segmentation, categorization, and abstraction described in the IDyOT (section 2.4 and chapter 3). The second is to implement the theoretical results, with the third being to determine if the implemented system behaves as described in the theory when applied to a corpus of human speech audio.

Since the first goal is not so much a matter of empirical testing, but of theoretical explication and derivation, there is no real hypothesis for this objective. That being said, if those derivations, when implemented in the system, yield poor results, it's possible that instead of the theory being deficient, it is actually the derivations. Therefore, the main hypothesis of the thesis is that the implemented system will produce results in line with those described in the theory. Specifically, as applied to audio data of human speech, the hypothesis is that at a few levels of abstraction, the categories corresponding to human speech syllables will emerge.

The research method is split primarily into two categories, though with significant overlap. The first category investigates the consequences of the choice of Hilbert spaces and the Fourier transform as the driving mathematical formalisms in an implementation of IDyOT. This portion of the thesis is highlighted in Chapter 3: Theory & Implementation and focuses on how the mathematics of the main processes of IDyOT shake out when using the aforementioned formalisms. The theoretical results described in this section are then used to create an implementation of the IDyOT system, which is trained on audio data of human speech. This implementation and analysis of results form the second category of the research method.

The results of training the implementation on human speech are then presented and discussed in Chapters 4 and 5. Since a quantitative comparison of the cognitive architecture implementation with the human mind is obviously impossible, we instead perform a qualitative analysis of the results. The visualizations presented in the section 4.2 provide a medium in which the results can be analyzed such that reasonable conclusions can be made. This is not to say that there are no quantitative techniques employed in the visualizations, just that the analysis of those visualizations is qualitative in manner.

1.3 Structure

Following this introduction, the structure of the thesis is as follows:

Chapter 2: Background: First, a few short sections of background information regarding conceptual spaces, Hilbert spaces, and information theory will be explained. These are necessary to understand the final background section, Information Dynamics of Thinking, of which this thesis is an implementation. This section will give the motivations for IDyOT theory as a cognitive architecture and explain its main principles of information dynamics and efficiency. It will also outline the main processes of segmentation, categorization, and abstraction core to the functioning of IDyOT.

Chapter 3: Theory & Implementation: Following the high-level description of IDyOT, more specific matters of theory stemming from the chosen formalisms will be covered. Explications of utilizing Hilbert spaces and the Fourier transform are explored in the Abstraction and Segmentation sections, whereas the Categorization and Interpolation sections round out the chapter in implementation, tying everything together.

Chapter 4: Empirical Analysis: The implemented system described in Theory & Implementation is then empirically tested on a corpus of human speech, with the results visualized to allow for qualitative evaluation against human speech. These visualizations are explained in the context of IDyOT with the preliminary results suggesting that the different processes of IDyOT are behaving as envisioned in the theory.

Chapter 5: Evaluation & Discussion: After looking at the specifics of the results in the Empirical Analysis, the overall strengths and deficiencies of both the theory and implementation are investigated in the Discussion, finding that though certain aspects of the results of the implementaion are inconsistent, overall, the implementation at least partially confirms the theory. After this, the broader implications of IDyOT as a general approach to perceptual representation will be examined.

Chapter 6: Conclusion: Finally, the contributions and limitations of not only the implementation but also the theory are summarized, finishing with prospects for future work that can expand the semantic range of the implementation through the use of different geometries and transformations.

Chapter 2

Background

2.1 Conceptual Spaces

In knowledge representation, the argument over the representation of cognition often falls into two camps: connectionism and symbolicism. As the lowest level of represention, connectionism is best exemplified by artificial neural networks (LeCun et al., 2015), where cognition emerges from myrid connections between neurons. At the highest level, symbolicism views the mind as a Turing machine (Turing, 2009), where cognition is equivalent to compution by symbol manipulation. Conceptual spaces theory (Gärdenfors, 2004) argues for a middle way, through the use of eponymous conceptual spaces. Conceptual spaces theory views cognition as the process of concept formation by means of similarity, so that the continuous representations of connectionism can be bridged to the discrete representations of symbolicism, hopefully gaining the best of both worlds.

Conceptual spaces theory states that representations in the mind are situated in conceptual spaces – semantically rich spaces with geometric properties – which allow for intuitive geometric reasoning about related objects. For instance, in the conceptual space of color with dimensions of hue, saturation, and brightness, one can formally make a geometric claim that "orange lies between yellow and red." This simple claim cannot be made by connectionism or symbolicism without imposing ad-hoc external semantics on the connection weights or symbols respectively, highlighting the explanatory power of conceptual spaces for knowledge representation.

2.1.1 Quality Dimensions

Quality dimensions are the basic building blocks of a conceptual space. They can be thought of as the axes that give meaning to the elements in the space. In three-dimensional Cartesian space, when referring to a point, we specify it by its placement on each of the x, y, and z -axes. By analogy, each of the xyz-axes would be a quality dimension in the 3D Cartesian space. However, quality dimensions are more than just orthogonal unit vectors, they can also have their own specific geometry that serves to constrain the dimension. For instance, a quality dimension may have the geometry of a circle, resulting in different behavior than the real number line. Quality dimensions also allow us to speak meaningfully about similarity between objects in a space since, by definition, they possess distance and

betweenness relations. Finally, it is important to remember that the 'quality' of the quality dimension is what gives it inherent semantic content beyond just being a descriptive dimension.

2.1.2   Domains and Conceptual Spaces

Integral quality dimensions require one another to exist. For instance, the three qualities of sound: pitch, timbre, and loudness, are all integral to one another (McAdams & Saariaho, 1985). It is impossible to identify a sound without specifying all three of these dimensions. On the other hand, most quality dimensions are separable, meaning that they are independent of one another. Though separable quality dimensions are independent, they may still be highly correlated, which may give the illusion that they are integral.

A domain is a set of integral dimensions that are separable from all other dimensions. In a sense, it is the minimum description needed for a given space. For example, the three qualities of sound form a domain. Often, different domains will be correlated with one another, and combining them will yield a richer description of a given object. This combination of multiple domains is what is referred to as a conceptual space, so that the specification of an object is nothing else but its location in a conceptual space.

2.1.3   Similarity, Distance, Betweenness

Humans have an innate sense of similarity without being able to fully describe why two things are similar (Tversky, 1977). In simple cases, this similarity can be made formally explicit, for example that a rectangle is more similar to a square than a circle. However, this intuition for similarity extends to even very abstract realms. For instance, most people would naturally agree that country music is closer to rock-n-roll than it is to classical Indian ragas, but would be hard-pressed to give an exact definition or method of why this is so.

Conceptual spaces allow this innate sense of similarity to be codified in the intuition of geometry, betweenness, and distance. Given a betweenness relation for a conceptual space, one can say that a given object is between two other objects, which allows us to say that one is closer to another than the third. In our case, we will study conceptual spaces equipped with a distance measure or norm. This

allows us to examine similarity between objects, such that more similar onjects will be closer to one another than more different objects.

2.1.4   Convexity of Properties and Concepts

Given that Conceptual Spaces Theory posits that cognition is equivalent to the formation of concepts (Gärdenfors, 2004), one should define a concept. Defining a property or concept as "an invariance across a range of contexts, [reifiable] so that it can be combined with other appropriate invariances" (Kirsh, 1991), it is immediately clear that these correspond to regions of a conceptual space. Since all of the objects in a given region are similar to each other, by grouping them together, we can see that the region corresponds to a property or concept. A property would be a region in a domain – for instance the red property corresponds to a region of the color space – and a concept would be a region in a conceptual space, with the property or concept being identified by something like its centroidal prototype (Rosch, 1983).

Gärdenfors posits that regions corresponding to properties and concepts are convex in nature (Gärdenfors, 2004). Though this does not fall directly from the theory itself, it is reasonable to think that the region of concept is not intruded upon by other concepts. For example, in the color space, the property of red is convex, since we don't see another color like blue interloping into the red region, which would appear as a small area of blue surrounded by a region of red.

2.1.5   Higher-order Conceptual Spaces

Higher-order conceptual spaces can be created from combinations and transformations of one or more lower-order conceptual spaces (Chella et al., 2008). The key notion of the higher-order nature of these spaces is that they are more abstract than the spaces they are generated from. This is to distinguish higher-order conceptual spaces from combinations that serve to more tightly constrain a space, similar to the intersection set operation. For instance, overlaying the full color space over the space of human phenotypes results in a restricted color space of human skin tones, not a more abstract space.

Since by nature, quality dimensions and domains often describe low-level quantities like color and sound, it is necessary to combine them into higher-level, more

abstract spaces possessing more explanatory power. Intuitively, the higher level a given conceptual space is, the richer its semantics, so to arrive at a space with sufficient descriptive capabilities for a given cognitive representation, it may be necessary to recursively abstract conceptual spaces into higher-order spaces to arrive at something nontrivial with interesting semantics.

2.2   Hilbert Spaces

"Hilbert spaces are the means by which the ordinary experience of
Euclidian concepts can be extended meaningfully into the idealized
constructions of more complex math." (Bernkopf, 2008)

Originally, conceptual spaces (Gärdenfors, 2004) were formalized by using Eu-
clidian or Manhattan distances to measure similarity in conceptual spaces modeled
in constrained Cartesian-like space. Though this is useful in building an intuition
as to how conceptual spaces operate in practice, it can be limiting in the de-
scription of relations between objects and the space itself. In order to allow for
more flexibility in this regard, we need a more general notion of a space than
a finite-dimensional Euclidian space. The generalization employed here is that
of Hilbert spaces (Kennedy & Sadeghi, 2013), which generalizes the pedestrian
finite-dimensional Euclidian space to infinite-dimensional spaces with arbitrary
geometry.

2.2.1   Complete Inner Product Space

A Hilbert space is defined as a complete inner product space. That is, a Hilbert
space is a vector space equipped with an inner product $\langle \cdot, \cdot \rangle$, but is also complete:
the space is big enough to include the norm of converging sequences. In the case of
an infinite-dimensional Hilbert space, this completeness criterion cannot be taken
for granted, but in the finite-dimensional case, the space is always complete. The
inner product of a Hilbert space induces a norm $\|f\| = \langle f, f \rangle^{1/2}$, which allows us to
talk about distances between vectors, something that we require in a generalized
formalism of conceptual spaces.

What makes Hilbert spaces powerful is the ability to represent a function as
a point in the space. With the aid of the inner product, one can produce an
(infinite) orthonormal sequence $\{\varphi_n\}_{n=0}^{\infty}$ for the Hilbert space. By decomposing
any function f into its Fourier series (equation 2.1) on that sequence, we can
recover a corresponding coefficient $\langle f, \varphi_n \rangle$ corresponding to each element of the
orthonormal sequence.

$$f = \sum_{n=0}^{\infty} \langle f, \varphi_n \rangle \varphi_n \tag{2.1}$$

By arranging each of these coefficients into a vector with dimensions φ_n, the function can be represented as a point in the Hilbert space. If that space is finite, we can equivalently represent a point x in the Hilbert space as an array of complex numbers, $x \in \mathbb{C}^n$ where n is the number of dimensions.

2.2.2 Application to Conceptual Spaces

Though the property of entanglement resulting from the use of Hilbert spaces has been used to study the combination of concepts (Aerts & Gabora, 2005) and meaning (Coecke et al., 2010), here, we use Hilbert spaces to define the geometry of the conceptual space by solely by specifying an inner product. This has two consequences. First, since for a given function, the Fourier series decomposition generates a vector representation, any number of different vectors can be produced from the same function for each inner product. This means that a given object can be represented in any number of conceptual spaces defined by their inner product.

Conversely, since it can be proven that all Hilbert spaces of the same number of dimensions are isomorphic (Kennedy & Sadeghi, 2013), given a vector, we can then choose an inner product to determine its representation. In this way, each inner product imposes a geometry that allows for different perspectives of the same "raw" data. By analogy, in figure 2.1 we see that coordinates represented on (x, y) in a 2D Cartesian space have a different meaning and location than if they are represented on (r, θ) in a 2D radial space. This allows for complete flexibility in representation, as an object can be placed in a conceptual space simply by interpreting its vector representation according to the inner product of that space. (Wiggins, 2018)
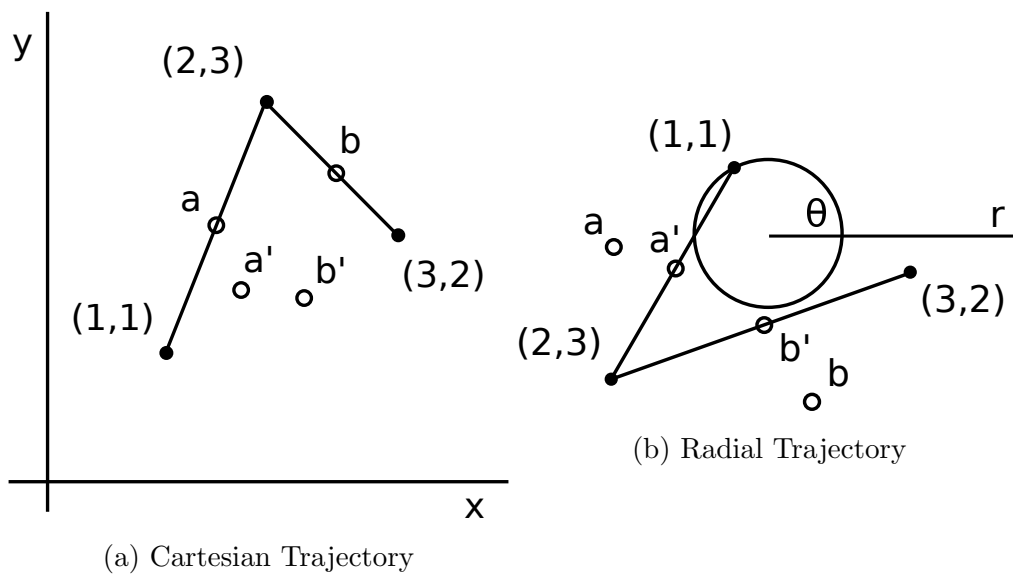
(a) Cartesian Trajectory

(b) Radial Trajectory

Figure 2.1: Effect of the inner product on spatial geometry

2.3　Information Theory

Shannon's information theory (Shannon, 1948), though originally aimed at providing a theoretical foundation to signal processing and communication, has become foundational in all aspects regarding information. According to information theory, the amount of information in a signal can be thought of as the amount of surprise at seeing a given quantity in that signal. Put more simply, suppose you come in late to work one morning, and upon arriving into the office, your boss tells you about the weather. No information was gleaned from the conversation, since you already knew the weather from being outside. If instead, upon entering the office late, your boss fires you on the spot, you would be very surprised, learning something unexpected, and gaining a lot of information.

2.3.1　Entropy

Entropy H is the measure of information in a probability distribution. If applied to a discrete signal of symbols, it can be used to represent the amount of information of the signal (equation 2.2), by taking it over the frequencies of each symbol s of the alphabet A. Entropy is then the number of bits required to represent each symbol without ambiguity.

$$H(A) = -\sum_{s \in A} p(s) \log_2 p(s) \tag{2.2}$$

2.3.2　Information Content

Though entropy is useful in describing the quantity of information of an entire distribution, to find the information content $h(s)$ of a single symbol in an alphabet, one must use equation 2.3 below (MacKay & Mac Kay, 2003). Since some symbols are less likely to occur than others, observing them gives more information than more likely ones, which is reflected here.

$$h(s) = -\log_2 p(s) \tag{2.3}$$

Whereas information content looks at a symbol in isolation, conditional entropy looks at symbols in pairs. Conditional entropy is equivalent to the information

content of a symbol, given that another symbol is known. If we are referring to a stream of symbols, the conditional entropy would be the information content of a symbol given that we just saw the previous symbol.

$$H(s|t) = -\log_2 p(s|t) \tag{2.4}$$

2.4 Information Dynamics of Thinking

2.4.1 Information Dynamics

In natural language processing, n-gram models are often employed to track the frequency of chains of symbols in a given signal (Sproat et al., 1996). For instance, if examining textual data, a unigram model would count the frequency of individual words, whereas a bigram model would track the frequency of pairs of words. It is clear that one can then utilize the information content measure on a unigram model and the conditional entropy measure on the bigram model in order to quantify the amount of information in that model (see section 2.3). Extrapolating further, higher-order n-gram models would model longer sequences of symbols, which could be analyzed using longer chains of conditional entropy. Considering a stream of symbols, one can then see that the information content and entropy of the stream will rise and fall according to the observed symbols. This is what is meant by *information dynamics*: the dynamic change in information over time and context.

2.4.2 Information-Efficient Cognitive Architecture

The Information Dynamics of Thinking (Wiggins, 2018) (IDyOT) is a cognitive architecture based on the principle that the human mind represents data according to the heuristic of information-efficiency. That is, in its internal representations, the mind seeks to use the least amount of bits to do so. In addition, IDyOT strives for a minimal set of processes and assumptions to describe cognitive phenomena, and yet still maintain the inherent dual quality of human memory: that representations in the mind are both learned and tied to temporal events with sequence and ordering, yet are also atemporal with regard to other situated concepts in the mind. IDyOT also does not make any assumptions about the form or structure of the data (Kemp & Tenenbaum, 2008), instead building semantics incrementally and hierarchically from the bottom-up as a probabilistic model of cognition (Chater et al., 2006).

An example of the IDyOT memory in action is shown in Figure 2.2 (Wiggins, 2018). In it, we can see the two parallel memories: sequential memory in the foreground and semantic memory in the background. These two memories are parallel in that the symbols seen in the sequential memory are tied to elements in
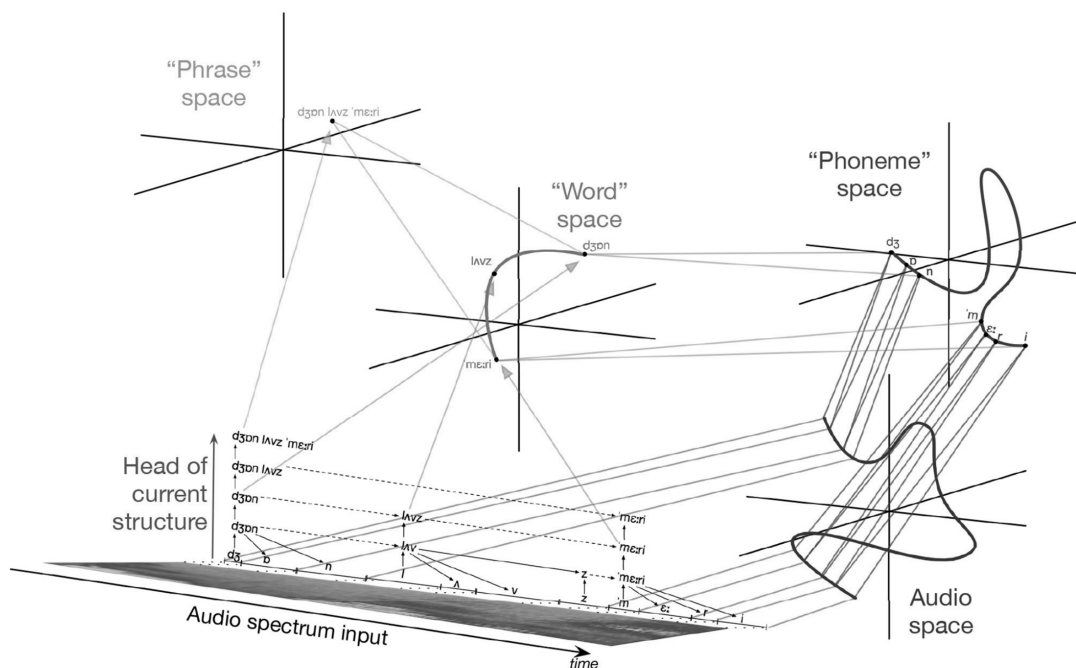
Figure 2.2: Illustration of IDyOT memory.
Reproduced from (Wiggins, 2018) with the author's permission.

the semantic memory such that the symbols maintain their temporal sequencing, but can be related atemporally in the semantic space. In addition, we see the hierarchical nature of representation in the diagram. Sequences of symbols in the sequential memory are chunked together in a subordinate layer into a single symbol in the superordinate layer. In the parallel semantic memory, the trajectory of this sequence through a semantic space is abstracted to a single element in the superordinate layer.

This hierarchical representation is driven by three processes, in the same vein as the non-conconscious processors of the Global Workspace Theory (Baars, 1993): segmentation, categorization, and abstraction. Segmentation operates on the sequential memory by splitting the stream of observed symbols into chunks, whereas categorization operates on the semantic memory by grouping similar instances in the space into categories. Abstraction ties these two processes together by taking the chunks formed in segmentation as trajectories of categories found in categorization, and transforming it as a single spectral representation in the superordinate

19

abstraction layer.

2.4.3 Boundary Entropy Segmentation

Since entropy represents the amount of uncertainty at what comes next, it makes sense that a jump in entropy would mark the beginning of a new semantic segment. For instance, when in conversation, at the beginning of a sentence, entropy is high because the listener has little idea what the speaker will say next. As the sentence proceeds, the listener will be better able to predict what comes next, maybe even being able to finish the sentence for the speaker, meaning entropy is steadily decreasing until the end of the sentence. Once the sentence is finished, the listener again is less sure what will come next, and so entropy rises. Therefore, at this rise in entropy, we would make a cut, resulting in a segment naturally representing the sentence just spoken. This method of segmenting a stream of data has been shown to be effective for cue prediction in music (Wiggins, 2010) and parsing language (Sproat et al., 1996).

2.4.4 Semantic Space Categorization

In the Information Dynamics of Thinking, the semantic memory – the portion of memory that represents time-invariant relationships – is represented as a collection of conceptual spaces (Gärdenfors, 2004). Elements in these spaces are spectral representations of trajectories of elements from less-abstract spaces (Chella, 2015), which themselves may be spectral representations. Being an instantiation of a conceptual space, regions in these semantic spaces correspond to properties or concepts, and so by grouping elements of the space together according to their similarity, the resulting categories correspond to the concepts of a conceptual space.

Since the IDyOT memory wants to be as information-efficient as possible in representation, the *maximum entropy reduction criterion* is used to ensure that the categorization of a space is maximally information efficient (Quinlan, 1983). However, since the most efficient categorization would be a single vacuous category, the *categorical convexity criterion* is used to ensure that categories are convex, for the same reasons concepts are convex in a conceptual space (Gärdenfors, 2004).

2.4.5 Hierarchy Construction

The hierarchical construction of abstraction layers (Chella et al., 2008) is performed by first taking the sequence of symbols of a chunk from the segmentation process in the sequential memory, then finding the trajectory of those symbols in the parallel semantic space. A spectral representation of that trajectory is then found and placed in the semantic space of the superordinate abstraction layer and given a label to serve as the symbol in the parallel sequential memory at that superordinate layer.

Since all three processes of segmentation, categorization, and abstraction occur at any level of abstraction, the result is a hierarchy of increasingly abstract layers, each composed of spectral representations of segments chunked by boundary entropy from the subordinate layer. In this manner, the higher layers are not only more semantically rich, being spectral abstractions from lower layers, but are information-efficient, being segmented and categorized according to their information-theoretic properties.

Chapter 3

Theory & Implementation

3.1 Abstraction

When a segment is produced in the segmentation process, it is abstracted to a single symbol in the superordinate abstraction layer. Since each dimension is composed of both an alphabet of symbols and a conceptual space in which those symbols live, a memory sequence is at once a segment of symbols in the sequential memory as well as a trajectory through the corresponding conceptual space in the semantic memory (Wiggins & Sanjekdar, 2019). Since here we will utilize the formalism of finite-dimensional Hilbert spaces to model conceptual spaces, this trajectory corresponds to a time-parametric curve in a high-dimensional space.

3.1.1 Fourier Transform

In the abstraction process, we would like to produce a spectral representation of a segment. Here, we will apply the discrete Fourier transform (DFT) operator (Cooley & Tukey, 1965) to the discrete trajectory (see equation 3.1), taking it from the time-domain n to the frequency-domain k. Though there are other spectral operators, there is evidence, especially in the auditory domain, that the brain operates on frequency transformations of time-varying signals from the Organ of Corti (Moore, 2012), and so it is employed here.

$$\mathcal{F}_D[x_n] = X_k = \sum_{n=0}^{N-1} x_n e^{-\frac{i2\pi}{N}kn} \tag{3.1}$$

By taking the Fourier transform (equation 3.2) of a time-varying signal $f(t)$, we produce the frequency-domain signal $F(\xi)$ for each orthogonal frequency. So by taking the Fourier transform of a curve in a Hilbert space, we end up with a spectral representation of that curve. The abstraction process will then consist of taking the Fourier transform of a trajectory through a Hilbert space, and then viewing the resulting frequency-domain signal as a point in the superordinate space.

$$\mathcal{F}[f(t)] = F(\xi) = \int_{-\infty}^{\infty} f(t) e^{-i2\pi\xi t} dt \tag{3.2}$$

3.1.2 Tensor Rank Promotion

The Fourier transform is an integral operator on Hilbert spaces (Kennedy & Sadeghi, 2013). Since operators, by definition, take a function as input and produce a function as output, when thinking in terms of finite-dimensional Hilbert spaces, an operator will take the input from one domain to another, but will not change the shape of that input. For example, if each point is represented as a vector, a sequence of those points would be a vector of vectors, i.e. a matrix. Since we view the result of the Fourier transform as a point in the superordinate space, in our example, the point is now represented as a matrix. Therefore, the result of abstracting a trajectory of points in a subordinate layer is a point in the superordinate space with one higher rank. Hence, each level of abstraction promotes the rank of its constituent tensors by one.

To clarify the first few stages of this process, consider Figure 3.1. The *Element* row represents the constituent elements of layer α. The *Trajectory* row strings those elements together, and the *Spectral* row shows the shape of the spectral representation of that trajectory after the Fourier transform. This spectral representation then becomes the constituent element of the superordinate abstraction layer $\alpha + 1$.

We begin at the base abstraction layer with a space filled with the raw discrete scalar signal values for each moment of time. By lining up these values and taking their Fourier transform, the resulting spectral representation is a vector. This spectral representation is placed into the $\alpha = 1$ abstraction layer, whose space is filled with points representing the frequencies of a signal of a short moment of time. These points are vectors (rank 1 tensors) that, when strung together into a trajectory, form a matrix. By taking the Fourier transform of this matrix and viewing the result as a point in the superordinate abstraction layer, that upper layer is now composed of points represented as matrices (rank 2 tensors). Again, we string these points together as a trajectory, but now forming a cube. By taking the Fourier transform, and viewing it as a point in the superordinate layer, that layer is now composed of points represented as cubes (rank 3 tensors). Repeating this process, we then get hypercubes of increasing rank at each layer of abstraction. In this manner, the rank of a representative tensor increases by one for each layer of abstraction.
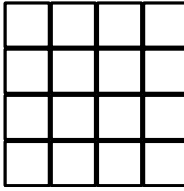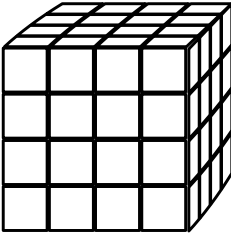
24
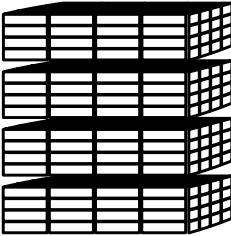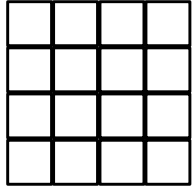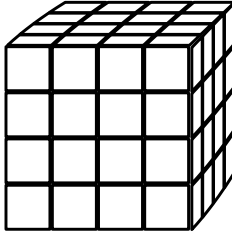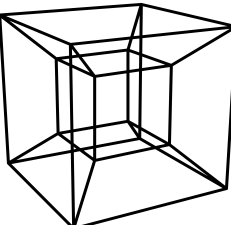
Layer	α = 0		α = 1	α = 2
Tensor	Scalar	Vector	Matrix	Cube
Element				
Trajectory				
Spectral				

Figure 3.1: An illustration of tensor rank promotion for the first 4 levels of abstraction, using an interpolation resolution $r = 4$.

Unfortunately, this leads to an exponential explosion in the number of elements constituting a point in a given layer. Formally, the number of elements is r^α, where r is the resolution of interpolation (see section 3.4), and α is the level of abstraction. Though this is not a problem in theory, it has consequences in terms of implementation.

3.1.3 Element-wise Independence

When performing the Fourier transform on a tensor of any rank, it should be noted that each component in the tensor is independent from every other component.

Though this is not necessarily true of tensors in general, due to the particular hierarchical construction of these spaces by means of the DFT, each component is decoupled from the rest. Starting at the bottom, the time-domain sound signal is transformed into a frequency-domain signal of coefficients in independent frequency bins. It is this independence that allows us to represent the short-term frequency signal as a point with those frequency bins as dimensions. Performing the DFT on the trajectory of frequency vectors results in independent frequency bins filled with vectors that possess independent entries, meaning all elements are independent one another.

Hierarchically performing the DFT at each abstraction layer on tensors with independent elements results in higher rank tensors with independent elements. This element-wise independence of the tensor means that the DFT should be taken element-wise – that is, across the base "time" axis – since all other cross-component terms would involve orthogonal dimensions.

3.1.4 Frobenius Norm

The norm chosen for this implementation is the Frobenius norm (Horn & Johnson, 1990). This norm corresponds with the familiar Euclidian norm, but since we are operating more generally on tensors, and not just vectors, the Frobenius norm is employed instead. This norm corresponds to a flat space where each dimension behaves linearly and equally in relation to the other dimensions. The Frobenius norm is intuitive for visualizing the space, and so serves as a good norm to begin exploration. Future research will investigate different inner products and their induced norms to impose different geometries on the representations.

$$\|A\|_F = \sqrt{\sum_{i=1}^{m} \sum_{j=1}^{n} |a_{ij}|^2} \tag{3.3}$$

In equation 3.3, the Frobenius norm is shown for a matrix. One can then imagine that for a cube, there would be three summations, four summations for a hypercube, and so on. Therefore, the Frobenius norm comes to be a sum over each element of the tensor.

3.2   Segmentation

Segmentation is the process operating on the sequential memory that divides an input stream into discrete chunks according to an information-theoretic difference function. (Wiggins & Sanjekdar, 2019)

3.2.1   Difference Function

The difference function operates on the stream of symbols entering a given dimension and decides where segments begin and end, and thus where a cut in the stream should be made. Primarily, we track the moving information content and conditional entropy of the stream. If the information content or entropy rises, then a cut should be a made, marking the end of the current segment and the beginning of a new one.

$$\Delta(\tau) = \begin{cases} true & \text{if } H(\tau - 1) < H(\tau) \vee h(\tau - 1) < h(\tau) \\ false & \text{otherwise} \end{cases} \tag{3.4}$$

This segmentation process results in two problems stemming from the two types of sparsity inherent in the segmentation and subsequent abstraction when working in continuous, high-dimensional spaces: symbol sparsity and content sparsity.

3.2.2   Symbol Sparsity

Symbol sparsity arises from the length of the segment produced in the segmentation process. Since the abstraction of the segment is its spectral transform, and we use the DFT to create the representative symbol in the superordinate abstraction layer, we run into precision problems due to the uncertainty principle of signal sparsity (Robertson, 1929). When performing a DFT, the signal precision is limited by the number of non-zero coefficients in either the time or frequency domain. Therefore, if the number of symbols in a given segment is small, its spectral representation will be imprecise. Therefore, it is necessary to perform interpolation (see section 3.4) to fill out the signal, so that high precision is maintained in the spectral transformation.

3.2.3 Content Sparsity

Content sparsity is due to the nature of the symbol contents being represented in a continuous, high-dimensional Hilbert space. Since in the case of audio perception, the content of a symbol is a high-dimensional tensor of complex coefficients, not only are there a high number of dimensions, each dimension has an uncountably infinite number of possible values. Therefore, it is incredibly unlikely that any two symbols have exactly the same value for every dimension in the conceptual space in which they live.

This poses a problem for determining the entropy and information content of symbols, which rely on the probability of seeing a symbol. Since we estimate this probability by counting the frequency of observed symbols, if every symbol is unique, in the limit, the probability of seeing one in the signal is 0, meaning the information content of each symbol in the signal would be 0 as well. In frequentist terms, talking about the the probability of a unique event is meaningless, and therefore, referring to a unique symbol's information content or entropy is also meaningless.

Though the space in which the symbols live is so sparse, we would still like to say that if two symbols are close enough together in space, for all intents and purposes, they are the same symbol. This is accomplished through categorization (see section 3.3), where a label is attached to each symbol according to its category. If two symbols have the same label, then they are equal, even though they may have different contents – that is, different values in their representative complex tensors.

3.3   Categorization

Beyond the necessity for categorization due to the problem of content sparsity (see section 3.2.3), categorization of representations is necessary to model the properties and concepts determined by regions of conceptual spaces (see section 2.1.4). Since we cannot *a priori* know the bounds of the regions of concepts in a conceptual space, we instead build them up through categorization, discussed here.

3.3.1   Information Content Reduction Criterion

Since the goal of an IDyOT is to be as information-efficient as possible in its representation of concepts, the primary way to do this is categorize two different symbols together if they lead to an overall reduction in information content of the space. However, if this reduction by information content measure was the only method used to determine categories, there would be nothing to stop all symbols from being categorized together. If all the symbols are the same, the information content is maximally reduced, but the result is a meaningless stream of monotony. Obviously, this is unrealistic and undesirable.

3.3.2   Categorical Convexity Criterion

To push back against the reduction by information content is the categorical convexity criterion. In section 2.1.4, we saw that categories correspond to convex regions in a conceptual space. In Hilbert spaces, categories would then translate to an infinite-dimensional hyperellipsoids. In IDyOT, the convexity criterion guarantees that for any two symbols in a given category, there is no symbol from a different category between those two symbols.

Though this criterion is simple in formulation, the definition of what *between* actually means can vary depending on the space in question. Even when the space is unidimensional, the definition of *between* is somewhat arbitrary. For instance, take a space that is wrapped around a circle. Any point is between any other two points, depending on which direction around the circle you move. Things get even less clear when moving into higher dimensions, where oftentimes, even a partial natural ordering is impossible.

Therefore, instead of looking at betweenness at all, we incrementally build up

categories by way of an inclusion radius ρ around each point (Wiggins & Sanjekdar, 2019). If another point falls within the inclusion radius of a category, that point is grouped into that category. In this way, we can ensure that there is never an interloper in a category, since if it was intruding on the region of the category, it would already be a member.

3.3.3 Adaptive Categories

Since we utilize an inclusion radius to determine the categorization candidates for a new point, this would naively result in a partition of the space in which all categories are the same size, since they all use the same radius. However, the categories of a given conceptual space are likely to be variable in size, and therefore we require a mechanism to adapt the size of a category according to the observations as they are added to the space.

In order to ensure this adaptability, we maintain a mean μ and variance σ^2 of a multidimensional Gaussian prior distribution $\mathcal{N}(\mu, \sigma^2)$ for each category. Whenever a new point x is added to the category c, we perform a posterior update, which becomes the new prior of the category in future categorization. In using a Gaussian prior, the centroid of the category corresponds to the prior mean, whereas the radius of the category corresponds to the $\|3\sigma\|$ (or 99.7% of the volume of the Gaussian) which can be found from the prior variance σ^2.

As more instances are added to a category, its mean will converge to a more somewhat stationary mean, and its variance will tend to decrease, corresponding to a reduction in the radius, though this will happen less if the constituent points of a category are spread out. This makes the size of the category adaptive to its observed members and allows different categories to have different volumes. By using the spherical Gaussian prior, the boundary-convexity of each category is ensured.

Since points are being added to the space incrementally, the posterior update of the category is also performed incrementally (Murphy, 2007) by maintaining not only the prior mean and variance, but also sample mean and variance for each category. Therefore, whenever a new point x_t is added to a category, the moments of the distribution are updated as follows:

$$m_t = m_{t-1} + \frac{x_t - m_{t-1}}{n_t} \qquad \text{(Sample Mean)}$$

$$s_t^2 = s_{t-1}^2 + \frac{(x - m_t)(x - m_{t-1}) - s_{t-1}^2}{n_t} \qquad \text{(Sample Variance)}$$

$$\mu_t = \frac{\mu_{t-1}s_t^2 + x\sigma_{t-1}^2}{\sigma_{t-1}^2 + s_t^2} \qquad \text{(Posterior Mean)}$$

$$\sigma_t^2 = \frac{\sigma_{t-1}^2 s_t^2}{\sigma_{t-1}^2 + s_t^2} \qquad \text{(Posterior Variance)}$$

3.3.4 Maximum Information Content Reduction

When a new point is added to the space, there may be multiple candidate categories for which the point falls within their inclusion radius. When this happens, the heuristic of maximum information content reduction is employed. Put simply, the new point is grouped with the category with the highest information content, as including the point in that category results in the largest reduction in information content out of the candidates. Unfortunately, this results in a violation of the categorical convexity criterion due to the restriction in this implementation that each point must belong to exactly one category.

Instead, the union of all candidate categories for the point could be combined to create a single category representing all the points of the constituent categories. This has the benefit of both greatly reducing information content while still maintaining categorical convexity. However, in practice, there is enough incidental overlap between what should be distinct categories that a large amount of points get categorized together which should not be, resulting in a small number of large, but meaningless groupings. This union categorization scheme is a likely avenue of future research.

3.4 Interpolation

When talking about trajectories through conceptual spaces, we refer to interpolation as the use of representative virtual points to perform the abstraction process instead of the actual values of the sequence. Since we formalized semantic spaces as Hilbert spaces, we can analogously talk about drawing a time-parametric regression curve through points in the space. Hence, the process of interpolation is simply sampling from a regression.

3.4.1 Signal Sparsity

As discussed in section 3.2.2, interpolation is necessary to deal with symbol sparsity in a given segment. To reiterate, this problem arises due to the use of the DFT in the abstraction process. Since the precision of the transform is determined by the number of non-zero coefficients describing that signal, if the segmentation process results in a relatively short segment, the number of coefficients will be small, and the resulting transformation imprecise. The way to deal with this imprecision is fill in coefficients such that the transform of the signal accurately represents the original signal, thereby producing a higher precision transform. This is valid because we can think of the sparse signal as a sequence of samples from a function, and by regressing on those samples, we can obtain an approximate, but representative function of that segment.

3.4.2 Hilbert Space Isomorphism

One fundamental aspect of Hilbert spaces is that all spaces of the same number of dimensions, even infinite, are isomorphic to all other Hilbert spaces of the same number of dimensions (Kennedy & Sadeghi, 2013). Since we are using finite-dimensional spaces, to allow comparison between two points, we need their representative tensors to not only be of the same rank, but for each rank of the tensor to be the same size, so that the compared points have an equal number of dimensions. Put another way, all points of a given space must have the same shape: that of the space.

Here, the problem arises during segmentation. Since each segment in a given abstraction layer can be of arbitrary length, but we want the spectral transforma-

tions of all segments in a subordinate space to land in a single superordinate space, the transformation of the each segment must be taken over the same number of points. Interpolation allows us to infer a curve through the points of a segment and place the required number of virtual points on that curve such that all segments have the same number of virtual points. Then by taking the Fourier transform of these virtual points, we can be assured that the resulting spectral representation has the exact shape required to land in the space of the superordinate abstraction layer.

3.4.3 Regression Sampling

In order to realize this process, we first regress through the available points of a segment. In order to regress through the points, each point is assigned an index according to its relative place in the trajectory, and regression is performed through that series. It is important to maintain the relative "lengths" of each point in this trajectory, where the length corresponds to the length of the segment of which this point is a spectral transformation in the subordinate layer. Since segments are likely to be of different lengths, in this way, points at higher abstraction layers still maintain a sense of their connection to the base layer's time domain, yet still being time-invariant. By spreading the index of a giving point according to its length for regression, this sense of relative length is maintained.

With the points now arranged according to their relative lengths, regression can be performed. Though the proper regression technique is still open to future research, in this implementation, Gaussian process regression (Williams & Rasmussen, 1996) is employed. Once the regression curve is found, the number of points according to the resolution hyperparameter r is taken at equal intervals from the curve. These virtual points will then be used in the spectral transformation of the abstraction process.

It should be noted that the value used for the the actual points in the regression are not of the points themselves, but rather of the centroid of the category that each point belongs to. Therefore, when performing the regression, we are drawing a curve through categories instead of drawing a curve through instances. This has the effect of making the spectral representations of similar trajectories consistent, thereby allowing them to land in the same region in the superordinate layer.

3.4.4   Gaussian Process Regression

One method for creating a smooth curve through these points is Gaussian process regression (Williams & Rasmussen, 1996). By thinking of a sequence of points of a trajectory as a being drawn from a multivariate Gaussian distribution, we can think of a distribution of functions, i.e. a process, through those points. The expectation of this process is the maximum a posteriori (MAP) function of the distribution, resulting in the highest likelihood trajectory through a sequence of points. This method automatically avoids problems of overfitting inherent in higher-order transformation linear regression methods, though has the problem being prohibitively computationally expensive for a large number of points. Fortunately, in this implementation the number of points is equal to the resolution of interpolation, which is quite small.

By utilizing the squared exponential kernel for Gaussian regression, the resulting trajectory will not only by the MAP curve, but also infinitely differentiable, meaning that it is maximially smooth. This smoothness pays dividends when taking the Fourier transform of the trajectory, as the less smooth a sampled signal is, the larger the amplitude of high frequency artifacts of the Fourier transform become. We would like to avoid these artifacts, making this method of regression ideal for an interpolated spectral representation.

Chapter 4

Empirical Analysis

Resolution r	Layer $\alpha = 0$	Layer $\alpha = 1$	Layer $\alpha = 2$	Layer $\alpha = 3$
16	0.25 KB	4 KB	64 KB	1 MB
32	0.5 KB	16 KB	512 KB	16 MB
64	1 KB	64 KB	4 MB	268 MB
128	2 KB	262 KB	34 MB	4.3 GB

Table 4.1: Memory Requirements for a single category with elements composed of 128-bit complex numbers

4.1 Methodology

4.1.1 Parameterization

The parameterization of the implementation comes from two parameters: the resolution of interpolation r, and the initial radii ρ_0 of a category for each abstraction layer. The resolution of interpolation is set to equal the sample window size used to slice the waveform, resulting in each rank of the tensor to be the same size at any given level. Though we expect that a larger resolution will result in semantically richer categories in the higher levels, the exponential explosion in the size of the representative tensor, due to tensor rank promotion (see section 3.1.2), results in exponentially increasing memory requirements (see table 4.1). This limits the resolution to $r = 16$ samples in our implementation. However, it was found that below a certain resolution, the results are somewhat similar. For instance, a 32-sample resolution behaves similarly to the 16-sample resolution, while having significantly higher processing and memory requirements.

The initial radius ρ_0 for a new category in a given dimension has the largest effect on categorization within that dimension, and therefore also affects the segmentation of that layer. Though the adaptive categorization method will alter the category volume according to its members, the initial radius will determine how willing a dimension as a whole is to categorize new instances. If this initial radius is too small, then nothing will be categorized, and therefore nothing can be segmented. If it is too large, then instances that should not be categorized together will be. Due to the exponential growth in the number of elements in a tensor at each level, the initial radius parameters also grow exponentially. Specifically, at

zero-indexed abstraction level α, the initial radius was manually set to $\rho_0 = 10^\alpha$.

4.1.2 Data

In order to evaluate the implementation on human speech, the TIMIT corpus (Garofolo et al., 1993) was used. This dataset has a large variety of American accents speaking a specific set of nonsense phrases that is meant to be used for natural language processing purposes. Though having a large variety of speakers on which to train in order to diversify the semantic memory is useful, more importantly, each speech clip in the dataset is accompanied by annotations connecting words and syllables to a specific sample in the clip. This allows us to compare the annotations and spectrogram of a clip with the categories of each layer in one unified visualization. This allows us to examine if, for instance, categories at higher levels are associated with certain words or syllables.

Since the word and syllable annotations mark qualitavely different sounds, these boundaries can be compared with the resulting categories at each level of abstraction to investigate if category boundaries align with word or syllable boundaries. If they do consistently, then this would be confirmation that the segmentation, categorization, and abstraction processes are working as envisioned in the theory.

4.1.3 Experiments

The IDyOT was trained on a 10-clip set from a single speaker, ranging from 1.5 seconds to 4 seconds, with most clips being around 3 seconds. With a resolution of $r = 16$ samples, this test set results in over 26000 distinct slices of speech to be analyzed. Though, theoretically the maximum abstraction layer is unknown, but likely quite large, here it was capped at the fourth level $\alpha = 3$ due to processing and memory limitations. The resulting categorization in semantic memory was then used to seed a sequential memory from a single clip of the same speaker.

Here the speaker is a young woman from somewhere in New England, perhaps a borough of New York or Boston, identified as TRAIN-DR1-FCJF0 in the TIMIT corpus. After training on all available audio for this speaker, the SA1 clip was used for analysis, which has the same speaker reading the nonsense line: "She had your dark suit in greasy wash water all year."

In an earlier iteration of the implementation, a larger 53-clip training set containing a variety of different accents and sentences was used to train the IDyOT, but the resulting visualization (see section 4.2) was not qualitatively different than the 10-clip set, while actually being more difficult to inspect due to the increased number of categories from the larger set. Therefore, the 10-clip training set was used for visualization.

Layer α	Instances	Categories	% Reduction
0	2841	605	78.7
1	1738	601	65.4
2	824	438	46.8
3	356	319	10.3

Table 4.2: Category Reduction

4.2 Results

4.2.1 Category Reduction

Looking at table 4.2, we see that at each layer, there is a reduction in the number of symbols represented at that abstraction level. At level 0, we see the largest reduction from instances to categories of 78.7%. A large portion of this reduction is due to the categorization of 'silence' – that is, inaudible white noise – that, from moment to moment, has distinct values that are extremely close together in space. This can be seen at the beginning of the clip in Figure 4.1.

As the level of abstraction increases, the number of instances in that layer decrease due to segmentation. In addition, categorization still occurs at each level, though the degree at which instances are categorized together also decreases. This is due most significantly to the 'silence' categories being hierarchically categorized together into fewer categories, though, as we will see in the following sections, other non-silence categories are being grouped together as well.

4.2.2 Category Flow

The Category Flow diagram (Figure 4.1) illustrates the hierarchical categorization process. At each level, the word and syllable annotations boundaries are overlaid on top of the categories to easily discern where a certain sound should approximately begin and end. The spectrogram is also displayed for visual comparison of the annotations and the actual signal. In the category flow, each category is represented for its duration in the clip for each abstraction level, where each category has a spot on the y-axis – that is, a category corresponds to a horizontal
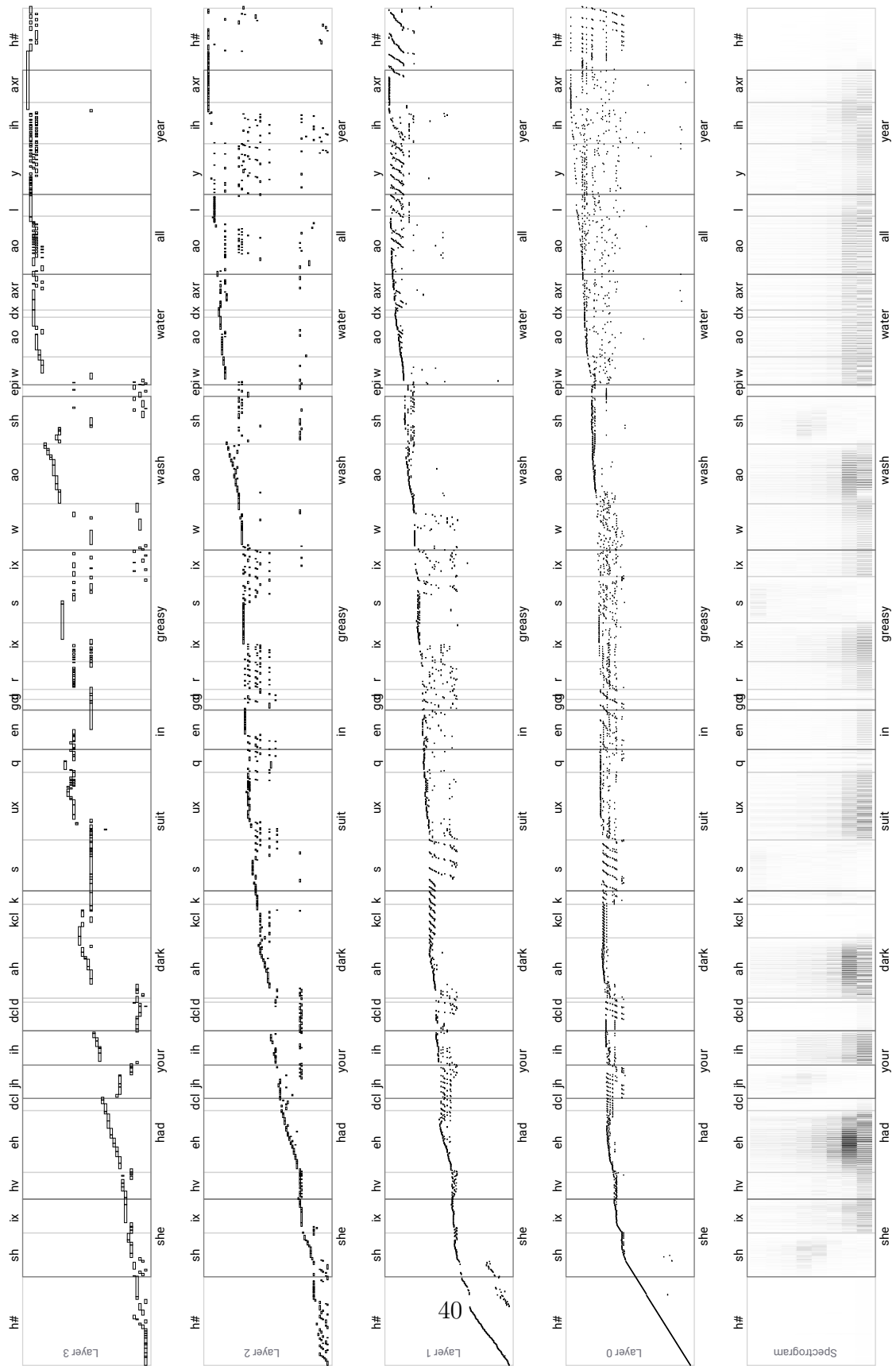
Figure 4.1: Category Flow of layers 0-3 with a Spectrogram

line on the plot. In order to illustrate how transitions in a subordinate layer are abstracted into categories in the superordinate layer, the categories are ordered according to their maximum pair frequency such that categories that often appear in sequence are next to each other vertically.

In the category flow, we see that lower levels are quite noisy, with nearly all categories persisting for a single moment. As we move up in abstraction, the flow flattens out and solidifies, meaning that trajectories from the subordinate layers are being categorized together in the superordinate layer. As the abstraction level increases, we see a tendency for transitions between categories to flatten out into a smaller number of categories, where at the higher levels, these transitions flatten out completely into just one category. By solidifying, we mean that the dispersed, noisy transitions found in the lower layers are categorized together in the upper layers, resulting in a "denser" category flow.

This process begins a bit in layer 2, but occurs much more frequently in layer 3. This is most easily seen in the silent portion, marked $h\#$, at the beginning of the clip. Since this syllable is not actually silent because it contains faint white noise, the lower layers categorize these different moments of white noise separately, resulting in the ramp in categories at level 0. Moving up the abstraction layers, their trajectories are categorized together into just a handful of categories at level 3. This is evidence that the spectral representations of trajectories are consistently being categorized together if the trajectories they are representing are similar.

We can also see a few sections at level 3 where the categories or trajectories nearly line up with the syllable annotation boundaries. For instance, the 's' syllable in the word 'suit' has been categorized into one category at this level, and the trajectory for the 'ao' syllable in the word 'wash' is clearly demarcated at the annotated boundaries. One would expect that at the next level of abstraction, these trajectories would then become categories.

When examining all the abtractions layers as a whole, one can see noisy category transitions at the bottom layers become more consistent at the upper layers. Even more, we can see that similar trajectories in lower layers are categorized together in upper layers. This is evident in the 'striping' seen in lower layers – perhaps a frequency beating artifact of the recording – becoming a single category in upper layers. This is seen most readily in the 's' syllable of the word 'suit', which exhibits striping at the lower levels. Since these transitions are very similar, their
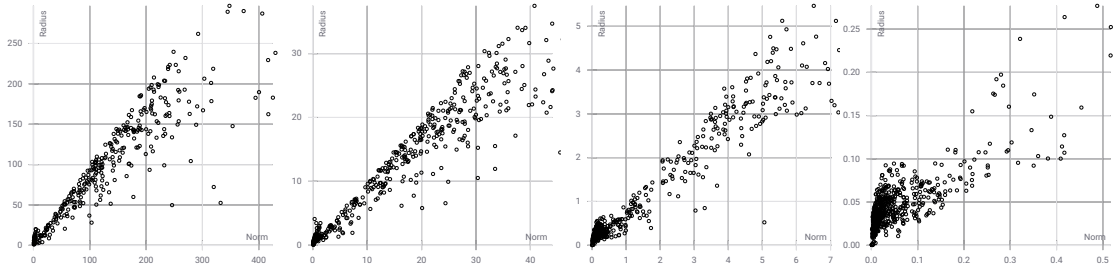
Figure 4.2: Category Distribution of layers 0-3 going from right to left

trajectories and therefore spectral representations should be similar. Indeed, we
see that at the upper levels these similar transitions are all categorized together, in-
dicating that the categorization of spectral representations of trajectories behaves
as expected as outlined in the theory.

4.2.3 Category Distribution

Since, especially at the higher levels, visualizing these high-dimensional categories
is nigh impossible, the category distributions plot (figure 4.2) shows the distribu-
tion of categories for a given dimension by showing how far away a given category
is from the origin (i.e. the norm of the centroid of a category) on the x-axis,
and the radius of the category on the y-axis. This means points to the right of
the distribution represent categories that are in general more distant from other
categories, and points toward the top will have a larger inclusion radius ρ and
volume.

Looking at all of the distributions, one can immediately see a positive corre-
lation between the remoteness (i.e. distance from the origin) of a category and
its volume. At each abstraction level, we see most categories are close together
with a small volume, and as they move away from the origin, there volume also
increases. This is to be expected since the adaptive categorization tends to reduce
the volume of categories with many members, and there are more instances near
the origin than away from it. Therefore, looking at the space as a whole, we see
that the category density is high near the origin, and more sparse as one moves
out.

It is also important to note that the adaptive categorization method employed
here is not overly constraining the radii of each dimension. If it were, instead of

42

the linear trend we see here, the trend would shoot up to initial radius ρ_0 of the dimension and flatten out in an inverted-L shape. This top-flattening not only results in most categories having nearly the same volume, but would also result in a larger amount of non-categorizable regions in the space; both of these qualities are undesirable. In fact, the maximum posterior radius for each level is well below the initial radius, indicating that it is not a limiting factor in the categorization. This shows that the adaptive categorization fills the space quite well, though more analysis would need be done to determine the analytical extent of the spatial covering, left to future research.

4.2.4 Category Similarity

Since it is difficult to discern in the category flow (figure 4.1) if there is consistent categorization of similar sounds across the whole clip, the category similarity matrix (figure 4.3) for abstraction level $\alpha = 3$ can be used instead. At each moment in the clip the current category is compared with the category of every other moment in the clip, finding their distance, and representing closer categories with lighter shades and distant categories with darker shades. Therefore, we see that diagonal is white, since the category of a moment has zero distance to itself. Both axes of the matrix are also annotated with the words and syllables, like in the category flow, so that the similarity between different words and syllables of the clip can be compared.

Looking for light areas off the diagonal, there are three main areas of interest. The first is at the intersection of 'she' and 'greasy'. Clearly, the syllables of these words sound similar, so it makes sense that this area is lightly shaded and white. Another white area is at the intersection of the words 'had' and 'in'. Though in most accents these syllables of these words sounds quite different, the particular New England accent of this speaker has these words sounding similar – something near 'hed' and 'en'. The other off-diagonal white area to call out comes at the intersection at the end of the words 'dark' and 'suit'. For this accent, the ends of both of these words are unpronounced and are replaced with a stop, meaning they also qualitatively sounds quite similar.

Since similar sounding syllables are either in the same category or very close categories, this is evidence that categories semantically representing human speech
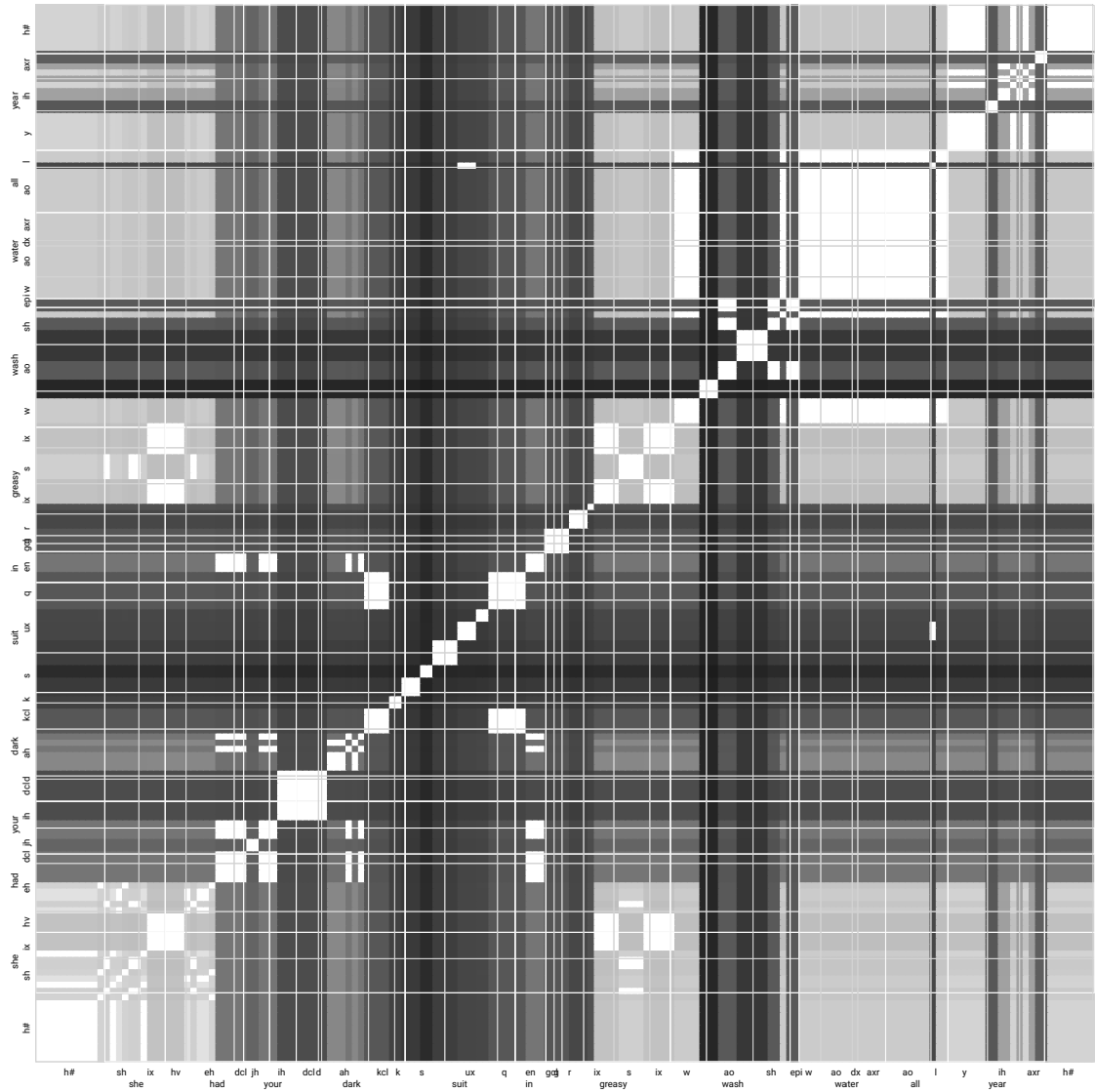
Figure 4.3: Category Similarity of Abstraction Layer 3

syllables are beginning to emerge at this level. Note that this emergence is occurring without any knowledge of this domain being human speech data, meaning that the general IDyOT processes behave as expected without specifying domain-specific information in the model.

Chapter 5

Evaluation & Discussion

5.1 Discussion

5.1.1 Addressing Categorical Inconsistencies

Though the category similarity matrix confirms that some similar-sounding syllables are either categorized together or at least near each other in the space, there are also confusing portions. For instance, the entire word 'water' is categorized into a few very similar categories. At higher levels of abstraction than seen here, one might expect that whole words would compose a single category, but since at abstraction level 3 we seem to be still mostly categorizing syllables and diphtongs, we would not expect the entire word to be one category.

This over-categorization may be due to two issues. The first may be that the categorization is too eager in this location, resulting in different points being grouped together that should not be. Though a significant amount of tuning was performed to find good initial radii for each layer, it's possible that a lower radius at this level would result in a more intuitive categorization for this particular region.

On the other hand, it may instead be due to the segmentation scheme employed in this implementation. Here, we segment the stream at either a rise in entropy *or* a rise in information content, resulting in numerous, relatively short segments. Though there are a large number of segments, any given segment will not contain much information, by nature of its short length. For a less eager segmentation, we could segment at a rise in just one of the measures. This would result in fewer but longer segments with more information per segment. Though there would be fewer resulting categories from this reduced scheme, each category would be richer in that it represents more information, and may result in categories that consistently represent full syllables at this level of abstraction.

5.1.2 Meaningful Categorizations

That being said, though not perfect, we do observe the emergence of a few consistent categories that represent syllables at abstraction layer 3. Since we also clearly see that similar trajectories are being categorized together in higher layers, it is a tenuous confirmation that the abstraction process as set forth in the theory behaves as intended. We can see that even at a relatively low abstraction

level 3, the hierarchical spectral representations of the speech signal are coalescing into categories that could be said to represent human speech sounds. One could imagine, given a few more levels of abstraction and significantly more training, that the higher-level abstraction categories would become consistent, meaningful representations of the human speech signals they are learned from.

5.1.3 Unified Approach to Perceptual Representation

What makes these results exciting is not just that we are beginning to see the emergence of syllables as discrete categories, but the general applicability of this method. Not only might we expect to see more complex elements of human speech such as words, sentences, and even syntactic structures (Chomsky, 2002) start to emerge as categories in higher levels of abstraction, but the processes of segmentation, categorization, and abstraction are agnostic about the domain of the data in operation. That is, instead of using human speech signals as the raw input, another audio domain such as music could be used, as in IDyOM (Pearce, 2005). In general, just about any time-varying signal could be used instead. Since other modes of perception such as vision can be modeled as time-varying signals, if consistent categories can be found for these human speech signals, perhaps semantically rich categories could be found for these other domains as well. Doing so would uncover related semantics from different areas of perception, resulting in a more holistic coginitive architecture that ties in the multitude of domains of human experience.

Chapter 6

Conclusion

6.1 Contributions

There are a number of significant contributions from this thesis that move toward a formal implementation of the Information Dynamics of Thinking theory, which up until now, and not counting other IDyOT-like systems (Forth et al., 2016), was almost entirely a theoretical proposition. Specifically, a significant portion of Chapter 3: Theory & Implementation is work contributed by the author, though of course with guidance from the promoter and advisor, Prof. Dr. Dr. Geraint Wiggins. A short review of specific contributions is presented in the following paragraph.

Tensor rank promotion (section 3.1.2) and element-wise independence (section 3.1.3) of the tensors produced in the abstraction process, though falling directly from the theory, were identified by the author. Though content sparsity was known prior in the theory, the problem of signal sparsity (section 3.4.1), in conjunction with the necessity for equal-shaped tensors falling from Hilbert space isopmorphism (section 3.4.2) led to the need for interpolation (section 3.4) and its proposed solution in regression sampling (section 3.4.3). In addition, the adaptive categorization (section 3.3.3) technique expanded on the idea of an inclusion radius and made it adaptive to the constituents of the category.

Beyond these theoretical contributions, this is the first, albeit limited, realized implementation of IDyOT that utilizes all three main processes: abstraction, segmentation, and categorization. As seen in the results, this is the first empirical analysis that spectral representations of meaning in the context of IDyOT, at least in human speech, can produce identifiable representations in a hierarchical manner. This highly novel and general approach to perceptual representation could have far-reaching consequences, but here was the first empirical result that the whole theory might be viable.

6.2   Limitations

6.2.1   Exponential Size of Category Representation

The primary limitation encountered was due to the problem of tensor rank promotion (see section 3.1.2). With the brain being massively parallel, its possible that this exponential growth in the size of representation and the resulting spectral transform, is not a problem for the human mind. However, even with significant parallelization of the categorization process, and using the Fast Fourier transform (Cooley & Tukey, 1965) to group the spectral representations, the memory requirements alone became beyond extraordinarily large.

To ameliorate this, we repeatedly reduced the resolution of interpolation to hinder this growth, as well as capping the highest abstraction level to 3. Unfortunately, this may have resulted in categories that are less informative overall, and we were not able to investigate higher level of abstraction, which would potentially yield consistent, semantically rich categories.

6.2.2   Categorization Schemes

The other major limitation was due to varying schemes that were tested in order to implement the categorization process. Though signficant effort was put into finding different methods to not only adapt to the expontial growth in representation, but also the nature of the raw signal input, this was found to be too difficult to do *a priori*. Therefore, the adaptive categorization scheme (see section 3.3.3) was devised with the initial radius hyperparameter to limit the amount of manual tuning necessary for reasonable categorization. That being said, the initial radii still had to be manually tuned to fit each level of abstraction for this particular type of input signal.

In addition, as larger training sets were run on the system, there was a corresponding increase in the number of categories produced. This has a marked effect on the speed at which a new instance is processed since all categories in a space must be checked for candidacy. In the future, consolidation techniques (Wiggins & Sanjekdar, 2019) can be employed to limit this monotonically increasing number of categories in a given layer.

6.3 Future Work

6.3.1 Inner Products and Spatial Geometry

One of the most interesting proposals of IDyOT theory is that, since the inner product determines the geometry of the space, we can impose different semantics on a space simply by employing a different inner product (Wiggins, 2018). In this implementation, the only inner product used corresponds to the Frobenius norm (see section 3.1.4), so future work is needed to examine how the categorization, interpolation, and abstraction processes implemented here will be effected by choosing inner products that correspond to different semantics of the space. For instance, there is evidence that humans represent the pitch space as a spiral (Deutsch, 2013) and represent the color space as a spindle (Sivik & Taft, 1994). Inner products could be chosen for a semantic space so that its geometry conforms to these semantics.

6.3.2 Spectral Projectors and Reification

One of the main problems of this formalism comes from the use of the Fourier transform operator as the method of producing a spectral respresentation in abstraction. Since tensor rank promotion results in exponentially larger representations of a category, the processing and memory requirements for a given category become huge after only a few levels of abstraction. If instead a different formalism is used for a spectral representation instead of the Fourier transform, but can avoid tensor rank promotion, then we retain the spectral time-invariance necessary for abstraction without the computational overhead. Specifically, if instead of using a spectral *operator* like the Fourier Transform, a spectral *projector* could be used. This would mean that the spectral representation of a trajectory would remain in the same rank Hilbert space at each level of abstraction, thereby avoiding the problem of tensor rank promotion.

An exciting corollary of using a projector instead of operator is that since the abstraction of a trajectory would land in the same Hilbert space, but separate conceptual space, a reification process could be defined. By mapping that spectral representation back onto its subordinate layer, one could examine an reified abstraction of a trajectory in the same space as that trajectory!

Bibliography

Aerts, D., & Gabora, L. (2005). A theory of concepts and their combinations ii: A hilbert space representation. *Kybernetes*, *34*(1/2), 192–221.

Baars, B. J. (1993). *A cognitive theory of consciousness.* Cambridge University Press.

Bernkopf, M. (2008). *Schmidt, erhardt.* Retrieved 2019-07-12, from https://www.encyclopedia.com/science/dictionaries-thesauruses -pictures-and-press-releases/schmidt-erhard

Chalmers, D. J. (1996). *The conscious mind: In search of a fundamental theory.* Oxford university press.

Chater, N., Tenenbaum, J. B., & Yuille, A. (2006). *Probabilistic models of cognition: Conceptual foundations.* Elsevier.

Chella, A. (2015). A cognitive architecture for music perception exploiting conceptual spaces. In *Applications of conceptual spaces* (pp. 187–203). Springer.

Chella, A., Frixione, M., & Gaglio, S. (2008). A cognitive architecture for robot self-consciousness. *Artificial intelligence in medicine*, *44*(2), 147–154.

Chomsky, N. (2002). *Syntactic structures.* Walter de Gruyter.

Coecke, B., Sadrzadeh, M., & Clark, S. (2010). Mathematical foundations for a compositional distributional model of meaning. *arXiv preprint arXiv:1003.4394*.

Cooley, J. W., & Tukey, J. W. (1965). An algorithm for the machine calculation of complex fourier series. *Mathematics of computation*, *19*(90), 297–301.

Deutsch, D. (2013). *Psychology of music*. Elsevier.

Forth, J., Agres, K., Purver, M., & Wiggins, G. A. (2016). Entraining idyot: timing in the information dynamics of thinking. *Frontiers in psychology*, *7*, 1575.

Gärdenfors, P. (2004). *Conceptual spaces: The geometry of thought*. MIT press.

Garofolo, J. S., Lamel, L. F., Fisher, W. M., Fiscus, J. G., & Pallett, D. S. (1993). Darpa timit acoustic-phonetic continous speech corpus cd-rom. nist speech disc 1-1.1. *NASA STI/Recon technical report n*, *93*.

Horn, R. A., & Johnson, C. R. (1990). Norms for vectors and matrices. *Matrix analysis*, 313–386.

Kemp, C., & Tenenbaum, J. B. (2008). The discovery of structural form. *Proceedings of the National Academy of Sciences*, *105*(31), 10687–10692.

Kennedy, R. A., & Sadeghi, P. (2013). *Hilbert space methods in signal processing*. Cambridge University Press.

Kirsh, D. (1991). Today the earwig, tomorrow man? *Artificial intelligence*, *47*(1-3), 161–184.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *nature*, *521*(7553), 436.

MacKay, D. J., & Mac Kay, D. J. (2003). *Information theory, inference and learning algorithms*. Cambridge university press.

McAdams, S., & Saariaho, K. (1985). Qualities and functions of musical timbre. In *1985 international computer music conference* (pp. 367–374).

Moore, B. C. (2012). *An introduction to the psychology of hearing*. Brill.

Murphy, K. P. (2007). Conjugate bayesian analysis of the gaussian distribution. *def*, *1*(2σ2), 16.

Pearce, M. T. (2005). *The construction and evaluation of statistical models of melodic structure in music perception and composition* (Unpublished doctoral dissertation). City University London.

Quine, W. V., & Quine, W. V. O. (1969). *Ontological relativity and other essays* (No. 1). Columbia University Press.

Quinlan, J. R. (1983). Learning efficient classification procedures and their application to chess end games. In *Machine learning* (pp. 463–482). Springer.

Robertson, H. P. (1929). The uncertainty principle. *Physical Review*, *34*(1), 163.

Rosch, E. (1983). Prototype classification and logical classification: The two systems. *New trends in conceptual representation: Challenges to Piaget's theory*, 73–86.

Shannon, C. E. (1948). A mathematical theory of communication. *Bell system technical journal*, *27*(3), 379–423.

Sivik, L., & Taft, C. (1994). Color naming: A mapping in the imcs of common color terms. *Scandinavian journal of psychology*, *35*(2), 144–164.

Sproat, R., Gale, W., Shih, C., & Chang, N. (1996). A stochastic finite-state word-segmentation algorithm for chinese. *Computational linguistics*, *22*(3), 377–404.

Sutton, R. S., et al. (1998). *Introduction to reinforcement learning* (Vol. 135).

Turing, A. M. (2009). Computing machinery and intelligence. In *Parsing the turing test* (pp. 23–65). Springer.

Tversky, A. (1977). Features of similarity. *Psychological review*, *84*(4), 327.

Wiggins, G. A. (2010). Cue abstraction, paradigmatic analysis and information dynamics: Towards music analysis by cognitive model. *Musicae Scientiae*, *14*(2_suppl), 307–331.

Wiggins, G. A. (2018). Creativity, information, and consciousness: the information dynamics of thinking. *Physics of life reviews*.

Wiggins, G. A., & Sanjekdar, A. (2019). Learning and consolidation as re-representation: Revising the meaning of memory. *Frontiers in psychology*, *10*.

Williams, C. K., & Rasmussen, C. E. (1996). Gaussian processes for regression. In *Advances in neural information processing systems* (pp. 514–520).