Generating layers of conceptual representations: a computational account of conceptual organisation in the brain.

5.1 Introduction

The previous chapters addressed the basis of categorical organisation in the brain as well as the representational capacities of the ventral stream in response to task demands. The hypotheses tested in these chapters were based on the view that conceptual representations are feature-based in nature and that object processing along the ventral stream is hierarchical and distributed. However, certain aspects of a theory, such as the CSA, need to be tested further in order to uncover the basic principles and mechanisms which underlie a specific cognitive process. Computational modelling in this respect becomes useful because the computational mechanisms are encoded within the model itself in the form of algorithms. Output derived from the model can be compared against empirical data and inferences can be drawn regarding the theory's validity. It is important at this point to recognise that a computational model is not in itself a theory but rather a formal instantiation of a specific aspect of a theory which is then tested against empirical or simulated data. Therefore the purpose of a computational model is to place the theory that it is testing on a more rigorous, quantitative footing and forces the explicit specification of the theory's assumptions which might otherwise be overlooked (Coltheart et al., 2001). In this chapter I will develop a computational model which aims to instantiate the principles of the CSA, and then test this against both behavioural and neuroimaging data in Chapter 6.

As described in previous chapters the CSA is a theory of semantic processing the central claims of which are, firstly, that the semantic system is organised in a distributed and hierarchical fashion (Taylor et al., 2007; Tyler and Moss, 2001) and, secondly, that high-level feature statistics play a crucial role in how concepts are organised in the brain. Experimental data in support of these claims have come from a number of behavioural and neuroimaging studies. In their fMRI study Tyler et al (2004) demonstrated that anterior regions including the perirhinal are only recruited during basic-level naming. When participants were asked to recognise objects at the domain-level (i.e. living/nonliving) only posterior regions were recruited. According to the authors, the study showed that there is a hierarchy, or gradient, of information richness ranging from coarse-grained representations within the posterior portions of the ventral stream (posterior IT / fusiform) to the most specific and fine-grained representations within the anteromedial temporal cortex. The hierarchy here refers to the representational capacity of specific regions within the ventral stream to carry out complex visuo-semantic processing. Within this scheme the perirhinal cortex is the hierarchical endpoint at which representations are maximally informative, enabling distinctive between similar, and, therefore, confusable objects.

Further support for the hierarchical organisation claim has come from an fMRI study using RSA (representational similarity analysis) conducted by Clarke and Tyler (2014). Here the authors used a large set of visual and semantic models (in the form of dissimilarity matrices or RDMs) to show that semantic information within posterior portions of the ventral stream is restricted to the category-level whereas anterior regions represent information which is more fine-grained and allows differentiation between exemplars within categories. Another fMRI study conducted by Tyler et al. (2013) showed that feature-statistic variables such as *sharedness* and the interaction between sharedness and feature correlation drive activity in the object processing system. The activation profile spanning the lateral-to-medial axis of the posterior fusiform gyrus was found to be uniquely sensitive to sharedness. In addition, the anteromedial temporal cortex exhibited activity which significantly correlated with the interaction measure.

Finally, a behavioural study by Taylor et al (2012) investigated the effect of feature statistics, sharedness and the interaction between feature correlation and sharedness, on naming latencies. The study involved two tasks where subjects had to identify objects at two different levels of specificity – basic-level and domain. Taylor et al showed that feature statistics had a differential effect on latency depending on the task. Sharedness correlated positively with naming latencies when identifying objects at the domain level and negatively when identifying objects at the basic-level. This finding showed that shared features were important when deciding whether an object was a living thing or not while distinctive features were more important when identifying *individual* objects. High sharedness, in this context, means that concepts are more embedded within their respective categories leading to better category recognition. However, when the participants had to identity the specific concept itself, high sharedness had an inhibitive effect. This was, they authors claimed, because high sharedness means a high degree of similarity with a high number of concepts thus making concepts more confusable, increasing the difficulty of the task.

Although the main claims of the CSA have been supported experimentally it is still unclear what types of neural computations take place that would give rise to a hierarchical structure and furthermore how high-level feature statistics (such as *sharedness* or *cLength*) relate to specific representations (either coarse-grained or fine-grained) at different levels of the hierarchy. As mentioned before one approach to this issue is to construct a computational model which embodies a plausible and testable mechanism under which a particular aspect of the CSA can emerge. The question then remains as to the type of computational model best suited to test the claims of the CSA. In addition, it would add to the explanatory power of the model if it were to have a degree of neurobiological plausibility.

A conventional approach would be to use a back-propagation neural network (McClelland and Rumelhart, 1986). Within the context of semantic processing this type of model has already been used successfully by Greer et al (2001) to test how a fully distributed network might give rise to category-level differences between concepts. The authors of the study modelled the naturally occurring differences in the distribution of shared and distinctive features that occurs across the domains of living and nonliving things (Tyler and Moss, 2001). They then showed that, when lesioned (i.e. a random set of connections was reset) the model exhibited a deficit for processing living things very similar to that shown by patients with a category-specific deficit for living things. However, this type of model (i.e. a 3-layer back-propagation network) does not allow for the testing of a representational, hierarchical structure which is an inseparable component of the CSA. Other models make the assumption of non-hierarchical organisation even more explicit. A ‘flat’ attractor network developed by O’Connor et al (2009) showed that feature-based, basic-level representations of concepts could give rise to superordinate categories without the need of a hierarchical structure. The model explicitly assumes a single unitary processing site which can form both basic-level and category-level representations; information within this attractor network is not forwarded to higher processing stages, which are thought be instantiated in other regions in a processing hierarchy, but statically remain within the same region. However, the particular nature of this model opposes the hierarchical architecture assumed in the early visual system as well as the principles of the CSA.

With these constraints in mind any model that is constructed to investigate semantic representations has to have the following properties:

1. **Neurobiologically plausible**: Both imaging methods and single-cell recordings are highly important in revealing the mechanisms by which neurons, whether at the single-cell or network level, process and learn information (Izhikevich, 2007; Kriegeskorte and Kreiman, 2012). This knowledge imposes a constraint on the possible number of models that can be used. Although a particular model could potentially mimic the internal processes which underlie semantic processing, its applicability would only be valid if its construction was constrained by neurobiology.
2. **Distributed encoding**: A key claim of the CSA is that the conceptual space consists of a unitary distributed system (Tyler and Moss, 2001), where information is represented and processed according to the same principles irrespective of the type of information involved (i.e. either visual or semantic). This means a neural ensemble can potentially encode a vast array of different categories in contrast to domain-specific accounts where individual regions are uniquely responsive to individual categories (see Chapter 1). This type of architecture also assumes non-symbolic, feature-based conceptual representations.
3. **Hierarchical architecture**: According to the Standard Model of object recognition (HMAX; Riesenhuber and Poggio, 1999) visual object processing follows a flow of information by which basic features, such as lines, are combined together to form more complex features such as shapes (also see

Chapter 1). The same principle is a central claim made by the CSA whereby objects are represented at different levels of specificity – from an abstract domain-level categorisation (is it living or non-living?) to a unique concept identification (e.g. 'duck' or 'fork'). The hierarchy in this case refers to a unidirectional flow from coarse-grained information (only sufficient for category judgements) towards more detailed and rich representations.

These properties are all satisfied by a set of computational models known as *deep belief networks* (DBN; Hinton et al., 2006). DBNs belong to the wider family of *generative models* (Hinton, 2007). The main aim of this type of model is to learn, in an unsupervised manner, a set of probabilistic transformations which allow the system to infer the probable causes of its input. This places them in distinct contrast to the more standard *discriminative models* where the aim is to classify input according to a pre-determined set of category labels (Bishop, 2006). Generative models have enjoyed a lot of popularity in recent years, with noted success in the field of computational vision (Kersten et al., 2004; Mumford and Lee, 2003; Dura-Bernal et al., 2012). The main attraction of this type of model is that the algorithms they use introduce a plausible principle of how the brain processes information. Specifically, vision in this regard is understood as a *process of inference*, whereby the system tries to find simplifying explanations for the incoming, rich sensory data. During this process, the system detects hidden features or patterns within incoming stimuli which allow it to categorise it and identify it accordingly. Perception, in this case, is not seen as a direct mapping between the stimulus and the representation; rather, it is an actively constructive process whereby sensory data guide the dynamic generation of hypotheses by the visual system in order to probabilistically *infer* what is the most likely stimulus (Rao et al., 2002). In other words, perception is the system's collective effort to create a coherent picture of its sensory input (Helmholtz, 1878/1971).

This framework has proven especially successful in computational models of low-level vision (Penny, 2010; Hinton, Osindero & Teh, 2006; Ranzato et al., 2007; Bengio et al., 2007). While most studies using generative models have been strictly constrained to vision research, the principles themselves can be applied to any cognitive function (Hinton, 2007). Specifically, the idea that a system can form its own internal representations which contain high-level, and non-linear, inter-correlations between the constituent features of the original input affords itself as a plausible, and testable, account of how the dynamic nature of conceptual representations can arise. I refer to conceptual representations as being dynamic in this context because, as suggested by the theoretical framework of the CSA, the brain needs to differentially parcellate information at varying degrees of detail in order to be able to efficiently process and organise the rich sensory input they receive from their environment. To create an effective internal, and coherent, model of the world the semantic system needs to extract multiple layers of representations. These layers are in essence high-level feature detectors whereby higher layers are capable of abstracting away from the original input in such a way that categories can form naturally and without supervision. Ultimately the system learns to categorise input based on the specific featural characteristics of the objects.

This chapter will detail the work I have conducted on constructing a generative model of semantic representations. The model itself is based on Geoff Hinton's work which established the formal basis of deep belief networks. Studies involving DBNs were primarily used to model early visual processes and, to date, have not been applied to conceptual processing.

The work undertaken can be split into three parts and I will go into each one in more detail for the rest of the introduction:

1. **Model architecture**: where I explain the inner mechanics of the model.
2. **Model analysis**: where I conduct an analysis of the layers that result from the model and relate the properties of the layers to CSA variables.
3. **Model testing**: where I compare model performance to previously-collected behavioural (Taylor et al, 2012) and neuroimaging data (Clarke and Tyler, 2014).

5.1.1 Overall architecture

A DBN is essentially a stack of learning modules or layers (Hinton, 2009). Learning in this type of model typically consists of training one pair of layers at a time. A pair of layers is comprised of a “visible” layer (which can be the initial data vector on which the network is trained on) and a “hidden” layer. Both layers are symmetrically connected with a set of weights (i.e. there is a statistical dependency between both sets of units). Each unit in one layer is connected with all the units in the *other layer*. However there are no connections between units *within* a layer meaning that their activity is statistically independent. The pair of layers and the general mechanisms of interaction between them are collectively referred to as a *restricted Boltzmann machine* (or RBM) and it forms the fundamental component of a DBN (**Figure 5.1**).

Figure 0.1: Architecture of the restricted Boltzmann machine

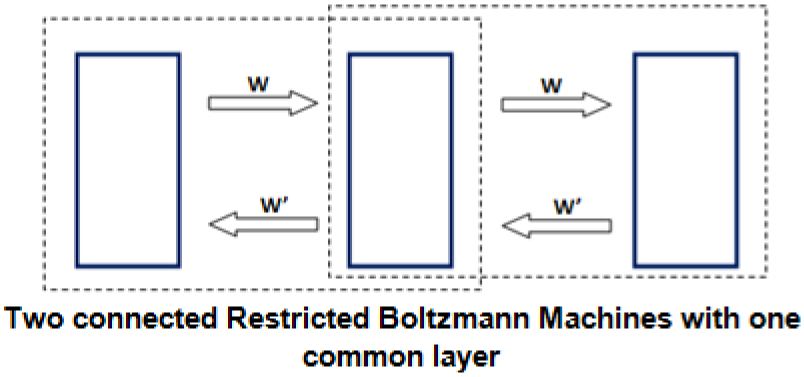
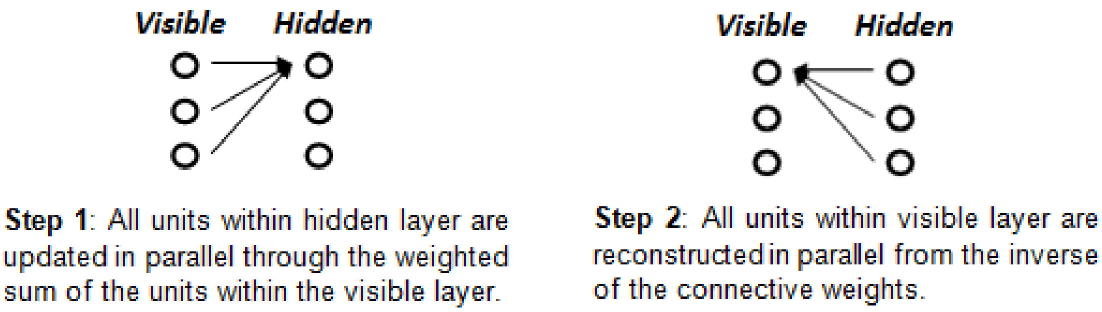

w
w
w'
w'
Two connected Restricted Boltzmann Machines with one common layer

Figure 5.1: *Architecture of the restricted Boltzmann machine*. It essentially consists of two layers which are symmetrically connected to each other. The overarching aim is to find an optimal set of weights (***W***) which allows for the seamless transformation of data vectors from one layer to the other.

Learning with RBMs follows two simple rules: first, all units within the hidden layer of the RBM are updated in parallel following a randomly initialised set of connective weights between itself and the visible layer. The key step following this is that the weights are evaluated on how well they can *reconstruct* the initial visible layer from the hidden layer. Based on the errors derived from this comparison the weights are updated again and the hidden unit values re-evaluated. The learning is unsupervised because there is no external objective or classification against which the error is derived. This process is repeated until an optimal set of weights is found which can accurately transform the visible layer unit values to hidden unit values and also, even more importantly, the hidden unit values back to the original visible layer. The overarching aim of the RBM is to learn a distribution of possible feature configurations (or states) that the visible layer can take. The basic intuition here is that by going through this learning process, the resulting units on the hidden layer capture high-order, latent features of the initial data structure which would have been otherwise very difficult to uncover (**Figure 5.2**; Hinton, 2009; Hinton and Salakhutdinov, 2006).

As an aside, one could argue that such a process could be undertaken deductively: utilising an algorithm which simply deduces the optimal weight-matrix in one pass and perfectly transforms one matrix (e.g. initial data layer) to another (e.g. hidden layer) instead of going through the more time-consuming iteration process described above. Such a learning algorithm was developed by Kosko (1988) and further developed by Chartier (2006) and is known as the bidirectional associative memory (BAM) model. However the BAM model, as well as other like-minded models, has certain limitations which narrow down the possible types of datasets that can be used. The most important limitation is that the number of vector-patterns that can be faithfully transformed is severely limited by the number of units (Kosko, 1988). This effectively prohibits cases in which the number of units vastly outnumbers the number of vector-patterns making the algorithm implausible for the present study at least. Furthermore, non-orthogonal (or correlated) vectors (which is the case with most concept vectors in the present study) lead to particularly poor results (Rojas, 1996). Although this limitation also holds, to a lesser extent, for DBNs it is not as prohibitive as the BAM model. This difference also highlights the necessity of the iteration process since it would be otherwise impossible to find the optimal weight-matrix.

Figure 0.2: Learning process within an RBM



Step 1: All units within hidden layer are updated in parallel through the weighted sum of the units within the visible layer.

Step 2: All units within visible layer are reconstructed in parallel from the inverse of the connective weights.

Figure 5.2: *The learning process within a restricted Boltzmann machine*. This cycle is repeated until there is an optimal transformation accuracy between the hidden to the visible layer.

Multiple RBMs can be stacked together. The crucial principle is that each RBM is trained *individually* with the hidden layer in the first pair becoming the visible layer in the second pair and so on. Once all RBMs have gone through a sufficient number of training cycles (i.e. the weights can transform patterns between layers with near-zero error) the network is now said to be trained. A *bottom-up* pass within the network consists of using the learned generative weights to get from the initial data vector to the final layer of the network. Conversely a *top-down* pass consists of using the inverse of the generative weights to re-construct the initial data vector from the top layer of the network (**Figure 5.3**). The top-down weights contain a probabilistic model of the training data (Hinton, 2008).

Figure 0.3: Deep belief Net with two passes

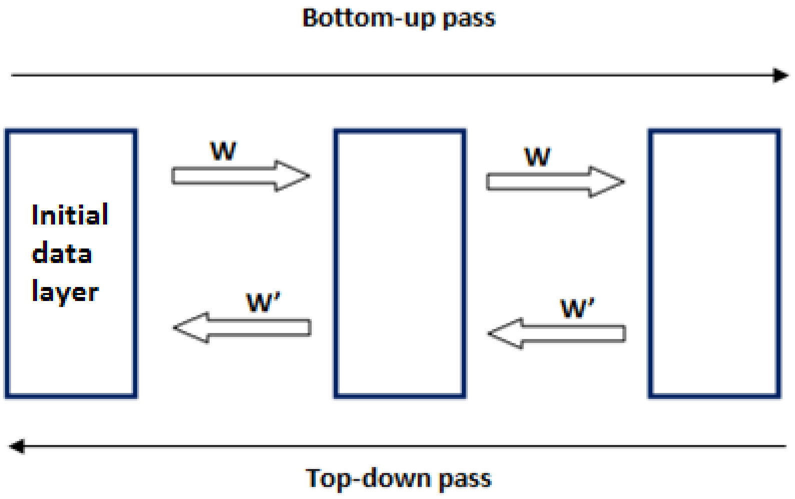


Figure 5.3: A Deep Belief Net with two passes: one originating from the bottom-layer (far left) towards the top-layer (far right) and one originating from the top-layer towards the bottom-layer. The set of weights connecting pairs of layers learned during the training process are inversed for the top-down pass.

In the model reported here training was based on the entire set of the McRae feature norms which consists of 517 concepts each represented by a 2341-element feature vector. The initial data layer therefore consisted of 2341 nodes. After training, I expected that the top layer of the DBN would capture the intricate semantic relationships between features resulting in cohesive category formations at the expense of fine-grained granularity (i.e. within-category concepts would be highly confusable). This is because the model would be able to uncover statistical regularities across different concepts of a particular category and thus highlight their overall similarities. The McRae dataset, on the other end, represents the finest grained end point of human semantic knowledge since all concepts have a unique combination of features resolving all possible sources of ambiguity. A top-down pass from the top-layer towards the bottom layer (i.e. the initial McRae feature dataset) would mean that representations become less and less categorically cohesive but gain more *distinctive* information on individual concepts as the original training dataset is re-constructed.

How does the architecture of the DBN relate to conceptual processing? First, the hierarchical architecture of the model is directly related to how conceptual processing is thought to take place in the CSA (Taylor et al, 2007; Tyler and Moss, 2001; Tyler et al, 2014). There are sequential stages of processing in which information is forwarded higher up the hierarchy. The representational capacities of each stage allow for increasingly more refined representations. Secondly, there is sufficient evidence to believe that the organisational hierarchy of semantic information begins with high-level categorisations. A behavioural study by Mace et al (2009) showed that coarse representations are retrieved faster than more individuated representations. They used a rapid go/no-go visual task whereby participants had to identify which of the two simultaneously presented matched a predefined label. The label could be either at super-ordinate level of specificity (e.g. ‘animal’) or basic-level (e.g. ‘dog’). When presented with two members of the same category participants took on average 40-65msecs more in making a basic-level decision. According to the authors this finding suggests that it is indeed high-level categorical representations which are accessed first leading to faster reaction times. The top layer (i.e. the layer with high category cohesion but low individual concept information) in this case would be at the most posterior end of the semantic system. The **top-down pass** would be the process by which the system *reconstructs* the uniquely identifiable concept from the coarse-grained representations at the beginning (i.e. the top-layer).

5.1.2 Model analysis

The aim of constructing the model was to uncover a plausible computational mechanism by which semantic processing might take place in the brain. In order to gain an understanding of this process it is necessary to have a clear picture of what the model is actually doing. In other words in order for the model's output to make any sense it must be contextualised within the framework of cognitive processing and more specifically the CSA.

The model in this case had 4 layers – each of which was built on top of the layer beneath it (**Figure 5.4**). As explained in the previous section each layer abstracted further and further from the initial training feature data uncovering high-level inter-correlations between individual features.

Figure 0.4: Overall model architecture

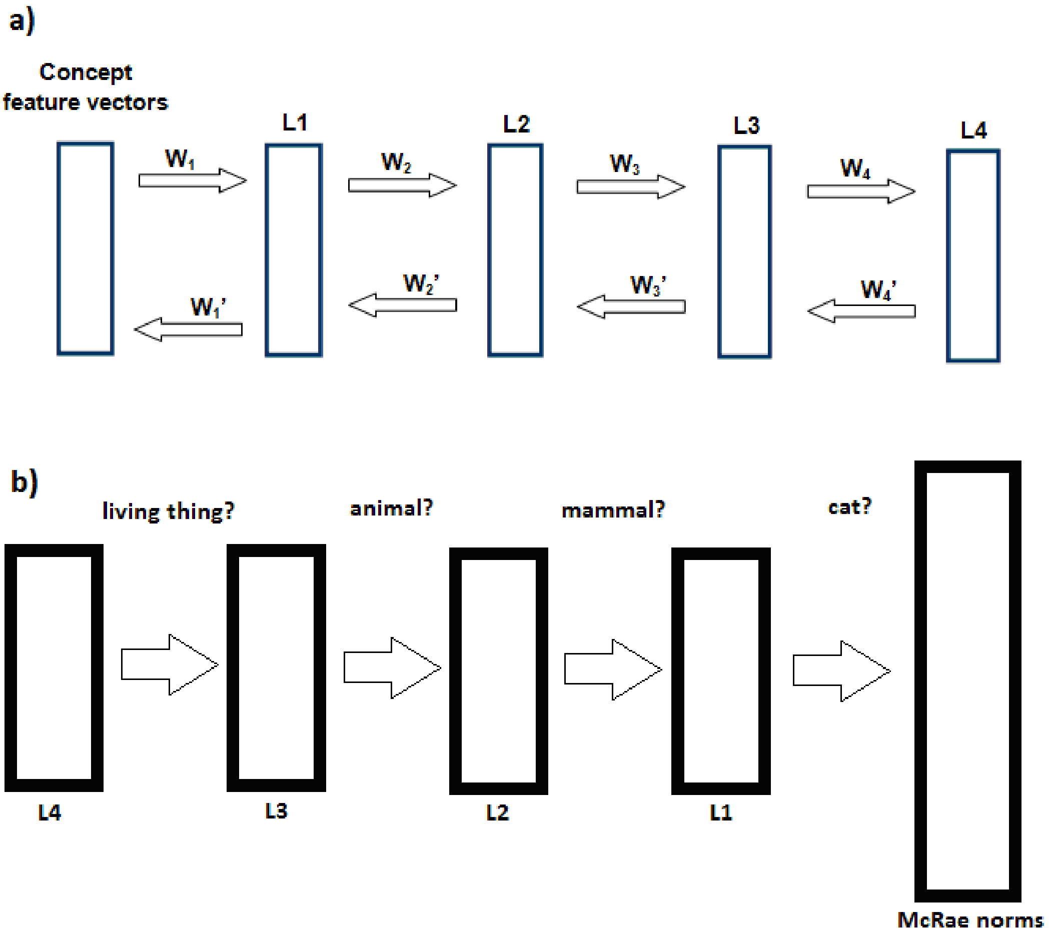


Figure 5.4: a) Overall architecture of the model. b) The evolution of representational specificity along the top-down pass from L4 towards the final reconstruction of training data. During the early stages, the model can make accurate decisions at the domain-level (is it living or non-living?) with specificity increasing towards the later stages.

Once the layers had been extracted, their representational structure was examined in detail. Concepts can be categorised at different levels of specificity: *superordinate categories* such as domain (e.g. 'living thing') and general category (e.g. 'animal'); *basic-level* (e.g. 'cat) and *subordinate* ('my pet cat') (Rosch et al, 1976) (note: given the availability of the data at hand subordinate categories cannot be taken into account). How does information content at these different levels of specificity change from layer to layer? I used Shannon's information theory to determine whether the pattern of response within individual layers contained information about concepts at these three different levels of specificity. In other words at which layer would we have maximal information about an individual concept? Is there a discernible trend of coarse-grained information in L4 towards more fine-grained information in L1 which would reflect the flow of information as explicitly predicted in the CSA?

I also needed to gain an understanding of how concepts were arranged in each layer's representational space. If we envision the flow of processing within the semantic system as moving from highly categorical types of representation towards more individuated representations then the top-down pass of the model should reflect a similar trend. To address this question I used RSA to determine:

a) The *categorical cohesion* within each layer's representational space,

b) Whether there was a *differential weighting on features* of concepts depending on their statistics. If there is a discernible trend of high to low cohesion (from L4 towards L1) within the representational space of successive layers then this could be attributed to a decreased weighting on distinctive features. This is because according to the CSA the hierarchical structure of semantic information is dependent on feature statistics. In general, giving more weight to shared features leads to concepts being less distinguishable within their respective categories. Conversely more weight on distinctive features leads to more fine-grained representations and concepts become more easily distinguishable and individuated.

I also computed the uncertainty of concept identification at each layer and for all three levels of specificity (basic-level, category, & domain). Uncertainty is formally defined as the entropy of the model's responses (Shannon, 1948; Trappenberg, 2009) and is essentially an indication of the model's confidence in making a response. For example, in situations where the model is faced with a multitude of equally-likely candidates would lead to high uncertainty. I reasoned that as the flow of processing progresses along the top-down pass of the model, uncertainty regarding the precise identification of a concept would *decrease* for each individual layer. This reasoning is essentially a reflection of how the CSA accounts for the increased dependence on anteromedial temporal structures during tasks which require fine-grained level information. The anteromedial temporal structures in this context are thought to possess the neural capacity to represent information at a high degree of detail. Representations in these structures would be detailed enough to resolve any uncertainty of identification between a concept and other competing semantic neighbours. For example in uniquely identifying a 'tiger' from a 'lion' the semantic system requires full access to the semantic features which constitute the 'tiger' concept. This is because there are many features in common between the two concepts. To resolve uncertainty in this case the semantic system would need to resort to those neural structures which carry sufficient information to make the distinction. Increased fine-grained information on individual concepts means decreased uncertainty regarding precise identification at a specific level of specificity. *Uncertainty*, in combination with identification *accuracy*, formed the quantitative indices of the model's performance. High performance in this case equates to highly accurate responses with low uncertainty.

Finally, I investigated how the performance of the model related to CSA-derived feature-based statistics and specifically *cLength* – which was extensively tested in Chapters 2 and 3. High values of cLength result from a large number of shared features while low values of cLength result from concepts having few, distinctive features. Depending on the type of the identification that the model has to make (basic-level, category, domain) cLength could either exhibit a positive or negative relationship. For example, a high-cLength concept (‘dog’) would be easier to categorise correctly given that it shares many features with its category co-members (i.e. ‘animals’: ‘has legs‘; ‘is furry’ etc). Low-cLength concepts by virtue of their few, distinctive features form categories which are less cohesive. This means that the model would perform *worse* for low-cLength concepts when identifying at either category or domain-level of specificity. This would lead to a **positive** relationship between the model’s **category/domain-level** performance and cLength: high-cLength = high accuracy/low uncertainty; low-cLength = low accuracy/high uncertainty. Conversely, when identifying a high-cLength concept at basic-level, performance would be low due to the high confusability with other neighbouring concepts. The opposite would be true for low-cLength concepts. This, in effect, would lead to a **negative** relationship with **basic-level** performance: high-cLength = low accuracy/high uncertainty; low-cLength = high accuracy/low uncertainty. I carried out a correlational analysis (cLength vs. performance) for all four layers of the model.

5.2 Methods

The aim of the experiment was to uncover a plausible computational mechanism by which semantic representations are organised in the ventral stream. I constructed a deep belief net consisting of four layers. The number of units within each layer is arbitrary although generally it is advised that it is smaller than the number of features in the training data set (Bengio, 2007). The expectation during the training process is that by forcing the initial data into a vector with fewer elements, hidden features of the data can be revealed. It is useful to imagine this process as a non-linear version of principle components analysis (PCA; Hinton and Salakhudinov, 2006; Bengio, 2007) where the principal components embody the main sources of variation within the data. In this sense, each layer unit captures a specific principal component of the data.

The network was trained on the full McRae feature norm dataset (517 object x 2341 features). The 517 objects were also split into 24 categories as well as into 2 domains (i.e. living / nonliving). This was done to test the model at different levels of specificity – basic-level (n = 517), category-level (n = 24), and domain-level (n = 2). Objects were grouped in categories and domains by hand at the CSLB (see **Appendix I**). All code was written in MATLAB (v8.3) and parts of it were based on Hinton and Salakhudinov (2006).

First, I will describe how the model was trained along with the basic rules of learning that were used. I will then describe how I analysed the representational content within each of the resulting layers in order to: 1) determine the level of specificity appropriate to each layer using Shannon information, 2) examine the workings of the top down pass using RSA, and 3) investigated the performance of individual concepts with uncertainty measures. In Chapter 6 I compare these extracted measures against both behavioural and imaging data.

5.2.1 Layer formation and analysis

Deep belief nets are trained in a sequential fashion where each trained layer is used as input for the layer above it. Each pair of layers forms an RBM and training is carried out one pair at a time. Each newly trained layer is then 'stacked' on top of the preceding one which is then subsequently used as the training set for another layer above it. At the end of the process the initial training data is transformed during a bottom-up pass, from layer to layer, by a set of trained weights. A top-down pass from the top-most layer will, conversely, re-construct the initial data set (i.e. the McRae norm dataset).

The model comprised of four layers (plus the initial training dataset) numbered **L4** (as the top and final layer) to **L1** (the first layer formed during training and the only layer in direct interface with the initial dataset). The reason there were only four layers was two-fold: first, there is an upper limit on the number of layers imposed by the nature of the dataset itself. Typically, models such as these are trained on a vast number of stimuli with a relatively small number of features (Bishop, 2006; Hinton, 2007). For example, the image training set used in Hinton and Salakhutdinov (2006) comprised of 20,000 images each made out of 784 pixels. The model itself comprised of one data layer and three hidden layers which was sufficient for the study's purposes. By contrast, the present dataset comprised only of 517 concept-vectors each with 2341-elements. In this case, having more than four layers would have resulted in poorer reconstruction accuracy. This is because the connecting weights between further layers would not be sufficiently trained given the nature of the training data. Secondly, given that this model is in an early stage of development, it is better to have fewer layers which are more manageable during analysis compared to a vast number which would have made a thorough analysis intractable. For the present study the flow of conceptual processing in the ventral stream was modelled as the flow of information from L4 to L1 (i.e. the ***top-down pass***). This means that both during testing and analysis the aim was to go from a highly abstract, categorical representation (in the top-most layer or **L4**) towards representations which resemble the highly differentiated, fine-grained nature of the initial training data.

Figure 0.5: a) model with one input layer and four hidden layers b) 4-unit layer which is fully connected to one unit on the layer above

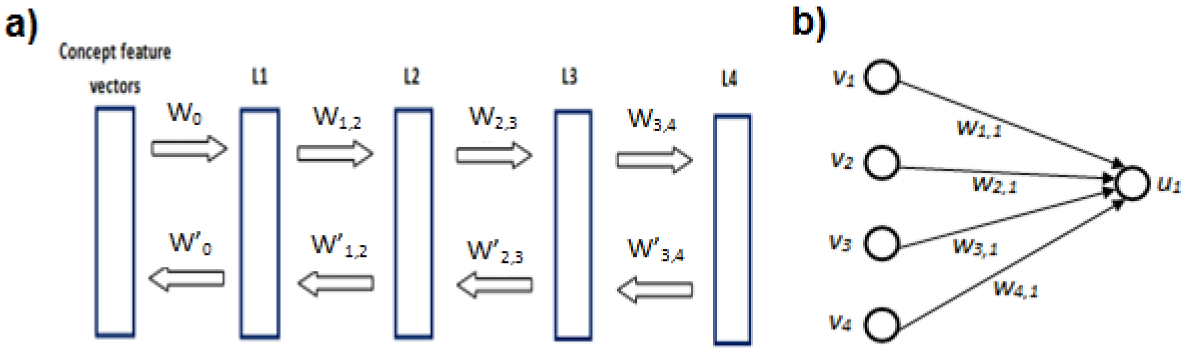


Figure 5.5: **a)** Overall architecture of the model which consists of one input layer (with 2341 features) and 4 'hidden' feature layers (each consisting of 750 units). In the present study the flow of processing is simulated as going from L4 towards L1. **b)** A simple 4-unit layer which is fully connected to a unit (u_1) belonging to the layer above it. None of the units (v_1, v_2, v_3, v_4) within the layer are connected to each other but they are all connected to the unit above making the probability of its activation statistically dependent.

I ran the model on three different unit numbers [500, 750, 1250] where 750 was the setting with the smallest amount of reconstruction error. The number of units was kept constant for all layers. I did this to remove any implicit assumptions regarding the neural size of the processing regions along the ventral stream.

There was no dependency between units within layers, because there were no connections between units within a layer. A unit can take a value of either '0' and '1' with a certain probability, $p(u = 1)$. However, as the equations below and **Fig. 5.5a** show, the activity of units across two adjacent layers *is* dependent. In other words the conditional probability, $p(u = 1|v_1, \dots, v_n)$ (see **Fig. 5.5b**), is directly computable if we know the states of the layer units *beneath*. However we cannot compute such a probability for units belonging to the *same layer*. This is because there are no connections between neighbouring units within a layer but any given unit receives a collective input from all the units belonging to the layer beneath it. The weights were learned between layers over 250 epochs, with the weights at the 250th epoch being used as the final set of weights. The number of training epochs was derived during initial trial runs. Further epochs did not substantially improve in reconstruction errors. The probability with which a certain unit was activated was computed by:

Equation 0.1: Probability of activation for individual unit

$$P(u = 1) = \sigma(b_j + \sum_i w_{ij} v_i)$$

(Eqn. 5.1)

The **σ** function in this case signifies the logistic function:

Equation 0.2: Definition of σ function in Eqn. 5.1

$$\sigma(x) = \frac{1}{1+\exp(-x)}$$

(Eqn. 5.2)

This function confers a non-linear stochasticity on the overall activity pattern for each layer. ***b_j*** is the bias for unit u_j and ***w_{ij}*** is the weight between units v_i and u_j. The result of these equations is that unit activity is: **a)** stochastic and **b)** defined by the linearly weighted sum of all the unit states of the layer beneath it.

The training for the connective weights between layers was derived from the following formal learning rule:

Equation 0.3: Learning rule

$$\Delta w_{ij} = \varepsilon(< v_i u_j >_{bottom-up} - < v_i u_j >_{top-down})$$

(Eqn. 5.3)

For every training cycle within a single RBM there is one bottom-up pass for every top-down pass. $< v_i u_j >_{bottom-up}$ is the correlation between the two units during training from initial data. The objective in this case is to find a weight value which ensures that this correlation is maintained when doing a reverse, top-down pass. When the result of the subtraction operation is zero or near-zero then the weight values have reached their target. All the information necessary to transform representations from one layer to the next is contained within these weights. Even when there is a loss of information or 'corruption' during training it can be recovered because of the weight-matrix transformations. ε is the learning rate which for the present modelling study was an exponential function with the initial value set to 1. This was done to give decreased importance to later stages of the learning process and avoid over-training.

5.2.2 Information theory

A concept is represented as a particular pattern of activation within each layer (**Figure 5.6**).

Figure 0.6: Transformation of layer unit patterns

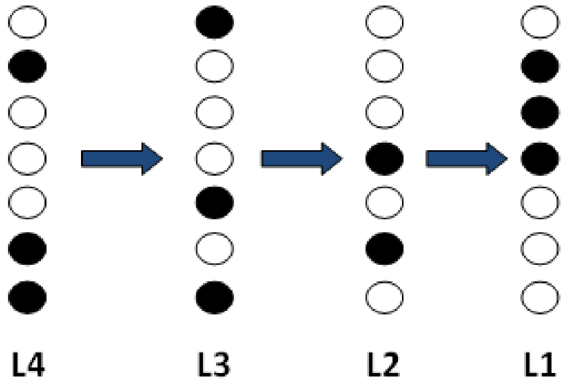


Figure 5.6: Each layer represents a unique pattern of activation for a particular concept. Arrows denote the top-down pass of information. L4 contains the most 'abstract' representations while L1 in the layer in which representations are the closest to the initial training data.

These patterns contain information about a particular concept. I needed a way to quantify this information and assess it for different levels of specificity (basic-level, category, domain). The goal was to investigate whether there was a meaningful trend of information flow from L4 (top-most layer) to L1. Specifically I asked: how much information could I extract about a concept at a given level of specificity (e.g. which level can answer the question, 'is it a living thing?' / 'is it an animal?' / 'is it a dog?') just by observing its associated activity pattern on a particular layer? This question is formally defined as the *mutual information* between concept C_i and the model layer response R_i. Mutual information*, I(C; R)*, is defined as (Mackay, 2003):

Equation 0.4: Mutual information

$$I(C;R) = H(S) - \ H(C|R)$$

(Eqn. 5.4)

$H(S)$ is Shannon entropy (or maximum entropy; Cover and Thomas, 1991) which is the entropy when each category is equally likely and is the average amount of information contained within a particular dataset. This value is directly related to the number of possible choices – high numbers give rise to higher entropy values since there is a larger margin for error.

Equation 0.5: Stimulus entropy

$$H(S) = \ -\sum_i (1/n) \log_2(1/n)$$

$$\text{where } \boldsymbol{i} \text{ refers to concept } \boldsymbol{i} \text{ and } \boldsymbol{n} = \text{number of categorical distinctions}$$

(Eqn. 5.5)

Given this equation, if the model is faced with a particular response pattern and asked to make a guess regarding the precise basic-level identification of the concept there would be a total of 517 possible candidates (i.e. the total number of unique concepts within the McRae norms set all with the same probability) giving rise to high stimulus entropy. On the contrary, domain-level decisions (living vs. nonliving) are effectively binary and as such have relatively low entropy. Fewer choices means that one is much less likely to make an error. The higher the value in ***n*** (the number of possible distinctions; see **Equation 5.5**), the higher the stimulus entropy.

The second term in **Equation 5.4**, $H(C|R)$, is the *conditional entropy*. It quantifies how much information remains unaccounted, regarding the identity of concept ***C*** after observing response ***R***. It is generally defined as:

Equation 0.6: Conditional entropy

$$H(C|R) = -\sum_i \sum_n P(u_i)P(C_n|u_i)\log_2 P(C_n|u_i)$$

$$\text{where } \boldsymbol{i} \text{ refers to } response\ \boldsymbol{i} \text{ and } \boldsymbol{n} \text{ to } concept\ \boldsymbol{n}$$

(Eqn. 5.6)

$P(u_i)$ is the overall un-conditional probability of getting a particular pattern of response across the entire set of units of a layer, i.e. $P(u_i) = \prod_m P\left(u_{i,m} = 1\right)$. **Equation 5.1** provides the value for $P\left(u_{i,m} = 1\right)$ after the model is fully trained. $P(C_n|u_i)$ is the *conditional probability* of observing a particular concept C given the specific response u_i. How this probability was computed is described in the following section.

5.2.3 Bayesian inference

In making a decision regarding a concept's identity the model needs to infer a probability distribution across the entire set of possible candidate concepts. This means that the model has to compute the conditional probability of a concept given the specific activity pattern elicited across a layer, formally defined as $P(C|F_1, \ldots, F_n)$. The winning candidate is the concept with the highest conditional probability (**Figure 5.7**).

Figure 0.7: Example of how the model makes a decision by computing a set of conditional probabilities for each concept in the set

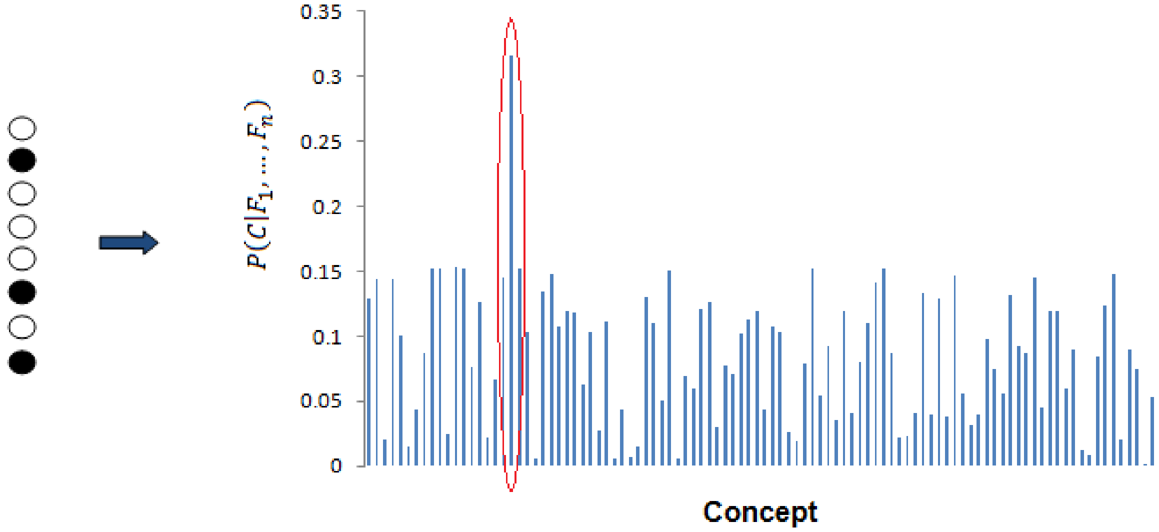


Figure 5.7: A simplified example of how the model makes a decision given a particular activity pattern. It will compute a conditional probability over the entire set of concept and then pick the winning candidate scoring the highest (circled in red).

The conditional probability can be generally defined as:

Equation 0.7: Conditional probability (Bayesian estimation) (1)

$$P\left(C\middle|F_1, \dots, F_j\right) = \frac{P(C)P(F_1,\dots,F_j\,|C)}{P(F_1,\dots,F_j)}$$

(Eqn. 5.7)

$P(C)$ in the above equation is the unconditional probability of getting a particular concept which is computed as *1 / n* (n = number of concepts = 517). Although this is the standard equation for Bayesian estimation of a conditional probability it was preferable to use the following equation mainly because there were a smaller number of terms involved:

$$= \frac{P(C)\prod_{j=1}^{n} P(F_j\,|C)}{P(C)\prod_{j=1}^{n} P\left(F_j\,|C\right) + P(\neg C)\prod_{j=1}^{n} P(F_j\,|\neg C)}$$

(Eqn. 5.8)

$P(F_j \,|C)$ is the *likelihood* of observing an activated unit, F_j , if presented with concept C. This probability is already known through **Equation 5.1** which explicitly computes the activation function of an individual unit during training. Each concept vector within a layer comprises of a set of probabilistic values. I used these values to extract $P(F_j \,|C)$. $P(F_j \,|\neg C)$, in this case, is simply: $1 - P(F_j \,|C)$.

Equations 5.7 and **5.8** are both instantiations of a *naive Bayes classifier.* This type of classifier is specifically referred to as 'naive' because of the strong assumption that the units within a layer are independent (Bishop, 2006) which is actually the case in this model. This method was crucial in determining the accuracy of concept identification within each layer of the model because the winning candidate was defined as the concept with the highest posterior probability. During each test run (model testing is described in section 2.2) each concept was run through each layer of the model. The winning candidate was then recorded to assess the layer's performance. Accuracy was then defined as the number of times the model's winning candidate was the correct concept divided by the total number of runs.

The method so far only extracts a probability distribution over basic-level categorisations (e.g. 'dog', 'peacock' etc). The model also needs to make a decision at two further levels of specificity: general category ('mammal', 'bird') and domain ('living',' nonliving'). In order to do this I converted the computed conditional probability for a basic-level identification, $P\left(C\middle|F_1, \dots, F_j\right)$, into a conditional probability for a category-level identification $P\left(CAT\middle|F_1, \dots, F_j\right)$. In other words given a particular activity pattern, $\{F_1, \dots, F_j\}$, what is the most likely category to which a particular concept might belong to?

$P(CAT|F_1,\dots,F_j)$ was defined by the following matrix operation:

Equation 0.9: Conditional probability (Bayesian estimation) for category-level identifications

$$P(CAT|F_1,\dots,F_j) = P \times M$$

(Eqn. 5.9)

P is an n-element vector containing all the probabilities across the entire concept set ($P(C_n|F_1,\dots,F_j)$). M is a $c \times n$ matrix containing all category membership information across the entire concept set (the *membership matrix*). c in this case is the number of categories and n the number of concepts. The membership matrix was normalised across each (category) column by the total number of concepts within each category. This was done to make sure that all probability vectors for each category had a total sum of 1. This function is effectively a summing operation: if the model produces a large number of high-probability candidates within a particular category then they will collectively give rise to a high probability for that category being the winning candidate. As **Appendix I** shows, there was one non-standard category containing 92 objects ('miscellanea') which were not easily put into a recognisable category. This was removed from any further analysis. For the present study, the membership matrix for general category was a 23 x 425 matrix; 23 being the total number of categories within the concept set and 425 being the total number of concepts. For domain, the membership matrix was a 2 x 425 binary matrix; 2 being the number of domains (living and nonliving).

5.2.4 Uncertainty and confidence measures

Uncertainty is closely related to information as defined in **Equation 5.4** as well as the naive Bayes classifier described earlier. Using the probability vector, $P\left(C_n \middle| F_1, \dots, F_j\right)$, as a starting point we can derive a measure of uncertainty as:

Equation 0.10: Uncertainty

$$H(C_n) = -\sum_i (x_n) \log_2(x_n)$$

where $\boldsymbol{x_n} = P\left(C_n \middle| F_1, \dots, F_j\right)$ and $\boldsymbol{i}$ refers to response $\boldsymbol{i}$ (where response $= \left\{F_1, \dots, F_j\right\}$)

(Eqn. 5.10)

$H(C_n)$ measures the degree of entropy across the concept probability vector. High entropy means that the pattern activity derived from the model layer contains little information. Low entropy means that there are only a few competing high-probability candidates. Zero entropy means that the probability distribution has collapsed to just one winning candidate at a probability of 1 with all other candidates having a probability of 0. The maximum value for $H(C_n)$ is equal to the stimulus entropy $H(S)$ defined in **Equation 5.5**. Entropy, as calculated and defined in this section, is what I refer to as *uncertainty* throughout this chapter.

Uncertainty is measured in bits which might be difficult to interpret intuitively. For this reason I also computed a further measure where values ranged from 0 to 1. I simply divided a concept’s uncertainty score by the stimulus entropy (see **Eqn. 5.5**):

Equation 0.11: Confidence

$$N(C_n) = \frac{H(C_n)}{H(S)}$$

(Eqn. 5.11)

Confidence, $N(C_n)$, in this sense denotes the degree of variance within the entire concept set that is accounted for by the response pattern within a layer – when there is little variance, confidence will be high and vice verse. In combination with accuracy, confidence provides a quantifiable measure of the model's overall performance – high performance means high accuracy with high confidence / low uncertainty.

5.2.5 Analysing representational content using RSA

The methods described earlier revolve around the idea of information as reduction in entropy. However I also needed a methodology which addressed the representational structure of each layer. Specifically, I asked: what is the relationship between high-level feature statistics and the representational space of each individual layer? I used two methods derived from RSA to assess how concepts are organised within the model's representational space. There methods were category cohesion and RSA. The former is a measure of the overall similarity amongst different members of a particular category. When the average within-category similarity is high and the across-category similarity is low, the cohesion of the category will be high. This particular method was described in more detail in Chapter 4.

RSA is based on the comparison of information content as captured by dissimilarity matrices (or RDMs). In my case, this RDM takes the form of the semantic distances between all 425 concepts found in the initial dataset. In this form, the RDM is a reflection of the overall semantic overlap between concepts. However, what I wanted to know was whether the specific characteristics of a concept's features (i.e. sharedness / correlational strength) had any bearing on how the representational space was organised – something which is not taken into consideration when an RDM

$$w_i$$

is computed on just the binary vectors of concepts. A key claim of the CSA is that highly-shared features are given more weight during early stages of semantic processing giving rise to more cohesive category formations. Within the context of the model this means that feature sharedness should have a stronger effect on the representational space of layers with highly abstract categorical representations. Conversely for layers with more fine-grained representations sharedness should play a smaller role since concepts now need to be distinguished from one another within their category. The aim of RSA for the present study was to determine how feature statistics are driving similarity between concepts across layers.

In Chapter 2 I geometrically derived a set of measures which described specific aspects of a particular feature, such as *sharedness* and *correlational strength*, along with a novel measure, *feature length* (φ). Feature length captures both sharedness and correlational strength. The objective was to include information regarding these feature-specific characteristics (as contained in feature length) within the calculation of the metric distance between concepts. I computed the feature length for each feature within the McRae concept set. A high score of feature length means that the feature is highly shared across concepts and also co-occurs with a high number of other features.

Each concept vector is represented as a 2341-element binary vector where each element denotes the presence or absence of a particular feature. I multiplied each feature ***i*** within a particular concept-vector by a specific weight computed by:

Equation 0.12: Exponential term for k-RSA

$$w_i = e^{k\varphi_i}$$

(Eqn. 5.12)

φ in this case signifies the feature length of the feature itself. ***k*** is a scaling factor with continuous values (if ***k*** = 0 then **w*i*** = 1). At high ***k***-values, more weight is given to features with high ***φ*** (i.e. feature length). If there is a '1' for feature ***i*** in the original concept vector, then this is replaced by **w*i*** in the new vector. This means that when computing distance metrics on these re-weighted vectors, concepts will be drawn disproportionately closer to neighbouring concepts within the category, i.e. representational spaces formed at high ***k***-values will become more *categorically ordered*. I used the exponential function here to give a non-linear weighting to features with high feature length. A linear function would have maintained the ratio of the feature lengths between any two features. This in turn would prevent the representational space from changing since metric distances between concepts would remain the same. The function used was:

Equation 0.13: Weighted distance function

$$D^w(F_1, F_2) = D(w_{ij}F_j, w_{ik}F_k)$$

(Eqn. 5.13)

Where the metric function $D(F_1, F_2)$ is defined as the cosine distance between two weighted concept vectors:

Equation 0.14: Cosine distance between weighted concept vectors

$$D(F_1, F_2) = 1 - \frac{F_1 \cdot F_2}{|F_1||F_2|}$$

(**Eqn. 5.14**)

A set of 200 RDMs (or ***k***-RDMs) were produced each with a different value of ***k*** ranging from 0 to 20 at intervals of 0.1. At ***k = 0*** the function was equivalent to the un-weighted cosine distance ($e^0 = 1$). The figure below visually shows an example of how differences in the categorical structure arise at different ***k***-values (**Figure 5.8**).

Figure 0.8: Example k-RDMs

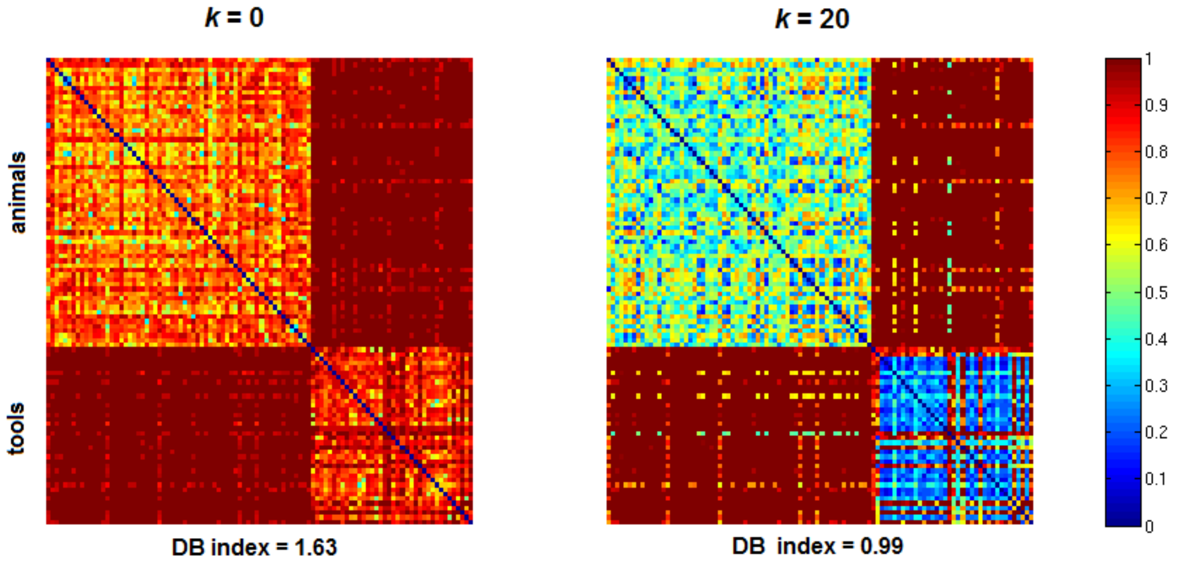


Figure 5.8: Two ***k***-RDMs set at **0** and **20** respectively. The preponderance of bluish-hued colors within the right RDM indicates that dissimilarities within categories are closer to zero compared to the RDM on the left. Similarly, DB indices for the two k-RDMs also reflect their differences in categorical cohesion with the left ***k***-RDM (**k = 0**) having **lower** cohesion compared to the right ***k***-RDM (**k = 20**).

Each of the 200 RDMs was then correlated to the corresponding layer RDM. A concept on each layer is represented as a vector of 750 values (= number of units) each indicating the probability of the unit being ‘turned on’ (i.e. taking a value of ‘1’). A layer RDM was formed by computing the cosine distance between these probability vectors. A set of four layer RDMs was created in this manner which were then correlated against the 200 ***k***-RDMs described earlier.

5.2.6 Model testing

The model was tested in two ways: firstly by a top-down pass of concepts from L4 to L1 and secondly by an analysis of each layer individually. For the top-down pass the reconstruction accuracy measure at L1 was the measure of interest. Category cohesion, information-theoretic measures and RSA were used to analyse the specific properties of each layer.

Model test-run

As explained above the model was trained in a bottom-up fashion and then ran in a top-down pass from L4 to L1. The aim was to re-construct the data at the bottom-most layer (L1). Representations in L4 would be abstracted away from the initial training data. These higher-level layers perform better in classification analyses (Hinton, 2008) exactly because concepts with shared information are drawn together more readily within the representational space of the model's layer. This comes at a cost however since precise information about the concept receives less weighting at this stage. Because of this, it is important to make sure that the weight-matrix operations during the top-down, re-construction process have the ability to recover the more-detailed information at layer L1.

All layers consist of 425 probability vectors (one for each concept). Although the identity of each vector, at each layer, is known to the experimenter – for example, which vector corresponds to the 'dog' concept – the model can only resort to its layer's responses to make a decision. During the top-down pass (i.e. the model is run from L4 to L1) a specific concept vector at L4 was chosen (e.g. 'dog'). This vector consisted of unit probability values which were the result of the training cycles described in **Section 5.2.1**. I then imposed a threshold value (derived from a uniform distribution of values ranging from 0 to 1) on the layer vector in order to 'binarise' it into a vector consisting only of 1's and 0's - whether a specific L4 unit will be activated or not is entirely dependent on a random threshold which is set at each run. Given the stochastic nature of the units it is important to note that at different model runs a particular concept can have multiple, yet similar patterns, of activation within a specific layer.

Figure 0.9: Overview of model's top-down run

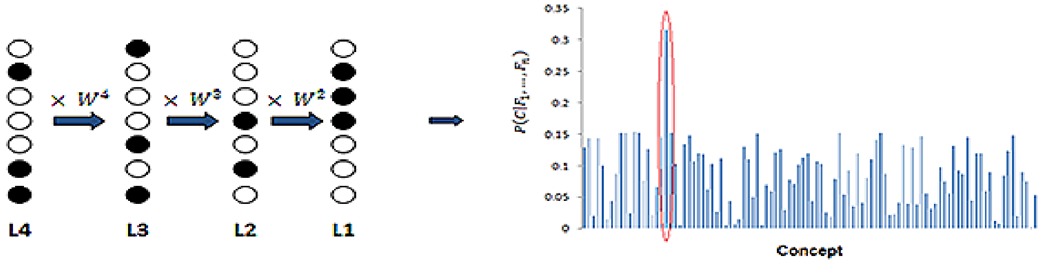


Figure 5.9: *Overview of the top-down model-ru*n. An activity pattern is transformed through a sequence of weight-matrix multiplications. The end-product at L1 is then evaluated through a Bayesian classifier to determine the precise identity of the concept entered at L4.

The top-down pass consisted of a sequential transformation of the L4 concept vector from one layer to the next (see **Figure 5.9**). The objective at the end of the process was to recover the precise identity of a concept at L1. Bayesian estimation (see **Section 5.2.3**) was used at L1 in order for the model to determine the concept's precise identity.

The overarching purpose of this procedure was to test the connecting weights between layers. If the re-construction accuracy at L1 was found to be particularly low (<50%) then it would have necessitated a re-training of the entire set of layers using a different set of parameters (number of epochs and/or units, learning rate). The entire set of concepts was run through the top-down pass of the model a total of 25 times. Accuracy for each concept was defined as the number of times the model produced the correct identification divided by the number of runs (n = 25). If the candidate concept with the highest posterior probability at L1 was the one I chose initially at L4 (the initial stage of the top-down pass) then this was scored as a correct identification.

Layer-by-layer testing

This method of testing was different from the top-down run described above in one fundamental way: the learned weights were not used at any stage of the process. Within a layer, each concept is represented as a vector of probabilities. To test the performance of each layer I ran the entire set of concepts through *each layer individually* a total of 25 times and extracted measures of performance at identifying this concept at three different levels of specificity – basic-level, general category and domain-level. At each run, a random threshold was set to produce a ‘binarised’ vector for a specific concept. It was necessary to repeat this process a number of times to get a reliable estimate of the layer’s performance. I extracted *accuracy* and *uncertainty* at different levels of specificity and averaged across runs. I also extracted the *type of errors* the model could make at basic-level. I expected that even though the model would make errors the misidentified concepts would be within the same category or the same domain, for the majority of cases. The degree of within-category and within-domain errors would be dependent on the representational capacities of each layer. For example, if the model were to make an misidentification given the concept ‘dog’, general, coarse representations would lead to errors within the broader semantic space of the correct concept (i.e. domain – ‘living’ such as ‘cockroach’) while more refined representations would lead to errors which are within the close semantic neighbourhood of the concept (i.e. category – ‘animal’ or ‘mammal’ such as ‘cat’). I extracted the average number of errors across the 25 runs for each of the three types – *within-category* errors, *within-domain* errors and *across-domain* errors.

Figure 0.10: Accuracy and uncertainty

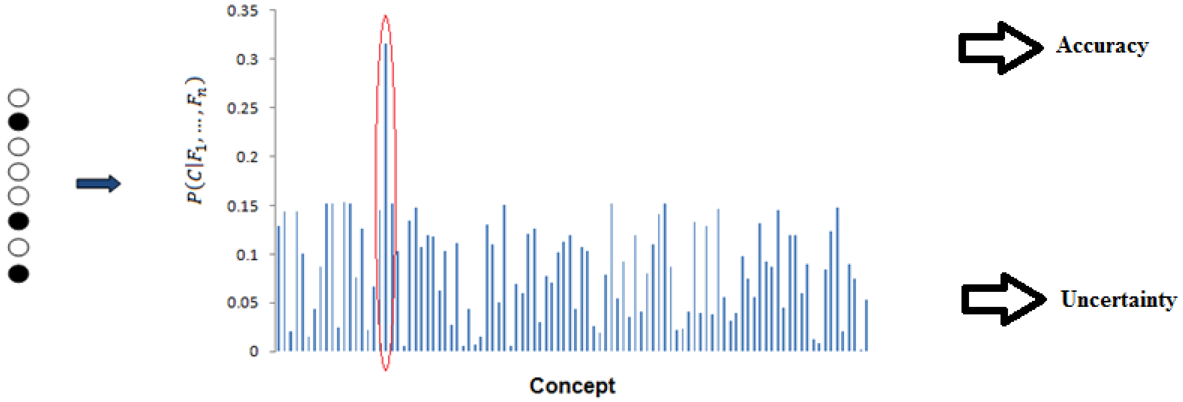


Figure 5.10: Individual patterns of activation for each concept are analysed within one layer at a time without running through the top-down pass of the model.

As mentioned in the Introduction, I also used Pearson's correlation to determine any relationship between *cLength* and the model's performance. Specifically I asked whether the performance at different layers was differentially affected by *cLength*. *cLength* quantifies both the sharedness and semantic richness of a concept (see **Chapter 2**). For example, if a particular layer contains highly abstract coarse-grained representations then high-cLength concepts would result in poor performance for basic-level identification. This is because a high degree of shared features would draw similar concepts together thus making them more confusable. Low-cLength concepts, which have a higher proportion of distinctive features, would remain relatively isolated within their semantic neighbourhood. Uncertainty would also correlate with *cLength* in this case. A high degree of shared features (high cLength) would result in a high number of possible candidates (high uncertainty) whereas a high degree of distinctive features (low cLength) would minimise possible candidates (low uncertainty).

However, feature statistics can affect uncertainty in different ways. For example, if the representational space is sparse and diversified, categorical members are unlikely to have an effect on a concept's uncertainty - semantic distances between category members are already large enough. What is more likely to have an effect are concepts within a concept's immediate semantic neighbourhood. It was necessary then to determine the exact source of uncertainty within each layer. This was the reason I computed a variation of *cLength* where I could vary the weight given to concepts with high feature overlap. This measure, *cLength[m]*, was computed in an identical fashion to conventional *cLength* with the simple addition of an exponential term. As described in Chapter 2 cLength is the geometric norm of a concept vector in concept space. A concept vector in this case consists of 425-elements each denoting the number of features they share with each other. When computing the norm all elements are typically given the same weight. The exponential term is a weighting on these elements which is proportional to its magnitude. An element, $\boldsymbol{C_i}$, within the concept vector is replaced with the value, $\boldsymbol{w_i}$, from:

Equation 0.15: Exponential term for cLength[m]

$$w_i = e^{\boldsymbol{m}C_i}$$

(Eqn. 5.15)

As ***m*** increases, concepts which share a lot of features with the concept in question will be given a disproportionately high weight. *cLength[m]* at high ***m*** is driven more strongly by immediate neighbours; *cLength[m]* at ***m*** = *0* is equivalent to cLength where *all* concepts in the set are given equal weighting. A total of 1000 values for ***m*** were used, ranging from 0 to 1 at an interval of 0.001. For each value of ***m*** there was a corresponding vector of 425 *cLength[m]* values (one for each concept). I then correlated (using Pearson's) each of these vectors against the corresponding uncertainty values at each layer. I then recorded the value of ***m*** which produced the maximum correlation across the entire set. For each layer there was a corresponding maximal ***m***-value. I reasoned that given that there are distinct representational properties for different layers of the model they would exhibit different optimum ***m***-values for *cLength[m]*.

5.3 Results

I first ran a top-down pass to determine reconstruction accuracy at L1. This was found to be **98%**. This high level of accuracy means that the model was able to recover information closest to the original training data set. Most importantly, this is a strong indication that the weights have been trained adequately and the layers are suitable for further analysis.

There were four analyses in total conducted on the model to assess: a) information content b) overall performance (i.e. accuracy and uncertainty) c) representational structure and d) the relationship between feature statistics and performance. The following sections are devoted to presenting the results from the layer-by-layer analysis.

5.3.1 Information content

I used mutual information between layer responses and concept identities to assess the information content at each layer and for all three levels of specificity.

Figure 0.11: Line plot showing the trend of information content from L4 to L1.

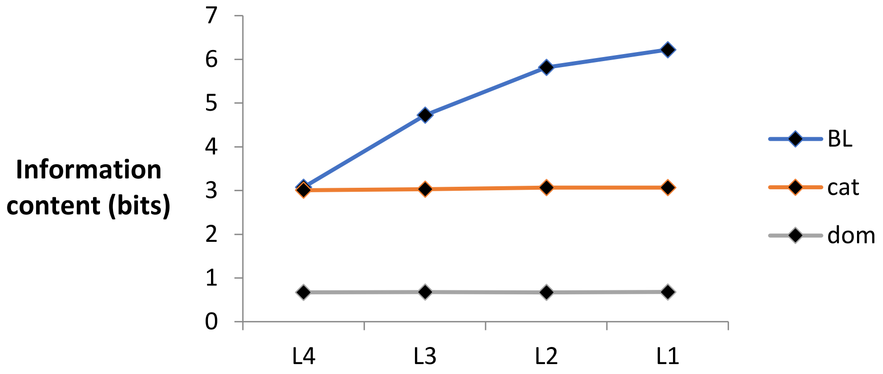


Figure 5.11: Line plot showing the trend of information content from L4 to L1.

The averages across layers for different levels of specificity were: 4.96 bits for basic-level, 3.04 bits for general category and 0.67 bits for domain. This difference in bits across levels of specificity was due to differences in stimulus entropy. Unsurprisingly, the large number of possible choices for basic-level (*n = 425*) meant that it had the highest stimulus, or theoretical maximum, entropy with domain-level the lowest. I decided to normalise each specificity level by the stimulus entropy. A score of 1 in this case would be the ceiling of how much information content is possible to represent. **Figure 5.12** shows that all layers are near-ceiling for domain-level and category identifications (mean: domain = 0.97, category = 0.98) but only L1 reaches a comparably high degree of information for basic-level (0.99). Overall, information content for levels of specificity *increases* from L4 to L1 but this trend is much more pronounced for basic-level.

Figure 0.12: Line plot showing the trend of information content normalised by stimulus entropy from L4 to L1.

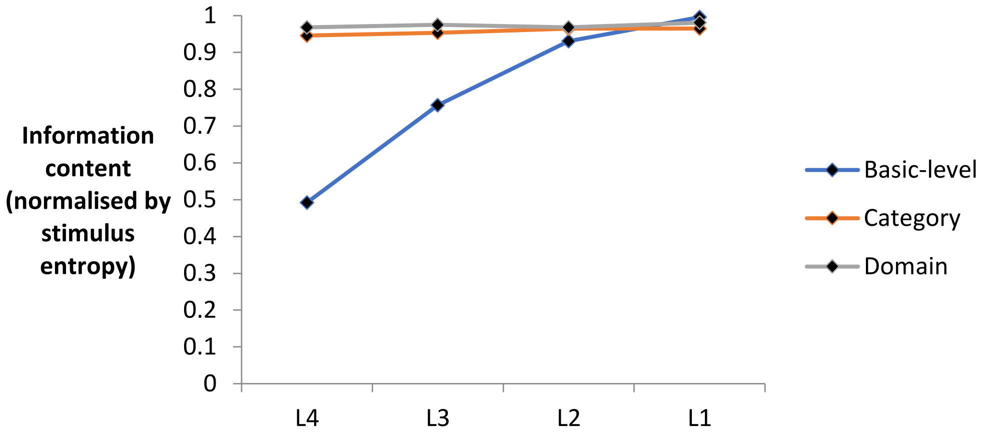


Figure 5.12: Line plot showing the trend of information content *normalised by stimulus entropy* from L4 to L1.

5.3.2 Performance: accuracy and uncertainty

Accuracy

Accuracy was defined as the proportion of correct predictions made by a particular layer while uncertainty is a characterisation of the probability distribution, $P(C|F_1, \ldots, F_n)$ (see Section 5.2). I collapsed the performance measures across all 25 runs and 517 concepts into a grand mean for each layer and specificity level. This resulted in a total of 12 values (4 layers x 3 levels of specificity) for each measure. L4 was the worst performing layer overall with accuracy reaching chance-levels (54.5%) (see **Figure 5.13**) for basic-level and moderate levels (70.5%) for general category. Domain-level performance remained near ceiling across the entire top-down pass (mean: 98%) while category and basic-level performance were lower (88% and 81% respectively).

Figure 0.13: (top) accuracy scores per layer (bottom) domain and L1 accuracy

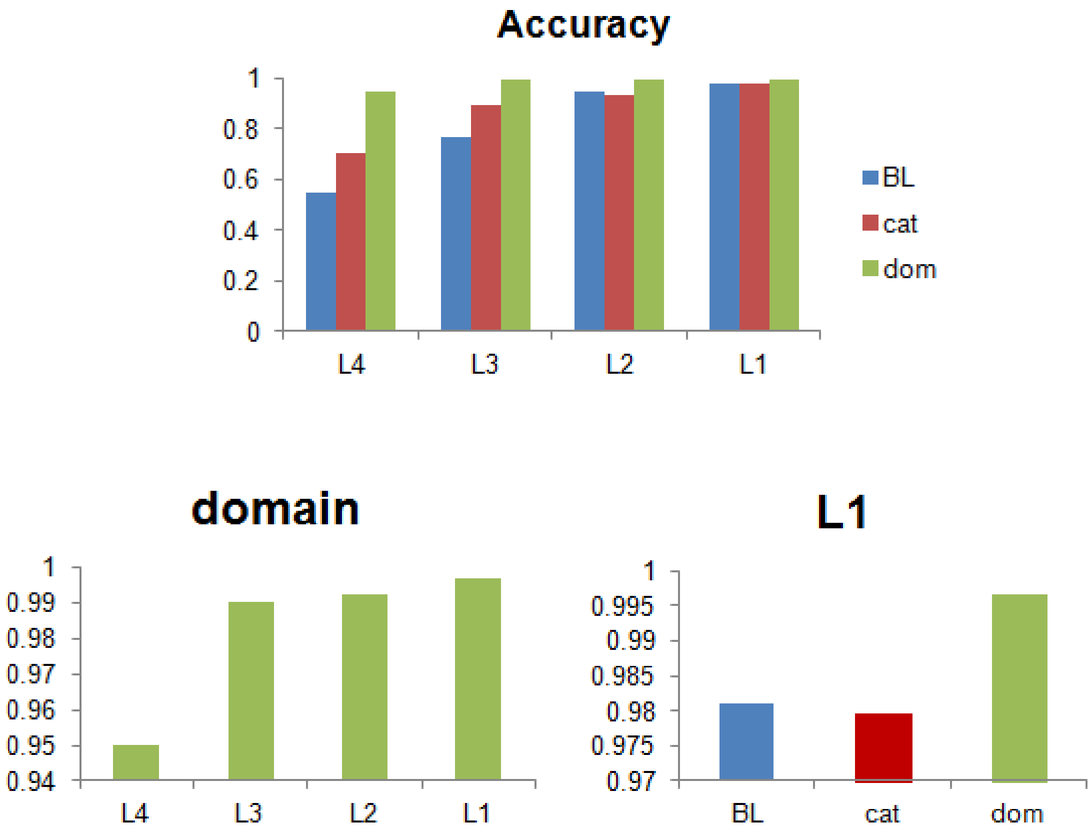


Figure 5.13: (**top**) Bar plot indicating the average accuracy scores for all layers at three different levels of specificity. L4 is the first stage of the top-down pass carrying the most general, coarse representations while L1 is the layer most closely related to the initial concept dataset. (**bottom**) Bar plots for domain and L1 accuracy at different axis ranges in order to highlight the fine difference across layers for the former and across specificity levels for the latter. Standard error bars were not included because of their extremely small size (<<0.01).

Types of error for basic-level identification

There were three types of error the model could make when identifying concepts at the basic-level – *within-category*, *within-domain* and *across-domain*. L4 made the highest, overall, number of errors with L1 the lowest (see **Figure 5.14**).

Figure 0.14: Average number of the three types of errors across 25 runs for all four model layers

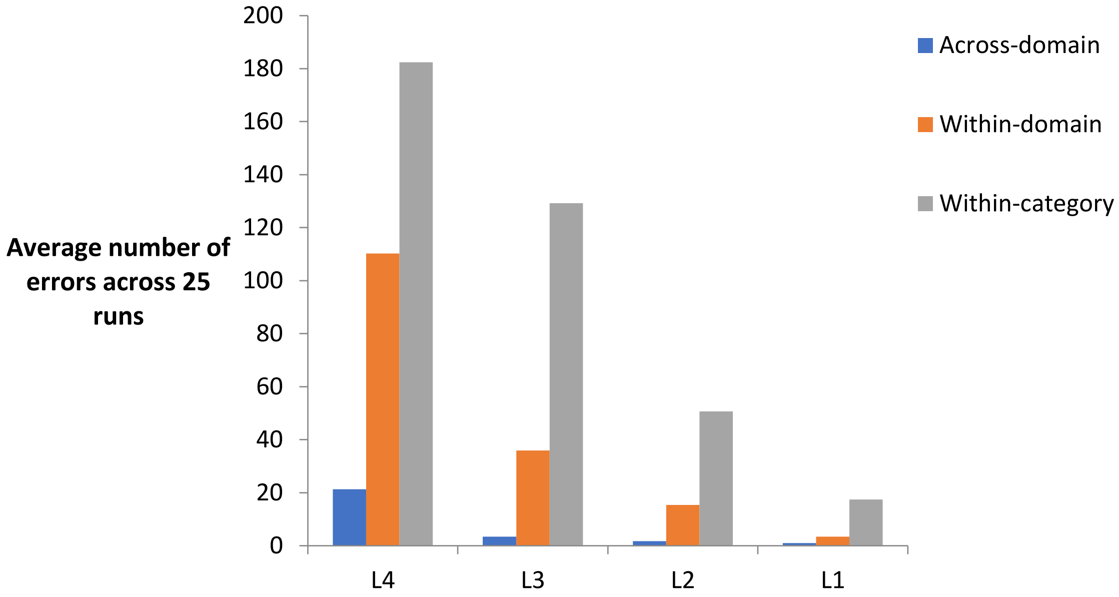


Figure 5.14: Average number of the three types of errors across 25 runs for all four model layers.

Most importantly, L4 had the highest proportion of within-domain errors (**35%**) vs. L1 which had the lowest (**16%**). On the other hand, L1 had the highest proportion of within-category errors (**80%**) vs. L4 which had the lowest (**58%**). This means that the specificity of the responses became increasingly more refined: if the model made an error at L4 it was likely to be broadly within the correct domain but not necessarily within the correct category. Conversely at the end-point of processing, at layer L1, errors are more likely to be within the correct category. Proportion for within-domain/within-category errors for layers L3 and L2 was comparatively the same with **21%**/**77%** and **23%**/**75%** respectively.

Effect of layer on overall accuracy

I computed the overall, average accuracy (across **all** specificity levels) within each layer: L4, 73.5%; L3, 88.4%; L2, 95.8%; L1, 98.6%. I then ran a Kruskal-Willis test (nonparametric one-way ANOVA which does not assume a normal distribution) to determine whether there was an overall effect of layer on the average accuracy (χ^2 (1550) = *1351.77*; p << 0.01).

Effect of layer on accuracy within individual specificity levels

I ran three Kruskal-Willis tests, one for each specificity level, to determine the effect of layer on the accuracy of *individual* specificity levels. I found that layer had an effect on accuracy for all specificity levels (***basic-level**: χ^2(2067) = 1271.6, p << 0.01;* ***category**:* χ^2 *= 737.66, p << 0.01,* ***domain**:* χ^2 *= 408.43, p << 0.01*). For domain, although the effects are significant the actual effect size is small (see Figure 3.3). This is probably due to the extremely small standard error values for each layer (**L4** = 0.0024; **L3** = 0.0015; **L2** = 0.0012; **L1** = 0.0008).

Effect of specificity level on accuracy within individual layers

I ran four Kruskal-Willis tests, one for each layer, to determine the effect of specificity level on accuracy within individual layers. I found that specificity level had an effect on accuracy within all layers (***L4**: χ^2(1550) = 766.34, p << 0.01;* ***L3**: χ^2 = 706.46, p << 0.01;* ***L2**: χ^2 = 332.3, p << 0.01;* ***L1**: χ^2 = 99.21, p << 0.01*). L1 performs very highly for all levels of specificity – as with domain the actual effect size is small but still yields a significant effect across levels due to small standard errors (**basic-level** = 0.0013; **category** = 0.0019; **domain** = 0.0008).

Uncertainty

Figure 0.15: Average uncertanity per layer

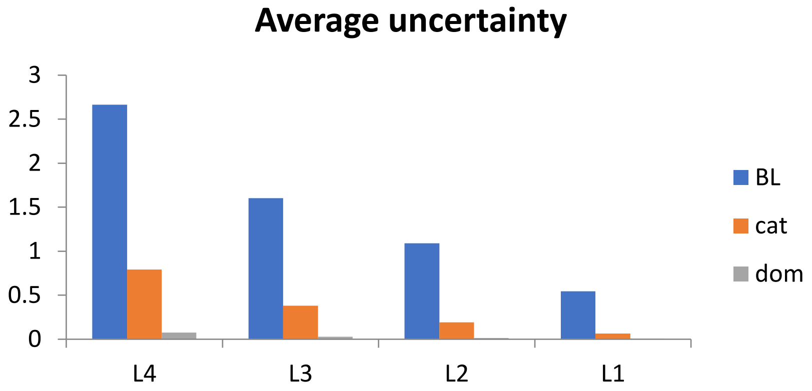


Figure 5.15: Bar plot indicating the average uncertainty scores for all layers at three different levels of specificity.

Uncertainty is closely related to information so it was expected that the average score for each layer (**Figure 5.15**) would follow the trends shown in **Figure 5.11** for all levels of specificity. However, it must be noted that the trend for general category did show a smooth decrease towards L1, unlike the information content trend where it remained fairly stable. Again the most pronounced change across layers was exhibited for uncertainty at the basic-level reaching its lowest point at L1 (0.55 bits).

Effect of layer on overall uncertainty

Again, as with accuracy, I averaged all uncertainty values within layers. A Kruskal-Willis test revealed a strongly significant overall effect of layer on average uncertainty (χ^2 (1550) = *3410.95*; p << 0.01).

Effect of layer on uncertainty within individual specificity levels

Similarly when the same test was run *within* individual levels of specificity there was a strong effect of layer on uncertainty (***basic-level****:* χ^2 *(2067) = 1220.62; p << 0.01;* ***category****:* χ^2 *= 1044.84; p << 0.01,* ***domain****:* χ^2 *= 1141.7; p << 0.01*).

Effect of specificity level on uncertainty within individual layers

I ran four Kruskal-Willis tests, one for each layer, to determine the effect of specificity level on accuracy within individual layers. I found that specificity level had an effect on accuracy within all layers (***L4**: χ^2 (1550) = 1335.29, p << 0.01;* ***L3**: χ^2 = 1171.64, p << 0.01;* ***L2**: χ^2 = 1098.91, p << 0.01;* ***L1**: χ^2 = 938.64, p << 0.01*).

Summary

Taken collectively both performance measures indicate a clear pattern of performance from L4 to L1. L4 overall had the greatest differentiation in performance across levels of specificity on both uncertainty and accuracy. L1 provided highly confident and accurate responses across the entire hierarchy of conceptual specificity. This is not surprising if the results in **Figure 5.11** are taken into account since they clearly indicate an increasing trend of information content for all levels of specificity which is especially pronounced for basic-level.

5.3.3 Representational structure

Category cohesion

Category cohesion was measured by the Davies-Bouldin index – the lower the index the higher the cohesiveness (see Appendix IV). I also included the initial McRae feature vectors in the result for comparison. There was a trend of decreasing cohesion from L4 to L1 (see **Figure 5.16**).

Figure 0.16: Category cohesion measures per layer and original training data

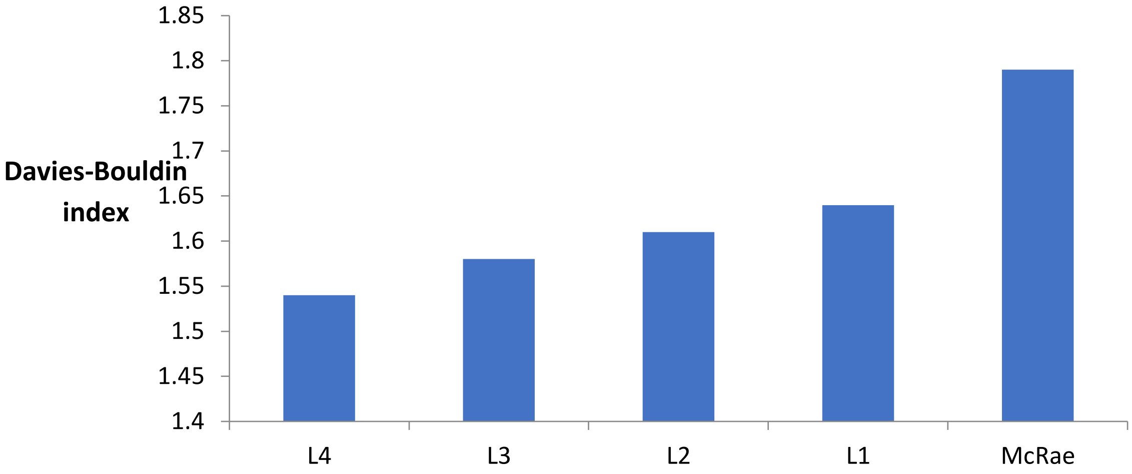


Figure 5.16: Category cohesion measures for all model layers and initial training dataset.

RSA

I computed a semantic distance RDM (using the cosine distance metric) for each layer and correlated this with a similar semantic distance RDM derived from the initial McRae vectors (see **Figure 5.17**). As shown in **Figure 5.18**, all layer representations maintained a highly significant correlation with the McRae RDM. However, this relationship waned from L4 and L1. In combination with the category cohesion results this means that fine-grained conceptual similarities are disrupted from L4 to L1 with a concurrent increase in category cohesion.

Figure 0.17: RDMs for layers and original McRae concept matrix

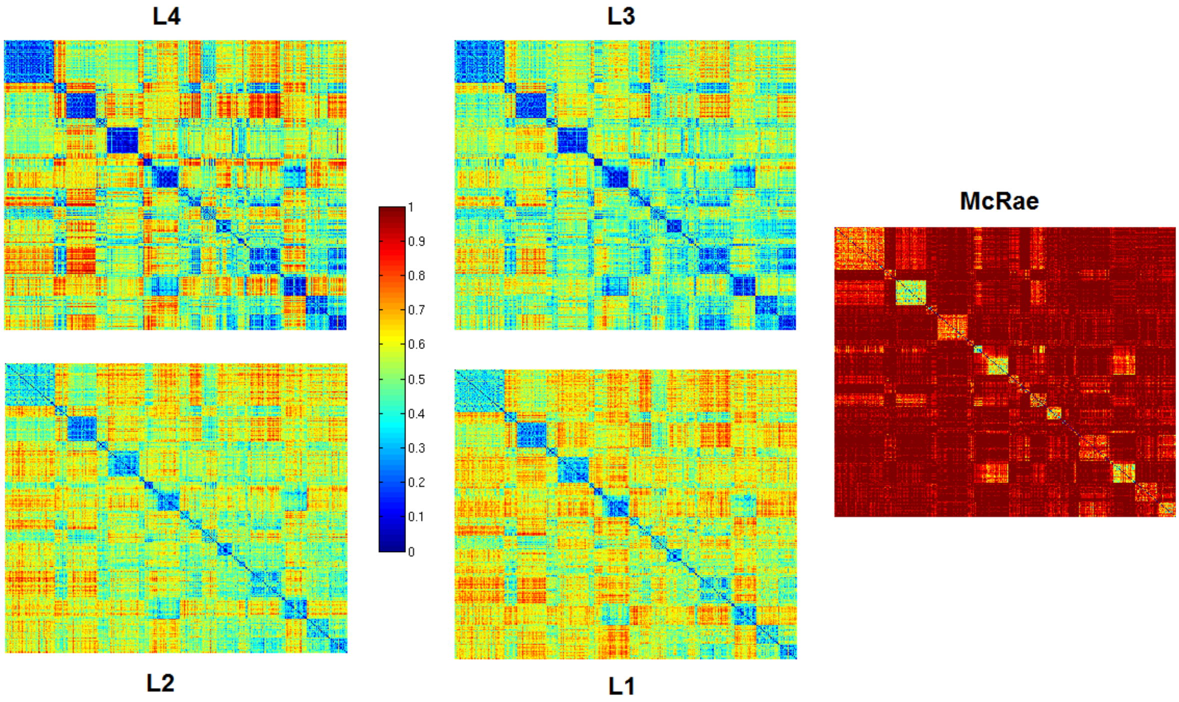


Figure 5.17: Model layer 425 x 425 RDMs (based on the 1-cosine distance metric) plus the RDM from the initial training dataset.

Figure 0.18: Second-order comparisons between layers and initial McRae vector RDM

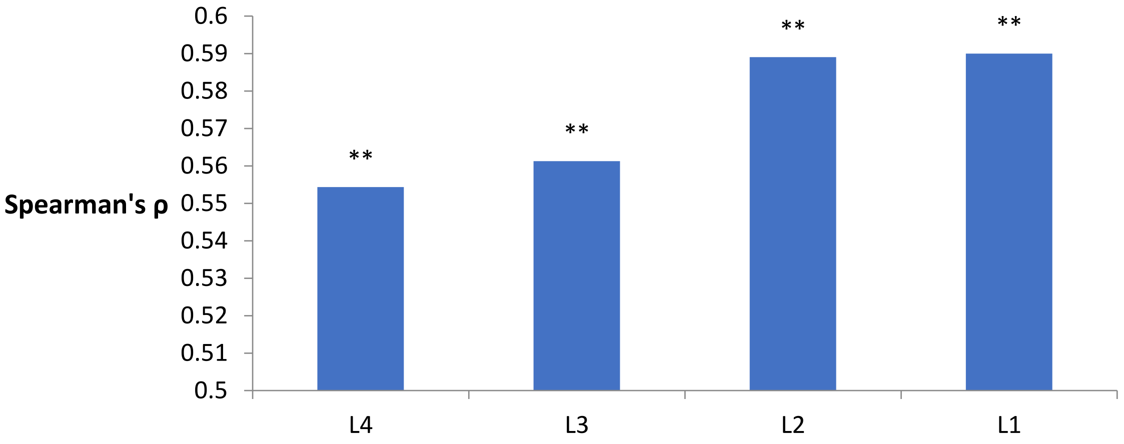


Figure 5.18: Second-order comparisons between the semantic distance RDM from the McRae dataset and each of the four layer RDMs.

A corresponding trend was also found from the RSA analysis using the 200 ***k***-weighted semantic RDMs. ***k***-values ranged from 0 to 20 (see **Figure 5.19**). High k-weights mean that semantic similarities are driven primarily by highly shared features (high feature length). As ***k***-weights decrease there is less dependence on highly shared features which also accounts for the decrease in category cohesion.

Figure 0.19: a) k-RDM correlation values b) optimal k-values for each layer

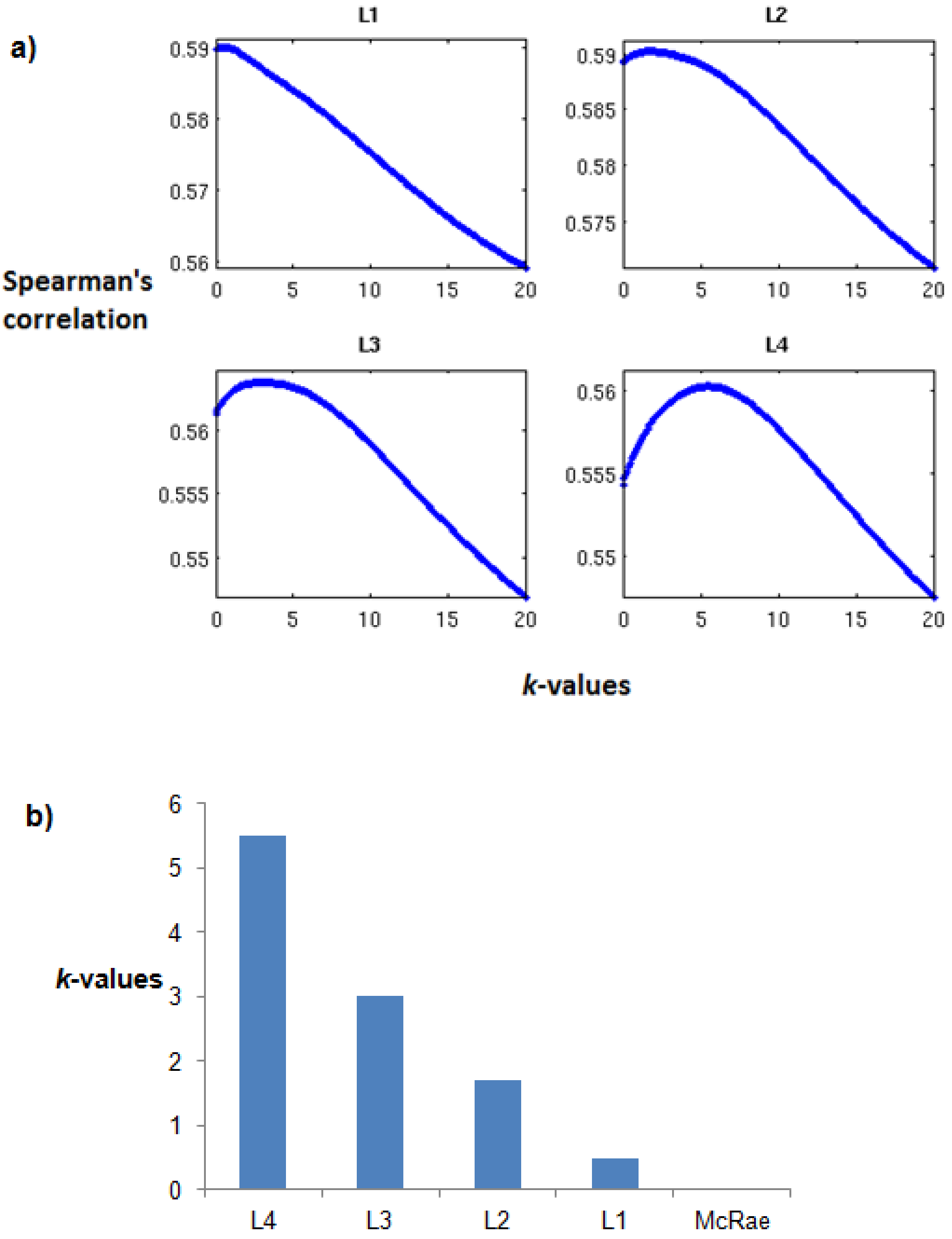


Figure 5.19: **a)** Correlation values between layer RDMs and the entire set of k-weighted semantic RDMs. **b)** Optimal *k*-values for all model layers and initial training dataset (McRae; *k = 0*).

Figure 5.19a depicts a ‘tuning curve’ of preference for each layer. Peaks along the curves indicate the ***k***-value at which the layer RDM produces the maximal correlation. The optimal ***k***-values indicate which features are driving conceptual similarities within a particular layer’s representational space (**Figure 5.19b**). This means that there is a loss of fine-grained information, as layers are trained from L1 to L4, largely because of a disproportionate weighting on highly shared features.

Summary

The purpose of this analysis was to assess the properties of each layer’s representational space. This is necessary because performance of the model alone is not sufficient to understand how information is organised within each layer. Category cohesion decreases monotonically from L4 to L1. This decrease in cohesion coincided with a decrease in the importance of shared features in structuring the representational space.

5.3.4 Performance versus feature-statistics

This analysis addresses whether *cLength* (a CSA-derived measure) had an effect on the performance of the model. As described in the Introduction, I generally expected a **negative** effect on **basic-level** performance and a **positive** effect on **category/domain-level** performance. I computed *cLength* for all concepts in the concept dataset and then correlated them against the two performance measures (i.e. accuracy and uncertainty). For each layer and specificity level performance measures were averaged across all 25 runs leading to a total of 12 sets of data (4 layers x 3 specificity levels). *cLength* was then correlated against each set (see **Figures 5.20** and **5.21**).

Accuracy vs. *cLength*

Figure 0.20: accuracy vs. cLength

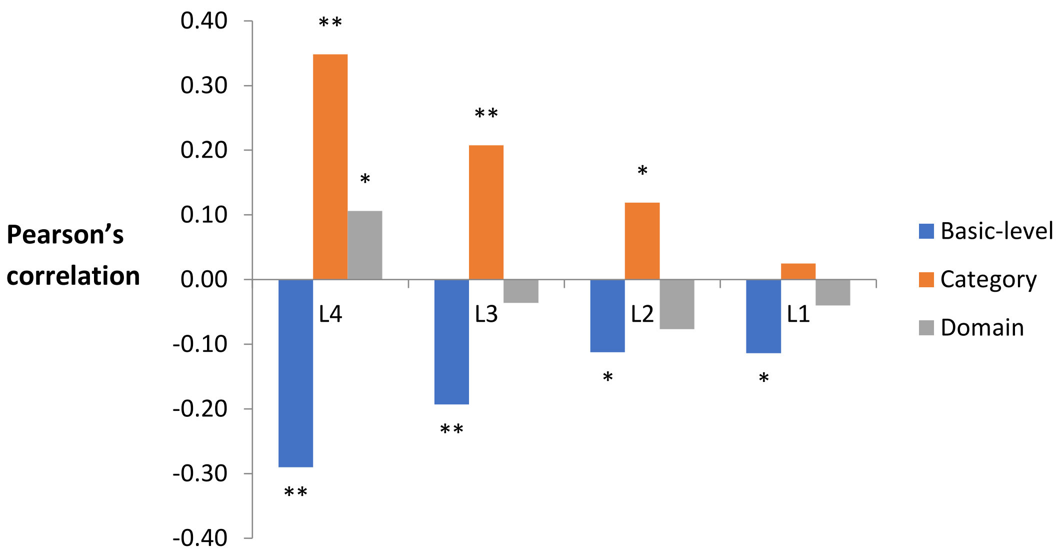


Figure 5.20*:* Correlation values for **accuracy vs. cLength** for each layer and level of specificity

As predicted, cLength and accuracy at **basic-level** were found to be significantly **negatively** correlated for all layers. However, this effect waned from L4 to L1. Similarly, there was a significant **positive** correlation between cLength and accuracy at **category-level** which decreased from L4 to L1. However even though there was a significant effect, in the expected positive direction for L4, there were no significant effects between cLength and **domain-level** accuracy for all the remaining layers.

Interaction between *cLength* and layer on accuracy

There was a visible interaction between *cLength* and layer for both basic-level and category identifications. Specifically as processing flowed towards L1 there was a decreased relationship between accuracy and *cLength*. This means that *cLength* had less of an effect on how well the model performed during the top-down pass. To test if this interaction was significant, for all levels of specificity, I ran a regression model with accuracy as the dependent variable and both *cLength* and layer as independent variables with an appended interaction term (*cLength* **x** *layer*). I then extracted a t-statistic for the resulting coefficients by dividing them with their standard error (see **Table 5.1**).

Table 0.1: t-statistics for regression model (accuracy)

t-statistic (d.f. = 2068)			
Accuracy	Basic-level	Category	Domain
Overall effect of cLength *across all layers*	2.43*	-3.63**	-2.38*
Overall effect of layer	-7.17**	-14.52**	-5.52**
Interaction cLength x layer	-6.43**	8.24**	2.87*

Table 5.1: t-statistics for regression model. **Dependent variable** = *accuracy* / **Independent variables** = *cLength, layer and interaction term.* * denotes p < 0.05; ** denotes p < 0.01

There was a strongly significant interaction for both basic-level and category-level identifications – as representations were transformed from L4 towards L1 there was a concurrent decrease in the dependence of accuracy on *cLength* (see **Table 5.1**). However the effects are in opposite directions: *cLength* has an initially negative effect on basic-level identification while category-level identification has a positive correlation. The interaction effect was less pronounced for domain-level – there was only an initial significant positive effect at L4.

Collectively these results show that although cLength does have an effect on the model's accuracy it is largely dependent on the layer (as shown in **Table 5.1**). At the earliest stages of processing (i.e. L4) the model can make better predictions for category when concepts have high cLength values while the opposite is true for basic-level. As the flow of processing progresses towards L1, cLength has less of an effect. Interestingly, the relationship between category accuracy and cLength does not reflect the relationship between domain accuracy and cLength. This was something not predicted initially since I expected cLength to facilitate domain-level identifications in the same manner it does with category.

Uncertainty vs. *cLength*

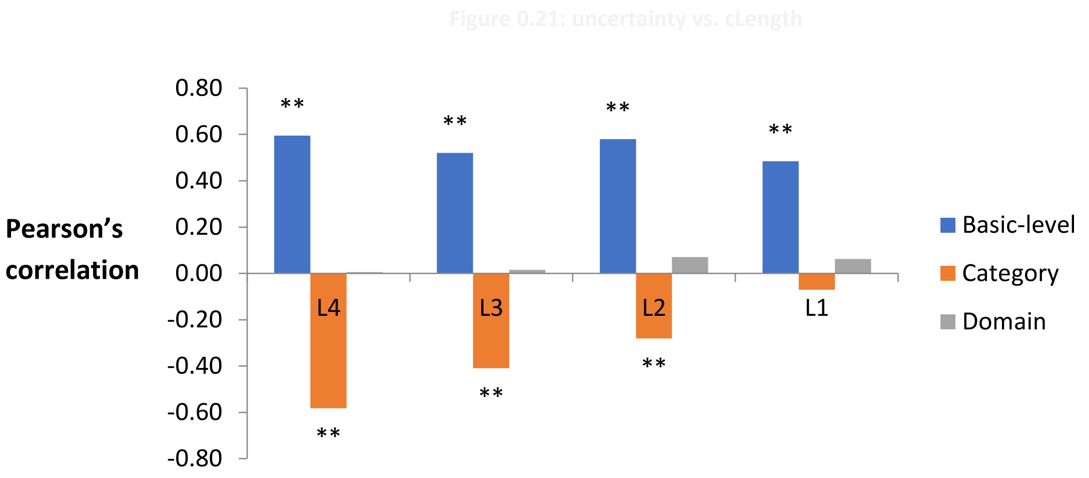


Figure 5.21: ***Uncertainty vs. cLength*** for each layer and level of specificity

As predicted, **basic-level** uncertainty was positively correlated with cLength across all layers (**Figure 5.21**). High uncertainty in this case equates to low performance so this means cLength had a negative effect on performance which corroborates the findings reported for accuracy in **Figure 5.20**. Conversely, **category** uncertainty was negatively correlated with cLength especially in layers L4 to L2 – again validating the initial predictions. High-cLength concepts result in category-level responses with low uncertainty. This effect decreased (as with accuracy) from L4 to L1. As with accuracy, cLength had no effect on **domain-level** identification uncertainty.

Interaction between *cLength* and layer on uncertainty

As with accuracy I ran the same regression model but with uncertainty as the dependent variable.

Table 0.2: t-statistics for regression model (uncertainty)

t-statistic (df = 2068)			
Uncertainty	Basic-level	Category	Domain
Overall effect of cLength *across all layers*	10.12**	4.8**	1.08
Overall effect of layer	18.74**	26.14**	4.57**
Interaction cLength x layer	***1.56***	***-13.62****	***-0.6***

Table 5.2: t-statistics for regression model. **Dependent variable** = *uncertainty* / **independent variables** = *cLength, layer and interaction term.* * denotes p < 0.05; ** denotes p < 0.01

Uncertainty at basic-level remained highly correlated with *cLength* at all layers. This means that layer played no role in determining the strength of the relationship between basic-level uncertainty and cLength. By contrast, there was a significant interaction between *cLength* and uncertainty at the category level. *cLength* had no effect on domain-level uncertainty (**Table 5.2**).

cLength[m] vs. uncertainty

I correlated uncertainty at basic-level and category-level against *cLength[m]* for all values of ***m*** (n = 1000; ranging from 0 to 1). This was done to specifically determine the extent of the semantic neighbourhood around a concept which affects identification uncertainty. As ***m*** tends towards one, there is more weight given to the closest semantic neighbours of the concept. Optimal ***m***-values provide an insight, beyond the standard formulation of cLength, into what's driving uncertainty at specific layers: high ***m***-values mean that uncertainty is driven mostly by immediate semantic neighbours; for ***m***-values closer to zero, uncertainty is driven by a wider expanse of semantic space which potentially includes most members of a category.

The winning ***m***-value was the one which produced the strongest correlation with a specific set of uncertainty values. L1 had the highest ***m***-value for *cLength[m]* while L4 had the lowest (see **Table 5.3; Figure 5.22**).This means that uncertainty at L1 tends to be higher for concepts which have close and immediate semantic neighbours (high ***m***-values). Conversely in L4, uncertainty is mainly driven by representational density within a broader semantic neighbourhood (low ***m***-values)).

There was a similar increasing trend for category-level. Concepts within cohesive categories resulted in lower uncertainty for L4. Layers L3 and L2 had lower m-values which meant that uncertainty was driven by competitors within a narrower semantic neighbourhood. As with basic-level L1 scored the highest ***m***-value but the correlation was non-significant.

Table 0.3: Optimal *m*-values – cLength[m] vs. uncertainty

	Basic-level		**Category-level**	
	Optimal *m*-value	**Pearson's r**	**Optimal *m*-value**	**Pearson's r**

L1	0.172	0.61**	0.245	0.08
L2	0.143	0.67**	0.135	0.3**
L3	0.115	0.58**	0.095	0.42**
L4	0.061	0.61**	0.078	0.61**

Table 5.3: Optimal ***m***-values (out of a thousand possible values) and corresponding correlation measures for both basic-level and category-level uncertainty.

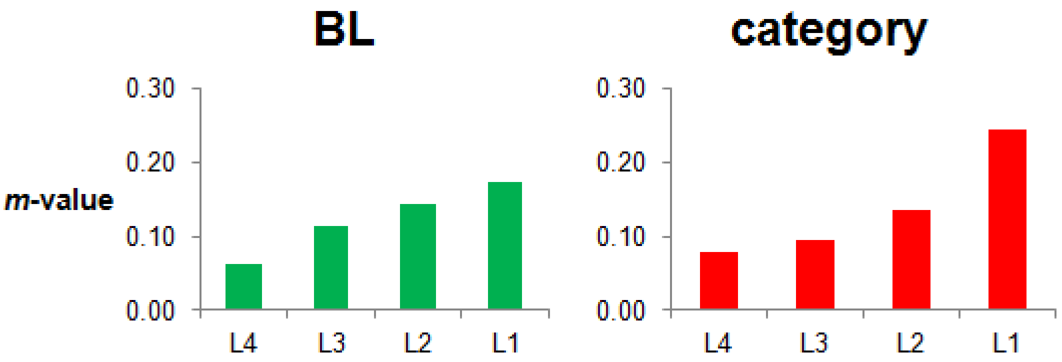


Figure 5.22: Bar plot for cLength[m] *m*-values for both basic-level and category

With respect to domain-level there was no overall effect of *cLength* on uncertainty across layers so I did not run any further analyses using *cLength[m]*.

5.4 Discussion

The aim of all the analyses carried out in this chapter was to determine whether the emerging representations within the model had any cognitive relevance to the claims made by the CSA. If we view the top-down pass of the model as a computational instantiation of how concepts are processed in the ventral stream during object recognition then there should be a meaningful pattern in how layer representations evolve from L4 to L1. This is particularly important when comparing the output derived from the model to empirical data. Specifically, it would be highly problematic if the test-run, described in Section 2.2.1, revealed an irreversible disruption of semantic information during the internal processes of the model. It would cast serious doubts as to how relevant or appropriate it would be to suggest that this particular computational framework can account for the actual conceptual processing that takes place in the brain.

5.4.1 Information and representational structure

There was one encompassing phenomenon observed across the entire set of results: an increasing monotonic (but not necessarily linear) trend in the richness of the representational capacity in each layer (see Figure 3.1). Information for basic-level especially exhibited a pronounced increase compared to both category and domain-level. From the viewpoint of training, this means that there is a *loss* of fine-grained information with each added, newly-trained layer (training progresses in a bottom-up fashion building one layer at a time on top of the training data). However, the nature of the remaining information is sufficient to keep performance at the category and domain level high. Response patterns across units become less distinguishable between members of the same category. Interestingly enough, response patterns are informative for lower levels of specificity across all layers of the model. This would suggest that the model does not profoundly distort the semantic space but rather abstracts away from detailed information regarding specific objects. Higher layers such as L4 still provide information about a concept but only on the category / domain level (e.g. ‘animal’ / ‘living thing’).

Although fine-grained distinctions amongst concepts are given less weight during training they can be recovered through the inverse top-down transformation. The robustness of the weights resulted in a near-perfect 98% reconstruction accuracy at

L1 during the initial full model battery of testing. It is important to stress a pertinent limitation of the model at this point: adding further layers resulted in a significant drop in reconstruction accuracy as well as severe distortions in the representational space of the higher layers. This in itself imposed a limit on the number of layers during model construction. This limitation makes a precise mapping of the layers on the actual neuroanatomy of the ventral stream intractable. It would be highly inefficient then, in light of this, to try and equate a particular *layer* with a particular *region* within the ventral stream. Rather, the most important aspect of the model should be the way representations change during the top-down pass and how the extracted performance measures relate to behaviour. It would be interesting to re-build and re-train a model with a similar architecture but with a dataset with a larger number of items which is more typical of those used to train deep belief networks (Hinton, 2007). This might result in more layers revealing more latent properties of the data. Hypothetically, the information trend observed (where fine-grained information diminishes) could be generalised to encompass all three levels of specificity given a sufficiently high number of layers.

The tendency of the model towards an increasingly categorical organisation of concepts was further corroborated by the results from the category cohesion analysis. Categorical cohesion *increased* from L1 to L4. In other words, the metric distances of concepts within a category decreased while distances between concepts from different categories increased. This would explain the loss of fine-grained information at the basic-level as the reduction in distance between within- category members makes categorical exemplars increasingly confusable. Results from the ***k***-RSA analysis shown in **Figure 5.19** suggest that this effect might be due to an increased weighting on features with high feature length. This is not surprising given how the model is trained. Feature length is dependent on both the sharedness and the correlational strength of the feature. During training of a particular RBM, features which appear frequently and are highly correlated will have more of an effect on the weights. The probability of activation (see **Equation 5.1**) will be more influenced, comparatively, by the summative weighting of these high-scoring features.

The representational and informational composition of the layers appears to be in close agreement with the CSA. Representations at the early stages of processing (i.e. layer L4 in the model) are unrefined in terms of basic-level specificity yet highly ordered categorically. Processing further down the top-down pass results in richer representations, as evidenced by the information-theoretic measures. Furthermore, these types of representations and the morphing of semantic space emerged with no external, a priori guidance; rather, they were the result of an entirely unsupervised learning process.

5.4.2 Performance measures

Accuracy

Given that performance is highly linked to the specific capacities of each layer it is not surprising that accuracy follows a similar trend to the information content trend across layers (see **Figures 5.11** and **5.12**). There was both an effect of layer and specificity level on accuracy. Basic-level performance was the most affected by layer (as with information content at this level) and domain-level identification was, overall, the most accurate across all layers.

However, although the effect of layer was significant for domain-level identification, all layers produced highly similar mean accuracy scores (L4: 95%; L3: 99%; L2: 99.2%; L1: 99.6%) which was not the case for basic-level and to a lesser extent, category-level. The reason for the strength of this effect might be because domain-level performance for one specific layer is highly consistent across concepts with minimal deviation from the overall mean. This would lead to differences being significant even though the difference in mean accuracy is small.

Compared to both domain- and category-level the effect of layer on basic-level accuracy was much stronger. The large variation in accuracy across layers indirectly reflects the same effects observed in the information content results (**Figure 5.11**): a pronounced decrease of fine-grained information leads to a decrease in basic-level accuracy. Similarly it also highlights the specific ways in which the representational spaces are arranged within individual layers. Taking L4 as an example, which also scored the highest *k*-weight during the ***k***-RSA analysis (see **Figure 5.19**), it can be reasoned that a strong weighting on highly shared / highly correlated features within concepts of a particular category would inhibit performance at basic-level. This is because distinctive features would receive less importance making distinctions between semantically similar concepts difficult. For example, the concept '*chicken*', which has many highly shared / correlated features within its category (*cLength* = 54.8) scored an average accuracy of 0.48 at L4. Comparatively, the concept '*crowbar*', which has fewer, more distinctive features (*cLength* = 27.5) scored an average accuracy of 0.92. The types of errors made by the model at different layers also lend more support to the view that the representational space does indeed become more refined towards L1. If shared features do receive more weight at the earliest layers (e.g. L4/L3) then it is not surprising that the errors the model would make would be within the correct category. Although this higher weighting on shared features should not negatively affect performance at less specific levels of identification (category / domain) layer L4 seems to be more prone at making category-identification errors but within the *correct domain*. The fact that L4 performs at a moderate accuracy level (70%) for category-level suggests that even distinctions between categories are blurred. For example, features such as '*has_legs*' might have received a higher weighting leading to conflations between categories such as 'mammals' and 'reptiles' (see **Appendix I** for category listing). The model is less likely to make within-domain errors as representations become more differentiated and responses more refined. This explains why at L1 the vast majority of errors are within-category and rarely within-domain.

Once *k*-values decrease for a particular layer there is also an increase in category-level performance – the next layer following L4, L3, exhibited an increased performance at 89%. Intuitively, one might expect a straightforward linear relationship between feature length and category cohesion, that is highlighting similarities bring concepts closer together making categories tighter and more distinguishable. Although this might be indeed true, up to a certain point, this does not necessarily imply a concurrent improvement in performance. I computed the average accuracy for each domain (see **Figure 5.23**). My reasoning was that if the aforementioned was true then living things, which generally possess many highly shared/correlated features, would have less accurate basic-level identification. Conversely, category-level identification would be more accurate for living things exactly because their type of features would pull similar concepts together thus increasing cohesiveness. I found that this was indeed the case especially for L4. It appears then that the low category-level accuracy scores for L4 are mainly driven by nonliving things. This could be accounted for by highly shared but not categorically unique features such as 'made_of_metal' which increases similarity between concepts which belong to different categories (e.g. 'fork' - *utensil* and 'car' - *vehicle*).

Figure 0.23: Mean accuracy scores for both nonliving and living things

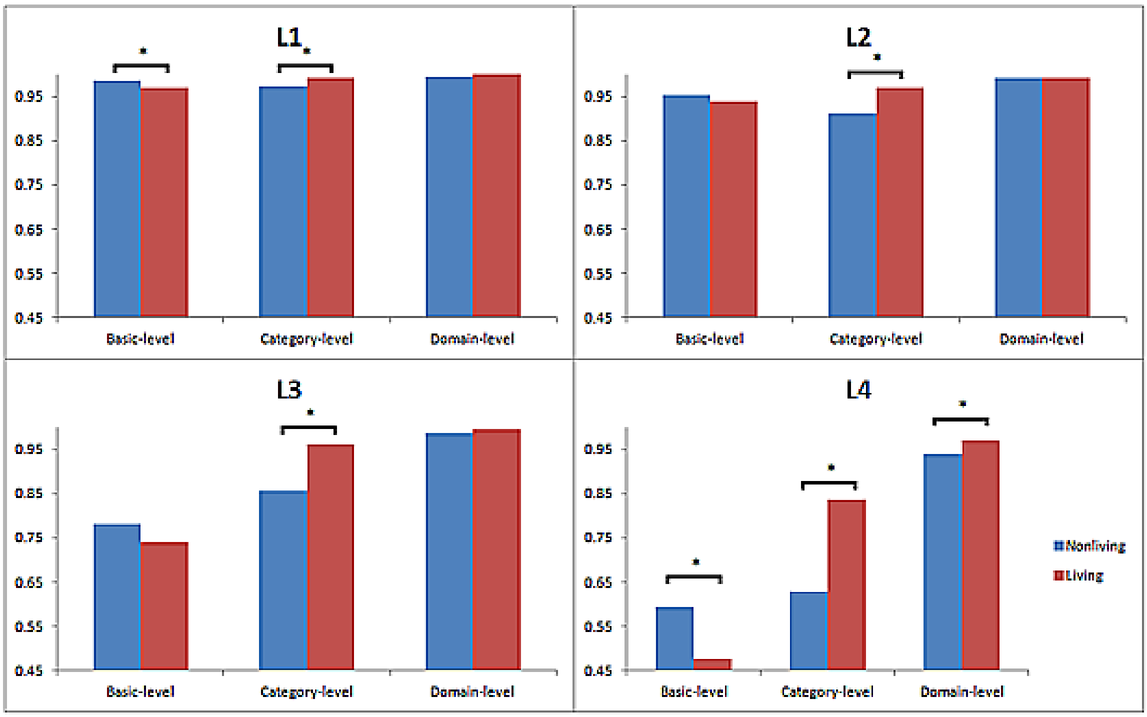


Figure 5.23: Mean **accuracy** scores for both nonliving and living things. * indicates significant difference between groups ($p < 0.05$; Mann-Whitney test).

Uncertainty

High uncertainty means that there are a high number of high-probability candidates. Although in this case the model could still make a correct guess, the stochastic nature of the units means that future performance would be unreliable. Uncertainty would be highly dependent on how precise pattern responses are with respect to specific concepts. Unsurprisingly, layers with high overall uncertainty also perform poorly in terms of accuracy.

Uncertainty scores at different levels of specificity are not necessarily correlated. The high uncertainty for basic-level at L4 does not necessarily explain why it also has the highest uncertainty scores for both category and domain-level. This particular layer could have produced a high number of possible candidates, thus increasing basic-level uncertainty. At the same time all the candidates could have been within the concept’s category which would result in a near-zero uncertainty score for category. It would be conceivable then to imagine a scenario where a layer has a very high basic-level uncertainty but a near-zero category-level uncertainty. The fact that this is not the case provides an insight into the nature of the emergent representations of the model. It means that during identification, candidates are not confined within one particular category. This also, at least partially, explains why this layer produced low accuracy scores for category-level identifications – candidates spread over a wide expanse of semantic space leading to identification inconsistencies across repeated runs and high uncertainty. The cause could very well be the same one underlying the poor accuracy scores: highly shared/correlated features draw together many concepts across categories thus morphing the semantic space into being less differentiated.

I computed the mean uncertainty with each domain (living vs. nonliving). As with accuracy, I reasoned that L4 would have a lower overall basic-level uncertainty for nonliving things, given their high number of distinctive features. L1 would also show the same relationship but less pronounced comparatively. For category-level uncertainty the effect of feature statistics on the representational space could either lead to a merging of categories, thus *increasing* uncertainty because category boundaries become less pronounced, or increased cohesiveness thus decreasing uncertainty.

Figure 0.24: Mean uncertainty scores for nonliving and living things

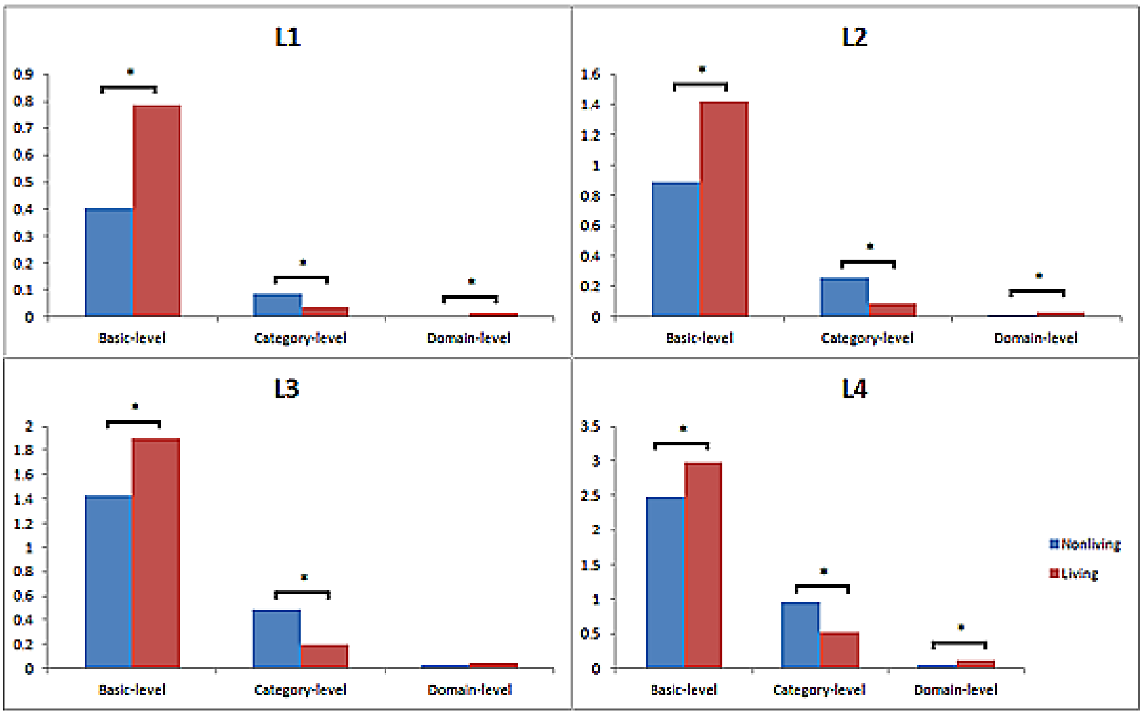


Figure 5.24: Mean **uncertainty** scores for both nonliving and living things. * indicates significant difference between groups ($p < 0.05$; Mann-Whitney test).

I found that this was indeed the case (**Figure 5.24**). Basic-level uncertainty was significantly higher for living things across all layers. The reverse was true for category-level. This would suggest that the effect of feature statistics had a positive effect on category cohesiveness within living things. For nonliving things, highly shared/correlated features led to a disruption of the general category structure which explains the increased category-level uncertainty compared to living things.

5.4.3 Feature statistics effects on performance

The RSA analysis of the individual model layers revealed that representations become more condensed within their categories making them increasingly more confusable. Although this analysis has shown that there is indeed an effect of feature statistics on how the representations change from layer to layer there is still a necessity to understand how these measures affect performance. Specifically, how dependent is the model’s performance on the featural make-up of a specific concept? I will address this question separately for both accuracy and uncertainty.

***cLength* vs. accuracy**

It has already been shown (**Figure 5.13**) that accuracy steadily increases from L4 to L1 for all specificity levels. I surmised that the primary cause of this trend was the increase in fine-grained information within each successive layer along the top-down pass. Patterns became more distinguishable and thus more informative regarding distinct properties of a particular concept. However how does the accuracy within each layer relate to cLength at different levels of specificity?

cLength measures the degree of overall sharedness of a concept. A high number of highly shared and correlated features leads to a high *cLength* score. If, as suggested by the results specifically in **Figure 5.19**, there is an increasingly disproportionate weighting on shared features then basic-level accuracy for high-cLength concepts would be much lower compared to low-cLength concepts. This is because the featural make-up of high-cLength concepts makes them more confusable with other members of their category. Conversely, concepts with low cLength, would be expected to be less affected by this series of transformations. The relationship between accuracy and cLength would be dependent on how much weight is placed on shared/correlated features. L4 for example would fare much worse in basic-level identification for high-cLength concepts and its performance would therefore exhibit the strongest dependency on cLength. The results in **Section 5.3.4** show that this is the case: cLength has a significantly negative effect on how accurate is L4 in making basic-level identifications. High-cLength leads to low accuracy and vice versa.

To illustrate this point further, I ranked all categories by both *average cLength* and *average accuracy* at L4 (see **Figure 5.25**). It can be seen that the worst performing categories mostly come from the living domain, such as '*fish*', '*vegetables*' and '*birds*', in which members possess a high degree of shared features amongst them. Conversely, the best performing concepts come from loosely bound categories, such as '*container*', '*natural objects*' and '*outdoor artifacts*', which are mostly non-living things which have a relatively small portion of feature overlap. It must be noted however that the 'animals' category, which possesses the second highest overall *cLength* score, performed at 50.6%. Although a low score it was still higher than lower-cLength categories such as 'clothing' and 'food'. This would suggest that although there is a strong correlation there are still factors not captured by the *cLength* measure which account for performance. What is equally important is that the effect of cLength on accuracy *decreases* as a function of layer – in other words there is a significant interaction (see **Table 5.1**) between cLength, accuracy and layer. Again this appears to be a natural outcome of the representational capacities of each layer. As highly shared features become successively de-emphasised from layer to layer there is a subsequent increase in fine-grained information. This means that at the point where conceptual representations reach L1 they are already sufficiently drawn away from any semantic neighbours and *cLength* would have less of an effect.

Figure 0.25: Ranking of categories by average cLength and accuracy at L4

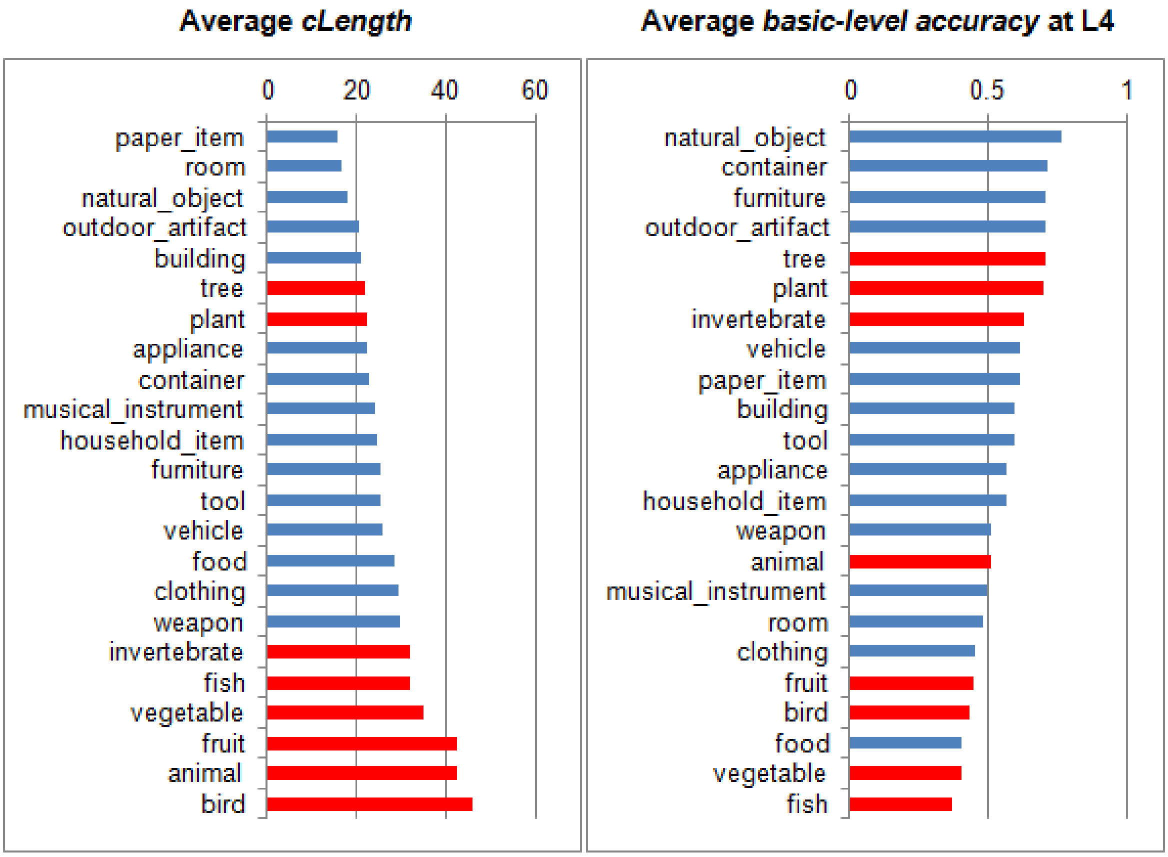


Figure 5.25: Ranking of categories by average cLength and accuracy at **L4**. Red color indicates living things.

However the same trend was not observed for both category- and domain-level identifications which exhibited an overall positive effect of cLength. Again, this is directly relevant to the representational capacities of the layers. If representations within a layer are weighted more towards shared features it follows that they play a crucial role in determining category membership as opposed to distinctive features only found in a few concepts. *cLength* in this respect enhances categorisation: if a concept possesses a large degree of shared features within a category it is more likely to be identified by the model as a member. This explains why the correlational trend between category-level accuracy and cLength follows the same trend observed for *k*-values in the RSA analysis. In the case of domain-level identification however there is no clear linear trend as with the previous specificity levels (only L4 shows significant positive correlation with *cLength*). This might be because the domain-level is largely an artificial construct whereby categories with largely non-overlapping features are 'forced' into the same categorisation. For example, '*tools*', '*vehicles*' and '*musical instruments*' have no overlap either in terms of perceptual or functional features but are nevertheless classified as '*non-living*'. *cLength* accentuates highly shared *overlapping* features between concepts and so in this case would have little effect on cohesion across the domain-level categorisation. In fact, there is a *negative* (although non-significant) effect on domain-level accuracy for all layers except L4.

5.4.4 *cLength* vs. uncertainty

Uncertainty for all levels of specificity decreased in a monotonic fashion from L4 to L1 (see Section **5.3.4**). In other words there was a clear effect of layer on uncertainty: conceptual representations became increasingly more refined and detached from their categorical neighbours. However, how does *cLength* relate to uncertainty levels within layers?

As shown in **Figure 5.21** and **Table 5.2** there is a maintained, strong correlation between basic-level uncertainty and *cLength* throughout all the layers of the model with no effect of layer. Conversely, for category-level uncertainty there is a clear interaction between *cLength* and layer ($t = -13.62$; $p < 0.01$). Finally, similar to accuracy, domain-level uncertainty is largely unaffected by *cLength*.

There is a high degree of correlation between basic-level uncertainty and cLength for all layers. This is a sensible finding because, as described earlier, uncertainty is effectively a reflection of the number of probable candidates during the model's processing. If a concept has a large degree of overlap with many concepts (a property which is captured by cLength) then it follows that there will also be large number of likely and semantically similar candidates. This is especially the case for layers where representations are dependent on shared features such as L4. However, as noted previously, the nature of the representations changes along the top-down pass. When the model reaches the final layer of the top-down pass, L1, the representational space becomes sparse and diversified allowing for fine-grained identification. Uncertainty at this point would largely depend on the closest semantic neighbours of a concept rather than overall categorical similarity. This would explain the strong correlation between uncertainty at L1 and *cLength*, even though there was no significant correlation for accuracy. The results using cLength[m] also strongly suggest that this is the case: ***m*** values steadily increase from L4 to L1. This means that possible candidates become more and more restricted within the representational space. At L1, uncertainty is driven mainly by immediate semantic competitors. For example a concept such as '*tiger*' shares many features with other members of the '*animals*' category. At L4 semantic competitors would span most members of the category because of the increased weight on shared features such as '*has_legs*', '*is furry*', etc. However as the concept is transformed throughout the top-down pass the more distinctive features ('has_stripes') would gain more weight. The pool of semantic competitors then becomes increasingly restricted to just a few concepts which share a large number of both shared and distinctive features with '*tiger*' (e.g. 'lion' etc). This explains why uncertainty L1 has the highest *m*-value of all layers – at this level of granularity only the *most* confusable concepts have any bearing on how confident the model is in making a basic-level identification.

In contrast with basic-level, the effect of layer is significant on the correlation between category-level uncertainty and *cLength* with a weaker (yet significant) overall effect (t = 4.8; p < 0.01). The strongest correlation was found at L4 where increased cLength correlated with decreased uncertainty. This effect at this particular layer is not surprising since L4 has a strong dependency on shared features (see **Figure 5.19**). High-*cLength* concepts become more cohesively bound together within their respective category reducing the probability of other competing categorical candidates (i.e. reducing uncertainty). The availability of distinctive information increases as conceptual representations move forwards towards L1. At this layer, richer representations minimise the likelihood of having possible candidates outside the concept's general category. This leads to a concurrent decrease in the effect of *cLength* on categorical uncertainty since L1 does not rely exclusively on shared features to extract relevant information. As such, there are few high-probability candidates within the category of a particular concept. This result comes into contrast with basic-level uncertainty where *cLength* had a significant effect. The differential dependence on *cLength* highlights the different ways in which information is represented within each layer. Basic-level is the highest level of specificity and so any number of highly-similar candidates would increase uncertainty. These candidates would be concepts which have a very high degree of feature overlap with the target concept. Dense semantic neighbourhoods such as these lead to high values of *cLength* which ultimately lead to an increase in basic-level uncertainty.

There was no significant effect of *cLength* on domain-level uncertainty. As mentioned earlier, domain-level categorisation groups concepts with very little feature overlap into the same domain. This is especially the case for the nonliving domain. When the same correlational analysis was run within each domain separately there was indeed a significant effect for living things across all layers (see **Figure 5.26** below). Nonliving things on the contrary had a positive correlation within layers L2 and L1.

Figure 0.26: Correlation values between domain-level uncertainty and cLength

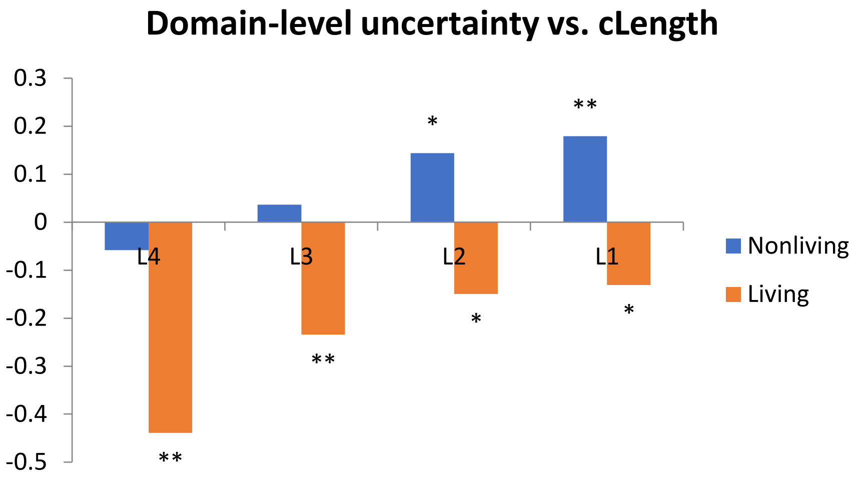


Figure 5.26: Correlation values between domain-level uncertainty for both nonliving and living things and their corresponding cLength values. */** denote p < 0.05 / p < 0.01 respectively.

This is because living things share a number of features across different categories (e.g. ‘has_legs’) which in turn differentiate the entire domain as a whole from nonliving things. This is the main reason why *cLength* would have an effect at L4 for living things only since nonliving things can be highly dissimilar from each other leading to ambiguity. Shared features in the case of nonliving things would only be confined within the particular category of the concept. For example members of the ‘*utensils*’ category share functional features which are unique to them; high-*cLength* concepts within this category would be drawn further apart from neighbouring categories. This particular aspect of nonliving categories means *cLength* does not contribute in reducing uncertainty in the same way in does for living things. As the representational space becomes more sparse and diversified so does the effect of *cLength* increase for nonliving things. This differential effect of cLength on uncertainty did not affect accuracy scores however since all layers scored above 95% (see **Figure 5.13**). A significant difference between the two domains in terms of accuracy was only found at L4.

5.4.5 Conclusions

The aim of the present study was to try and uncover a plausible computational mechanism by which semantic representations might arise in the brain. The findings of the model gave a number of insights into how conceptual processing might take place in the brain and more specifically how they might relate to the CSA. The model described in this chapter was meant to be both neurophysiologically and computationally plausible. The layers resulting from the model's training are stages of processing each with its own representational capacities. Representations at the earliest stages (e.g. L4) can be thought to reflect the types of representations found at the most posterior edges of the semantic processing stream namely posterior fusiform / posterior temporal regions (see **Chapter 1** for review). The end-point of this process (L1) and the representations associated with it can be thought to reflect the type found within the anteromedial temporal portions of the stream. Most importantly, I have shown that CSA-derived measures directly relate to the performance of the model at different layers and that this relationship depends on the representational characteristics of each layer. Finally, this model also introduced a further measure, *uncertainty*, which in this chapter I quantifiably defined as the entropy of the model's responses and related it to feature-statistics. Within the context of processing the aim of layer-to-layer transformations, as information flows from L4 to L1, is to *resolve uncertainty* regarding the precise identity of a particular concept. This principle can also be tested against human behaviour. As such, uncertainty forms an important cornerstone of the model which in the following chapter will be used to directly relate against both behavioural and imaging data. There are however certain limitations in the model which I will address separately.

1. I interpret the top-down pass in this model to be a simulation of how semantic processing takes place within the ventral stream. However, the bottom-up, training process might involve regions outside the ventral stream (including those found in Chapter 3) and it is not something explicitly addressed in this study.

2. The model is not interfaced with the visual system. In other words there is no connection between perceptual information, as received from V1 onwards, with conceptual information. It would have been possible to use an arbitrary mapping between visual information and the analogous semantic representation. For example, a back-propagation algorithm could have been trained to map a specific pattern of C2 responses (theorised to be high-level perceptual representations) unto a specific pattern of activation within L4. However, this process would not have been in any way illuminating as to what actually takes place in the brain. As it stands now, the model can be interpreted as a computational instantiation of processes that take place along the posterior-anterior axis of the ventral stream *beyond* the posterior fusiform after which basic visual processing has already taken place.

3. As mentioned earlier there was also an inherent limitation with respect to the number of layers that proved to be best for training the model. Consequently, relating any of the layers to an actual portion of the neuroanatomy of the ventral stream was not possible. Furthermore, units within layers were independent yet there is no reason to believe why this should be the case in the brain. Spike-dependent plasticity and Hebbian learning are examples of how individual neurons can interact with each other and influence the ways in which they respond to incoming stimuli. However, this does not necessitate a fundamental

change in the computational mechanism inherent in the model and DBN in general. Future improvements could include lateral inhibition between units thus introducing a form of dependency and making the model more relatable to neurophysiology.

Generating layers of conceptual representations: testing predictions against behavioural and imaging data

6.1 Introduction

As described in the previous chapter, computational modelling is a formal instantiation of a specific feature within a theoretical account. To this end, I constructed a DBN in an effort to capture how the hierarchical architecture of semantic representations might arise in the brain. Model analysis revealed that the flow of semantic processing from L4 to L1 follows a trend from a highly-organised categorical structure towards more diversified and differentiated representation of individual concepts. The feature-statistics of individual concepts, namely *cLength*, also had a direct impact on how well the model performed (namely accuracy and uncertainty). This effect was dependent on both the specificity level as well as the model layer. For example, basic-level identification is hindered when a concept has high *cLength*; conversely low-*cLength* concepts fare better in terms of performance (see **Chapter 5**; **Results** section).

This gradient of increasing information richness from L4 to L1 is analogous to how the ventral stream is thought to process semantic information. Specifically, Tyler et al (2004), Tyler et al., (2013) and Clarke and Tyler (2014) revealed that there are distinct representational capacities at different portions of the ventral stream – posterior regions such as the posterior fusiform are restricted to category-relevant information while specialised regions such as the perirhinal cortex represent information at a much finer scale. If the model's computational processes reflect the way in which semantic information is processed in the brain, then measures derived from the model (including uncertainty) should exhibit a relationship with both behaviour and neural representations along the ventral stream.

Figure 6.1 depicts this scenario whereby L4 of the model roughly maps unto the more posterior aspects of the ventral stream/earliest stages of semantic processing after which the flow of information reaches the fine-grained detail captured by L1 which maps onto the more anterior parts of the ventral stream. The perirhinal cortex, in this case, possesses the neural capacity to represent the most complete conceptualisation of an object at a level of detail which is closest to the semantic knowledge encapsulated within the McRae feature norm dataset (as seen in Clarke and Tyler, 2014). Within this context, high-level representations which abstract away from individuated information, such as general category and domain-level identifications, are accessed at an *earlier* stage compared to basic-level (Tyler et al., 2004).

Figure 0.1: Flow of processing and top-down pass of model

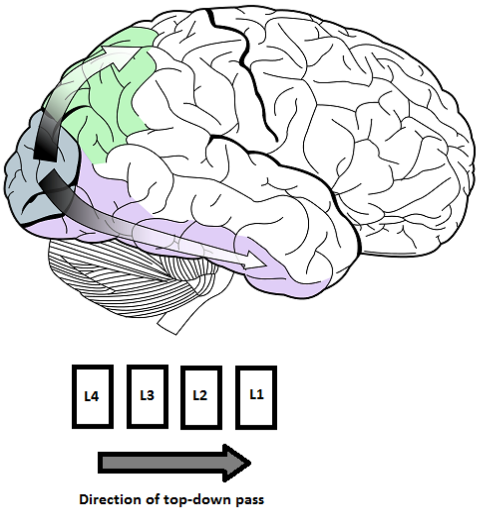


Figure 6.1: The flow of processing within the ventral stream (bottom stripe in light purple) juxtaposed against the top-down pass of the model. Processing in the brain is modelled as a process of *re-constructing* the most detailed representation about a concept as information passes through different stages along the ventral stream. (Brain image courtesy of Lokil Profil, permission for use granted under the GNU Free Documentation Licence)

As described in the preceding chapter, the model's architecture is explicitly hierarchical in nature with respect to representational capacity: L4, which is the first layer during the top-down stream, possesses the least degree of information richness while L1, the end-point of processing, the highest. This is mainly reflected in the uncertainty of concepts at each layer. Generally, there is high uncertainty at L4 which gradually decreases towards L1. In this chapter, I take the view – implicit in the model's architecture – that the modus operandi of the ventral stream is to decrease uncertainty regarding the identity of an object by going through successive stages of processing each increasingly capable of representing information at a finer-grained level of detail. This view is directly related to the work conducted by Karl Friston in Friston (2003; 2005; 2006a; 2006b). The position broadly taken by the authors in these studies is that neuronal responses probabilistically encode the range of possible causes which give rise to a particular sensory input. Uncertainty, similar to how it is quantified in the previous chapter, is based on the probability distribution of these causes (e.g. possible line orientations). These probability distributions are updated continually - meaning improvements in uncertainty - as representations go through increasing levels of complexity during information processing (Lee and Mumford, 2003). Although these studies are conducted within the framework of low-level visual processing, it has been suggested as an overarching principle underlying information processing in general. Thus, it can be argued that the same principle applies to high-level, conceptual representations. In my case, each layer encodes a particular probability distribution in response to a particular concept. These distributions give rise to specific values of uncertainty which gradually decrease across the top-down pass of the model. This gradient of high to low uncertainty provides a crucial testing ground for the model's basic function which is to resolve uncertainty of a concept's identity at a particular level of specificity.

In this chapter I aim to determine whether the model, and the view implicit in its architecture, had any valid relevance with respect to experimental findings. If there is a relationship then there is reason to expect that similar mechanisms might underlie both the model and semantic processing in the brain. Specifically, I compared model performance against two distinct datasets. The first dataset was derived from a behavioural study, Taylor et al. (2012), and the second from an imaging study, Tyler and Clarke (2013). I will address the manner in which I tested the model for each dataset separately.

6.1.1 Testing against behavioural data

The behavioural study was conducted to find out whether feature-statistics have an effect on how conceptual representations are processed in the brain under different task-demands. It comprised of two experiments: a basic-level naming task and a domain decision task. For the first experiment participants had to verbally identify a set of 412 images at the basic-level of specificity, while for the second a different group of participants had to press a button which denoted whether the depicted concept was a living or nonliving thing. They showed that concepts with a higher proportion of shared features elicited faster responses (reaction times; **RTs**) for domain decisions, while responses for concepts with relatively more distinctive features were faster during basic-level naming. In other words, they showed that *sharedness* (quantified in the study as the number of concepts in which a feature occurred) had a facilitative effect on domain-level identification while *distinctiveness* (the inverse of sharedness described above) had a facilitative effect on basic-level identification. Furthermore, a second measure, which quantified the interaction between sharedness and correlational strength, had a strong effect on RTs but only for basic-level naming. The authors argued that these differential effects of *sharedness* and the second measure on task performance show that feature-based statistics are highly important during conceptual processing. Most importantly it offered support to the CSA which states that concepts with highly shared information will require further processing (in this case captured by slower RTs) for specific identification. Less specific identification (e.g. living vs. non-living?), on the other hand, would be facilitated by a concept's sharedness (i.e. faster RTs) since shared features are highly informative of category/domain membership.

In this study, I tested whether the time taken for a participant to make an identification (either at basic or domain) was related to model performance. Processing at each layer in the model involves the prediction of the most likely concept given a particular response. The confidence of the model in its prediction at each layer would determine the extent of processing that needs to be undertaken. High uncertainty would necessitate further processing until a sufficiently refined probability distribution yields a confident prediction. In this case, I reasoned that the time taken to produce a response (behavioural RTs) in humans would be directly related to concept uncertainty derived from the model. This is because, according to the view outlined above, the overarching purpose of both the model and the brain's object processing system is to resolve uncertainty: a concept with high uncertainty will require further processing – or more time in the participant's case - to disambiguate it from semantic neighbours and thus maximally increase confidence in its prediction. To test this claim I compared human reaction times against two performance measures from the model: *earliest layer response (ELR)* and *uncertainty*. My expectation was that if a particular aspect of human behaviour can be reproduced through the model's responses then it would strongly suggest that uncertainty, and its resolution over the course of processing, play a decisive role in object recognition.

ELR was defined as the *earliest* layer (with L4 being the absolute earliest and L1 the latest) at which the model made a correct response (for basic-level and domain-level) at >90% confidence. Confidence was defined as uncertainty normalised by stimulus entropy at a particular level of specificity (see **Chapter 5**, **Methods section**). ELRs were used as a proxy for the model's "reaction time". The model had four layers each effectively a stage of processing. Low ELRs mean that the model is able to make highly confident, correct responses with information available within the initial stages of the top-down pass (i.e. either L4 or L3). High ELRs mean that the model requires more fine-grained information only available within the later layers, or even a full reconstruction of the McRae feature norm vector. I then tried to determine whether the model ELRs were able to capture a signification portion of the variance within the actual reaction times of human participants. When identifying objects at the domain level the model would only recruit information at the earliest layers along its top-down pass. Conversely, when identifying objects at the basic-level the model would recruit information at further layers towards L1. Overall, concept-specific ELRs should correlate with concept-specific RTs.

Uncertainty is a measure of central importance within the computational mechanisms of the model. It was shown (see **Chapter 5**; **Section 5.3.4**) that uncertainty across all levels of specificity decreases from L4 to L1. It was also shown that uncertainty at specific layers is dependent on various aspects of a concept's semantic neighbourhood (***cLength[m]***). Specifically, uncertainty within L4 and L3 is mostly driven by categorical neighbours of the concept. If a concept is cohesively embedded into its category by means of a high number of shared features it will generally have a comparatively high (basic-level) uncertainty. At layers L2 and L1, as information becomes more refined and detailed, a concept becomes detached from its category and only the most similar concepts remain within its immediate neighbourhood.

Uncertainty in this case is driven by only the closest, immediate semantic neighbours of a concept.

I argued that these different types of layer-specific uncertainty would have a differential effect on reaction times. In other words, reaction times in this case are effectively a reflection of the types of uncertainty that need to be resolved. When participants are making a domain-level decision their reaction times would largely correlate with domain uncertainty at L4. This is because L4 stands as the earliest stage of processing which corresponds to the posterior entry-point of the ventral stream that is known to relate to categorical and domain level information (Tyler et al., 2004, Clarke & Tyler, 2014). At this layer, information is coarse-grained but highly ordered categorically. Domain-level uncertainty at L4 signifies how well the system can ascertain whether a concept is living or nonliving. The extent of processing will depend on the degree of uncertainty at this early stage. Low uncertainty will result in short reaction times while high uncertainty will necessitate further processing and so longer reaction times. However uncertainty at L1, which signifies the end-point of processing within the model, will have no relationship to behavioural performance for domain decisions. This is because L1-uncertainty, as mentioned earlier, is highly dependent on the closest, immediate semantic neighbours of a concept. It is not necessary for the system to disambiguate a concept at this level of granularity.

I also tested whether there was any relationship between basic-level naming reaction times and the model’s uncertainty when making basic-level identifications. Within the context of the model, basic-level naming necessitates resolving uncertainty at a very fine-grained level of specificity. I reasoned that to make a basic-level distinction the semantic system must be able to disambiguate a concept from other competitors which are very similar to it semantically – considerably more similar compared to other members of its category. This means that uncertainty at this level will not be captured in L4, which only has the capacity for general, abstract representations, but rather in L1 – the end-point of processing in the model. I argued it would take more time for participants to process and identify an object with high L1-uncertanity as derived from the model. L4-uncertainty, in this respect, has no relevance since L4 is at a stage of processing which has a limited representational capacity. The demands of the task necessitate further processing beyond the stage encapsulated by L4.

In summation, I carried out two different tests against behavioural data:

1. Extraction of **ELRs** at two different levels of specificity (basic-level and domain) and comparison against the corresponding behavioural reaction times.
2. Correlation between **uncertainty** at **domain-level** (for layers L4 and L1) and the corresponding reaction times in the second task of the study. Similarly, a correlation between **uncertainty** at **basic-level** (for layers L4 and L1) and the corresponding reaction times in the first task of the study.

6.1.2 Testing against imaging data

Each stage of processing, within the model, signifies a change in the representational arrangement of concepts. As information flows, from L4 to L1, concepts are re-arranged within the semantic space reflecting the differential weighting on specific features according to their statistical properties (e.g. sharedness). This in turn leads to a decrease in categorical cohesion as the overall semantic distance between category members increases and allows for within category individuation. Representations follow a set of computational transformations which differentiate a concept from its neighbours. If a concept is rigidly embedded within its category (i.e. there is high overall similarity between itself and all other category members) it will readily dissociate from its category over the course of processing in order to facilitate identification. By dissociation I mean that similarity with other category members will decrease. Likewise, if a concept has a low degree of overall similarity with its category members, dissociation will be *less pronounced* comparatively since the semantic space surrounding the object is already sparse and differentiated. Highly similar semantic neighbours will take longer to dissociate in the model. The nature of representations in the *earliest* stages of processing, in this case, largely determines the degree of dissociation the concept will undergo over the course of being run through the successive layers. As shown in the previous chapter (**Section 5.3**), high uncertainty at L4 (which in this case signifies the earliest processing stage) arises from concepts which belong to categories that are tightly-bound semantically, meaning they have a large number of semantic neighbours. This in turn leads to a large number of competitors and thus high uncertainty. For example, concepts with high uncertainty such as ‘dog’ (because this particular concept belongs to a relatively large, semantically tight-bound category) would undergo a more pronounced dissociation (along the ventral stream) in an effort from the semantic system to resolve uncertainty and individuate the concept away from its neighbours. On the contrary there would be less pressure to disambiguate concepts with low uncertainty. For these concepts, which don’t belong in a cohesive category, their position within the representational space will not change as dramatically.

The Clarke and Tyler (2014) study addressed how representations are organised during basic-level naming. Given its rich set of stimuli (n = 131) taken from 6 semantic categories it provided a suitable platform for testing. Using this dataset, I tested whether uncertainty had any relationship with regards to how individual concept representations would dissociate from their category along a posterior to anterior axis in the ventral stream. This representational gradient (abstract, categorical to fine-grained, specific information) from the posterior towards the more anterior parts of the ventral stream (including the perirhinal) reflects how information is transformed across the layers of the model. With respect to object processing, this means that as information flows from these posterior regions along the ventral stream, categorical cohesion is reduced but fine-grained granularity increases (Clarke and Tyler, 2013).

I argued that the trend of dissociation (i.e. how readily the overall similarity between a concept and other category members decreases) would be directly dependent on the system's uncertainty at the initial stages of processing (L4 in the model). L4, in this case, corresponds to the posterior entry-point of the ventral stream including the posterior fusiform which according to Clarke and Tyler (2014) contains information which is highly organised in cohesive categories. As information flows through the ventral stream the effect of L4-uncertainty (which is influenced by categorical similarity) would decrease since representations become more refined and differentiated. I therefore expected that at this processing stage, hypothesised to be instantiated within the more anterior regions of the stream, L1-uncertainty would have the strongest effect on the degree of dissociation. The rationale behind this expectation was that the structure of categories at later stages of processing (**L1** in the model; **anteromedial regions** in the brain) is already sufficiently differentiated to minimise ambiguity between category members. At this point, it is only the closest, immediate semantic neighbours that will have an effect on the system's uncertainty in identifying a concept. This means that for a particular concept, the degree of how readily the overall category similarity will decrease depends on its similarity with its immediate categorical neighbours – high similarity leads to high uncertainty which in turns leads to a more pronounced trend of dissociation; low similarity leads to low uncertainty which in turn leads to a more ‘flat’ or stable trend.

In summation, there were two key predictions regarding the relationship between the model and imaging data: 1) **categorical cohesion** of semantic representations should decrease along the posterior to anterior axis in a similar fashion to the model, 2) the **trend of dissociation** of concepts along the ventral stream should be directly correlated with specific types of uncertainty at different processing stages.

6.2. Methods

6.2.1 Behavioural study

Behavioural data

Inverse-transformed reaction times were collected from each participant from both tasks performed during the Taylor et al. study. There were a total of 412 stimuli for basic-level and 475 stimuli for domain-level. For basic-level, stimuli which had below 70% identification accuracy across all participants were automatically removed from the entire dataset before any further analysis. Stimuli consisted of coloured pictures, centred against a white background. All stimuli were derived from concepts found in the McRae norm database (see Taylor et al., 2012 for full details).

Participant responses were analysed separately. For each participant, stimuli with incorrect responses, stuttering (for basic-level task), or reaction times faster than 300msec or longer than 4s were removed and analysis was performed on the remaining data (average number of stimuli across participants: Basic-level = 220.5 / Domain = 410.5).

Extracting Earliest Layer Responses (ELRs)

I ran the full model a total of 25 times for both basic-level and domain-level identifications. It was noted in Chapter 5 that concepts belonging to the 'miscellanea' category which contained un-categorised objects were removed from the analysis. However, I had to include the category in the present study because a high number of these concepts were included in the domain-level task in Taylor et al (2012). Including this category meant that the total number of concepts during the model run totalled to 517 (vs. 425 in the previous chapter). For each concept, the run was stopped if a layer fulfilled the following two conditions: 1) correct identification and 2) confidence level above 90%. The number of the layer, at which these conditions were satisfied, was then recorded. The chosen confidence threshold means that there is a 90% reduction in the overall uncertainty of the model layer given the response elicited by the concept. Lower thresholds, tending towards 0%, would mean that if a layer produces a correct response it becomes increasingly likely that it was made at random. Responses at lower confidence thresholds would be highly inconsistent in this respect. I chose a confidence threshold of 90% to ensure that responses would be consistent across runs. Layer numbers were recorded according to their order in the sequence of processing (L4 = 1; L3 = 2; L2 = 3; L1 = 4; full re-construction = 5). The ELR for a particular concept was then computed as the **median layer number** across runs. There were two sets of ELRs (each comprising of 517 concepts), one for basic-level and one for domain-level. I then extracted the values for concepts which were found in either task.

Uncertainty measures

Uncertainty measures were already computed in Chapter 5 so I derived a value for each concept at layers L4 and L1 for each corresponding specificity level. This resulted in a total of four variable sets (2 x 412 uncertainty values for the basic-level task; 2 x 475 for the domain-level task).

Mixed effects model

To assess the relationship between model-derived measures and behavioural reaction times I performed a linear mixed effects model across all participants using the same procedure as in the Taylor et al study within the R programming environment (R Development Core Team, 2008). Behavioural responses (inverse-transformed reaction times) were the dependent variable for all subjects. Independent variables, for each subject, comprised of either ELRs or uncertainty (i.e. the variable of interest) plus a further 8 control variables which were computed and used in the original study, namely *phonological length, visual complexity, naming agreement, lemma frequency, familiarity, cohort size* and *H-statistic*. This was done to remove any visual, phonological or lexical effects from the responses. Mixed effects models incorporate both fixed (i.e. individual variables) and random effects (i.e. subjects) within the same statistical framework. The method takes into account individual fluctuations across the entire group as well as variation in the overall mean of individual predictors (Baayen, 2008).

6.2.2 Imaging study

Imaging data

Imaging data was acquired from an fMRI study (Clarke and Tyler, 2014) where 16 participants had to perform an overt basic-level naming task with 131 objects that were one of six different superordinate categories (‘animals’,). Each object was a coloured image presented on a white background that depicted concepts from the McRae norms dataset.

The fMRI preprocessing consisted of slice-time correction and spatial realignment only, and all images were un-smoothed and un-normalised. SPM8 was used to generate statistical maps for each individual concept, combining data over the six repetitions. This resulted in a dataset comprising of 131 individual statistical brain maps (one per concept) for each participant.

Using the RSA toolbox (Nili et al., 2014), I extracted a set of spherical searchlights (Kriegeskorte et al, 2006) (radius = 8mm) over the entire brain surface for each participant. Within each searchlight-neighbourhood, I computed the Pearson’s correlational distance between voxel patterns for all pairs of concepts, resulting in a 131x131 RDM. This resulted in an individual RDM-map for each participant. I weight-averaged the RDM-maps across participants which I then projected on to the compromise space (DISTATIS; Abdi et al, 2009; Chapter 4). All analyses were performed on the single compromise-RDM map. For the purposes of this study, I only required the voxels which contained a significant degree of semantic information. To select these voxels I correlated each searchlight-RDM (within the compromise-RDM map) with a semantic RDM derived from the cosine distances between McRae feature norm vectors of the 131 concepts. I also computed a visual RDM as the correlational distance between C1 responses for each stimulus-image. C1 responses were used as a model of early visual processing (in V1/V2; Serres et al., 2007) and were extracted in the same manner as in previous chapters. I then regressed out any variance accounted by the visual RDM from the semantic RDM before performing the correlational analysis with the compromise-RDMs. This was done to remove any confounds in the metric relationships between concepts which could be attributed to low-level visual properties. I used the Spearman's correlation for second-order comparisons between semantic and compromise-RDMs. The resulting correlation map was masked using a ventral stream mask (as used and constructed in Acres, 2008) and only showed significant effects relating to the semantic RDM ($p < 0.01$; FDR-corrected). The mask contained the following areas: **calcarine sulci**, **lingual**, **fusiform**, **inferior occipital** and **inferior temporal gyrii** (including the **perirhinal cortices**) as well as the **temporal poles**. I used the indices of the remaining voxels (see **Figure 6.2**) to create an index-map which indicated which voxels were to be used in further analyses.

Figure 0.2: Searchlight voxel neighbourhoods with significant semantic information

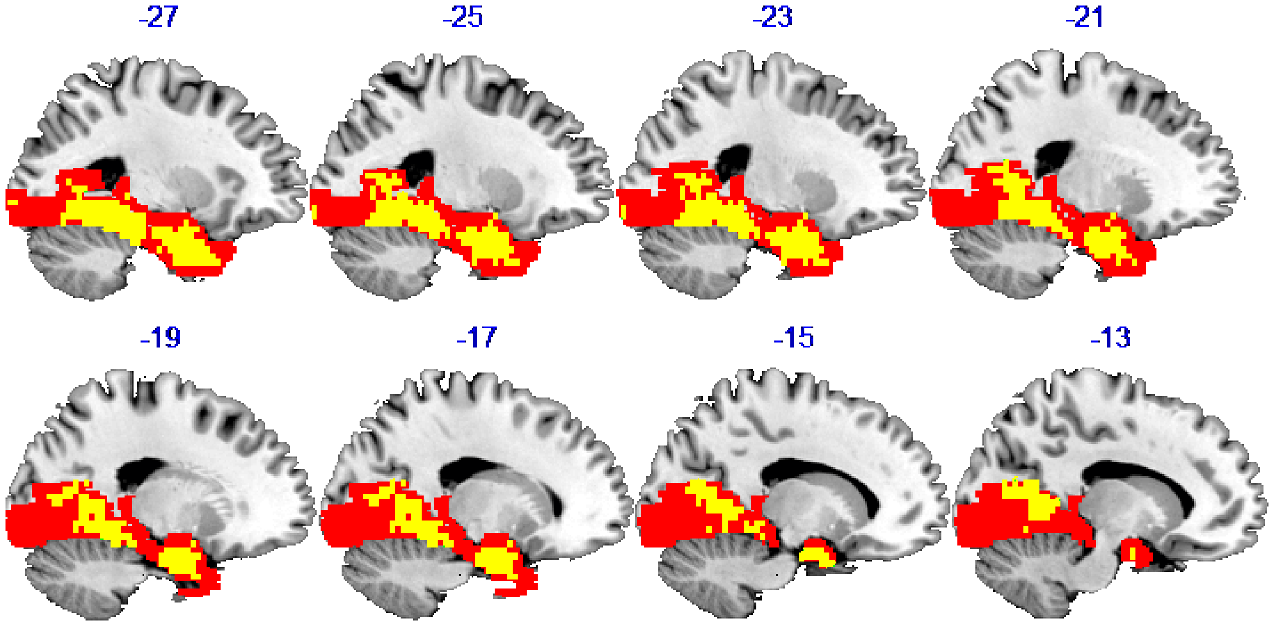


Figure 6.2: Searchlight voxel neighbourhoods (in yellow) with significant semantic information ($p < 0.01$; FDR) using a ventral stream mask (in red) and controlled for C1 information. The y-co-ordinates of the sagittal slices are numbered on top in blue.

The normalised MNI template space was split into a set of 63 coronal slices covering the entire posterior-anterior axis of the ventral stream (posterior start co-ordinate: **y = -104**; anterior end co-ordinate: **y = 27**). I then used the index-map (**Figure 6.2**) to collect all semantically relevant voxels within each coronal slice. At each voxel, I computed each of the concept's overall category similarity. This was computed as the norm of the distances between a concept and its category members (I chose the norm instead of the mean because I did not want to exclude the effect of category size). The objective was to find the slope of the straight-line which best described how overall category similarities changed across a particular slice range. This was done by running the following regression equation *for each individual concept* at a particular slice range:

Equation 0.1: Regression equation for trend of dissociation

$$Y^s = \beta X^s + \varepsilon$$

(Eqn. 6.1)

Y is a vector containing the entire set of category similarities for a particular concept across a particular slice range ***s***, slice-specific values are stacked together in a single column; ***X*** is a vector containing the corresponding slice numbers; ***β*** denotes the slope of the best-fit straight line. The trend of dissociation (***ToD***) was defined as the value, ***β***, multiplied by the **R-square** statistic (the latter term was added to penalise poor-fit lines). The reasoning behind the latter operation was that it is possible to have a high value for a regression ***β***-coefficient even though it does not capture a sufficient portion of the variance (Cohen et al, 2003). Such cases need to be taken into consideration and this is why I decided to standardise coefficients with the goodness-of-fit (R-square statistic) for each concept. For more than one independent variable dividing coefficients by their standard error would be a standard, and more appropriate, practise (Harrell and Frank, 2001) – however this was not the case in the present study (there was only one independent variable: X which contained the slice numbers). **Figure 6.3** depicts one example concept, ‘dog’, to better illustrate how ToD was computed.

Figure 0.3: a) coronal slices from which values were extracted b) sample trend line to indicate derivation of trend of dissociation

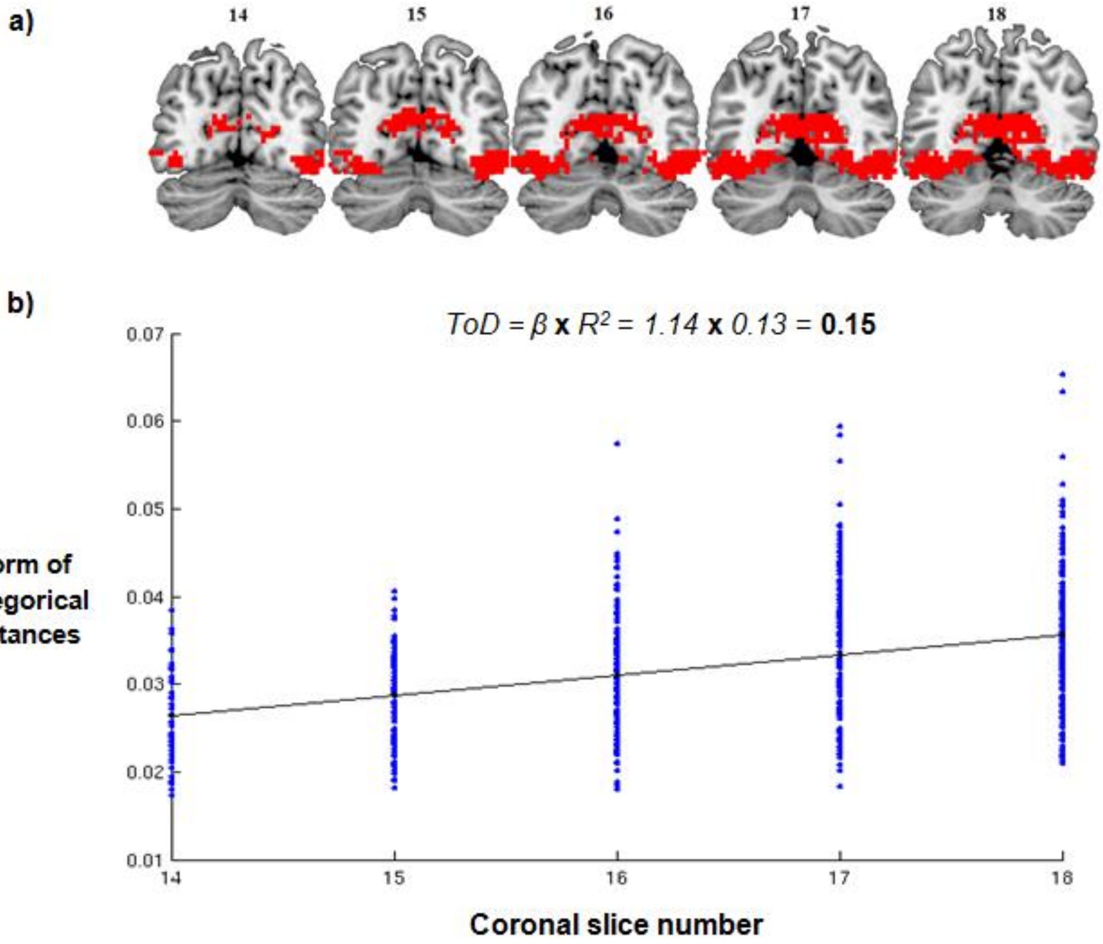


Figure 6.3: a) Coronal slices from which values were extracted. b) Dots in blue denote individual voxels while the trend line running through the values indicates the best-fit straight line as computed through the regression equation (Eqn. **6.1**).

In the above example ***Y*** contains the categorical distances from each slice stacked unto one column vector. ***X*** is also a column vector (the same size as ***Y***) containing the stacked slice indices corresponding to the categorical distances in ***Y***. The fitted ***β*** – coefficient, which signifies the slope of the line, is then multiplied by the R-square statistic to obtain the trend of dissociation value. This resulted in a set of 131 ToD values (one per concept) for a particular slice range. This set was then correlated against both L4- and L1-uncertainty resulting in a single correlational value for each slice range.

The selection criterion for including a particular slice was a minimum of 30 semantically sensitive voxels. The voxel number criterion was chosen to remove slices with particularly small semantic effects. Under this criterion only slices numbered between **14** and **42** were included in the analysis (most posterior slice MNI co-ordinate: **y = -73**; most anterior slice MNI co-ordinate: **y = 12**). Furthermore, no slice range could be shorter than 5 coronal slices. The slice range criterion was chosen to provide a sufficient number of ordinal values in the independent variable over which to determine the linear slope. Computing the slope over just two or three slices, for example, could result in spurious results – I needed a sufficient number of slices to produce a reliable result. The regression analysis was carried out across all possible slice ranges which fulfilled the aforementioned criteria.

Categorical cohesion

Categorical cohesion (Davies-Bouldin index; see **Appendix IV**) was calculated for each of the voxels within each of the 29 slices described above (i.e. those numbered between 14 and 42). I then calculated the **average index value across all the voxels** within each slice. I reasoned that the slice-average cohesion indices would *increase* (high DB-indices signify low cohesion) along the posterior-anterior axis of the stream. This would be, first and foremost, a reflection of the trend observed in the model in the previous chapter, and secondly a confirmation of the findings found in Clarke and Tyler

(2014) which showed that semantic information becomes more distinctive for specific concepts and less categorically cohesive.

Uncertainty measures

I collected two sets of uncertainty values for basic-level identification, at layers L4 and L1 from the analysis conducted in the previous chapter. As mentioned before, I used the resulting values to correlate against the corresponding ToD measures of a particular slice range. This ultimately resulted in a single correlational value for each slice range.

6.3 Results

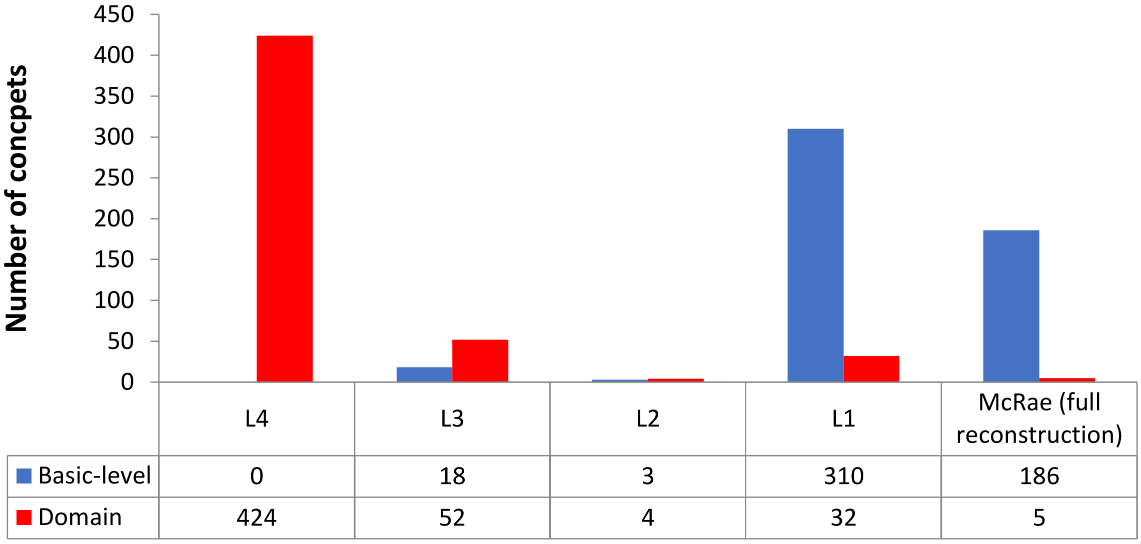
6.3.1 Behavioural data

The analysis was split in two parts. First, I investigated whether the speed at which participants made a response, for either basic-level or domain, was related to the earliest layer at which a model made an accurate and confident response. For the second analysis, I assessed the relationship between uncertainty, derived from two different layers (L4 and L1), and the reaction times. For both analyses, I made use of a linear mixed effects model.

Earliest Layer Responses (ELRs)

I extracted ELRs for both basic-level and domain-level for all 517 concepts in the McRae feature norm dataset. First, I plotted the distribution of responses for each specificity level (see **Figure 6.4**). I then used a χ^2-test of independence to determine whether there were any significant differences between the two response distributions. I found that there was indeed a strongly significant difference ($\chi^2(4) = 838.16$; $p < 0.01$): overall the model was “faster” in performing domain-level identifications since it was able to make accurate and highly-confident responses at L4. Only 1% of concepts required a full reconstruction of the McRae feature vectors meaning for the vast majority of concepts (82%) the information contained in L4 was sufficient to make a domain decision. For basic-level, 96% of concepts required the fine-grained information contained at the very latest stages of processing with 36% reaching full reconstruction.

Figure 0.4: ELR distribution for both basic-level and domain



	L4	L3	L2	L1	McRae (full reconstruction)
🟦 Basic-level	0	18	3	310	186
🟥 Domain	424	52	4	32	5

Figure 6.4: Histogram of ELR distributions for both basic-level and domain.

Model ELRs vs. behaviour

There were two sets of ELRs – one for each specificity level – extracted over the whole 517-concept dataset. Given the skewed distribution of the ELRs (**Figure 6.4**) I decided to transform by squaring each value to make the data more suitable for statistical analysis. The results summarised in **Table 6.1** are based on the transformed ELRs.

Table 0.1: Mixed-effects model: ELRs vs. behaviour

	Basic-level				**Domain-level**			
	Estimate	Std. Error	t-value	p-value	Estimate	Std. Error	t-value	p-value
(Squared) ELRs	***-0.004***	***0.001***	***-2.977***	***0.003***	***-0.011***	***0.002***	***-5.114***	***0.000***
Additional control variables								
Phonology	0.004	0.007	0.593	0.553	-0.006	0.009	-0.699	0.485
Visual complexity	0.000	0.006	0.020	0.984	0.024	0.009	2.749	0.006
Naming agreement	0.115	0.020	5.595	0.000	-0.030	0.009	-3.371	0.001
Frequency	0.049	0.007	7.353	0.000	-0.029	0.009	-3.225	0.001
Familiarity	0.079	0.007	11.832	0.000	0.025	0.009	2.762	0.006
Cohort size	-0.017	0.007	-2.551	0.011	-0.019	0.009	-2.103	0.036
H-stat	-0.019	0.008	-2.340	0.020	-0.025	0.009	-2.774	0.006
NOF	0.028	0.007	4.192	0.000	0.033	0.009	3.729	0.000

Table 6.1: Results from the mixed effects model showing coefficients for all control variables and ELRs. P-values were estimated in the R programming platform using the Satterthwaite approximation (Satterthwaite, 1946).

Squared-ELRs for both basic-level and domain were found to be significantly correlated with behavioural reaction times (t = -2.98** / t = -5.11** respectively). The direction of the effect in both cases is negative because the dependent variables in all cases are the *inverse*-transformed reaction times (had it been the raw reaction times there would be a positive correlation). In follow-up analyses, the raw ELRs were also found to have a significant effect, t = **-2.6**** and **-4.9**,** for basic-level and domain respectively. Overall, these results show that concepts which engage later layers in the model also correspond to longer RTs for participants.

Uncertainty vs. behaviour

Table 0.2: Mixed effects model: Uncertainty vs. behaviour

	Basic-level				**Domain-level**			
	Estimate	Std. Error	t-value	p-value	Estimate	Std. Error	t-value	p-value
uncertainty.L4	***0.0221***	***0.0160***	***1.3810***	***0.1684***	***-0.0637***	***0.0113***	***-5.6360***	***0.0000***
uncertainty.L1	***-0.0611***	***0.0176***	***-3.4770***	***0.0006***	***-0.0034***	***0.0114***	***-0.2960***	***0.7677***
Additional control variables								
Phonology	0.0021	0.0065	0.3180	0.7507	-0.0049	0.0087	-0.5670	0.5709
Visual complexity	0.0008	0.0063	0.1270	0.8987	0.0234	0.0086	2.7300	0.0066
Naming agreement	0.1075	0.0205	5.2520	0.0000	-0.0288	0.0087	-3.3060	0.0010
Frequency	0.0458	0.0067	6.7880	0.0000	-0.0282	0.0087	-3.2320	0.0013
Familiarity	0.0790	0.0067	11.8650	0.0000	0.0277	0.0089	3.1230	0.0019
Cohort size	-0.0182	0.0067	-2.7350	0.0066	-0.0183	0.0087	-2.0960	0.0368
H-stat	-0.0223	0.0081	-2.7360	0.0065	-0.0252	0.0087	-2.8890	0.0041
NOF	0.0271	0.0066	4.0810	0.0001	0.0320	0.0087	3.6720	0.0003

Table 6.2: Results from the mixed effects model showing coefficients for all control variables and both uncertainty values for the respective layers. P-values were estimated in the R programming platform using the Satterthwaite approximation (Satterthwaite, 1946). Values highlighted in red indicate the type of uncertainty which was predicted to have an effect at that particular specificity level.

L1-uncertainty had a significant effect on basic-level responses, where greater uncertainty was associated with longer RTs (t = -3.48**) but no effect on domain-level responses (t = 1.38). In the same fashion, L4-uncertainty had a significant effect on domain-level responses (t = -5.64**) but no effect on basic-level responses (t = -0.3). Most importantly, the type of uncertainty was selectively correlated with the type of task taking place: L1-uncertainty with basic-level and L4-uncertainty with domain. As with ELRs (**Table 6.2**) the direction is negative meaning that there is a positive dependence between the actual non-transformed reaction time and uncertainty. These results show that both types of uncertainty can reliably predict the time taken for participants to respond.

6.3.2 Imaging data

The aim of the imaging analysis was firstly to establish that categorical cohesion within semantic representations decreased from posterior to anterior regions, as was the case within the model and Clarke and Tyler (2014). This is a fundamental property of how representations are believed to be organised along the ventral stream according to the CSA and the model itself provides a computational, and thus testable, instantiation of this property. **Figure 6.5** shows the Davies-Bouldin index increased consistently from the earliest slice at *1.754* (numbered **14** at *y = -104*) until it reached its maximum of *1.816* at slice number **36** (at *y = -7*). The overall trend was found to be significantly well-approximated by a linear fit (r-square = 0.87; p = 0.00) which means that, on average, *categories become consistently less cohesive* along the ventral stream in a linear fashion.

These results are in broad agreement with the predictions of the CSA and the specific behaviour of the model. Specifically, semantic representations are maximally cohesive within the most posterior regions of the ventral stream (including fusiform and inferior temporal gyrii) which equate to the earliest layers along the top-down pass of the model. Furthermore, they become minimally cohesive within the most anterior regions of the ventral (which include the perirhinal cortex) which equates to layer L1 of the model.

Figure 0.5: Categorical cohesion along the ventral stream

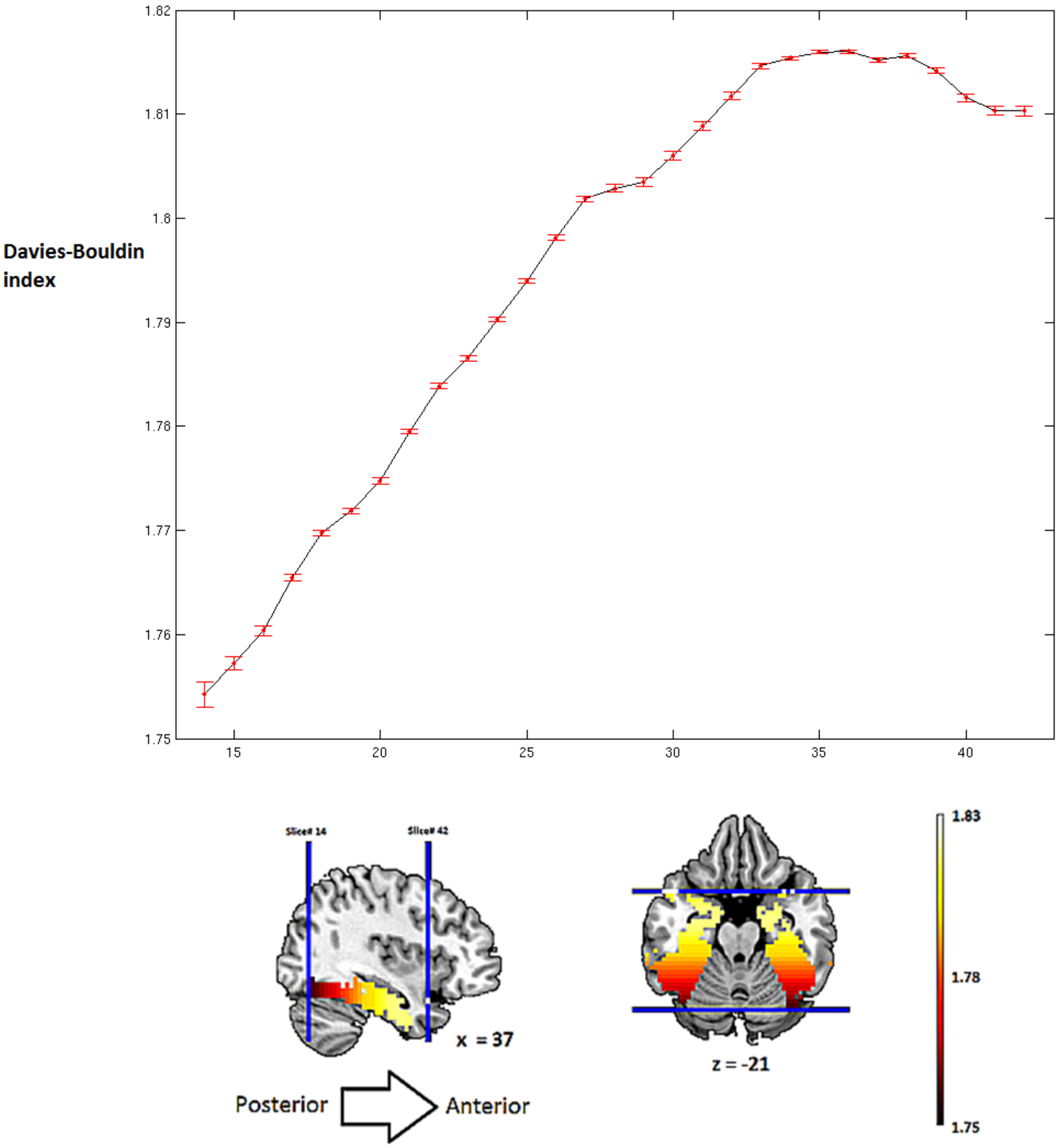


Figure 6.5: Trend of categorical cohesion along the ventral stream. Error bars at each slice-point are designated in red. Color bar indicates the values of Davies-Bouldin index. High indices are indicative of low categorical cohesion. The blue bars traversing the images designate the start- and end-points of the slice range (**14** and **42** respectively; see Methods).

The second aim of the analysis was to determine whether uncertainty correlated with how readily a concept dissociates from its category. This resulted in a single correlation value for each possible slice range (see **Figure 6.6**).

Figure 0.6: Color matrix indicating correlation values between trend of dissociation and uncertainty

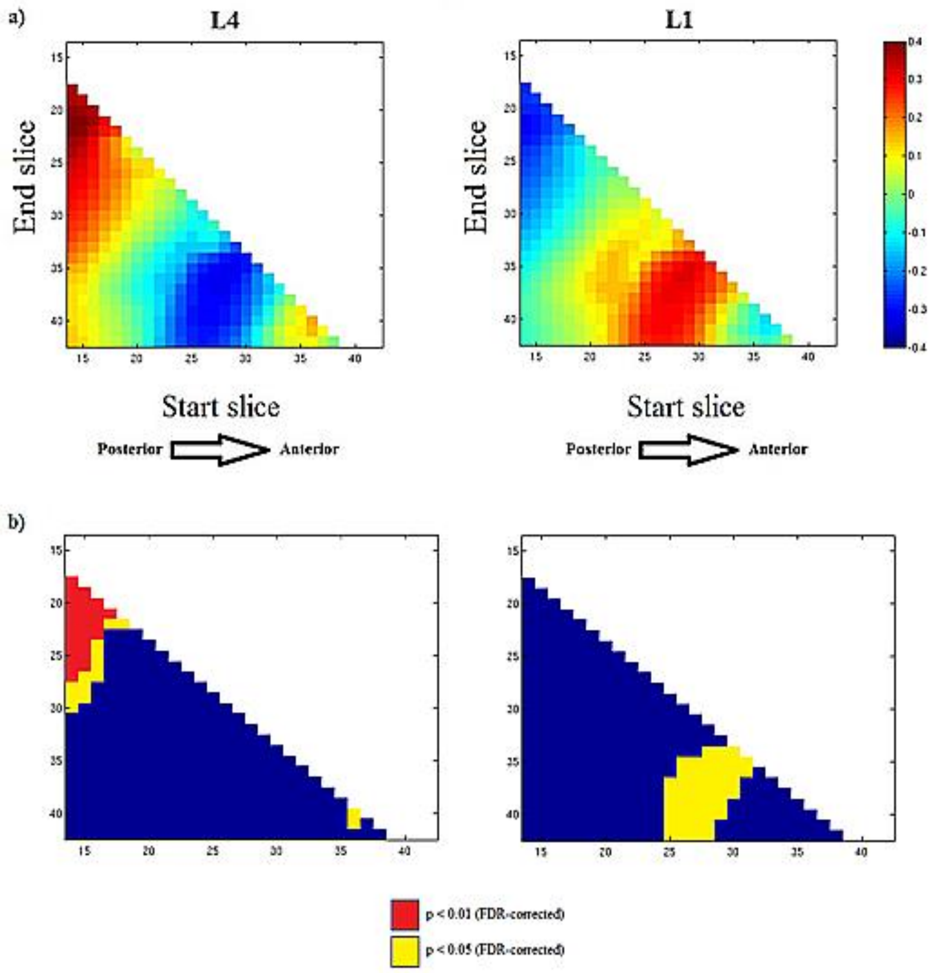


Figure 6.6: a) Color matrices depicting Pearson’s correlation values for each slice range. Vertical axis is the start slice of the range while the horizontal axis denotes the end slice. b) p-values for the corresponding correlations. Red indicates p < 0.01 (FDR); yellow indicates p < 0.05 (FDR).

The **strongest** correlation between L4-uncertainty and the ToD was found to be in the most posterior slices of the semantic processing system (slice range: 15 to 21 / MNI y-coordinates: -69 to -53; Pearson's r = 0.4, p < 0.05; **Figure 6.6**). This included the **posterior fusiform** and **inferior temporal gyrii**. There was also a mildly significant effect within the slice range 36 to 40 which mostly included the perirhinal cortex (r = 0.18; p < 0.05). The strongest effects for L1-uncertainty were within more anterior parts of the ventral stream (slice range: 29 to 35 / MNI y-coordinates: -29 to -11; r = 0.31, p < 0.05) including **anterior inferior temporal** and **perirhinal cortices**.

There were also strong negative correlations – these signify a situation where high uncertainty is predictive of low ToD values ('flatter' trends). This was the case for L4-uncertainty most notably within slices 27 to 35 (anterior half of ventral stream) as well as for L1-uncertainty within slices 15 to 20 (posterior half of ventral stream). These two types of uncertainty are sensitive to different aspects of the representational space (categorical vs. immediate semantic neighbourhoods) – this explains why negative correlations for one type of uncertainty correspond to positive correlations for the other within the same slice range.

In order to summarise my results, I collected all slice ranges which exhibited a significant effect for either L4-uncertainty or L1-uncertainty. From this set, I computed the median start and end slice for each of the two uncertainty type variables (see **Figure 6.7**). I found that as semantic information flowed along early portions of the ventral stream, specifically contained within coronal slices numbered 15 to 24, the trend of dissociation was highly correlated with L4-uncertainty. As information flowed further down the more anterior regions of the ventral stream, the effect of L4-uncertainty waned with a concurrent increase in the effect of L1-uncertainty (slice range: 27 to 38; **Figure 6.7a**). This slice range contained the more anteromedial portions of the ventral stream, including the perirhinal cortex (**Figure 6.7b**), believed to be involved in fine-grained processing of semantic information (Clarke and Tyler, 2014; Tyler et al, 2004, Kiviisaari et al., 2012; Bussey and Saksida, 2002). Both these findings conformed to the predictions made regarding the relationship between the *change of conceptual representations* along the ventral stream and *uncertainty*.

a) **L4**: Median start/end slices: *15 → 24 / y = -69 → -44*

L1: Median start/end slices: *27 → 38 / y = -33 → y = -2*

Figure 0.7: a) Median slice ranges b) Effects of uncertainty on categorical dissociation along the perirhinal

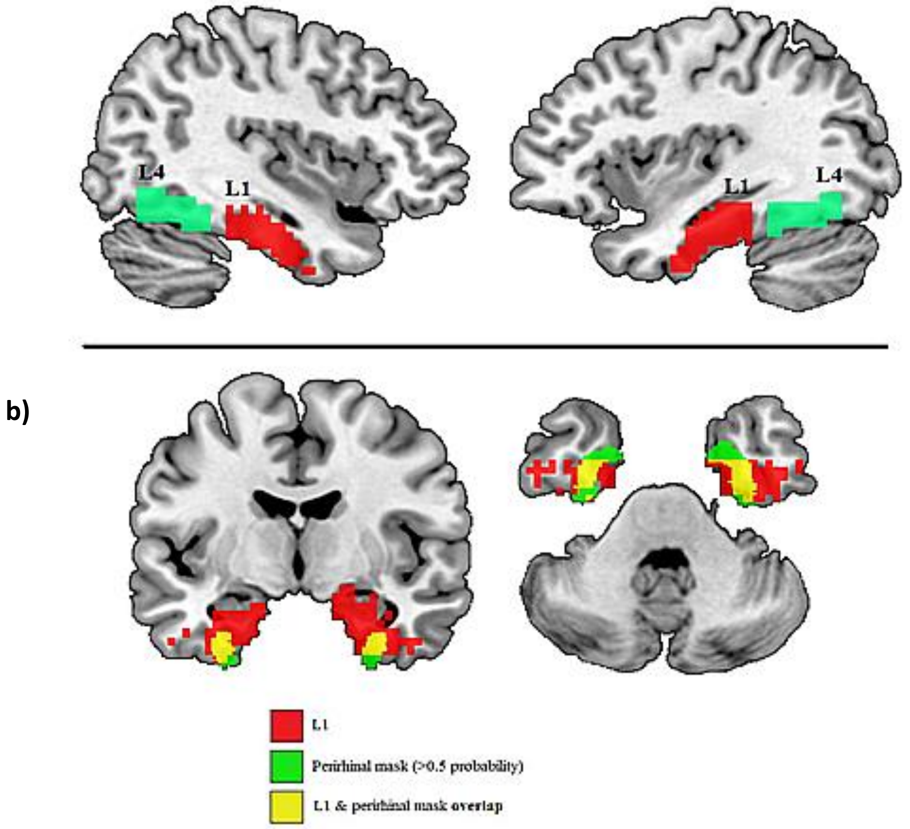

L4
L1
L1
L4
b)
L1
Perirhinal mask (>0.5 probability)
L1 & perirhinal mask overlap

Figure 6.7: **a)** Brain map indicating the indexed voxels (see **Figure 2.1**) which were found within the reported median slice ranges. Labels on images indicate the model layer from which uncertainty was extracted. **b)** Overlap between the voxels within slice range 27 to 38 and a perirhinal mask. This was done to highlight that the effect of L1-uncertainty on the trend of dissociation extended into the perirhinal cortex.

6.4 Discussion

The aim of the present study was to determine whether the model had any meaningful relevance to both behavioural and imaging data. This is an important step towards making the case that the model provides a useful insight into the basic computational mechanisms of object recognition. Overall, the DBN model could successfully capture both behavioural performance and the nature of how conceptual representations unfold along the ventral stream. I will address each analysis separately before discussing the importance of these findings with respect to the nature of conceptual representations.

6.4.1 Behavioural data analysis

Model ELRs and uncertainty vs. human behaviour

The aim of this analysis was to determine whether the principles instantiated in the model has any relevance to human behaviour. Model responses (ELRs) were found to be strongly associated with human responses within the two tasks. Furthermore, the uncertainty measures also had strong predictive power with respect to response times. This finding supports the view of the object processing system as a representational hierarchy where each level provides more refined and confident predictions regarding the identity of a stimulus. This view takes into account both the feature-statistics of a concept as playing a role during processing (Moss et al, 2007) as well as the representational capacities of each participating stage (Clarke and Tyler, 2014).

The distributions of model-derived ELRs were found to be significantly different between tasks. Specifically, the vast majority of domain-specific responses engaged only the first layer (L1) of the model during identification while basic-level responses required further processing with 36% reaching full re-construction. This means that overall the system could make accurate, highly confident responses at the domain-level with the information available at L4 while basic-level identification required more fine-grained representations at L1. The Taylor et al study did not address this question directly, i.e. whether response times were different between tasks, since the very nature of the two different experiments (basic-level **naming** vs. domain-level **button-press**) would have confounded any statistical tests. However, in the case of the model, there is no such confound. ELRs in both tasks utilised the exact same information across layers. The number of possible responses had no effect on ELR values. As such, the reported difference is a reflection of the representational capacities in each layer. This finding is particularly relevant to a study by Mace et al (2009), who conducted an experiment where participants had to identify objects at either the superordinate or basic-level of specificity using a go/no-go experimental framework. It was found that participants took longer to make basic-level decisions (dog vs. non-dog or bird vs. non-bird) of objects compared to superodinate-level judgements (animal vs. non-animal). In line with the discrepancy in ELRs between tasks, this would suggest that to identify an object at the basic-level requires a further stage of processing where representations are more fine-grained than those needed for domain decisions. This observation is in direct agreement with previous studies conducted on human participants (Taylor, 2007; Tyler and Moss, 2001, Moss et al, 2005).

ELRs and cLength

As a follow-up analysis I also correlated the ELRs, from both specificity levels, with *cLength*. This was done to assess the effect of feature statistics on how “fast” the model was able to make an accurate response and whether the nature of this relationship reflected the findings of the Taylor et al study. I expected that basic-level ELRs would be positively correlated with cLength (more shared features, more processing required for disambiguation) while domain-level ELRs should be negatively correlated (more shared features would facilitate processing leading to small ELR values). Basic-level ELRs were found to be **positively** correlated with cLength (r = 0.52**) while domain-level ELRs were also mildly correlated but in the same direction (r = 0.1*). With respect to the latter, given the findings from Taylor et al (2012), it would be expected that there would be a **negative** correlation between the two variables. cLength would have a *facilitatory* effect on domain-level responses leading to high-cLength concepts scoring low ELRs.

I decided to pursue the matter further by trying to determine whether there were domain-specific effects. I correlated domain-level ELRs and cLength but this time separately for each domain. I found that there was indeed a significantly strong **negative** correlation between cLength and ELRs for living things, as expected as per Taylor et al, (r = -0.28; p<0.01) but **no significant correlation** for nonliving things (r = 0.07). This discrepancy between living and non-living things serves as an indication of what might drive ELRs for domain-level responses. In the case of living things, high *cLength* brings concepts together across the entire domain (by virtue of highly shared features such as ‘*has_legs*’) thus allowing the model to make accurate responses early on during the processing stages. However, in the case of nonliving things, high cLength has no effect exactly because nonliving things tend to have more distinctive features, compared to living things, and fewer features which are shared across the entire domain. In other words, the sharedness of features across nonliving things, in general, is not sufficiently high enough to facilitate domain-level responses. This reasoning is closely related to the one given in the previous chapter in order to explain a similar discrepancy in correlations between domain-level uncertainty (at L4) and cLength (see **Sections 5.3.4** and **5.4.3**). As with domain-level ELRs, cLength was correlated with domain-level uncertainty but *only* in the case of living things.

Ultimately, what does this relationship mean in terms of human behaviour? Given that domain-level ELRs are only mildly correlated with cLength the strength of the relationship between model and human responses cannot be fully dependent on CSA-relevant variables. This suggests that the model ELRs might be capturing an aspect of processing, in human participants, which is *not* accounted by the CSA, or at least cLength. One possible explanation is that the model extracts high-level, statistical feature properties, which have yet to be uncovered and are orthogonal to the ones described and understood so far (e.g. cLength, sharedness, correlational strength). Domain-level L4-uncertainty, for example, is not correlated with *cLength* either (see **Chapter 5**; **Figure 5.21**) and yet it produces a strongly significant correlation with human reaction times. Further research is required to properly characterise these properties and their relevance with behaviour.

6.4.2 Imaging data analysis

For this analysis, I hypothesised that the manner in which concepts dissociate from their category along the ventral stream should have some relevance to uncertainty (as computed by the model). Uncertainty is a way to quantify the system's degree of confidence in making a decision. If a system has a small degree of confidence then this means that subsequent stages in processing will readily dissociate the concept from its category in an effort to facilitate decisions. This seems to be the case during the experimental task in Clarke and Tyler (2014). Specifically, the slope of the trend of the dissociation correlates strong with uncertainty, at L4, within posterior regions of the ventral stream. As concepts become more and more dissociated from their category, the particular type of uncertainty quantified at L4 (sensitive to categorical similarity) becomes less and less relevant. Uncertainty becomes increasingly more driven by the semantically closest concepts. At the most anterior parts of the ventral stream the trend of dissociation becomes more dependent on uncertainty at L1 which is driven by the representational density of a concept's immediate semantic neighbourhood.

Results from the imaging study have also shown that the effect of L1-uncertainty on representational change extends well into the perirhinal cortex which is the neuroanatomical terminus of the ventral stream. As mentioned previously, the area has been specifically hypothesised to be involved in representing information at a very high degree of detail. With this in mind, it is not surprising that L1-uncertainty (which is high for concepts with dense, immediate semantic neighbourhoods with highly similar concepts) will drive the manner in which representations change within the area. Given that the perirhinal cortex has the capacity to represent information at a fine level of detail, concepts scoring high on L1-uncertainty will undergo a more pronounced dissociation from their category to facilitate identification. Interestingly enough L4-uncertainty also produced an effect, although only mildly significant, within portions of the perirhinal which were even more anterior to the L1-uncertainty effects. This could either be the result of a spurious correlation (given the weakness of the effect) or a reflection of inter-regional differences in functionality. Such a division, with respect to object naming, has already been reported by Kivisaari et al (2012) but it is organised across the medial-lateral axis of the region. Specifically, they reported that performance in naming objects was selectively associated with medial, and not lateral portions, of the perirhinal cortex. In a review by the same authors (Kivisaari et al, 2013) they also report that anterior perihinal cortex is tightly connected with its neighbouring anterior entorhinal cortex – itself an integral part of the semantic memory system (Davies et al, 2004). A more neuroanatomically relevant study by Rosen et al (1992) has shown that the anterior perirhinal cortex is an important gateway linking neural projections from the ventral stream with amygdaloid structures. In addition, a study conducted on rats by McIntyre et al (1996) have shown that there are direct connections from the most anterior tips of the perirhinal cortex with frontal regions. Collectively these studies suggest that the neuro-functional organisation of the region is not homogeneous which might explain why there are different effects of uncertainty. In the case of the anterior portion, representations here might reflect gated information to higher or memory-related areas.

In relation to the distinct representational capacities of the perirhinal cortex, previous studies have also made reference to ‘feature ambiguity’ which describes a situation where a feature is associated with reward within one stimulus but not in another (on monkeys: Bussey et al, 2002; on rats: Norman and Eacott, 2004; Bartko et al, 2007; Clark et al, 2011). These studies have contended that the perirhinal cortex is crucial for the resolution of feature ambiguity in tasks which are particularly difficult in this regard because of its capacity to represent complex conjunctions of features. This view of the perirhinal cortex is not unlike to what layer L1 of the model is thought to encapsulate. Resolving either feature ambiguity or uncertainty requires a rich representation which takes into account all the constituent features (be it semantic or visual) of a stimulus. In the present model, this type of representation is only instantiated in the latest stages of the top-down pass. A related connectionist model by Bussey and Saksida (2002) comprised of two layers – a feature layer and a feature conjunction layer. Similar to the present model, the architecture was hierarchical and representations were emphatically feature-based. It was shown that virtually 'lesioning' the latter layer of the network resulted in profound deteriorations of model performance when undertaking visual discrimination tasks. Can the 'feature conjunction' layer of this particular model be related to layer L1? A comparable future step would be to test the present model under similar discrimination tasks. The expectation would be that removing L1 would severely affect the model's performance in carrying out the task in question similar to the findings of Bussey and Saksida. However, equating the two models, along with their underlying principles (feature ambiguity vs. uncertainty), would be quite premature at this point. This is because there are fundamental differences in the mechanisms which characterise the two models: the Bussey-Saksida model is a connectionist model trained on classifying inputs into different targets whilst the DBN used herein is a *generative* model where the formation of representations contained in each layer is not guided by any external input. Furthermore uncertainty (as defined and used in this dissertation) is derived differently from feature ambiguity: the former is concept-specific and explicitly extracted from the model itself while the latter is not strictly a quantitative concept-specific measure but rather a qualitative way to describe visual discrimantion tasks. Nonetheless, such an endeavour still provides an interesting avenue for further research in an effort to subsume previous modelling and imaging studies within an all-encompassing theoretical framework.

6.4.3 Uncertainty and conceptual representations

My hypothesis on the onset of this study was that conceptual processing in the brain goes through different levels of representation (Hinton, 2007). Each level brings to bear a degree of refinement on the prediction being made by the previous one thus decreasing uncertainty. Uncertainty in this sense plays a key role since it is what the system is trying to optimally minimise (Friston, 2003). I tried to test this view in two ways, first by directly comparing model-derived variables (ELRs and uncertainty) against human behavioural responses and secondly by assessing the correlational relationship between uncertainty and the specific manner in which representations change along the ventral stream. In both cases I have shown that the model has a significantly strong predictive power over specific aspects of processing during object recognition. This suggests that the computational mechanisms of the model can provide a principle with which to characterise conceptual representations: representational organisation along the posterior-anterior axis of the ventral stream is determined according to the capacity to resolve uncertainty. This view fits with the one held by the CSA as outlined in Moss et al (2007) and Taylor et al (2011) where different tasks demand the engagement of different stages along the stream depending on the representational specificity required.